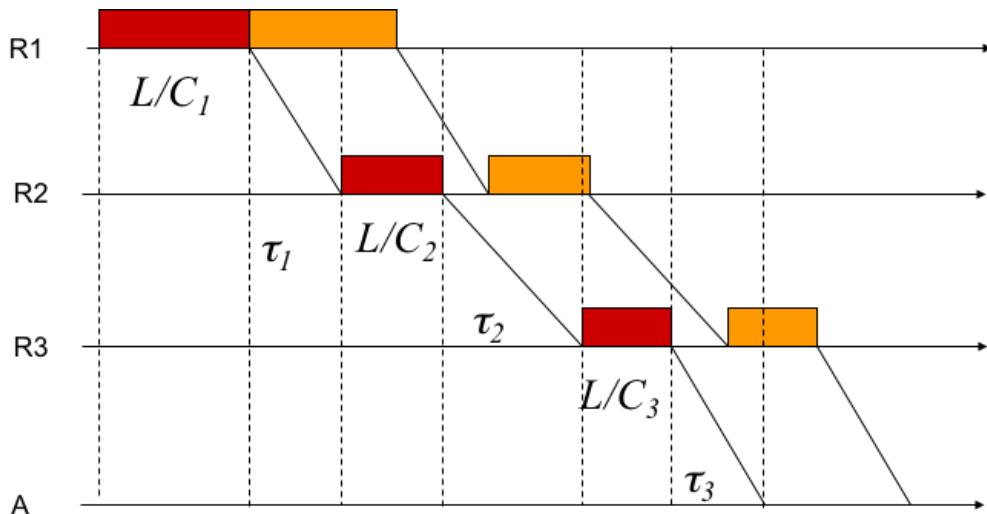
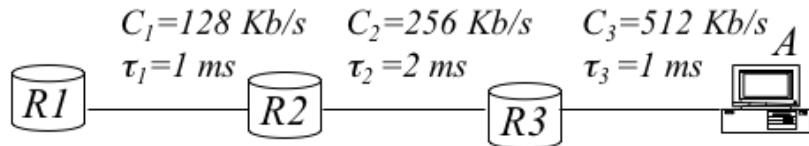


1. Esercizi su ritardi e tempi di trasmissione

Esercizio 1.1

Si consideri la rete in figura. Al tempo t=0 la coda di uscita di R1 ha 2 pacchetti diretti ad A. Assumendo lunghezza dei pacchetti di L=512 [bits], si indichi per ciascun pacchetto l'istante in cui viene completamente ricevuto a destinazione.



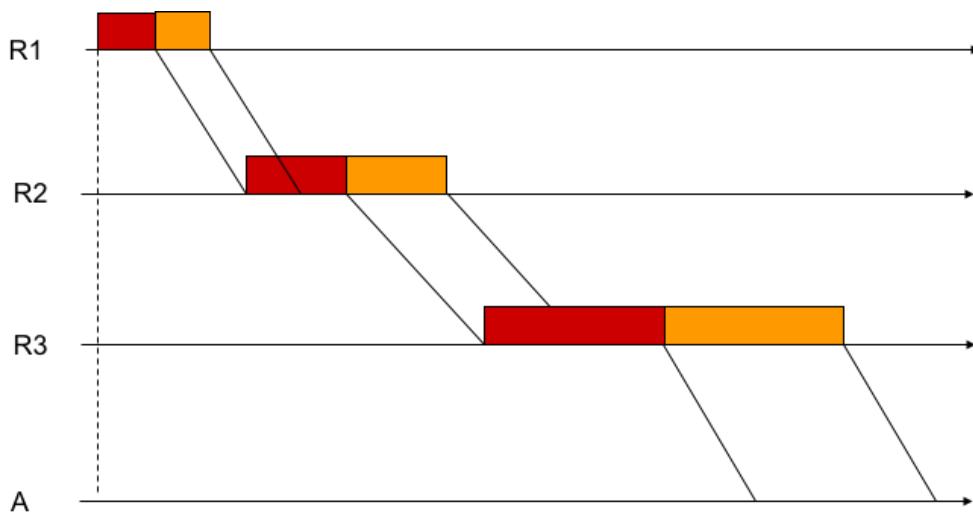
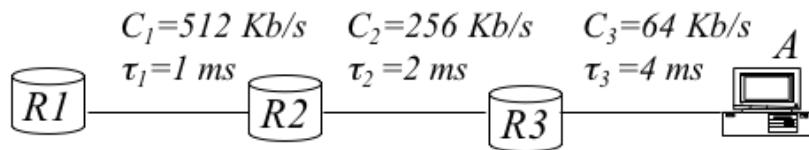
Non abbiamo casi di accodamento. Ad ogni nodo attraversato, il secondo pacchetto finisce la propria ricezione dal nodo precedente dopo che il primo pacchetto ha finito la propria trasmissione verso il nodo successivo. In questo modo, appena finita la ricezione, il secondo pacchetto può essere immediatamente ritrasmesso verso il nodo successivo.

$$T_1 = \frac{L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 + \frac{L}{C_3} + \tau_3 = 4 + 1 + 2 + 2 + 1 + 1 = 11 \text{ ms}$$

$$T_2 = \frac{2L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 + \frac{L}{C_3} + \tau_3 = 15 \text{ ms}$$

Esercizio 1.2

Si consideri la rete in figura. Al tempo $t=0$ la coda di uscita di R1 ha 2 pacchetti diretti ad A. Assumendo lunghezza dei pacchetti di $L=512$ [bits], si indichi per ciascun pacchetto l'istante in cui viene completamente ricevuto a destinazione.



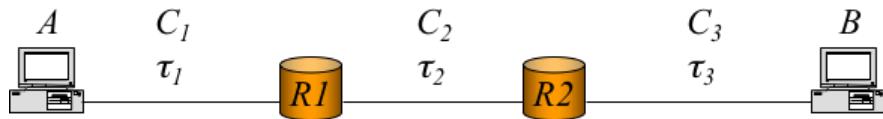
Il link tra R2 e R3 ha un rate trasmissivo minore del link tra R1 e R2, dunque il secondo pacchetto finisce la propria ricezione nel nodo R2 mentre il primo pacchetto è ancora in trasmissione da R2 a R3. Il secondo pacchetto non può essere immediatamente ritrasmesso verso R3, ma deve attendere la fine della trasmissione del primo pacchetto, che sta occupando l'interfaccia trasmissiva tra R2 e R3. Il secondo pacchetto viene dunque accodato in attesa che l'interfaccia si liberi. Lo stesso accade nell'hop successivo.

$$T_1 = \frac{L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 + \frac{L}{C_3} = 1 + 1 + 2 + 2 + 8 + 4 = 18 \text{ [ms]}$$

$$T_2 = \frac{L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 + \frac{2L}{C_3} = T_1 + \frac{L}{C_3} = 26 \text{ [ms]}$$

Esercizio 1.3

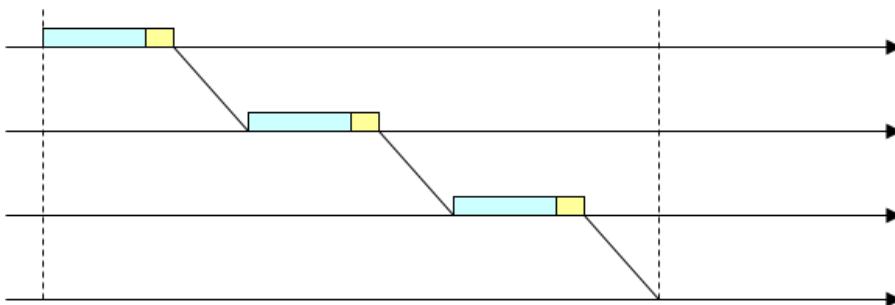
Si consideri la rete in figura.



- Si calcoli in forma parametrica il tempo necessario a trasmettere un pacchetto da A a B (header h, dati D).
 - Si assume di dividere il pacchetto in 2 frammenti. Si calcoli in forma parametrica il tempo necessario per trasmettere tutti i frammenti. Si assuma $C_2 \leq C_1 \leq C_3$
 - Qual è il numero di frammenti che minimizza il ritardo?
-

- La lunghezza di ogni pacchetto è data dalla somma dei bit di header e di dati

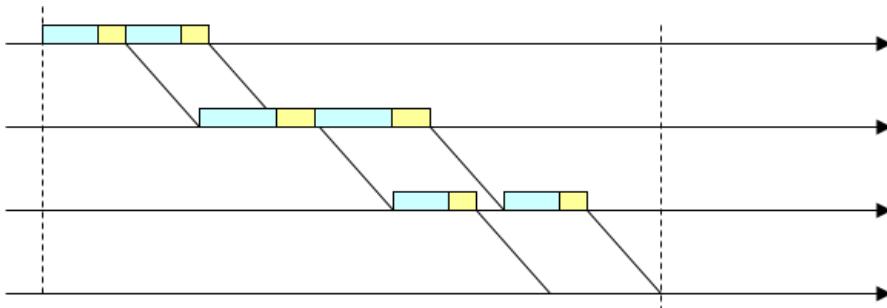
$$T = \frac{h+D}{C_1} + \tau_1 + \frac{h+D}{C_2} + \tau_2 + \frac{h+D}{C_3} + \tau_3$$



- Ogni frammento sarà formato da una parte dati pari alla metà della parte dati del pacchetto originario, mentre la lunghezza dell'header rimane uguale. Ogni frammento avrà bisogno del proprio header per poter essere processato dai nodi intermedi.

$$d = D/2$$

$$T = \frac{h+d}{C_1} + \tau_1 + \frac{2(h+d)}{C_2} + \tau_2 + \frac{h+d}{C_3} + \tau_3$$



- Nell'espressione parametrica la dimensione di ciascuno degli n frammenti sarà pari ad $1/n$ della dimensione del pacchetto originario

$$\begin{aligned}
T &= \frac{h+D/n}{C_1} + \tau_1 + \frac{n(h+D/n)}{C_2} + \tau_2 + \frac{h+D/n}{C_3} + \tau_3 = \\
&= \left(\frac{h}{C_1} + \tau_1 + \frac{D}{C_2} + \tau_2 + \frac{h}{C_3} + \tau_3 \right) + \frac{D}{nC_1} + \frac{nh}{C_2} + \frac{D}{nC_3}
\end{aligned}$$

Troviamo il punto di stazionarietà

$$\frac{\partial T}{\partial n} = \frac{h}{C_2} - \frac{D}{n^2 C_1} - \frac{D}{n^2 C_3} = 0$$

$$n^* = \sqrt{\frac{C_2}{h} \left(\frac{D}{C_1} + \frac{D}{C_3} \right)}$$

Esempio numerico

$$C_1 = 1 \text{ Mbit/s}$$

$$C_2 = 900 \text{ Kbit/s}$$

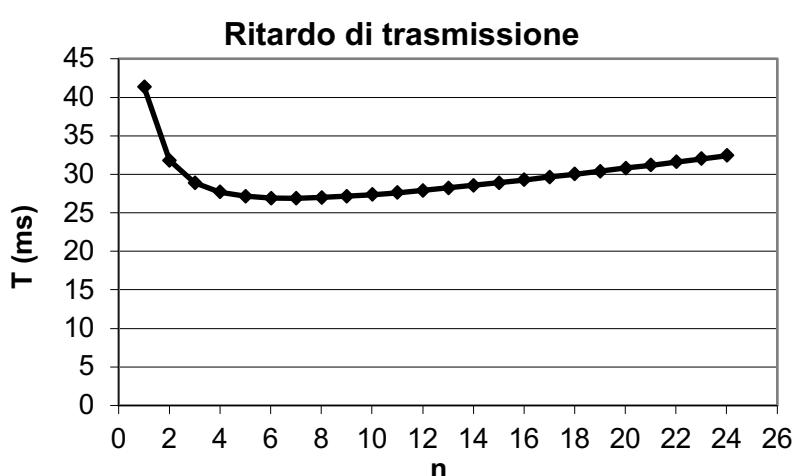
$$C_3 = 1 \text{ Mbit/s}$$

$$\tau_1 = \tau_2 = \tau_3 = 3 \text{ ms}$$

$$h = 400$$

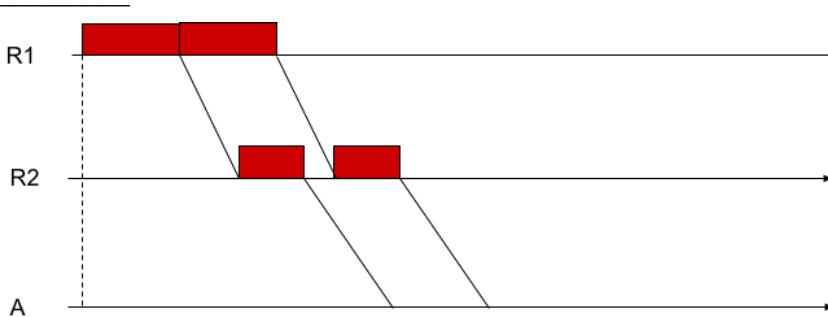
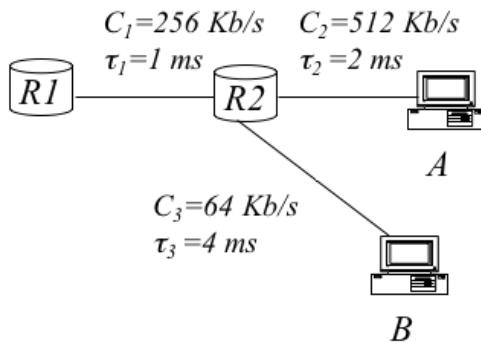
$$D = 10000$$

$$n^* = \sqrt{\frac{C_2}{h} \left(\frac{D}{C_1} + \frac{D}{C_3} \right)} = 6,71$$



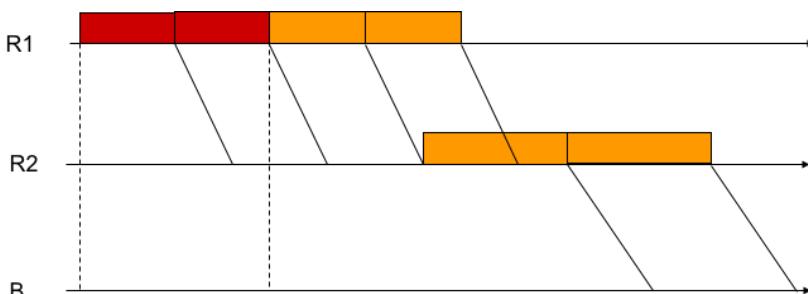
Esercizio 1.4

Si consideri la rete in figura. Al tempo $t=0$ la coda di uscita di R1 ha 4 pacchetti diretti rispettivamente A, A, B, B. Assumendo lunghezza dei pacchetti di $L=512$ [bits], si indichi per ciascun pacchetto l'istante in cui viene completamente ricevuto a destinazione.



$$T_1 = \frac{L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 = 2 + 1 + 1 + 2 = 6 \text{ ms}$$

$$T_2 = \frac{L+L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 = 8 \text{ ms}$$



Una volta ricevuti al nodo R2, i pacchetti diretti ad A e B verranno gestiti in maniera indipendente. Infatti, i primi verranno accodati in uscita all'interfaccia tra R2 e A, mentre i secondi all'interfaccia tra R2 e B.

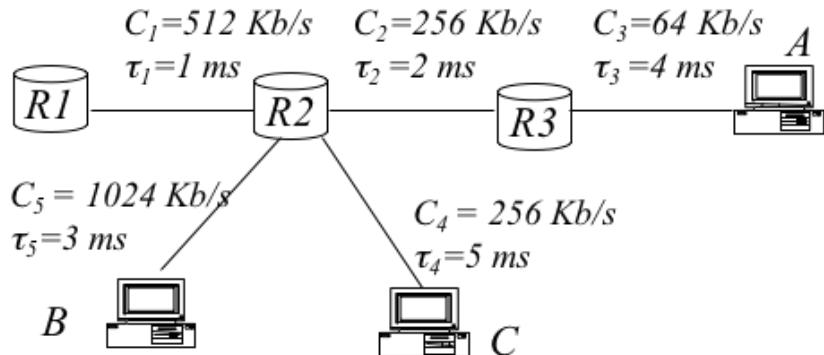
Dato che il link R2-B ha un rate trasmissivo minore del link R1-R2, i pacchetti diretti a B verranno accodati in uscita da R2

$$T_3 = \frac{L+L+L}{C_1} + \tau_1 + \frac{L}{C_3} + \tau_3 = 6 + 1 + 8 + 4 = 19 \text{ ms}$$

$$T_4 = T_3 + \frac{L}{C_3} = 19 + 8 = 27 \text{ ms}$$

Esercizio 1.5

Si consideri la rete in figura. Al tempo $t=0$ la coda di uscita di R1 ha 6 pacchetti diretti rispettivamente A, A, B, B, C, C. Assumendo lunghezza dei pacchetti di $L=512$ [bits], si indichi per ciascun pacchetto l'istante in cui viene completamente ricevuto a destinazione.



Il primo pacchetto arriva al nodo A senza incontrare altri pacchetti in rete. Il secondo pacchetto, sempre diretto a A, verrà accodato, dato che i link successivi al primo hanno un rate trasmissivo minore.

$$T_1 = \frac{L}{C_1} + \tau_1 + \frac{L}{C_2} + \tau_2 + \frac{L}{C_3} + \tau_3 = 18 \text{ ms}$$

$$T_2 = T_1 + \frac{L}{C_3} = 26 \text{ ms}$$

I pacchetti diretti a B vengono trasmessi da R1 dopo quelli diretti a A, e da R2 in poi ne diventano indipendenti. Non c'è accodamento tra i pacchetti di B perché il link R2-B ha un rate trasmissivo maggiore di R1-R2

$$T_3 = \frac{3L}{C_1} + \tau_1 + \frac{L}{C_5} + \tau_5 = 7.5 \text{ ms}$$

$$T_4 = \frac{4L}{C_1} + \tau_1 + \frac{L}{C_5} + \tau_5 = 8.5 \text{ ms}$$

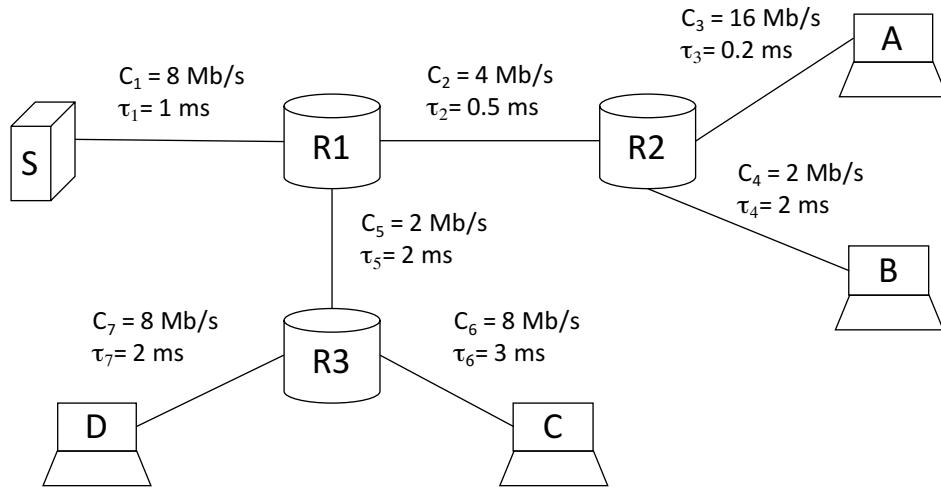
I pacchetti diretti a C vengono trasmessi da R1 dopo quelli diretti a A e B, e da R2 in poi ne diventano indipendenti. Dato che R2-C ha un rate trasmissivo minore di R1-R2, abbiamo accodamento tra i pacchetti diretti a C.

$$T_5 = \frac{5L}{C_1} + \tau_1 + \frac{L}{C_4} + \tau_4 = 13 \text{ ms}$$

$$T_6 = T_5 + \frac{L}{C_4} = 15 \text{ ms}$$

Esercizio 1.6 (E)

In una rete a commutazione di pacchetto al tempo $t=0$ sono presenti 8 pacchetti in S diretti rispettivamente alle seguenti destinazioni: A, A, B, A, C, C, D, D. Calcolare il tempo di ricezione di ciascuno dei pacchetti assumendo che i pacchetti abbiano le seguenti dimensioni: pacchetti verso A, $L_A=1000$ [byte]; pacchetti verso B, $L_B=2000$ [byte]; pacchetti verso C, $L_C=500$ [byte]; pacchetti verso D, $L_D=1000$ [byte].



$$T_1^A = \frac{L_A}{C_1} = \frac{8 \cdot 10^3}{8 \cdot 10^6} = 1 \text{ ms}$$

$$T_1^C = \frac{L_C}{C_1} = 0.5 \text{ ms}$$

$$T_2^A = \frac{L_A}{C_2} = 2 \text{ ms}$$

$$T_5^C = \frac{L_C}{C_5} = 2 \text{ ms}$$

$$T_3^A = \frac{L_A}{C_3} = 0.5 \text{ ms}$$

$$T_6^C = \frac{L_C}{C_6} = 0.5 \text{ ms}$$

$$T_1^B = \frac{L_B}{C_1} = 2 \text{ ms}$$

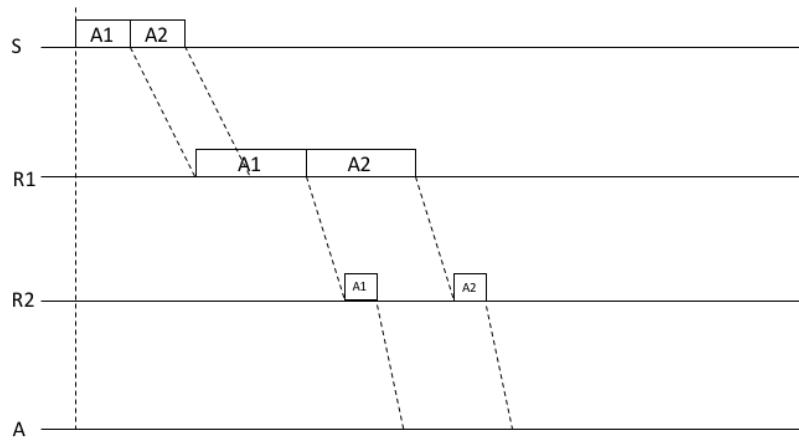
$$T_1^D = \frac{L_D}{C_1} = 1 \text{ ms}$$

$$T_2^B = \frac{L_B}{C_2} = 4 \text{ ms}$$

$$T_5^D = \frac{L_D}{C_5} = 4 \text{ ms}$$

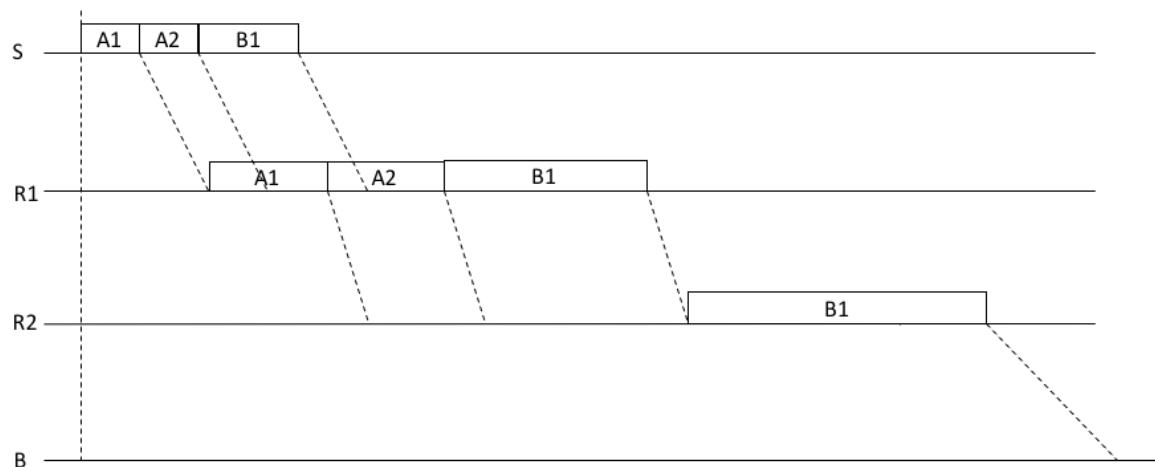
$$T_4^B = \frac{L_B}{C_4} = 8 \text{ ms}$$

$$T_7^D = \frac{L_{CD}}{C_7} = 1 \text{ ms}$$

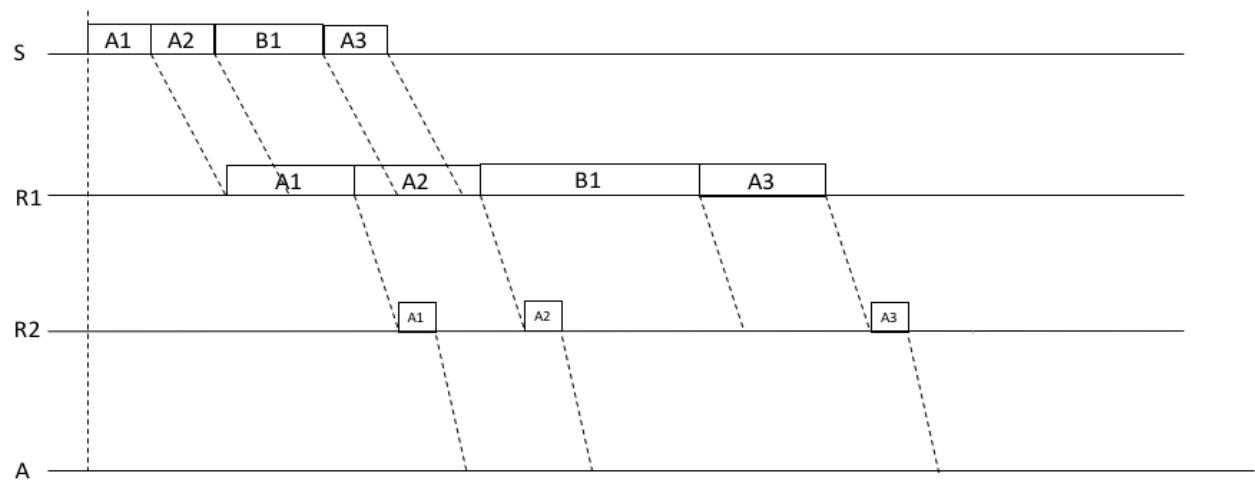


$$T_{A1} = T_1^A + \tau_1 + T_2^A + \tau_2 + T_3^A + \tau_3 = 5.2 \text{ ms}$$

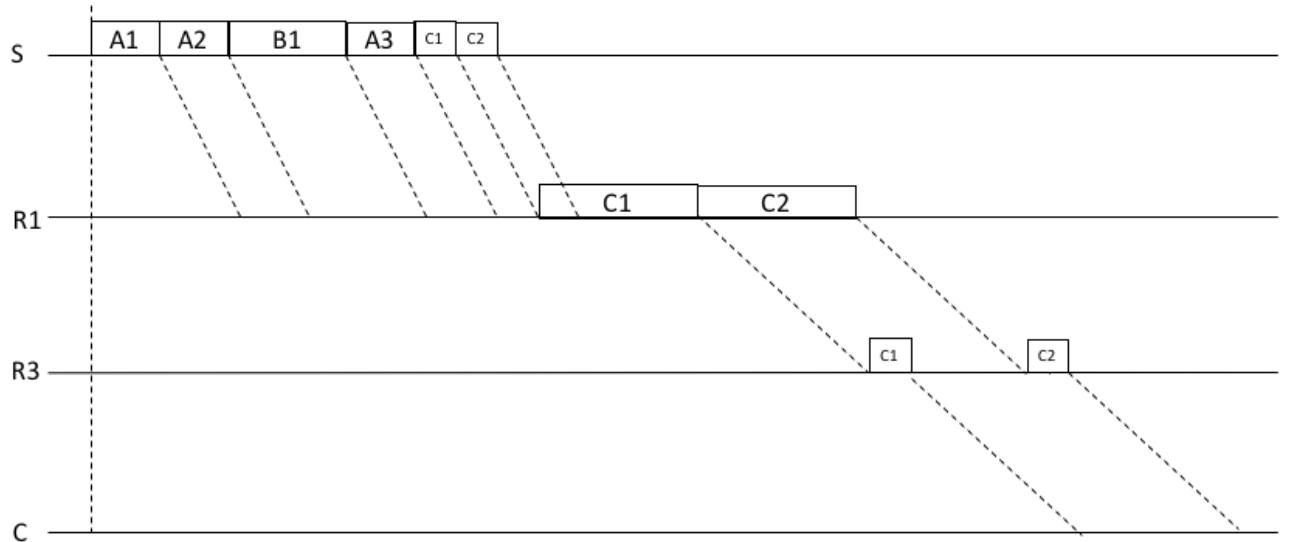
$$T_{A2} = T_{A1} + T_2^A = 7.2 \text{ ms}$$



$$T_{B1} = T_1^A + \tau_1 + 2T_2^A + T_2^B + \tau_2 + T_4^B + \tau_4 = 1 + 1 + 2 \times 2 + 4 + 0.5 + 8 + 2 = 20.5 \text{ ms}$$

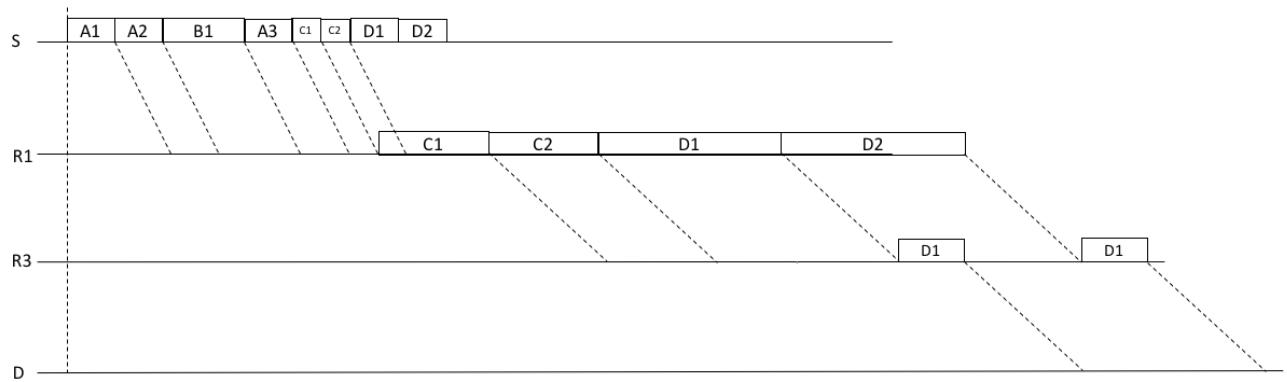


$$T_{A3} = T_1^A + \tau_1 + 3T_2^A + T_2^B + \tau_2 + T_3^A + \tau_3 = 1 + 1 + 3 \times 2 + 4 + 0.5 + 0.5 + 0.2 = 13.2 \text{ ms}$$



$$T_{C1} = 3 T_1^A + T_1^B + T_1^C + \tau_1 + T_5^C + \tau_5 + T_6^C + \tau_6 = 14 \text{ ms}$$

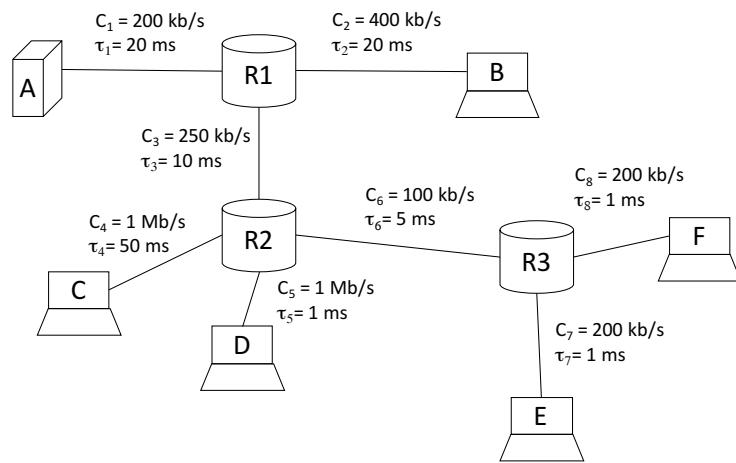
$$T_{C2} = T_{C1} + T_5^C = 16 \text{ ms}$$



$$T_{D1} = 3 T_1^A + T_1^B + T_1^C + \tau_1 + 2T_5^C + T_5^D + \tau_5 + T_7^D + \tau_7 = 19.5 \text{ ms}$$

$$T_{D1} = T_{D1} + T_5^D = 23.5 \text{ ms}$$

Esercizio 1.7 (E)



Nella rete a commutazione di pacchetto in figura, al tempo $t=0$ sono presenti 5 pacchetti in A diretti rispettivamente alle seguenti destinazioni: C, D, E, F, E. Calcolare l'istante di fine ricezione degli ultimi 3 pacchetti a destinazione assumendo che i pacchetti abbiano le seguenti dimensioni: pacchetti verso C, $L_C = 375$ [byte]; pacchetti verso D, $L_D = 250$ [byte]; pacchetti verso E, $L_E = 375$ byte; pacchetti verso F, $L_F = 125$ [byte].

$$T_1^C = \frac{L_C}{C_1} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_3^C = \frac{L_C}{C_3} = \frac{375 * 8 \text{ bit}}{250 \text{ kbps}} = 12 \text{ ms}$$

$$T_1^D = \frac{L_D}{C_1} = \frac{250 * 8 \text{ bit}}{200 \text{ kbps}} = 10 \text{ ms}$$

$$T_3^D = \frac{L_D}{C_3} = \frac{250 * 8 \text{ bit}}{250 \text{ kbps}} = 8 \text{ ms}$$

$$T_1^E = \frac{L_E}{C_1} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_3^E = \frac{L_E}{C_3} = \frac{375 * 8 \text{ bit}}{250 \text{ kbps}} = 12 \text{ ms}$$

$$T_6^E = \frac{L_E}{C_6} = \frac{375 * 8 \text{ bit}}{100 \text{ kbps}} = 30 \text{ ms}$$

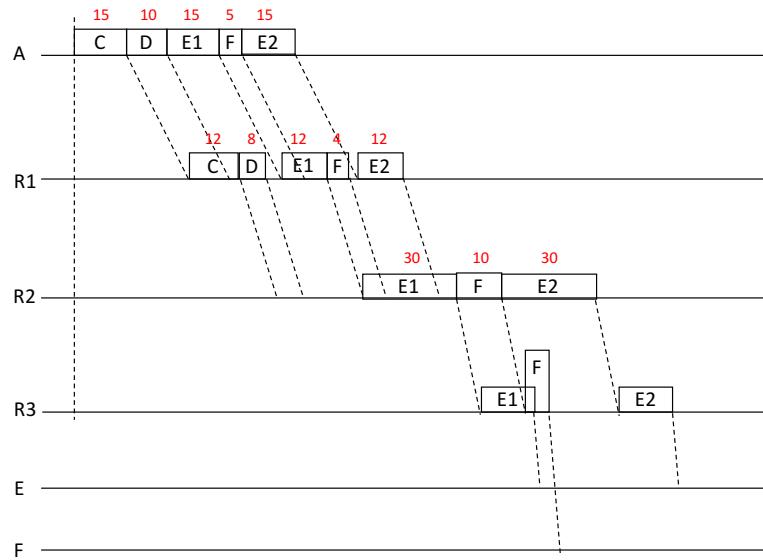
$$T_7^E = \frac{L_E}{C_7} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_1^F = \frac{L_F}{C_1} = \frac{125 * 8 \text{ bit}}{200 \text{ kbps}} = 5 \text{ ms}$$

$$T_3^F = \frac{L_F}{C_3} = \frac{125 * 8 \text{ bit}}{250 \text{ kbps}} = 4 \text{ ms}$$

$$T_6^F = \frac{L_F}{C_6} = \frac{125 * 8 \text{ bit}}{100 \text{ kbps}} = 10 \text{ ms}$$

$$T_8^F = \frac{L_F}{C_8} = \frac{125 * 8 \text{ bit}}{200 \text{ kbps}} = 5 \text{ ms}$$



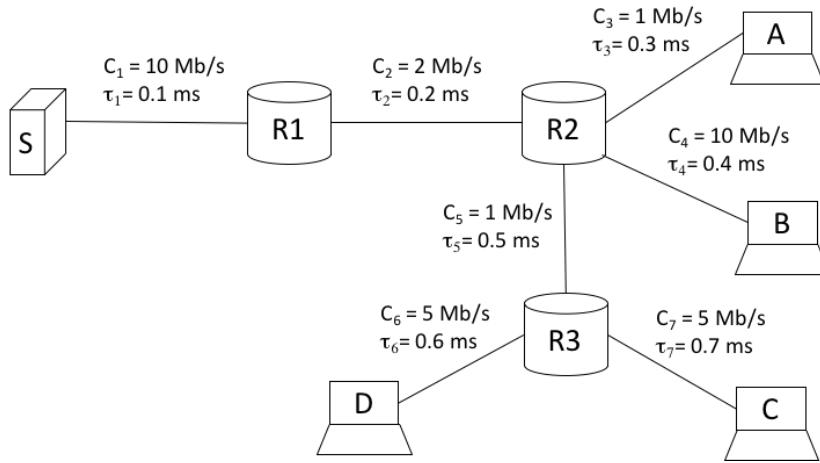
$$\begin{aligned}
 T_{E1} &= T_1^C + T_1^D + T_1^E + \tau_1 + T_3^E + \tau_3 + T_6^E + \tau_6 + T_7^E + \tau_7 \\
 &= 15 + 10 + 15 + 20 + 12 + 10 + 30 + 5 + 15 + 1 = 133 \text{ ms}
 \end{aligned}$$

$$\begin{aligned}
 T_F &= T_1^C + T_1^D + T_1^E + \tau_1 + T_3^E + \tau_3 + T_6^E + T_6^F + \tau_6 + T_8^F + \tau_8 \\
 &= 15 + 10 + 15 + 20 + 12 + 10 + 30 + 10 + 5 + 5 + 1 = 133 \text{ ms}
 \end{aligned}$$

$$T_{E2} = T_{E1} + T_6^E + T_6^F = 133 + 30 + 10 = 173 \text{ ms}$$

Esercizio 1.8 (E)

In una rete a commutazione di pacchetto al tempo t=0 sono presenti 6 pacchetti in S diretti rispettivamente alle seguenti destinazioni: A, A, B, B, C, D. Calcolare il tempo di ricezione di ciascuno dei pacchetti assumendo che i pacchetti abbiano le seguenti dimensioni: pacchetti verso A, L_A=1250 [byte]; pacchetti verso B, L_B=250 [byte]; pacchetti verso C, L_C=1250 [byte]; pacchetti verso D, L_D=1250 [byte].



$$T_{A1} = T_1^A + \tau_1 + T_2^A + \tau_2 + T_3^A + \tau_3 = 16.6 \text{ ms}$$

$$T_{A2} = T_{A1} + T_3^A = 26.6 \text{ ms}$$

$$T_{B1} = T_1^A + \tau_1 + 2T_2^A + T_2^B + \tau_2 + T_4^B + \tau_4 = 12.9 \text{ ms}$$

$$T_{B2} = T_{B1} + T_2^B = 13.9 \text{ ms}$$

$$T_{C1} = T_1^A + \tau_1 + 2T_2^A + 2T_2^B + T_2^C + \tau_2 + T_5^C + \tau_5 + T_7^C + \tau_7 = 31.5 \text{ ms}$$

$$T_{D1} = T_1^A + \tau_1 + 2T_2^A + 2T_2^B + T_2^C + \tau_2 + T_5^C + T_5^D + \tau_5 + T_6^D + \tau_6 = 39.4 \text{ ms}$$

2. Esercizi sul LIVELLO FISICO

Esercizio 2.1

Un segnale analogico ha una banda che si estende fino alla frequenza di 120 Hz. Dopo il campionamento il segnale è quantizzato con 256 livelli. Il segnale è infine convertito in una sequenza digitale e trasmesso in banda base attraverso un mezzo trasmissivo. Si calcoli la banda del mezzo necessaria per trasmettere il segnale nel caso di:

1. modulazione di ampiezza a 2 livelli di tipo 2-PAM.
 2. modulazione di ampiezza a 8 livelli di tipo 8-PAM.
-

Il rate di campionamento che rispetta il teorema di Nyquist è $f_{\text{campionamento}} = 2 f_{\text{max}}$.

I 256 livelli di quantizzazione sono rappresentati mediante $\log_2(256) = 8$ bit.

Il rate di trasmissione all'uscita del convertitore analogico/digitale è quindi è:

$$R_b = (240 \text{ sample/s}) \times (8 \text{ bits/sample}) = 1,92 \text{ kb/s}$$

Nel caso 1 il formato è binario: si trasmette quindi 1 bit per simbolo. Il rate di trasmissione resta $R_{TX} = 1,92 \text{ kb/s}$. La banda passante necessaria al mezzo trasmissivo deve quindi essere maggiore di $B_{\text{canale}} = 1,92 \text{ kHz}$.

Nel caso 2, ogni livello di ampiezza del formato di modulazione 8-PAM corrisponde a $\log_2(8) = 3$ bit. Vengono trasmessi simboli, ognuno dei quali corrisponde a 3 bit. Quindi il rate di trasmissione dei simboli è $R_{TX} = 1920/3 = 640 \text{ b/s}$. La banda passante necessaria al mezzo trasmissivo deve quindi essere maggiore di $B_{\text{canale}} = 640 \text{ Hz}$.

Esercizio 2.2

Un segnale musicale analogico con banda $B = 20 \text{ kHz}$ è quantizzato su 1024 livelli dopo essere stato campionato e trasmesso in banda base su un canale trasmissivo sfruttando il formato di modulazione binario 2-PAM. Il campionamento rispetta il teorema di Nyquist. Si calcoli la banda del mezzo trasmissivo richiesta per la trasmissione digitale del segnale.

Per soddisfare il teorema di Nyquist la frequenza di campionamento deve essere maggiore del doppio della banda massima del segnale -> $f_{\text{campionamento}} > 2 \times 20 \text{ kHz} = 40 \text{ kHz}$.

I 1024 livelli di quantizzazione sono rappresentati mediante $\log_2(1024) = 10 \text{ bit}$.

Il rate di trasmissione all'uscita del convertitore analogico/digitale è quindi è:

$$R_b = (40 \text{ ksample/s}) \times (10 \text{ bits/sample}) = 400 \text{ kb/s}$$

In caso di modulazione 2-PAM, ciascun impulso corrisponde ad 1 bit.

Il rate di trasmissione degli impulsi resta $R_{TX} = 400 \text{ kb/s}$. La banda passante necessaria al mezzo trasmissivo deve quindi essere maggiore di $B_{\text{canale}} = 400 \text{ kHz}$.

Esercizio 2.3

Un segnale analogico con banda $B = 4 \text{ kHz}$ è campionato alla frequenza 10 kHz e quantizzato con 256 livelli. I campioni sono convertiti in segnale binario e trasmessi in banda traslata su un canale con banda passante $B_c = 50 \text{ kHz}$. Si valuti se la banda passante del canale trasmissivo è sufficiente a trasmettere il segnale in forma digitale nel caso di:

1. modulazione a 16 livelli di fase e di ampiezza di tipo 16QAM.
2. modulazione di fase a 4 livelli di tipo QPSK.

IMPORTANTE: nel caso di trasmissione in banda passante, si consideri che la banda passante del canale sia almeno il doppio rispetto a quella necessaria nel caso di trasmissione in banda base.

Il rate di campionamento rispetta il teorema di Nyquist poiché $f_{\text{campionamento}} > 2 f_{\text{max}}$.

I 256 livelli di quantizzazione sono rappresentati mediante $\log_2(256) = 8 \text{ bit}$.

Il rate di trasmissione all'uscita del convertitore analogico/digitale è quindi è:

$$R_b = (10 \text{ ksample/s}) \times (8 \text{ bits/sample}) = 80 \text{ kb/s.}$$

In caso 1 di modulazione 16QAM, ciascun impulso corrisponde a $\log_2(16) = 4 \text{ bit}$. Il rate di trasmissione degli impulsi è quindi $R_{\text{TX}} = 80/4 = 20 \text{ kb/s}$. La banda passante necessaria al mezzo trasmissivo deve quindi essere $> 40 \text{ Hz}$ e quindi B_c risulta sufficiente.

In caso 2 di modulazione QPSK, ciascun impulso corrisponde a $\log_2(4) = 2 \text{ bit}$. Il rate di trasmissione degli impulsi è quindi $R_{\text{TX}} = 80/2 = 40 \text{ kb/s}$. La banda passante necessaria al mezzo trasmissivo deve quindi essere $> 80 \text{ Hz}$ e quindi B_c risulta non sufficiente.

Esercizio 2.4

Un CD audio è un sistema binario, dove il segnale musicale è campionato a 44,1 kHz e poi quantizzato per mezzo di 16 bit.

1. Si trovi la banda ideale (in banda base) del canale di trasmissione necessaria nel caso di modulazione binaria.

2. Si consideri che sul supporto ottico del CD (di dimensioni 86 cm²) siano registrati 20,4 Gbit di informazione (che includono i 2 canali stereofonici e anche il codice di correzione dell'errore). Sono necessari 74 minuti per riprodurre tutti i bit registrati. Si trovi la banda reale utilizzata dal canale di riproduzione del CD audio nel caso di trasmissione binaria.

Il rate di trasmissione ideale dopo il campionamento e la quantizzazione sarebbe:

$$R_b = (44,1 \text{ sample/s}) \times (16 \text{ bits/sample}) = 705,6 \text{ kb/s}$$

Nel caso di modulazione binaria il rate di trasmissione degli impulsi resta $R_{TX} = 705,6 \text{ kb/s}$. La banda ideale deve essere almeno 705,6 kHz.

Nel CD sono memorizzate però molte più informazioni da trasmettere. Il rate di trasmissione reale è infatti:

$$R_{TX} = (20,4 \text{ Gbit}) / (74 \times 60 \text{ sec}) = 4,594 \text{ Mb/s}$$

In banda base la banda reale necessaria deve quindi essere maggiore di 4,594 Mb/s.

Esercizio 2.5

Sfruttando un formato di modulazione multilivello di tipo M-QAM si vuole trasmettere in banda base un segnale numerico con un bit rate di 100 Mb/s usando un mezzo trasmissivo con banda disponibile $B_c = 25$ MHz. Quanti livelli M del formato di modulazione M-QAM sono richiesti?

La banda passante richiesta deve essere almeno $B_c = R_b / \log_2(M)$.

Quindi si ottiene $M = 2^k = 16$ livelli del formato di modulazione di tipo QAM.

Esercizio 2.6

Si consideri un collegamento in fibra ottica lungo $L = 20 \text{ km}$, caratterizzato da un'attenuazione per km $\alpha = 0,2 \text{ dB/km}$. Se la potenza del segnale in ingresso è $P_{IN} = 10 \text{ mW}$, si calcoli la potenza ottenuta in uscita.

Ragioniamo in dB. Convertiamo la potenza da mW a dBm, considerando 1mW come riferimento:

$$P_{IN_{dBm}} = 10 \log_{10} \frac{P_{IN}}{1\text{mW}} = 10 \log_{10} \frac{10}{1} = 10 \text{ dBm}$$

Le perdite complessive di tratta dovute all'attenuazione sono pari a $L = \alpha \times l = 0,2 \times 20 = 4 \text{ dB}$

$$P_{OUT} \text{ in dBm} = P_{IN} - L = 10 - 4 = 6 \text{ dBm}$$

$$P_{OUT} = 1\text{mW} \cdot 10^{\frac{6}{10}} = 10^6 \cdot 1\text{mW} = 4\text{mW}$$

Esercizio 2.7

Si consideri una rete in fibra ottica punto-a-punto caratterizzata da un'attenuazione per km $\alpha = 0,2 \text{ dB/km}$, le cui perdite complessive della tratta corrispondono a $L = 30 \text{ dB}$. Quanto è lunga la rete? Se la potenza del segnale in uscita è pari a -17 dBm , quanta potenza ha il segnale in ingresso alla rete?

Se le perdite complessive L sono 30 dB , allora la rete è lunga $L = 30 / 0,2 = 150 \text{ km}$.

$$P_{OUT} = 1mW \cdot 10^{\frac{P_{dB}}{10}} = 10^{\frac{-17}{10}} \cdot 1mW = 20\mu W$$

$$P_{IN_{dBm}} = P_{out_{dBm}} + perdite = -17 + 30 = +13dBm$$

$$P_{OUT} = 1mW \cdot 10^{\frac{P_{dB}}{10}} = 10^{\frac{+13}{10}} \cdot 1mW = 20mW$$

Esercizio 2.8

Si consideri una rete in fibra ottica punto-a-punto lunga $l = 45$ km. La potenza del segnale all'uscita del trasmettitore è pari a $P_{TX} = -8$ dBm. Il segnale è trasmesso su una portante ottica a 1310 nm, dove la fibra ottica presenta un'attenuazione per km pari a $\alpha = 0,35$ dB/km. A causa della presenza di connettori e filtri, la rete presenta ulteriori $A = 7,2$ dB di perdite. La potenza necessaria al ricevitore per ottenere le prestazioni volute è pari a $P_{RX}^* = -35$ dBm. Se il costruttore vuole tenersi almeno $M = 3,8$ dB di margine per future modifiche, la rete sta funzionando con le prestazioni richieste dal punto di vista fisico?

E' necessario calcolare il power budget della rete in dB.

La condizione richiesta è che la potenza ottenuta al ricevitore dopo la propagazione sia maggiore del valore richiesto per ottenere certe prestazioni.

Quindi: $P_{RX} = P_{TX} - \alpha l - A - M \geq P_{RX}^*$

Svolgendo i calcoli, si ottiene: $P_{RX} = -8 - 15,75 - 7,2 - 3,8 = -34,75$ dBm

La potenza ricevuta alla fine della rete $P_{RX} = -34,75$ dBm risulta maggiore della potenza richiesta $P_{RX}^* = -35$ dBm

Quindi la rete sta funzionando dal punto di vista fisico secondo le prestazioni richieste.

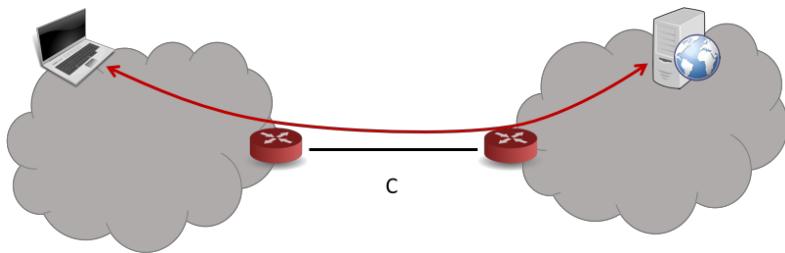
3. Esercizi sul LIVELLO APPLICATIVO

Esercizio 3.1

Un *client* HTTP richiede ad un *server* HTTP una pagina *web* costituita da un oggetto base (file HTML) e 10 altri oggetti. Ogni oggetto ha una dimensione $L=200$ [kbit]. Il collegamento tra *client* e *server* HTTP è in grado di trasferire informazione ad una velocità di $C=100$ [kbit/s] in entrambe le direzioni. I messaggi di controllo usati per aprire una connessione TCP tra *client* e *server* ed il messaggio di GET HTTP (di richiesta dell'oggetto base e gli altri oggetti) hanno lunghezza $l=100$ [bit]. Il ritardo di propagazione è trascurabile.

Calcolare il tempo totale per ricevere interamente la pagina *web* richiesta nei tre casi seguenti:

- il *client* HTTP apre in parallelo in modalità non persistente tutte le connessioni TCP necessarie per scaricare la pagina *web* (si assuma che il data rate della singola connessione sia $r=C/N$, con C data rate del collegamento e N numero di connessione aperte in parallelo)
- il *client* HTTP apre un'unica connessione TCP persistente per scaricare tutti gli oggetti della pagina *web*.
- Il *client* HTTP apre in serie 11 connessioni TCP in modalità non persistente.



- a) il client funziona in modo non persistente. Il client apre una connessione TCP non persistente per richiedere il file HTML (oggetto base) della pagina web. Ricevuto il file HTML chiude la connessione, “legge” il file HTML, scopre che la pagina web è costituita da 10 altri oggetti ed apre quindi 10 connessioni TCP non persistenti per scaricare ciascuno dei 10 oggetti. Il tempo richiesto per ottenere il file HTML (oggetto base) è:

$$T_{open+HTML} = 2 \frac{l}{C} + \frac{l}{C} + \frac{L}{C}$$

dove il primo termine rappresenta il tempo per aprire la connessione TCP, il secondo termine il tempo per inviare al server il messaggio di GET HTTP ed il terzo termine il tempo per scaricare il file HTML.

Una volta ottenuto il file HTML, il client apre in parallelo 10 connessioni TCP che condividono lo stesso collegamento. Il tempo per ottenere ciascuno dei 10 oggetti attraverso le connessioni TCP, ciascuna delle quali necessita di una fase di apertura, è

$$T_{open+oggetti} = 2 \frac{l}{r} + \frac{l}{r} + \frac{L}{r}$$

Il tempo complessivo sarà quindi dato da:

$$T_{totale} = T_{open+HTML} + T_{open+oggetti}$$

Sostituendo i valori di l , L , C e r nelle formule, si ottiene: $T_{open+HTML} = 2 [ms] + 1[ms] + 2000 [ms] = 2.003[s]$ e $T_{open+oggetti} = 20[ms] + 10[ms] + 20000[ms] = 20.03 [s]$ e quindi $T_{totale} = 22.033 [s]$

b) Il *client* apre una connessione persistente per scaricare in serie tutti gli 11 oggetti che comprendono la pagina *web*:

$$T_{totale} = 2 \frac{l}{C} + 11 \frac{l+L}{C}$$

dove il primo termine rappresenta il tempo per aprire la connessione TCP, ed il secondo termine rappresenta il tempo per inviare la richiesta GET HTTP e ricevere il singolo oggetto per tutti gli 11 oggetti della pagina web. Sostituendo i valori nella formula si ottiene: $T_{totale} = 22.013 [s]$.

c) Il client apre 11 connessioni TCP in serie in modalità non persistente. Si ha quindi:

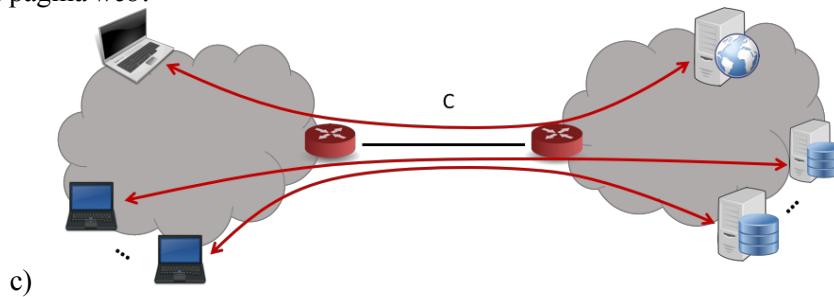
$$T_{totale} = 11 \left(2 \frac{l}{C} + \frac{l+L}{C} \right)$$

da cui si ottiene: $T_{totale} = 22.033 [s]$.

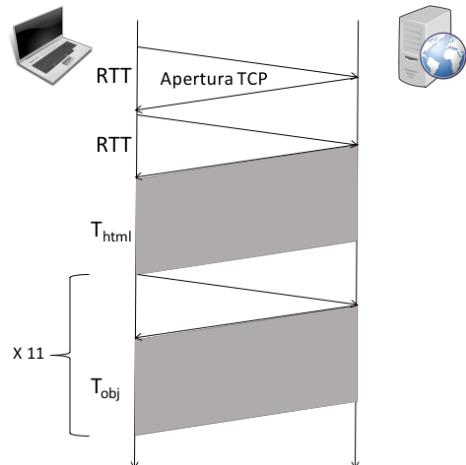
Esercizio 3.2

Un *client* HTTP richiede ad un *server* HTTP una pagina *web* costituita da un oggetto base (file HTML) e 11 altri oggetti. Ogni oggetto ha una dimensione $L=50 \text{ kB}$. Il collegamento (collo di bottiglia) tra *client* e *server* HTTP è in grado di trasferire informazione ad una velocità di $C=1 \text{ Mbit/s}$ in entrambe le direzioni. I messaggi di controllo usati per aprire una connessione TCP tra *client* e *server* ed il messaggio di richiesta della pagina hanno lunghezza *trascubabile*. Il ritardo di andata e ritorno (RTT) sia pari a 150 ms. Il collegamento è condiviso da altri 9 trasferimenti file di lunga durata verso altri server. Assumendo che tutti i trasferimenti condividano in modo equo la capacità del collegamento ottenendo un rate medio pari a C/n , dove n è il numero di trasferimenti paralleli, calcolare in tempo totale per ricevere interamente la pagina *web* richiesta nei due casi seguenti:

- il *client* HTTP apre un'unica connessione TCP persistente per scaricare tutti gli oggetti della pagina *web*.
- il *client* HTTP apre in parallelo in modalità non persistente tutte le connessioni TCP necessarie per scaricare la pagina *web*.



- Nel primo caso si ha il seguente schema temporale:



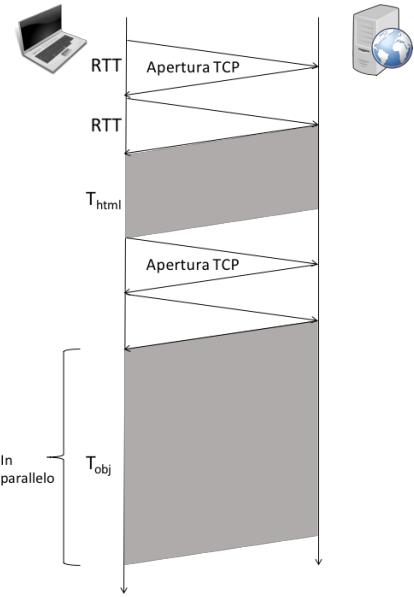
Il rate effettivo è dato dalla suddivisione delle capacità del collegamento tra i 9 flussi interferenti e il flusso relativo allo scambio HTTP, dunque 10 flussi.

$$r_a = \frac{C}{10} = 0.1 \text{ [Mb/s]}$$

$$T_{html} = T_{obj} = \frac{L}{r_a} = \frac{50 \cdot 8 \text{ [kbit]}}{100 \text{ [kb/s]}} = 4 \text{ [s]}$$

$$T_{tot} = 2RTT + T_{html} + 11(RTT + T_{obj}) = 49.95 [s]$$

b) Nel secondo caso si ha il seguente schema temporale:



Il tempo di trasferimento della pagina HTML è uguale al precedente perché la situazione rimane invariata

$$T_{html} = 4 [s]$$

Il rate effettivo con cui vengono scambiati gli oggetti invece è diverso, poiché vengono aperte 11 connessioni in parallelo. Esso è dato dalla suddivisione delle capacità del collegamento tra i 9 flussi interferenti e il flusso delle 11 connessioni, dunque 20 flussi.

$$r_b = \frac{C}{20} = 0.05 [Mb/s]$$

$$T_{obj} = \frac{L}{r_b} = \frac{50 \cdot 8 [kbit]}{50 [kbps]} = 8 [s]$$

Quindi il tempo totale di trasferimento è

$$T_{tot} = 2RTT + T_{html} + 2RTT + T_{obj} = 12.6 [s]$$

Esercizio 3.3

Un'azienda possiede una rete locale con un *proxy HTTP* con *cache* locale (vedi Figura 1). I client sono collegati al *proxy HTTP* tramite collegamenti dedicati con capacità $C=1$ [Gbit/s]. La probabilità che il contenuto (pagina web) richiesto dal generico client sia presente nella cache del *proxy* locale (cache hit rate) sia $P=0.4$ (dualmente, la probabilità che la pagina web richiesta non sia presente in cache del *proxy* locale sia $Q=0.6$). Trovare il ritardo medio sperimentato dal generico client da quando invia richiesta HTTP per una pagina web a quando ottiene la pagina web richiesta. Assumere che:

- i messaggi di richiesta HTTP siano di 100 [byte],
- la pagina web richiesta sia di 100 [kbyte],
- il *proxy HTTP* abbia un canale di comunicazione con capacità equivalente di $c=100$ [Mbit/s] verso il server web che ospita le pagine *web* richieste,
- il tempo di apertura delle connessioni TCP tra client e proxy e tra proxy e web server sia trascurabile,
- i ritardi di propagazione siano trascurabili.

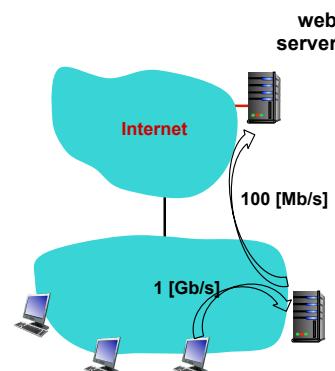


Figura 1 Topologia di riferimento

Nel caso in cui la pagina web richiesta sia disponibile presso il *proxy* locale, il tempo per ottenere la stessa è uguale al tempo necessario al client per inviare il messaggio di richiesta HTTP al *proxy* e per ricevere il messaggio di risposta HTTP dal *proxy* contenente la pagina web stessa. In formule:

$$T_1 = \frac{l}{C} + \frac{L}{C}$$

Sostituendo i valori ai parametri, si ottiene: $T_1 = 0.8008$ [ms]. Si noti che non sono stati considerati nel computo del tempo di ottenimento della pagina web i tempi di processing della richiesta HTTP da parte del *proxy*.

Nel caso in cui la pagina web richiesta non sia disponibile presso il *proxy* locale, il tempo per ottenere la stessa è uguale al tempo necessario al client per inviare il messaggio di richiesta HTTP al *proxy*, il tempo necessario per il *proxy* per inviare il messaggio HTTP di richiesta al web server, il tempo che il web server impiega per inviare il messaggio HTTP di risposta contenente la pagina web al *proxy* ed il tempo che il *proxy* impiega per inviare la pagina web al client. In formule:

$$T_2 = \frac{l}{C} + \frac{l}{c} + \frac{L}{c} + \frac{L}{C}$$

dove i parametri sono gli stessi del caso precedente, fatta eccezione per c che è la capacità del collegamento tra il *proxy* ed il web server. Sostituendo i valori ai parametri, si ottiene: $T_2 = 8.08$ [ms]. Si noti che non sono stati considerati nel computo del tempo di ottenimento della pagina web i tempi di processing della richiesta HTTP da parte del *proxy* e del server web.

Il tempo medio complessivo per ottenere una pagina web è quindi:

$$T_{medio} = PT_1 + QT_2$$

uguale a $T_{medio} = 5.6[ms]$

La Figura 2 di seguito riporta l'andamento di T_{medio} al variare del parametro di cache hit rate, P . Come era lecito aspettarsi, il tempo medio per ottenere una pagina web diminuisce linearmente al crescere della probabilità che la pagina web sia disponibile “più vicina” all’utente finale.

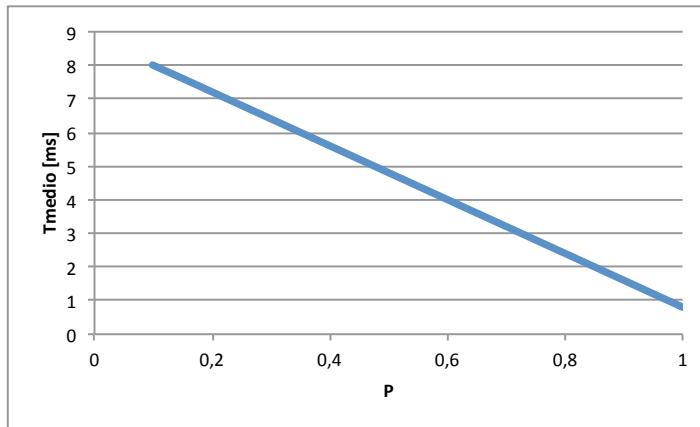


Figura 2 Andamento di T_{medio} al variare della cache hit rate, P .

Esercizio 3.4

Si consideri la trasmissione di un file video di dimensione $D=600$ MByte ad $N=1000$ utenti. Il video è originariamente distribuito da un Server con capacità di upload pari a $C_s=1$ Gbit/s, ciascuna dedicata ad uno degli N utenti. I vari utenti sono dotati di interfacce di ricezione/trasmissione con capacità di download e di upload rispettivamente pari a $C_d=20$ Mbit/s e $C_u=2$ Mbit/s.

Si calcoli il tempo necessario per trasferire il video a tutti gli N utenti nei seguenti casi:

- la trasmissione del video avviene secondo l'architettura *Client-Server* e **in serie** verso gli utenti, utilizzando tutta la capacità disponibile per la trasmissione del video;
- la trasmissione del video avviene secondo l'architettura *Client-Server* e **in parallelo** verso gli utenti (si ipotizzi che la capacità di upload del Server, C_s , sia distribuita *equamente* su N interfacce di trasmissione distinte, ciascuna dedicata ad un utente);
- la trasmissione del video avviene secondo l'architettura Peer-to-Peer; si ipotizzi che all'istante $t=0$ il video sia disponibile soltanto ad uno degli N utenti e che il video può essere messo a disposizione da ciascun peer agli altri peer solo dopo averlo ricevuto per intero;

-
- Il server trasmette il video verso gli N utenti, servendoli uno dopo l'altro, ed utilizzando l'intera capacità C_s per il trasferimento del file. Tuttavia, la massima capacità che può essere sfruttata da ciascun utente coincide con la propria capacità di download, C_d . Il tempo totale necessario per servire tutti gli utenti è quindi dato da:

$$T_{tot(S,serie)} = N \frac{D}{C_d} = 1000 \frac{600 * 8 [Mbit]}{20 [Mb/s]} = 240000 s$$

- Poiché la capacità del server è suddivisa equamente tra gli N utenti, la trasmissione del video verso ciascuno di essi potrà avvenire ad un bit rate pari a

$$C_{s,parallelo} = \min\left(C_d, \frac{C_s}{N}\right) = \min(20 Mb/s, 1 Mb/s) = 1 Mb/s$$

Infatti, bisogna tener conto che la massima capacità utilizzabile dipende sia dalla velocità delle interfacce di upload del server che dalla velocità di download dei client.

Pertanto, il tempo totale necessario per servire tutti gli utenti è dato da:

$$T_{tot(S,parallelo)} = \frac{D}{C_{s,parallelo}} = \frac{600 * 8 [Mbit]}{1 [Mb/s]} = 4800 s$$

- Per raggiungere tutti gli N utenti, sono necessari $\lceil \log_2 N \rceil$ "passi" (al 1° passo l'unico utente in possesso del video lo fornisce ad un solo altro utente, al 2° passo, questi due utenti forniscono il video ad altri due utenti, etc.). Ciascuno di questi passi, impiega un tempo di trasferimento pari a

$$T_{1,p2p} = \frac{D}{C_u} = \frac{600 * 8 [Mbit]}{2 [Mb/s]} = 2400 s$$

Pertanto, il tempo totale necessario per servire tutti gli utenti è dato da:

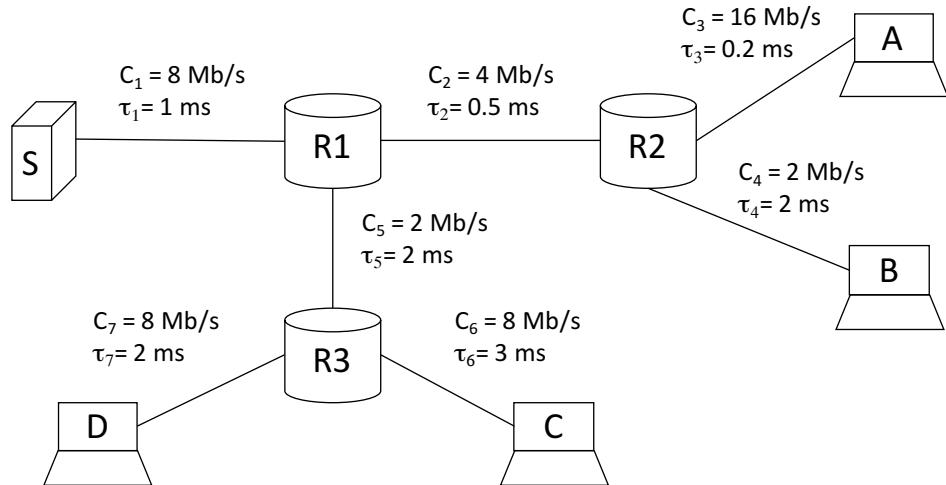
$$T_{tot(p2p)} = \lceil \log_2 N \rceil * T_{1,p2p} = 24000 s$$

Esercizio 3.5

Si assuma che un client http in A voglia scaricare una pagina web contenuta del server in S. La capacità del collegamento tra S ed A è limitata dal collegamento che costituisce il collo di bottiglia della rete, e che è condiviso con due flussi interferenti di lunga durata (file transfer) tra D e B e tra C e B. La pagina web è composta da un documento base (html) di 100 byte e da 8 immagini di 1 Mbyte. Si calcoli il tempo di scaricamento della pagina web:

- a) nel caso di connessione http persistente per il documento base e le immagini, e
- b) nel caso di connessione non persistente (prima il documento html e poi le 8 immagini con connessioni in parallelo).

N.B. Per il calcolo delle velocità di trasmissione utilizzabili dalle varie connessioni TCP, si consideri la capacità del “collo di bottiglia” del collegamento, assumendo il principio di **condivisione equa delle risorse**.



- a) Nel caso di una connessione persistente la condivisione equa tra i due flussi interferenti e il flusso http tra S e A porta quest'ultimo ad un rate $R=2 \text{ [Mbit/s]}$ sul link collo di bottiglia R1-R2. Infatti, i due flussi interferenti sono al loro volta limitati ad un totale di 2 [Mbit/s] attraversando il link R3-R1 (collo di bottiglia per i flussi interferenti), quindi sul link R1-R2 rimangono $4 \text{ [Mbit/s]} - 2 \text{ [Mbit/s]} = 2 \text{ [Mbit/s]}$ per il flusso tra S ed A.

$$RTT = 2(\tau_1 + \tau_2 + \tau_3) = 3.4 \text{ [ms]}$$

$$T_{html} = \frac{100*8 \text{ [bit]}}{2 \text{ [Mb/s]}} = 0.4 \text{ [ms]}$$

$$T_{obj} = \frac{8 * 10^6 \text{ [bit]}}{2 \text{ [Mbit/s]}} = 4 \text{ [s]}$$

$$\begin{aligned} T_{tot} &= T_{open} + T_{get} + T_{html} + 8(T_{get} + T_{obj}) \\ &= RTT + RTT + T_{html} + 8(RTT + T_{obj}) = 32.0344 \text{ [s]} \end{aligned}$$

- b) Nel caso di connessione non persistente e oggetti in parallelo, la parte del file html non cambia, per gli oggetti invece il rate di condivisione equa risulta 0.4 Mbit/s (4 Mbit/s condivisi da 10 flussi: 2 Mbit/s per ogni flusso).

interferenti + 8 immagini in parallelo). In questo caso, il link R1-R2 è il collo di bottiglia anche per i flussi interferenti.

$$T_{obj} = \frac{8 * 10^6 \text{ [bit]}}{0.4 \text{ [Mb/s]}} = 20 \text{ [s]}$$

$$\begin{aligned} T_{tot} &= T_{open} + T_{get} + T_{html} + T_{open} + T_{get} + T_{obj} \\ &= 2RTT + T_{html} + 2RTT + T_{obj} = 20.014 \text{ [s]} \end{aligned}$$

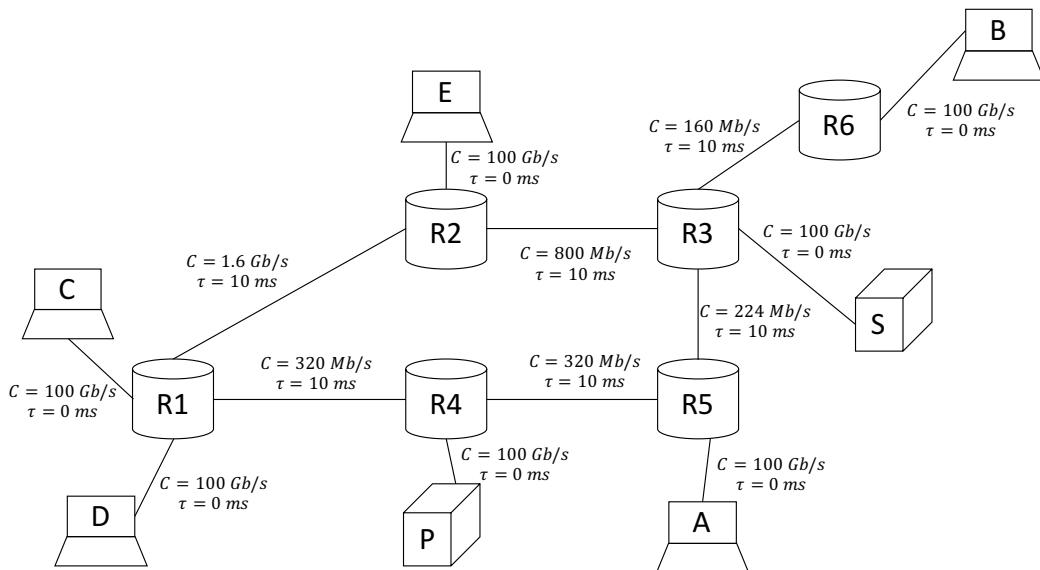
Esercizio 3.6

Nella rete in figura sono illustrati i client A, B, C, D, E, il server http S, il proxy http P ed i router R1, R2, R3, R4, R5, R6. Accanto ad ogni link (bidirezionale e simmetrico) sono indicati la capacità e il ritardo di propagazione. Sul client C è in esecuzione un client http che deve scaricare dal server S una pagina web composta da 1 oggetto HTML di 500 [kbyte] e 10 oggetti di 2 [Mbyte] ciascuno, richiamati dalla pagina HTML. Inoltre, nella rete esistono dei flussi interferenti di lunga durata: 4 flussi dal client A al client B, 4 flussi dal client D al client E. Si assuma ogni flusso in rete instradato secondo una politica che minimizza il numero di hop.

Si chiede di calcolare il tempo di trasferimento della pagina web nei tre seguenti casi:

- Il client C è configurato per usare il proxy P, le connessioni http sono non-persistenti e sia il client che il proxy aprono tutte le connessioni in parallelo possibili. Tutti gli oggetti della pagina si trovano nella cache del proxy.
- Il client C è configurato per non usare proxy e può aprire una connessione al massimo, ma in maniera persistente.
- Il client C è configurato per usare il proxy P, le connessioni http sono non-persistenti ed sia il client che il proxy aprono tutte le connessioni in parallelo possibili. Nessun oggetto della pagina si trova nella cache del proxy.

N.B. Per il calcolo delle velocità di trasmissione utilizzabili dalle varie connessioni TCP, si consideri la capacità del “collo di bottiglia” del collegamento, assumendo il principio di **condivisione equa delle risorse**.



- a) Tutti gli oggetti sono sul proxy, viene percorso il link R1-R4. Apertura della connessione e messaggi di GET avvengono solo tra client e proxy. Non ci sono flussi interferenti sul percorso, i 10 flussi in parallelo per gli oggetti condividono l'intera capacità di R1-R4, mentre il trasferimento iniziale della pagina html, essendo un unico flusso, può utilizzare tutta la capacità del link.

$$T = T_{open} + T_{get} + T_{HTML} + T_{open} + T_{get} + T_{ogg,10}$$

$$T_{open} = 2\tau = 20 \text{ [ms]}$$

$$T_{get} = 2\tau = 20 \text{ [ms]}$$

$$T_{HTML} = \frac{500*8 \text{ [kbit]}}{320 \text{ [Mb/s]}} = 12.5 \text{ [ms]}$$

$$T_{ogg,10} = \frac{2000*8 [kbit]}{\frac{320}{10} [Mb/s]} = 500 [ms]$$

Dunque $T = 592.5 [ms]$

b) Il proxy non viene usato, la connessione tra C e S è unica e persistente. Apertura della connessione e messaggi di GET avvengono solo tra client e server. La connessione usa i link R1-R2 e R2-R3. Su R1-R2 ci sono 4 flussi interferenti, oltre all'unico flusso per il trasferimento http.

$$T = T_{open} + T_{get} + T_{HTML} + 10(T_{get} + T_{ogg})$$

$$T_{open} = 4\tau = 40 [ms]$$

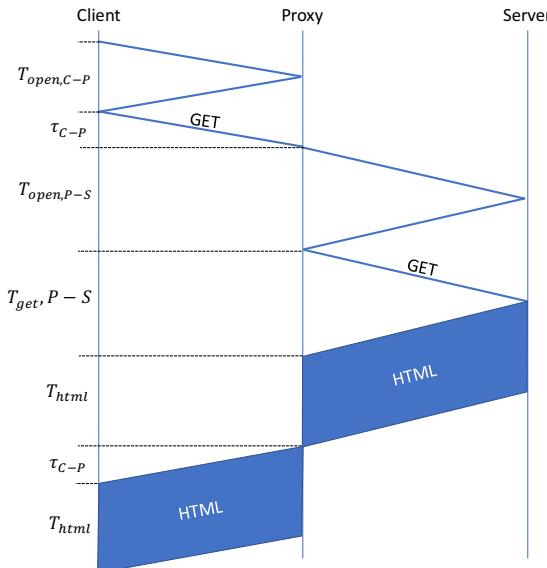
$$T_{get} = 4\tau = 40 [ms]$$

$$T_{HTML} = \frac{500*8 [kbit]}{\frac{1600}{5} [Mb/s]} = 12.5 [ms]$$

$$T_{ogg,10} = \frac{2000*8 [kbit]}{\frac{1600}{5} [Mb/s]} = 50 [ms]$$

Dunque, $T = 992.5 [ms]$

c) Nessun oggetto è nel proxy, per la sessione client-proxy viene percorso il link R1-R4, per la sessione proxy-server vengono percorsi i link R4-R5 e R5-R3. Apertura della connessione e messaggi di GET avvengono sia tra client e proxy che tra proxy e server.



$$T = T_{open,C-P} + \tau_{C-P} + T_{open,P-S} + T_{get,P-S} + T_{HTML,S-P} + \tau_{C-P} + T_{HTML,P-C} + T_{open,C-P} + \tau_{C-P} + T_{open,S-P} + T_{get,P-S} + T_{ogg,10,S-P} + \tau_{C-P} + T_{ogg,10,P-C}$$

$$T_{open,C-P} = 2\tau = 20ms, T_{open,P-S} = 4\tau = 40 [ms]$$

$$\tau_{C-P} = \tau = 10 [ms], \tau_{P-S} = 2\tau = 20 [ms]$$

$$T_{HTML,P-C} = \frac{500*8 [kbit]}{320 [Mb/s]} = 12.5 [ms]$$

$$T_{ogg,10,P-C} = \frac{2000*8 kbit}{\frac{320}{10} [Mb/s]} = 500 [ms]$$

Abbiamo 1 flusso lungo R4-R5 (flusso http proxy-server), 5 flussi lungo R5-R3 (flusso http proxy-server + 4 flussi A-B), 4 flussi lungo R3-R6 (4 flussi A-B) che corrispondono, rispettivamente, ai rate per flusso di 320 [Mbit/s], 44.8 [Mbit/s] e 40 [Mbit/s]. Quindi i 4 flussi da A a B sono limitati a 40 [Mbit/s] ciascuno e il flusso da P a S riceverà una capacità pari a $224 - 4 \cdot 40 = 224 - 160 = 64 [Mb/s]$ vincolato dal link collo di bottiglia R5-R3.

$$T_{HTML,S-P} = \frac{500 * 8 [kbit]}{64 [Mb/s]} = 62.5 [ms]$$

Abbiamo 10 flussi lungo R4-R5 (10 flussi http in parallelo), 14 flussi lungo R5-R3 (10 flussi http in parallelo + 4 flussi A-B), 4 flussi lungo R3-R6 (4 flussi A-B) che corrispondono, rispettivamente, ai rate per flusso di 32 [Mbit/s], 16 [Mbit/s] e 40 [Mbit/s]. Quindi i 4 flussi da A a B e i 10 flussi da P a S riceveranno ciascuno una capacità di 16 [Mbit/s] vincolati dal link R5-R3.

$$T_{ogg,10,S-P} = \frac{2000 * 8 [kbit]}{16 [Mb/s]} = 1000 [ms]$$

Dunque, $T = 1815 [ms]$

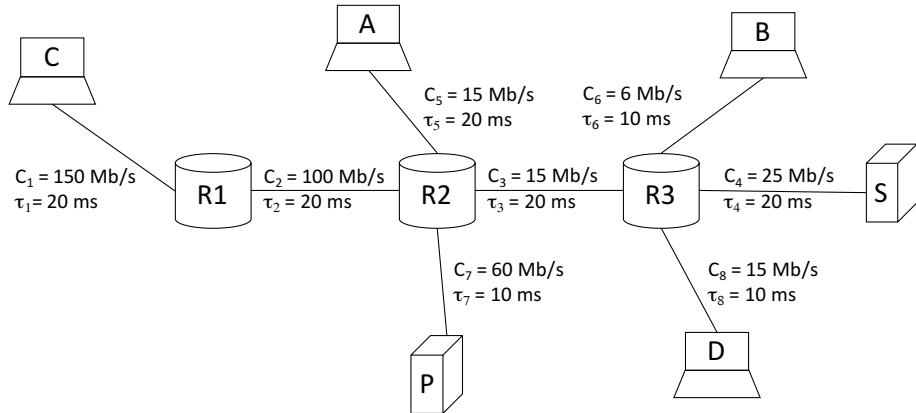
Esercizio 3.7

Nella rete in figura sono rappresentati 4 client (A, B, C e D), 3 router (R1, R2 e R3), un server HTTP (S) e un proxy HTTP (P). Il client C vuole trasferire un documento formato da una pagina HTML di dimensione $L_{HTML} = 30$ [kbyte] che richiama 11 oggetti di dimensione $L_{OGG} = 75$ [kbyte]. Nella rete sono presenti anche 4 flussi interferenti di lunga durata: 2 da A a D, 2 da A a B. Si supponga che: i) i messaggi di apertura della connessione e di richiesta HTTP siano di lunghezza trascurabile, ii) la connessione TCP tra proxy HTTP e server HTTP sia sempre aperta e non occorra mandare una richiesta di apertura.

Si calcoli:

- il tempo di trasferimento del documento nel caso in cui il client C senza proxy configurato apra in parallelo in modalità non-persistente tutte le connessioni TCP necessarie
- il tempo di trasferimento del documento nel caso in cui il client C con proxy configurato apra un'unica connessione TCP persistente per scaricare tutti gli oggetti, ipotizzando che la pagina HTML e solo i primi 6 degli 11 oggetti siano presenti nella cache del proxy
- nel caso b), il numero minimo di oggetti che occorre trovare nella cache del proxy per avere un tempo di trasferimento minore di 2 s.

N.B. Per il calcolo delle velocità di trasmissione utilizzabili dalle varie connessioni TCP, si consideri la capacità del “collo di bottiglia” del collegamento, assumendo il principio di **condivisione equa delle risorse**.



Punto a

Il client manda/riceve messaggi HTTP direttamente al/dal server.

Durante il trasferimento della pagina HTML (1 flusso tra C e S) sul link R2-R3 ci sono 5 flussi che condividono 15 [Mbit/s]. Il link è collo di bottiglia per tutti i flussi che ottengono 3 [Mbit/s] ciascuno. Quindi la capacità vista dal trasferimento C-S sarà $C_{eff-5} = 3$ [Mb/s].

$$\text{Il tempo di trasferimento della pagina HTML è: } T_{HTML} = \frac{L_{HTML}}{C_{eff-5}} = \frac{30*8 \text{ [kbit]}}{3 \text{ [Mb/s]}} = 80 \text{ [ms]}$$

Durante il trasferimento degli oggetti (11 flussi in parallelo tra C e D) sul link R2-R3 ci sono 15 flussi che condividono 15 [Mbit/s]. Il link è collo di bottiglia per tutti i flussi che ottengono 1 [Mbit/s] ciascuno. Quindi la capacità vista dal trasferimento C-S sarà $C_{eff-15} = 1$ [Mb/s].

$$\text{Il tempo di trasferimento di un oggetto è: } T_{OGG} = \frac{L_{OGG}}{C_{eff-15}} = \frac{75*8 \text{ [kbit]}}{1 \text{ [Mb/s]}} = 600 \text{ [ms]}$$

$$\text{Il RTT tra C e S è: } RTT_{C-S} = 2(\tau_1 + \tau_2 + \tau_3 + \tau_4) = 2 * 80 \text{ [ms]} = 160 \text{ [ms]}$$

Il tempo totale di trasferimento sarà:

$$T_{tot} = RTT + RTT + T_{HTML} + RTT + RTT + T_{OGG} = 4RTT + T_{HTML} + T_{OGG} = 1320 [ms]$$

Punto b

Il client manda/riceve messaggi HTTP al/dal proxy. Per i primi 6, riceve l'oggetto direttamente dal proxy, per gli ultimi 5, il proxy scarica l'oggetto dal server e poi lo invia al client.

La capacità del trasferimento tra client e proxy è determinata dal link P-R2 e dunque pari a $C_{eff} = 60 [Mb/s]$.

Il tempo di trasferimento della pagina HTML (diretta al proxy) è: $T_{HTML} = \frac{L_{HTML}}{C_{eff}} = \frac{30*8 [kbit]}{60 [Mb/s]} = 4 [ms]$

Il tempo di trasferimento di un oggetto (in cache) è: $T_{OGG-cache} = \frac{L_{OGG}}{C_{eff}} = \frac{75*8 k[bit]}{60 [Mb/s]} = 10 [ms]$.

La capacità del trasferimento tra proxy e server è determinata dal link R2-R3, su cui transitano 5 flussi, dunque pari a $C_{eff-5} = 3 [Mb/s]$.

Il tempo di trasferimento di un oggetto (nel server) è: $T_{OGG-nocache} = \frac{L_{OGG}}{C_{eff-5}} = \frac{75*8 [kbit]}{3 [Mb/s]} = 200 [ms]$.

Il RTT tra C e P è: $RTT_{C-P} = 2(\tau_1 + \tau_2 + \tau_7) = 2 * 50 [ms] = 100 [ms]$, mentre il RTT tra P e S è: $RTT_{P-S} = 2(\tau_7 + \tau_3 + \tau_4) = 2 * 50 [ms] = 100 [ms]$

Il tempo totale di trasferimento è:

$$\begin{aligned} T_{tot} &= RTT_{C-P} + RTT_{C-P} + T_{HTML} + 6(RTT_{C-P} + T_{OGG-cache}) \\ &\quad + 5(RTT_{P-S} + T_{OGG-nocache} + RTT_{C-P} + T_{OGG-cache}) \\ &= 100 [ms] + 100 [ms] + 4 [ms] + 6(100 [ms] + 10 [ms]) \\ &\quad + 5(100 [ms] + 200 [ms] + 100 [ms] + 10 [ms]) \\ &= 204 [ms] + 660 [ms] + 2050 [ms] = 2914 [ms] \end{aligned}$$

Punto c

Dall'ultima espressione del punto b) si può scrivere

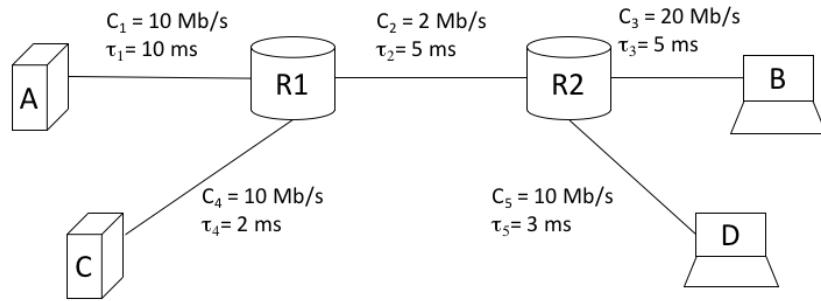
$$T_{tot} = 204 [ms] + x(110 [ms]) + (11 - x)(410 [ms])$$

dove x è il numero di oggetti trovati in cache.

Quindi da $T_{tot} = 204 + x(110) + (11 - x)(410) < 2000$, si ottiene $x > 9.046$. Occorre trovare in cache almeno 10 oggetti.

Esercizio 3.8

Si assuma A sia un server http e B un client http. Occorre trasferire un documento base html di $L_{html} = 100$ [kbyte] base e 9 immagini di $L_{obj} = 5$ [Mbyte] in presenza di 1 flusso interferente tra C e D. Si assuma che i messaggi di apertura della connessione e di richiesta HTTP siano di lunghezza trascurabile. Si calcoli il tempo necessario assumendo un ritmo medio di trasmissione (R_{html} e R_{obj}) pari al valore di condivisione equa delle risorse (capacità del link 2 diviso il numero di flussi che lo attraversano) nel caso di connessione http persistente e non persistente (con trasmissione in parallelo delle immagini).



$$RTT = 2(\tau_1 + \tau_2 + \tau_3) = 40 \text{ ms}$$

Persistente:

$$R_{html} = R_{obj} = \frac{C_2}{2} = 1 \text{ Mbps} \text{ (Condivisione equa del flusso http e del flusso interferente)}$$

$$\begin{aligned} T_{tot} &= T_{open} + (T_{GET} + T_{html}) + 9(T_{GET} + T_{obj}) \\ &= RTT + \left(RTT + \frac{L_{html}}{R_{html}} \right) + 9 \left(RTT + \frac{L_{obj}}{R_{obj}} \right) \end{aligned}$$

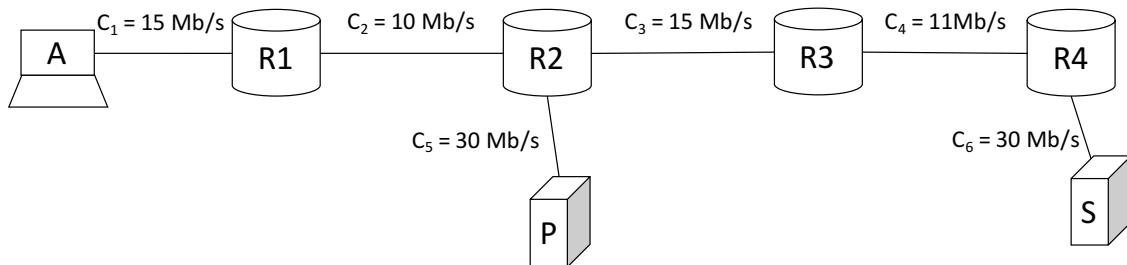
Non-persistente:

$$R_{html} = \frac{C_2}{2} = 1 \text{ Mbps} \text{ (Condivisione equa del flusso http e del flusso interferente)}$$

$$R_{obj} = \frac{C_2}{10} = 0.2 \text{ Mbps} \text{ (Condivisione equa dei 9 flussi http e del flusso interferente)}$$

$$\begin{aligned} T_{tot} &= (T_{open} + T_{GET} + T_{html}) + (T_{open} + T_{GET} + T_{obj}) \\ &= \left(RTT + RTT + \frac{L_{html}}{R_{html}} \right) + \left(RTT + RTT + \frac{L_{obj}}{R_{obj}} \right) \end{aligned}$$

Esercizio 3.9



Nella rete in figura sono rappresentati 4 router (R1, R2, R3 e R4), un client A, un HTTP proxy P e un HTTP server S. Accanto ad ogni collegamento è indicata la propria capacità, mentre il tempo di propagazione è pari a 10 [ms] su ciascun collegamento.

Il client vuole scaricare dal server un sito web composto da 1 pagina HTML di dimensione $L_{HTML}=80$ [kbyte] e 6 oggetti JPEG richiamati nella pagina HTML di dimensione $L_{OGG}=500$ [kbyte]. Nella rete sono presenti flussi interferenti di lunga durata: 4 tra R1 e R2, 10 tra R3 e R4.

Si chiede di calcolare il tempo di trasferimento del sito web a livello applicativo nei seguenti casi:

- Il client A non ha proxy configurato, apre connessioni non-persistent in parallelo (quando possibile e nel massimo numero possibile)
- Il client A utilizza il proxy P, apre al massimo una connessione alla volta in modalità non-persistent. Solo la pagina HTML ed i primi 2 oggetti JPEG sono presenti nella cache del proxy.

N.B. Per il calcolo delle velocità di trasmissione utilizzabili dalle varie connessioni TCP, si consideri la capacità del “collo di bottiglia” del collegamento, assumendo il principio di *condivisione equa delle risorse*.

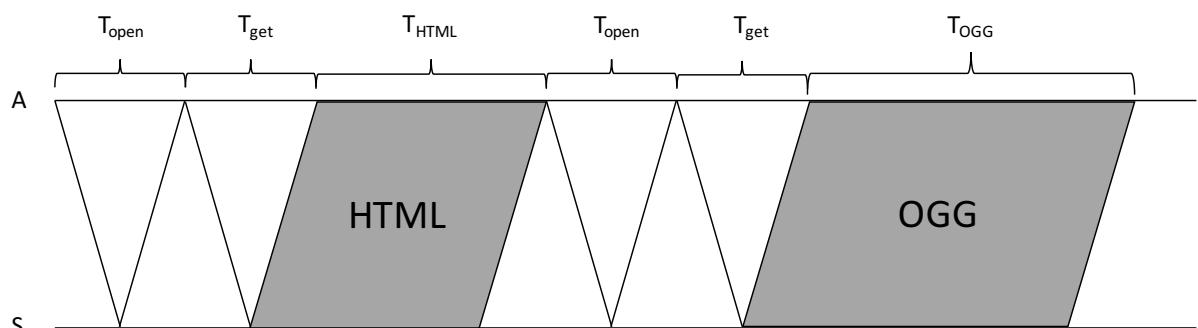
Punto a)

Il client C colloquia direttamente con il server S.

Per la pagina HTML, i collegamenti attraversati hanno le seguenti capacità effettive: A-R1 15 Mbit/s, R1-R2 2 Mbit/s (4+1 flussi), R2-R3 15Mbit/s, R3-R4 1Mbit/s (10+1 flussi). Dunque il trasferimento è governato dal collo di bottiglia R3-R4 a 1 Mbit/s. Il tempo di trasferimento della pagina HTML è pari a $T_{HTML} = (8 \cdot 80 \text{ [kbit]}) / 1 \text{ [Mb/s]} = 640 \text{ ms}$

Per gli oggetti JPEG, i collegamenti attraversati sono i medesimi, dunque il collo di bottiglia sarà il link R3-R4 con una capacità effettiva di 687.5 [kbit/s] (10+6 flussi). Il tempo di trasferimento di ogni oggetto JPEG è pari a $T_{OGG} = 8 \cdot 500 \text{ [kbit]} / 687.5 \text{ [kbit/s]} = 5.82 \text{ s}$.

Il tempo totale di trasferimento è pari a $T = T_{open} + T_{get} + T_{HTML} + T_{open} + T_{get} + T_{OGG}$ in accordo con la seguente figura. Per i tempi di propagazione, $T_{open} = T_{get} = 100 \text{ [ms]}$



$$T=6.86s$$

Punto b)

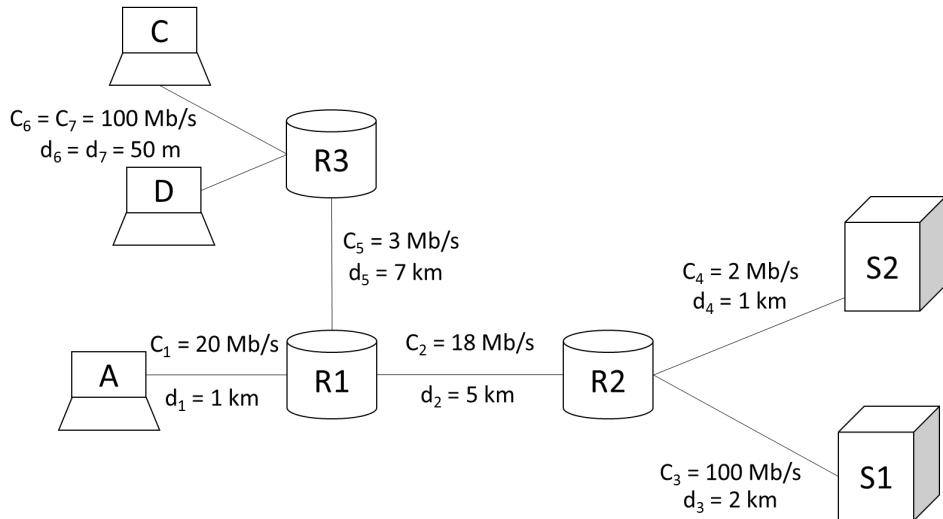
Il client C colloquia con il proxy P.

Il colloquio client-proxy è definito dal collo di bottiglia R1-R2 con una capacità di 2 Mbit/s (4+1 flussi), mentre il colloquio proxy-server ha R3-R4 come collo di bottiglia, dunque una capacità di 1 Mbit/s (10+1 flussi). I tempi di trasferimento sono $T_{HTML}^{CP} = \frac{640[kbit]}{2[Mb/s]} = 320[ms]$, $T_{OGG}^{CP} = \frac{4[Mbit]}{2[MB/s]} = 2[s]$ e $T_{OGG}^{PS} = \frac{4[Mbit]}{1[Mb/s]} = 4[s]$, rispettivamente, per il trasferimento della pagina HTML dal proxy al client, di un oggetto JPEG dal proxy al client e di un oggetto JPEG dal server al proxy.

I tempi di propagazione sono $T_{open}^{CP} = T_{get}^{CP} = 60[ms]$ per la connessione client-proxy e $T_{open}^{PS} = T_{get}^{PS} = 80[ms]$ per la connessione proxy-client.

Il tempo totale di trasferimento è pari a $T = T_{open}^{CP} + T_{get}^{CP} + T_{HTML}^{CP} + 2(T_{open}^{CP} + T_{get}^{CP} + T_{OGG}^{CP}) + 4(T_{open}^{CP} + T_{get}^{CP} + T_{OGG}^{CP} + T_{open}^{PS} + T_{get}^{PS} + T_{OGG}^{PS}) = 29.8 [s]$

Esercizio 3.10



Nella rete in figura sono rappresentati 3 router (R1, R2 e R3), tre client (A, B e C) e due HTTP server (S1 e S2). Accanto ad ogni collegamento (in fibra ottica) è indicata la rispettiva capacità e il ritardo di propagazione.

Il client A vuole scaricare dal server S1 un sito web composto da 1 pagina HTML di dimensione $L_{HTML} = 600 \text{ [byte]}$ e 5 oggetti JPEG richiamati nella pagina HTML di dimensione $L_{OBJ} = 1.3 \text{ [kbyte]}$. Le dimensioni dei segmenti che non trasportano dati siano trascurabili. Nella rete sono presenti flussi interferenti TCP di lunga durata: 2 flussi tra C e S2, 3 flussi tra B e S1.

Si chiede di calcolare il tempo di trasferimento del sito web a livello applicativo nei seguenti casi:

- Il client A apre connessioni non-persistenti in parallelo (quando possibile e nel massimo numero possibile)
- Il client A apre al massimo una connessione alla volta in modalità persistente, in assenza dei flussi interferenti (cioè, i flussi interferenti sono spenti)

N.B. Per il calcolo delle velocità di trasmissione utilizzabili dalle varie connessioni TCP, si consideri la capacità del “collo di bottiglia” del collegamento, assumendo il principio di **condivisione equa delle risorse**.

Punto a)

Calcoliamo la capacità equivalente delle sessioni al livello applicativo, tenendo conto dei flussi interferenti attivi. I link “candidati” ad essere bottleneck sono R3-R1, R1-R2 e R2-S2.

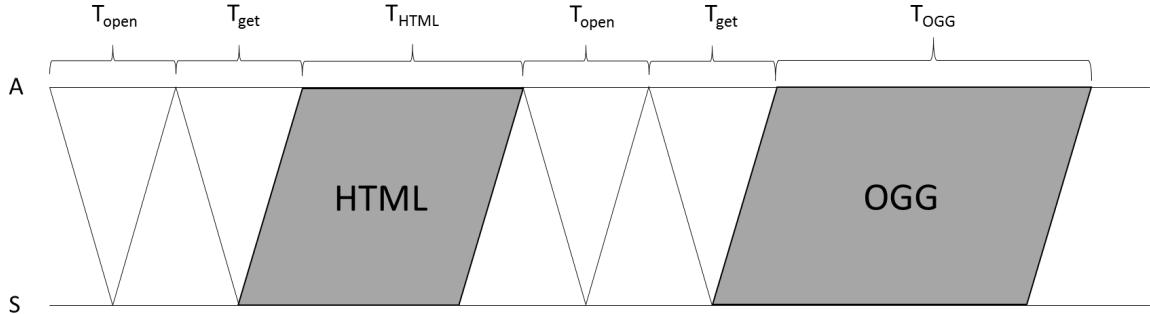
Nel caso vi sia una sola connessione attiva tra A e S1 (trasferimento file HTML), il numero di flussi sui 3 link è rispettivamente 5, 6, 2. Nell’ipotesi di equipartizione, il rate per flusso sarebbe rispettivamente 0.6 [Mbit/s], 3 [Mbit/s], 1 [Mbit/s]. Dunque, il bottleneck è il link R3-R1, determinando un rate uguale per tutti gli interferenti pari a $r_i = 0.6 \text{ Mbit/s}$. Di conseguenza, il rate equivalente per la connessione A-S1 è

$$r_{HTML} = C_2 - 5 \cdot (r_i) = 15 \text{ [Mb/s]}.$$

Nel caso vi siano 5 connessioni attive tra A e S1 (trasferimento oggetti in parallelo), il numero di flussi sui 3 link è rispettivamente 5, 10, 2. Nell’ipotesi di equipartizione, il rate per flusso sarebbe rispettivamente 0.6 [Mbit/s], 1.8 [Mbit/s], 1 [Mbit/s]. Dunque, il bottleneck è ancora il link R3-R1, determinando un rate uguale per tutti gli interferenti pari a $r_i = 0.6 \text{ [Mbit/s]}$. Di conseguenza, il rate equivalente per ciascuna connessione A-S1 (quando ce ne sono 5 in parallelo) è

$$r_{OBJ} = [C_2 - 5 \cdot (r_i)] / 5 = 3 \text{ [Mb/s]}.$$

Lo scambio a livello applicativo avviene secondo il seguente diagramma temporale:



$$T_{open} = T_{get} = 2(\tau_1 + \tau_2 + \tau_3) = 80 \text{ [us]}$$

$$T_{HTML} = L_{HTML} / r_{HTML} = (8 \cdot 600 / 15) \text{ [us]} = 320 \text{ [us]}$$

$$T_{OBJ} = L_{OBJ} / r_{OBJ} = (8 \cdot 1300 / 3) \text{ [us]} = 3466.7 \text{ [us]}$$

$$T = 2 T_{open} + T_{HTML} + 2 T_{open} + T_{OBJ} = 4106.7 \text{ [us]}$$

Punto b)

In assenza delle connessioni interferenti, e con una sola connessione attiva tra A e S1, il bottleneck è il link R1-R2. Quindi, $r_{OBJ} = r_{HTML} = C_2 = 18 \text{ [Mb/s]}$.

$$T_{OBJ} = \frac{L_{OBJ}}{C_2} = (8 \cdot 1300 / 18) \text{ [us]} = 577.8 \text{ [us]}$$

$$T_{HTML} = \frac{L_{HTML}}{C_2} = (8 \cdot 600 / 18) \text{ [us]} = 266.7 \text{ [us]}$$

$$T = 2 T_{open} + T_{HTML} + 5(T_{open} + T_{OBJ}) = 3715.7 \text{ [us]}$$

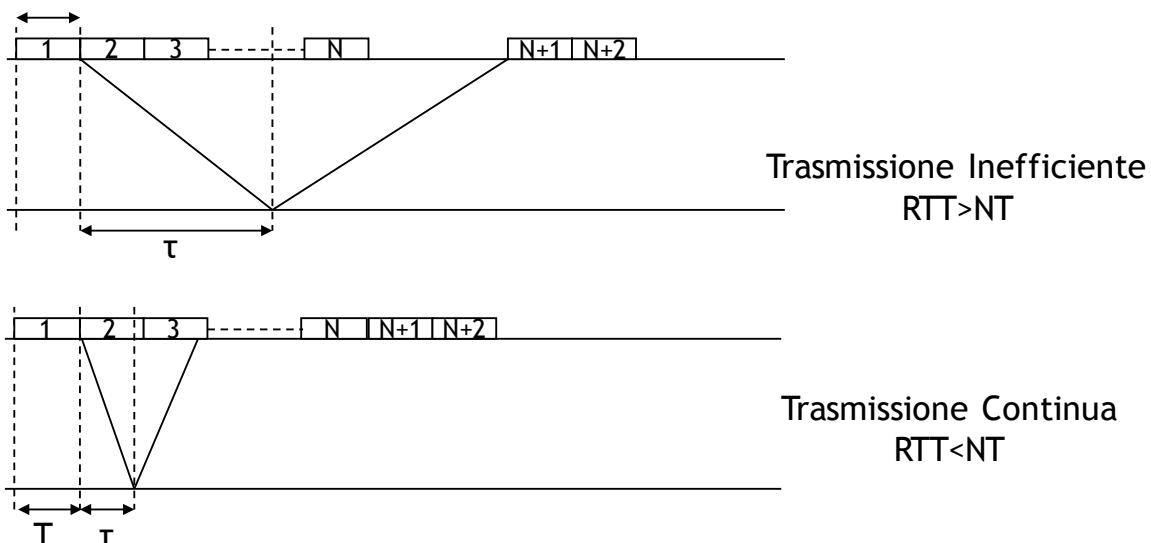
4. Esercizi sul LIVELLO di TRASPORTO – parte I

Esercizio 4.1

Si consideri un canale via satellite della capacità di 1 Mbit/s . Considerando che il tempo di propagazione attraverso un satellite geostazionario richiede 250 ms , si chiede di dimensionare la minima finestra di trasmissione di un protocollo Go-Back-n (con time-out) in modo che sia consentita la massima efficienza temporale del canale quando vengano trasmessi segmenti di 2000 bit in assenza di errori. Si suppongano gli ACK di dimensione trascurabile.

Si calcoli poi la massima efficienza trasmissiva che si avrebbe nel caso in cui il meccanismo ARQ fosse di tipo STOP & WAIT.

L'efficienza del meccanismo Go-BACK-N dipende dal rapporto tra RTT e lunghezza della finestra.



Per avere la massima efficienza, il numero di segmenti N nella finestra deve essere tale che il loro tempo di trasmissione copra il tempo di andata e ritorno del primo segmento. Detti:

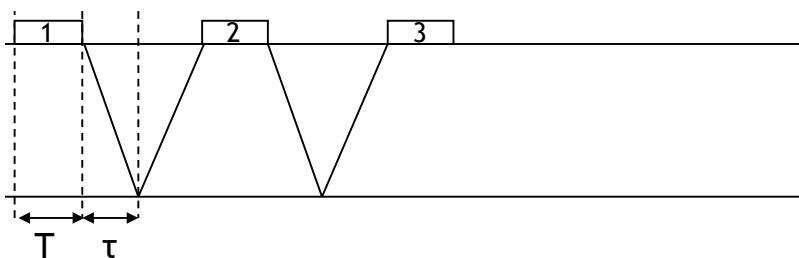
- $T = 2 \text{ [ms]} = 2000 \text{ [bit]} / 1 \text{ [Mb/s]}$, il tempo di trasmissione di un pacchetto
- $t = 250 \text{ [ms]}$, il tempo di propagazione

allora deve essere:

$$NT \geq T + 2t \rightarrow N \geq 1 + 2 \frac{t}{T} = 1 + 2 \cdot 250/2 = 251$$

Per finestre $N \geq 251$, la trasmissione risulta continua, dunque l'efficienza del meccanismo è 1.

Nel caso di meccanismo STOP&WAIT abbiamo:



L'efficienza del meccanismo STOP&WAIT si calcola considerando che il meccanismo trasmette 1 segmento (durata $T = 2 \text{ ms}$) ogni $T + 2t$, dunque l'efficienza: $\eta = T/(T + 2t) = 1/251$.

Esercizio 4.2

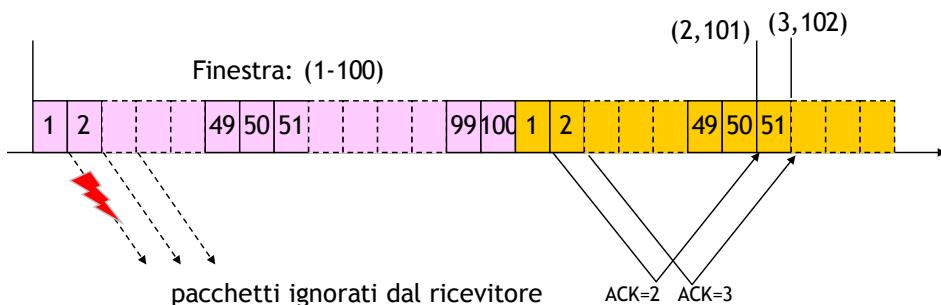
Un sistema GO-BACK-N presenta un ritardo di propagazione pari a 24 volte il tempo di trasmissione di un pacchetto e viene usato per inviare un file di 1000 pacchetti. Si assuma la durata del timeout pari al tempo di trasmissione dell'intera finestra di pacchetti (con inizio al primo bit del primo pacchetto di ogni finestra). Ipotizzando che tutti i pacchetti ricevuti correttamente siano riscontrati (tempo di trasmissione del riscontro = tempo di trasmissione del pacchetto), si calcoli il numero di pacchetti trasmessi inutilmente (errati o corretti ma scartati dal ricevitore) nel caso in cui la finestra sia lunga $W = 100$ quando:

1. si sbaglia il primo pacchetto del file
2. si sbagliano il primo e il 100-esimo pacchetto del file
3. si sbaglia l'ACK del primo pacchetto del file
4. si sbaglia l'ACK del primo e del 100-esimo pacchetto del file

Soluzione

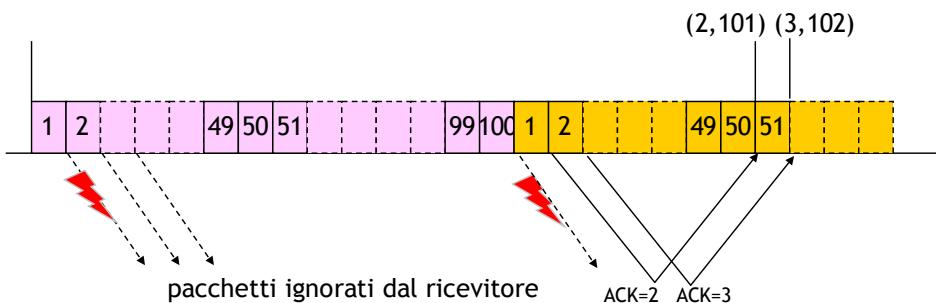
Tempo di ritorno dell'ACK, $RTT = T + 24T + T + 24T = 50T$

Caso 1



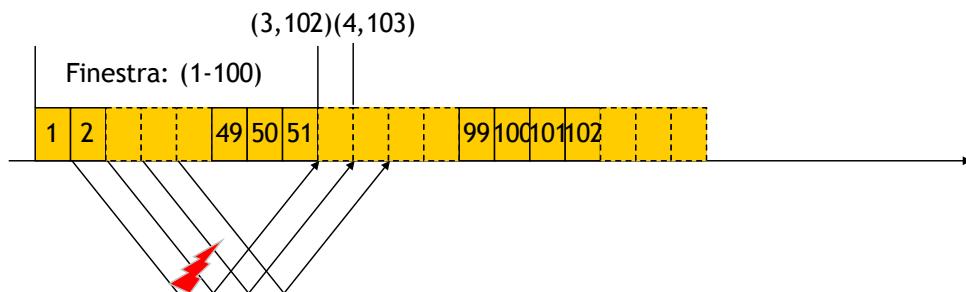
Trasmetto inutilmente i pacchetti dal 1 a 100

Caso 2



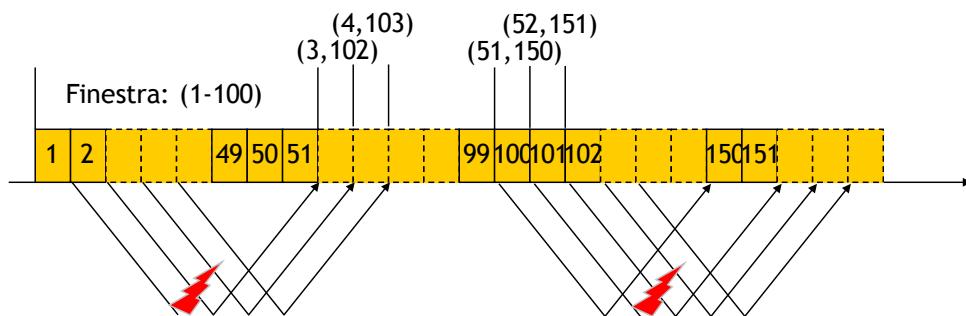
Come nel caso precedente trasmetto inutilmente i pacchetti dal 1 a 100

Caso 3



I pacchetti 1, 2, ... sono tutti ricevuti correttamente e viene inviato l'ACK. L'ACK del 2 riscontra implicitamente anche il pacchetto 1 e pertanto non si ritrasmette NESSUN pacchetto.

Caso 4



Anche in questo caso la perdita del 100-esimo ACK non ha effetto.

Esercizio 4.3

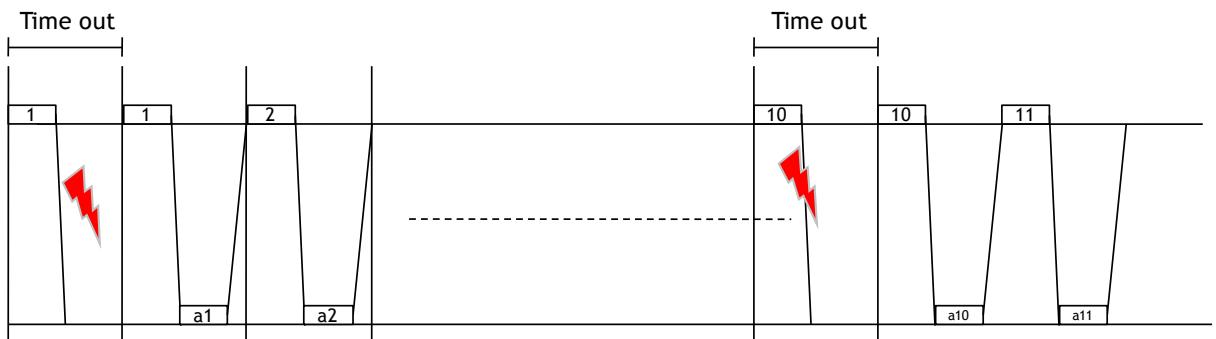
Un canale sbaglia la trasmissione di pacchetti in ragione di 1 ogni 10, mentre non sbaglia il ritorno degli ACK. Si sta usando un protocollo di tipo STOP and WAIT con tempo di propagazione pari a n volte il tempo di trasmissione di un pacchetto T e il tempo di trasmissione dell'ACK pari a T . Si consideri il time-out minimo pari a $T + TACK + 2\tau$ (con inizio al primo bit del pacchetto).

Si calcoli l'efficienza del meccanismo in termini di numero di pacchetti corretti / numero totale pacchetti trasmessi.

Si calcoli poi l'efficienza trmissiva temporale totale (tempo usato per trasmettere pacchetti corretti/tempo totale).

Soluzione

Con lo STOP and WAIT, ogni 10 pacchetti, ne sbaglio 1. Dunque l'efficienza è $\eta_{pkt} = 9/10 = 0.9$



Per quanto visto prima si trasmettono 9 pacchetti corretti (durata $9T$) ogni 10 round trip time. Un round trip time è pari a $T + nT + T + nT$. Dunque: $\eta_{tot} = 9T / 10T(2 + 2n) = 9 / 20(1 + 1n)$

Esercizio 4.4

Due stazioni A e B colloquiano attraverso due collegamenti in cascata con velocità rispettivamente di 100 Mbit/s e 200 Mbit/s e ritardo di propagazione di $500 \mu\text{s}$ su ciascun collegamento. Il forwarding fra i due collegamenti sia di tipo *Store and Forward* senza ritardo di processing. Un file di 1250 Mbyte viene trasferito fra i due nodi suddividendolo in segmenti di 10000 [bit] con header trascurabile. Si calcoli il ritardo con cui viene ricevuto l'ultimo bit del file in B nei seguenti casi in cui:

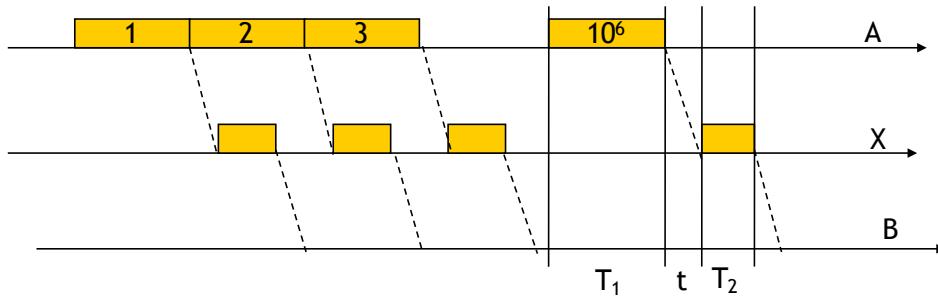
- 1) I segmenti vengono trasmessi sulle linee alla velocità massima
- 2) I segmenti vengono trasferiti attraverso un meccanismo di ritrasmissione *Stop and Wait* applicato end-to-end.
- 3) I segmenti vengono trasferiti attraverso un meccanismo di ritrasmissione *Stop and Wait* applicato su ciascuno dei due collegamenti separatamente

Si ipotizzi che le trasmissioni siano senza errori e che la lunghezza degli ACK sia pari a quella dei pacchetti.

Il numero di segmenti da trasmettere è pari a $N = (1250 \cdot 8 \cdot 10^6) / 10000 = 1000000$ segmenti.

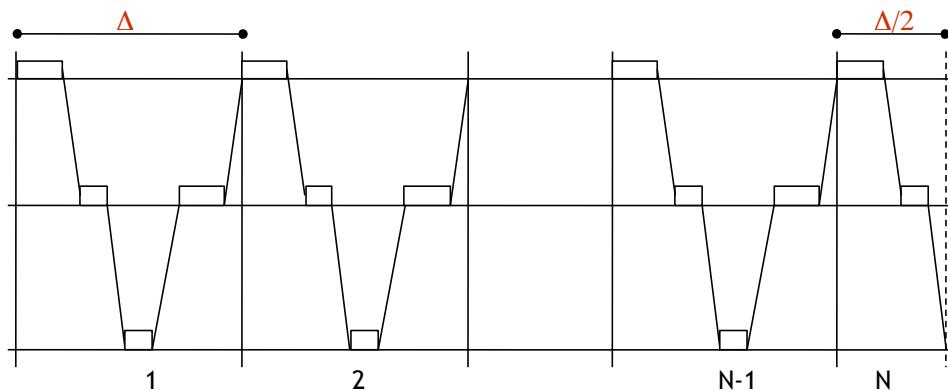
Il tempo di trasmissione sul primo link è $T_1 = 10000 \text{ [bit]} / 100 \text{ [Mbps]} = 100 \mu\text{s}$, mentre sul secondo link è $T_2 = T_1/2 = 50 \mu\text{s}$, infatti la velocità di trasmissione è doppia.

Caso 1



Il tempo totale di trasferimento è $T_{tot} = NT_1 + 2t + T_2 = 10^6 \cdot 100 \mu\text{s} + 2 \cdot 500 \mu\text{s} + 50 \mu\text{s} = 100.0015 \text{ [s]}$

Caso 2



Il tempo fra due trasmissioni consecutive sul primo link è pari a

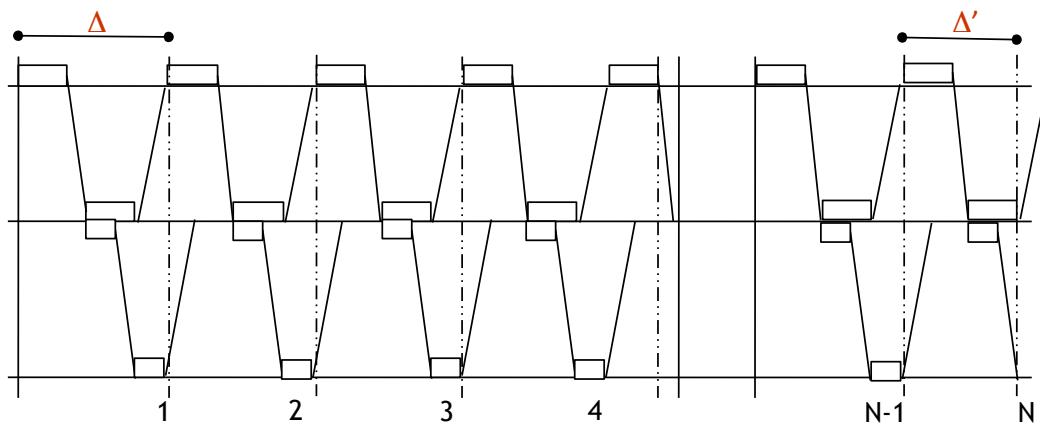
$$\Delta = T_1 + t + T_2 + t + T_2 + t + T_1 + t = 2T_1 + 2T_2 + 4t = 2.3 \text{ [ms]}$$

L'ultimo bit sul secondo link viene ricevuto dopo il tempo

$$(N - 1)\Delta + \Delta/2 = N\Delta - \Delta/2 = 2300 \text{ [s]} - 0.00115 \text{ [s]} = 2299.99885 \text{ s}$$

Caso 3

Il tempo fra due trasmissioni consecutive sul primo link è pari a $D_1 = 2T_1 + 2t = 1.2 \text{ [ms]}$, mentre il tempo fra due trasmissioni consecutive sul secondo link è pari a $D_2 = 2T_2 + 2t = 1.1 \text{ [ms]}$. Dunque i segmenti non fanno coda tra il primo ed il secondo link (quando arriva il secondo pacchetto sul primo link, il primo pacchetto è già partito sul secondo link).



La trasmissione dell'ACK sul primo link avviene contemporaneamente alla trasmissione del segmento sul secondo link, le interfacce coinvolte sono diverse.

Il tempo totale di trasferimento è pari a

$$\begin{aligned} (N - 1)\Delta + \Delta' &= (N - 1)\Delta + \Delta/2 + T_2 + t = N\Delta - \Delta/2 + T_2 + t = \\ &= ND_1 - D_1/2 + T_2 + t = 1200000 \text{ [ms]} - 0.6 \text{ [ms]} + 0.05 \text{ [ms]} + 0.5 \text{ [ms]} \\ &= 1199.99995 \text{ s} \end{aligned}$$

Esercizio 4.5

Due navi **A** e **B** si trovano alla distanza $d = 30$ km. Il transponder di identificazione di **A**, ricevendo il segnale radar da **B**, si attiva per fornire a **B** l'identità e i dati di navigazione di **A**. Il transponder di **B** a sua volta si attiva per inviare ad **A** i necessari riscontri. Per inviare un insieme completo di informazioni, **A** deve trasferire a **B** un blocco di $B = 480$ byte.

Si supponga che durante il trasferimento del blocco le due navi si possano considerare ferme. Le velocità di trasmissione sul canale da **A** a **B** e sul canale di ritorno da **B** ad **A** sono, rispettivamente, $C_{AB} = 8$ Mbit/s, $C_{BA} = 1$ Mbit/s.

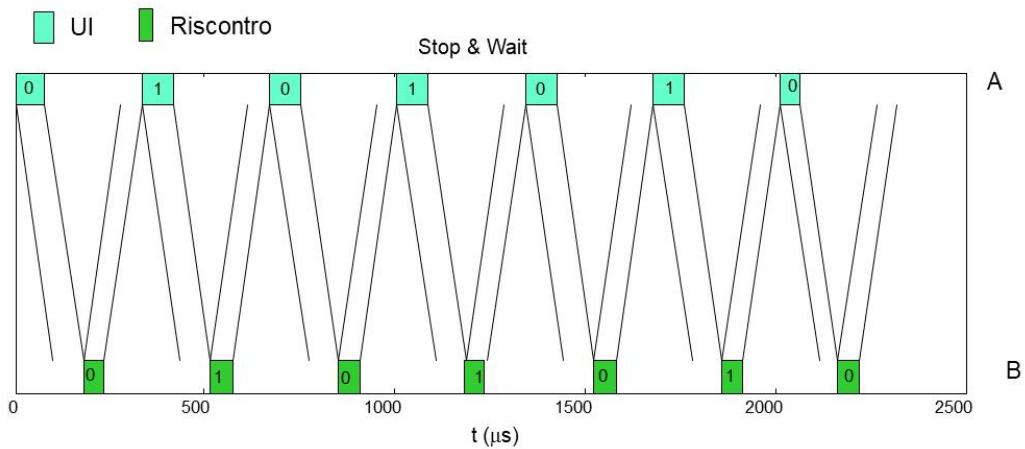
Le caratteristiche del protocollo *Stop&Wait* utilizzato per le trasmissioni sono le seguenti:

- dimensione fissa dei riscontri: $L_a = 7$ byte
- dimensione variabile dei segmenti, che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima $L_{fmax} = 80$ byte dei quali 7 byte rappresentano l'overhead del segmento
- i segmenti utilizzati per trasferire da **A** a **B** il blocco B abbiano tutti lunghezza massima ad eccezione eventualmente dell'ultimo
- tempo di elaborazione nelle stazioni **A** e **B** di un segmento o di un riscontro trascurabile

Se il protocollo è di tipo *stop-and-wait*, si calcoli in assenza di errori sul collegamento:

- a) il tempo di trasferimento $T_{S\&W}$ del segmento di dati (dall'inizio della trasmissione del primo segmento al termine della ricezione dell'ultimo riscontro);
 - b) il *throughput* dati di utente effettivo della connessione **A** → **B** $THR_{S\&W}$, misurato in [bit/s], e quanto questo vale in percentuale rispetto alla capacità del canale **A** → **B** (efficienza $\eta_{S\&W}$).
-

- Distanza A-B: $d = 30$ km
- Ritardo di propagazione A-B (e B-A) ($c = 300000$ km/s):
 $\tau = d / c = (30 / 300000) \text{ s} = 100 \mu\text{s}$
- $B = 480 \text{ byte} \Rightarrow 6 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
- $L_{fmax} = [(7 + 73) \cdot 8] \text{ bit} = (80 \cdot 8) \text{ bit} = 640 \text{ bit}$
- $L_{flast} = (7 + 42) \text{ byte} = 392 \text{ bit}$
- $L_a = (7 \cdot 8) \text{ bit} = 56 \text{ bit}$
- $C_{AB} = 8 \text{ Mbit/s}$
- $T_{fmax} = L_{fmax} / C_{AB} = 80 \mu\text{s}; T_{flast} = L_{flast} / C_{AB} = 49 \mu\text{s}$
- $C_{BA} = 1 \text{ Mbit/s}$
- $T_a = L_a / C_{BA} = (7 \cdot 8 / 1) \mu\text{s} = 56 \mu\text{s}$



Stop and wait

- $T_{S\&W} = 6 (T_{fmax} + T_a + 2 \tau) + (T_{flast} + T_a + 2 \tau) = [6 (80 + 56 + 200) + (49 + 56 + 200)] \mu\text{s} = 2.321 \text{ ms}$
- $THR_{S\&W} = B / T_{S\&W} = (480 \cdot 8 / 2.321) \text{ kbit/s} = 1.66 \text{ Mbit/s}$
- $\eta_{S\&W} = THR_{S\&W} / C = 1.66 / 8 = 0.21 \Rightarrow \eta_{S\&W\%} = 21.00 \%$

Esercizio 4.6

Un ricognitore “U2” (**A**) ad una quota di $h = 24860$ m sta sorvolando (sulla verticale) una postazione nemica che si trova a distanza $d = 86.5$ km da un mezzo terrestre d’attacco alleato **B**. Il velivolo deve inviare al mezzo **B** informazioni relative alla postazione contenute in un messaggio di 840 byte. Per determinare la quantità B di dati totale da trasmettere per inviare il messaggio si deve tener conto dell’operazione di cifratura dei messaggi che aggiunge un overhead del 20% all’informazione da criptare.

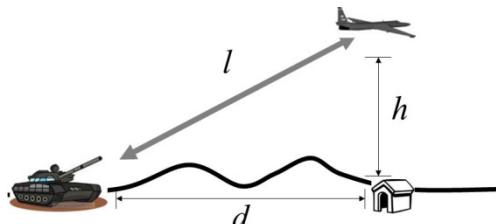
Si supponga che durante il trasferimento del blocco l’aereo sia fermo rispetto **B**. Le velocità di trasmissione sul canale da **A** a **B** e sul canale di ritorno da **B** ad **A** sono identiche e uguali a $C = 9.6$ Mbit/s

Le caratteristiche del protocollo *Stop&Wait* utilizzato per le trasmissioni sono le seguenti:

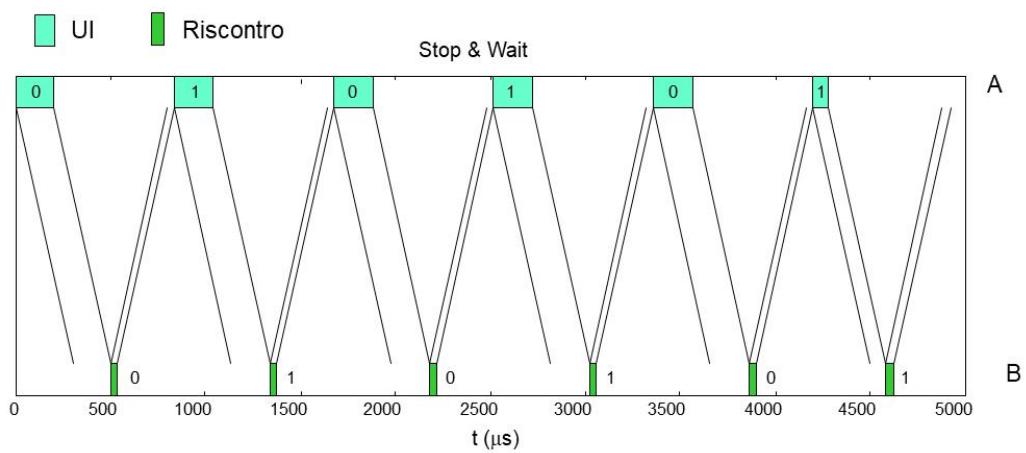
- dimensione fissa dei riscontri: $L_a = 48$ byte
- dimensione variabile delle unità informative (UI), che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{f\max} = 240$ byte dei quali 48 byte rappresentano l’overhead
- le UI informative utilizzate per trasferire da **A** a **B** il blocco B abbiano tutte lunghezza massima ad eccezione eventualmente dell’ultima
- tempo di elaborazione nelle stazioni A e B di una UI o di un riscontro trascurabile

Se il protocollo è di tipo *stop-and-wait*, si calcoli in assenza di errori sul collegamento:

- il tempo di trasferimento $T_{S\&W}$ del segmento di dati (dall’inizio della trasmissione della prima UI informativa al termine della ricezione dell’ultimo riscontro);
 - il *throughput* dati di utente effettivo della connessione $THR_{S\&W}$, misurato in [bit/s], e quanto questo vale in percentuale rispetto alla capacità del canale **A** → **B** (efficienza $\eta_{S\&W}$).
-



- Distanza A-B: $l = (d^2 + h^2)^{1/2} = 90$ km
- Ritardo di propagazione A-B (e B-A) ($c = 300000$ km/s):
 $\tau = l / c = (90 / 300000) \text{ s} = 300 \mu\text{s}$
- $B = 840 \cdot (1 + 0.2) \text{ byte} = 1008 \text{ byte} \Rightarrow 5 \text{ UI } L_{f\max} + 1 \text{ UI } L_{flast}$
- $L_{f\max} = [(192 + 48) \cdot 8] \text{ bit} = (240 \cdot 8) \text{ bit} = 1920 \text{ bit}$
- $L_{flast} = (48 + 48) \text{ byte} = 768 \text{ bit}$
- $L_a = (48 \cdot 8) \text{ bit} = 384 \text{ bit}$
- $C = 9.6 \text{ Mbit/s}$
- $T_{f\max} = L_{f\max} / C = 200 \mu\text{s}; T_{flast} = L_{flast} / C_u = 80 \mu\text{s};$
- $T_a = L_a / C = 40 \mu\text{s}$



Esercizio 4.7

Due rack A e B di un supercomputer comunicano tra di loro mediante un collegamento in fibra ottica con interfacce a $C = 10 \text{ Gbit/s}$. La linea in fibra ha lunghezza $d = 16 \text{ m}$.

Il rack A trasferisce un file a B utilizzando un protocollo di tipo Go-Back-N.

Le caratteristiche del protocollo GBN utilizzato sono le seguenti:

- dimensione fissa di riscontri: $L_a = 10 \text{ byte}$,
- dimensione variabile delle unità informative (UI), che dipende dalla dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{fmax} = 60 \text{ byte}$ dei quali 10 byte rappresentano l'overhead
- tempo di elaborazione nelle stazioni A e B di una UI o di un riscontro trascurabile

Si supponga che il file da trasferire sia di lunghezza $F = 375 \text{ byte}$ e che le UI informative utilizzate abbiano tutte lunghezza massima ad eccezione dell'ultima.

Si calcoli quanti pacchetti vengono trasmessi da A a B, indicando le dimensioni di ciascun pacchetto, nonché i tempi di trasmissione di ciascun pacchetto e di ciascun riscontro

- $L_{fmax} = (60 \cdot 8) \text{ bit} = 480 \text{ bit}$
- $F = 375 \text{ byte} = (7 \cdot 50 + 1 \cdot 25) \text{ byte} \Rightarrow 7 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (25 + 10) \text{ byte} = 280 \text{ bit}$

A invia a B 8 pacchetti in totale, di cui 7 di dimensione $L_{fmax} = 480 \text{ bit}$ ed 1 di dimensione $L_{flast} = 280 \text{ bit}$.

I tempi di trasmissione dei vari pacchetti e dei riscontri sono ottenuti come:

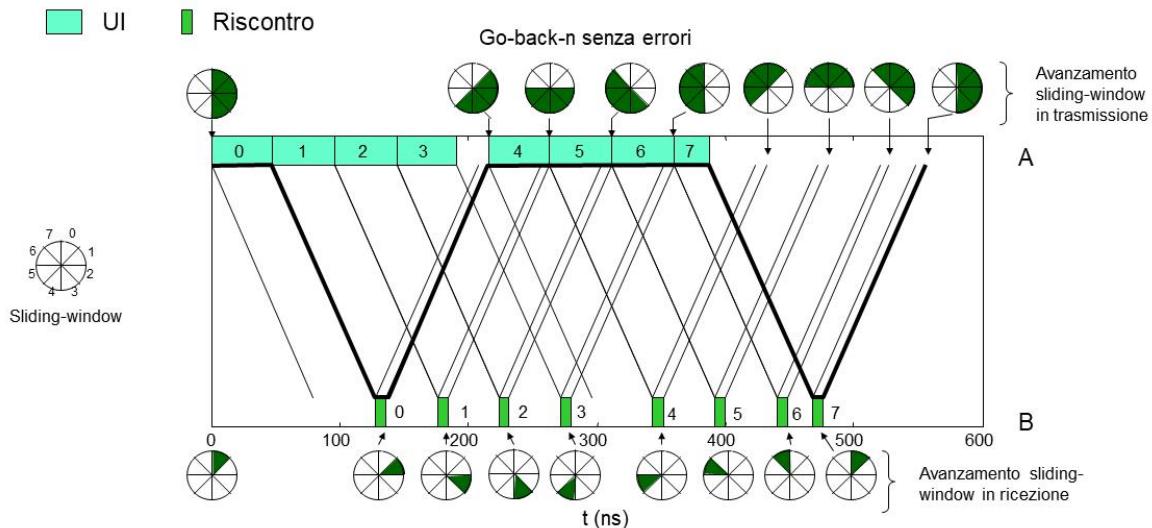
- $T_{fmax} = L_{fmax} / C = 48 \text{ ns}; T_{flast} = L_{flast} / C_u = 28 \text{ ns};$
- $T_a = L_a / C_d = 8 \text{ ns}$

Esercizio 4.8

Con riferimento allo scambio di dati dell'esercizio precedente, si consideri il protocollo di tipo *go-back-n*, con ampiezza della finestra di trasmissione $W_s = 4$ UI, finestra di ricezione, W_r , di ampiezza pari a 1, modulo di numerazione delle UI pari a $N=8$, timeout $T_O = 3$ s. Dopo aver rappresentato lo scambio di UI su un diagramma temporale, indicando per ogni UI (sia da A a B che da B ad A) il relativo valore di numerazione, si calcolino, in assenza di errori sul collegamento, i seguenti parametri:

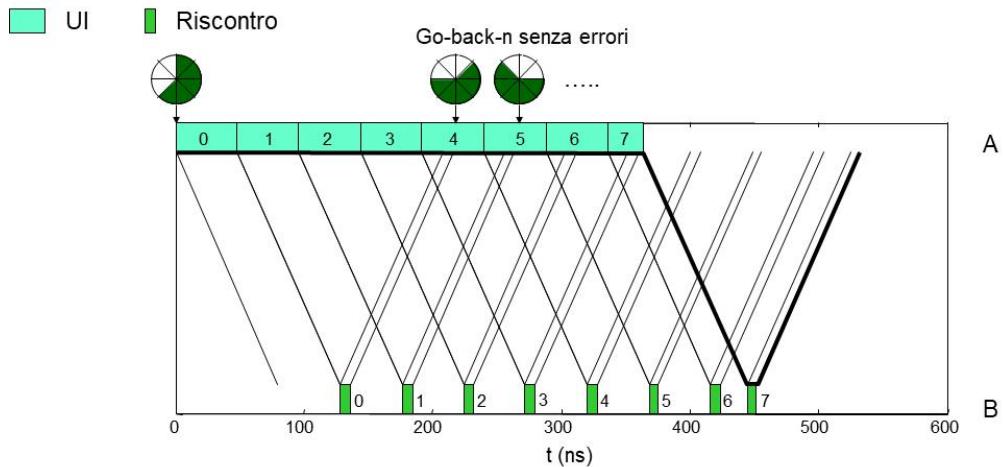
- il tempo di trasferimento T_{GBN} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo riscontro);
- il throughput dati effettivo THR_{GBN} della connessione, misurato in [bit/s], e l'efficienza η del collegamento $A \rightarrow B$;
- la dimensione minima W_s' della finestra di trasmissione per cui si raggiunge l'efficienza η_{max} massima possibile e il valore di η_{max} .

- Ritardo di propagazione in fibra = $\tau_U \cong 5 \mu\text{s}/\text{km} = 5 \text{ ns/m}$
- $l = 16 \text{ m}$
 $\Rightarrow \tau = l \cdot \tau_U = 80 \text{ ns}$
- $L_{fmax} = (60 \cdot 8) \text{ bit} = 480 \text{ bit}$
- $B = 375 \text{ byte} = (7 \cdot 50 + 1 \cdot 25) \text{ byte} \Rightarrow 7 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (25 + 10) \text{ byte} = 280 \text{ bit}$
- $L_a = 80 \text{ bit}$
- $C = 10 \text{ Gbit/s}$
- $T_{fmax} = L_{fmax} / C = 48 \text{ ns}; T_{flast} = L_{flast} / C_u = 28 \text{ ns};$
- $T_a = L_a / C_d = 8 \text{ ns}$



$$\text{Go-back-n} \Rightarrow W_s = 4; W_r = 1$$

- $W_s \cdot T_{fmax} = 192 \text{ ns} < T_{fmax} + T_a + 2\tau = 216 \text{ ns}$
- $T_{GBN} = T_{fmax} + \tau + T_a + \tau + 3T_{fmax} + T_{flast} + \tau + T_a + \tau = 4T_{fmax} + T_{flast} + 2T_a + 4\tau = 556 \text{ ns}$
- $THR_{GBN} = B / T_{GBN} = (375 \cdot 8 / 556) \text{ Gbit/s} = 5.4 \text{ Gbit/s}$
- $\eta_{GBN} = THR_{GBN} / C = 5.4 / 10 = 0.54 \Rightarrow \eta_{GBN\%} = 54 \%$



- Per avere la massima efficienza occorre che la finestra di trasmissione non “strozzi”: $W_s = \min \{W_s\} \mid W_s \cdot T_{fmax} > T_{fmax} + T_a + 2\tau = 216 \text{ ns} \Rightarrow$
- $W_s = \lceil (T_{fmax} + T_a + 2\tau) / T_{fmax} \rceil = \lceil 216 / 48 \rceil = 5$
- $T'_{GB} = 7 T_{fmax} + T_{flast} + T_a + 2\tau = 532 \text{ ns}$
- $THR'_{GB} = (375 \cdot 8 / 532) \text{ Gbit/s} = 5.64 \text{ Gbit/s} \quad \eta_{max} = 0.564 \Rightarrow \eta_{max\%} = 56.4 \%$

Esercizio 4.9

Due stazioni terrestri A e B sono collegate tramite una rete di trasmissione dati in fibra ottica che è così caratterizzata:

- distanza tra A e B $d = 4000 \text{ km}$,
- velocità di propagazione del segnale $v = 5 \mu\text{s}/\text{km}$,
- capacità del collegamento $C = 160 \text{ Mbit/s}$.

Il protocollo che controlla la trasmissione delle UI su questo collegamento sia così caratterizzato:

- dimensione fissa dei riscontri: $L_a = 80 \text{ byte}$,
- dimensione variabile delle UI informative, che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{f\max} = 125080 \text{ byte}$ dei quali 80 byte rappresentano l'overhead
- tempo di elaborazione di una UI o di un riscontro trascurabile

Si consideri il trasferimento da A a B di un segmento di dati di lunghezza 1205000 byte, imponendo che le UI utilizzate abbiano lunghezza massima ad eccezione eventualmente dell'ultima.

Si calcoli quanti pacchetti vengono trasmessi da A a B, indicando le dimensioni di ciascun pacchetto, nonché i tempi di trasmissione di ciascun pacchetto e di ciascun riscontro

- $L_{f\max} = (125080 \cdot 8) \text{ bit} = 1000640 \text{ bit}$
- $B = 1205000 \text{ byte} = (9 \cdot 125000 + 1 \cdot 80000) \text{ byte} \Rightarrow 9 \text{ UI } L_{f\max} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (80000 + 80) \text{ byte} = 640640 \text{ bit}$

A invia a B 10 pacchetti in totale, di cui 9 di dimensione $L_{f\max} = 1000640 \text{ bit}$ ed 1 di dimensione $L_{flast} = 640640 \text{ bit}$.

I tempi di trasmissione dei vari pacchetti e dei riscontri sono ottenuti come:

- $T_{f\max} = 6.254 \text{ ms}; T_{flast} = 4.004 \text{ ms}; T_a = 4.0 \mu\text{s}$

Esercizio 4.10

Con riferimento allo scambio di dati dell'esercizio precedente, si consideri il protocollo di tipo *go-back-n* che opera con riscontri positivi ACK e negativi NACK, con dimensione della finestra di trasmissione $W_s=5$ UI, finestra di ricezione, W_r , di ampiezza pari a 1, modulo di numerazione delle UI pari a $N=8$, timeout $T_O=60$ ms. Si calcolino, in assenza di errori sul collegamento, i seguenti parametri:

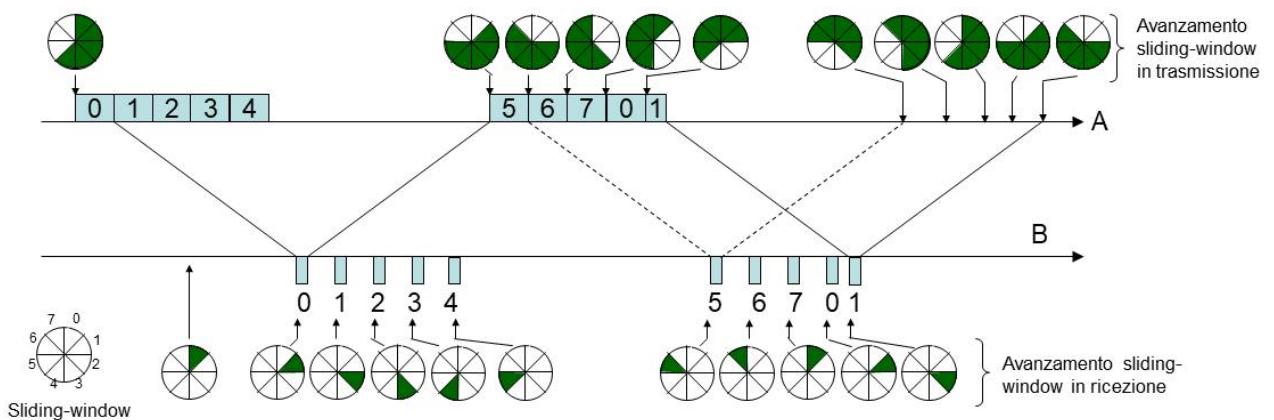
- il tempo di trasferimento T_{GBN} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo ACK);
- il throughput dati effettivo THR_{GBN} della connessione, misurato in [bit/s];

Sempre con protocollo di tipo *go-back-n*, ma nel caso il canale sia soggetto a errori, si calcoli

- a) il tempo di trasferimento T'_{GBN} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo ACK), nel caso la quarta UI trasmessa da A vada persa e non venga ricevuta da B;
- b) il numero totale N_{tot} di UI informative trasmesse, comprese quelle trasmesse due volte.

- $\tau = d \cdot v = (4000 \cdot 5) \mu s = 20 \text{ ms}$
- $L_{fmax} = (125080 \cdot 8) \text{ bit} = 1000640 \text{ bit}$
- $B = 1205000 \text{ byte} = (9 \cdot 125000 + 1 \cdot 80000) \text{ byte} \Rightarrow 9 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (80000 + 80) \text{ byte} = 640640 \text{ bit}$
- $L_a = 640 \text{ bit}$
- $T_x = L_x / C$
- $C = 160000 \text{ kbit/s}$
- $T_{fmax} = 6.254 \text{ ms}; T_{flast} = 4.004 \text{ ms}; T_a = 4.0 \mu s$

Go-back-n \Rightarrow

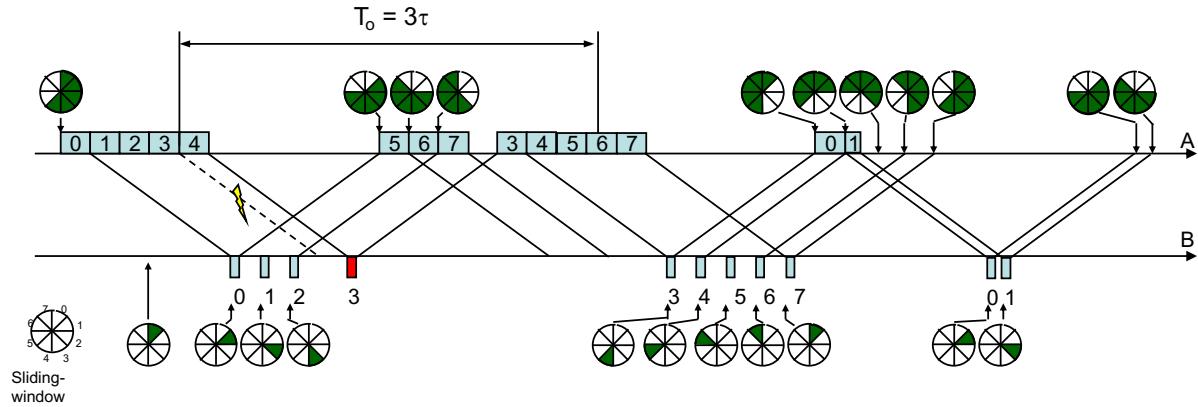


Go-back-n senza errori

- $T_{GBN} = T_{fmax} + \tau + T_a + \tau + 4 T_{fmax} + T_{flast} + \tau + T_a + \tau =$
 $= 5 T_{fmax} + T_{flast} + 2 T_a + 4 \tau = 115.282 \text{ ms}$

- $\text{THR}_{\text{GBN}} = B / T_{\text{GBN}} = (1205000 \cdot 8 / 115.282) \text{kbit/s} = 83.621 \text{ Mbit/s}$
- $\eta_{\text{GBN}} = \text{THR}_{\text{GBN}} / C = 83.621 / 160 = 0.523$

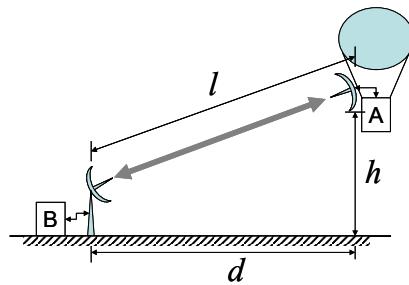
Go-back-n con errori



- La quarta UI (UI numero 3) trasmessa da A va perduta
- B si può accorgere dell'errore di trasmissione solo quando riceve la UI successiva (UI numero 4). A questo punto B la scarta ed invia il NACK relativo (NACK 3).
- Per effetto della finestra di trasmissione, A si blocca dopo la trasmissione della UI numero 4. Riprende appena riceve gli ACK delle 0, 1 e 2 e trasmette le UI numero 5, 6 e 7.
- Finalmente A riceve il NACK di 3 (prima che scada il timeout) e inizia la ritrasmissione di tutte le UI dalla 3 in poi, fino alla 7, in quanto la finestra resta bloccata dalla 3 alla 7.
- Intanto B scarta le UI 5, 6 e 7 e le riscontra rispondendo sempre col NACK relativo alla UI numero 3
- B inizia a ricevere UI dalla 3 alla 7 senza errori ed invia i riscontri
- Appena A riceve l'ACK relativo alla UI numero 4 può ruotare la finestra e trasmettere la UI numero 0. Di seguito trasmette la UI numero 1 (corta)
- La trasmissione si conclude regolarmente con la ricezione da parte di A degli ultimi due ACK sulle UI numero 0 e 1
- $T'_{\text{GBN}} = 5 T_{\text{fmax}} + \tau + T_a + \tau + T_{\text{fmax}} + \tau + T_a + \tau + T_{\text{fmax}} + T_{\text{flast}} + \tau + T_a + \tau = 7 T_{\text{fmax}} + 6 \tau + 3 T_a + T_{\text{flast}} = 167.794$
- $N_{\text{fot}} = 15$

Esercizio 4.11

Una stazione A a bordo di un pallone aerostatico deve trasmettere via radio alla stazione B di terra una sequenza di dati. Il pallone si trova ad una distanza $d = 12 \text{ km}$ (misurati al suolo) dalla stazione di terra e ad una quota $h = 9000 \text{ m}$ (si veda il disegno qui riportato). Nel calcolare la lunghezza l del collegamento si trascurino l'altezza dell'antenna della stazione di terra e le lunghezze dei collegamenti antenna-stazione e si supponga il pallone fermo rispetto alla stazione di terra almeno per tutta la durata del trasferimento dei dati.



Le velocità di trasmissione sono: $C_d = 16 \text{ Mbit/s}$ in direzione A \rightarrow B; $C_u = 4 \text{ Mbit/s}$ in direzione B \rightarrow A.

Il protocollo che controlla la trasmissione delle UI su questo collegamento sia così caratterizzato:

- dimensione fissa dei riscontri: $L_a = 8 \text{ byte}$,
- dimensione variabile delle UI informative, che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{fmax} = 56 \text{ byte}$ dei quali 8 byte rappresentano l'overhead
- tempo di elaborazione nelle stazioni A e B di una UI o di un riscontro trascurabile

Si consideri il trasferimento da A a B di un segmento di dati di lunghezza 350 byte, imponendo che le UI utilizzate abbiano lunghezza massima ad eccezione eventualmente dell'ultima.

Si calcoli quanti pacchetti vengono trasmessi da A a B, indicando le dimensioni di ciascun pacchetto, nonché i tempi di trasmissione di ciascun pacchetto e di ciascun riscontro

- $L_{fmax} = (56 \cdot 8) \text{ bit} = 448 \text{ bit}$
- $B = 350 \text{ byte} = (7 \cdot 48 + 1 \cdot 14) \text{ byte} \Rightarrow 7 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (14 + 8) \text{ byte} = 176 \text{ bit}$

A invia a B 8 pacchetti in totale, di cui 7 di dimensione $L_{fmax} = 448 \text{ bit}$ ed 1 di dimensione $L_{flast} = 176 \text{ bit}$.

I tempi di trasmissione dei vari pacchetti e dei riscontri sono ottenuti come:

- $T_{fmax} = L_{fmax} / C_d = 28 \mu\text{s}; T_{flast} = L_{flast} / C_d = 11 \mu\text{s};$
- $T_a = L_a / C_u = 16 \mu\text{s}$

Esercizio 4.12

Con riferimento allo scambio di dati dell'esercizio precedente, si consideri il protocollo di tipo *go-back-n*, con dimensione della finestra di trasmissione $W_s = 4$ UI e finestra di ricezione, W_r , di ampiezza pari a 1, modulo di numerazione delle UI pari a $N=8$, timeout $T_O = 150 \mu s$. Dopo aver rappresentato lo scambio di UI su di un diagramma temporale, indicando per ogni UI (sia da A a B che da B ad A) il relativo valore di numerazione, si calcolino, sempre in assenza di errori sul collegamento, i seguenti parametri:

- il tempo di trasferimento T_{GBN} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo riscontro);
- il throughput dati effettivo THR_{GBN} della connessione, misurato in [bit/s];

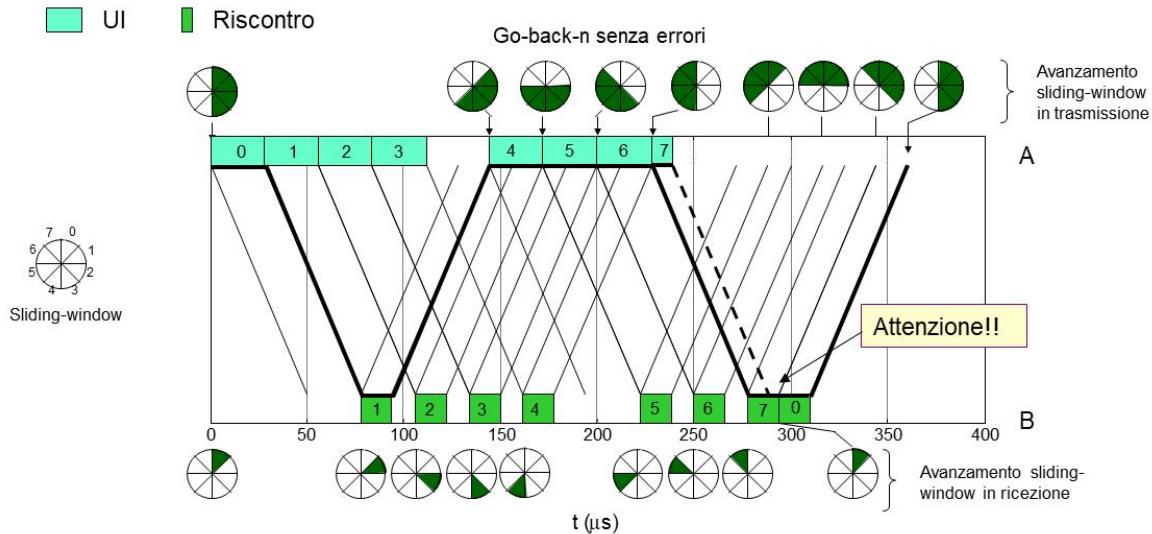
Sempre con protocollo di tipo *go-back-n*, i canali $B \rightarrow A$ e $A \rightarrow B$ siano ora soggetti ad errori. In particolare, si consideri il caso in cui il riscontro emesso da B relativo alla quinta UI trasmessa da A vada perso e non venga ricevuto da A; inoltre, la settima UI trasmessa da A vada perduta e non venga ricevuta da B. Rappresentare lo scambio di UI su di un diagramma temporale, indicando per ogni UI la relativa numerazione. Si calcoli

- il tempo di trasferimento T'_{GBN} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo riscontro);
- il numero totale $N_{f\text{tot}}$ di UI informative trasmesse, comprese quelle trasmesse due volte.

-
- Velocità propagazione \cong velocità luce vuoto = $v \cong 300000 \text{ km/s}$
 - $l = (h^2 + d^2)^{1/2} = (81 + 144)^{1/2} \text{ km} = 15 \text{ km}$
 - $\tau = l / v = (15 / 3 \cdot 10^5) \text{ s} = 50 \mu s$
 - $L_{f\text{max}} = (56 \cdot 8) \text{ bit} = 448 \text{ bit}$
 - $B = 350 \text{ byte} = (7 \cdot 48 + 1 \cdot 14) \text{ byte} \Rightarrow 7 \text{ UI } L_{f\text{max}} + 1 \text{ UI } L_{f\text{last}}$
 - $L_{f\text{last}} = (14 + 8) \text{ byte} = 176 \text{ bit}$
 - $L_a = 64 \text{ bit}$
 - $C_u = 4 \text{ Mbit/s}; C_d = 16 \text{ Mbit/s}$
 - $T_{f\text{max}} = L_{f\text{max}} / C_d = 28 \mu s; T_{f\text{last}} = L_{f\text{last}} / C_d = 11 \mu s;$
 - $T_a = L_a / C_u = 16 \mu s$

Go-back-n \Rightarrow

- $W_s \cdot T_{f\text{max}} = 112 \mu s < T_{f\text{max}} + T_a + 2 \tau = 144 \mu s$

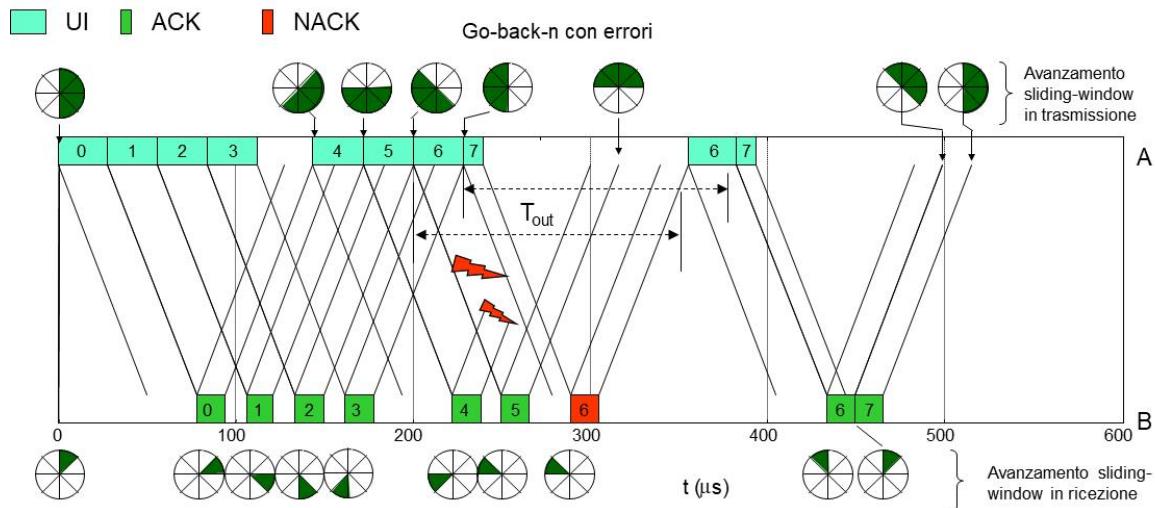


Go-back-n senza errori

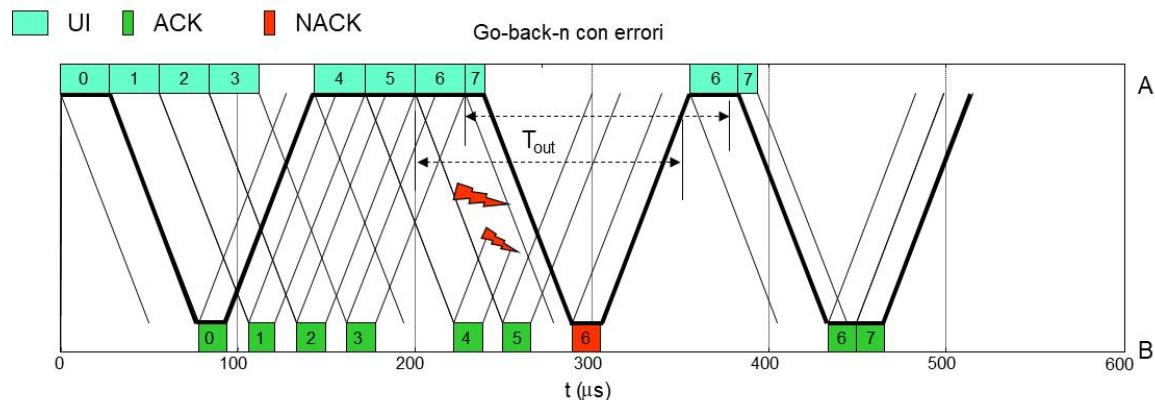
- Si nota una particolarità: la UI numero 7 viene interamente ricevuta da B prima che B abbia finito di trasmettere il riscontro della trama precedente (la 6). Dunque il riscontro della UI numero 7 viene trasmesso da B non appena terminata la trasmissione del riscontro della UI numero 6.
- $T_{GBN} = T_{fmax} + \tau + T_a + \tau + 3 T_{fmax} + \tau + 2T_a + \tau = 4T_{fmax} + 3T_a + 4\tau = 360 \mu s$
- $THR_{GBN} = B / T_{GBN} = (350 \cdot 8 / 360) \text{ Mbit/s} = 7.78 \text{ Mbit/s}$
- $\eta_{GBN} = THR_{GBN} / C_u = 7.78 / 16 = 0.486 \Rightarrow \eta_{GBN\%} = 48.6 \%$

Go-back-n con errori

- Errore sulla UI numero 6 e sul riscontro ACK 4



- La perdita del riscontro relativo alla UI 4 (la quinta) non ha alcuna influenza sulla trasmissione. Infatti prima dello scadere del T_{out} per tale UI (che scadrebbe a $t = 322 \mu s$) A ha già interamente ricevuto (a $t = 316 \mu s$) il riscontro ACK 6 relativa alla sesta UI. In virtù del valore cumulativo degli ACK in Go-back-n, A assume che tutte le UI fino alla UI numero 5 siano state correttamente ricevute da B
- Si noti che all'arrivo di ACK 5 la finestra di trasmissione scorre di due posizioni
- La UI numero 6 che va perduta (la settima) è la penultima trasmessa da A. Quando B riceve la UI successiva (fuori sequenza) invia un NACK 6.
- A si può accorgere dell'errore o allo scadere del T_{out} , ossia a $t = 378 \mu s$, o alla ricezione del NACK a $t = 355 \mu s$. Si verifica prima quest'ultimo evento
- Da questo istante inizia la ritrasmissione delle ultime due UI, seguita dalla trasmissione dei riscontri da parte di B (che avviene come in GBN senza errori)



- $T'_{GBN} = T_{fmax} + \tau + T_a + \tau + 3 \cdot T_{fmax} + T_{flast} + \tau + T_a + \tau + T_{fmax} + \tau + 2 \cdot T_a + \tau = 5T_{fmax} + 4T_a + T_{flast} + 6\tau = 515 \mu s$
- $N_{fot} = 10$

Esercizio 4.13

Si consideri un collegamento (bidirezionale simmetrico) in fibra ottica tra due apparati A e B all'interno di una centrale di commutazione. La lunghezza l del collegamento è di 80 m.

La velocità di trasmissione in entrambe le direzioni ($A \rightarrow B$, $B \rightarrow A$) è: $C = 2.56 \text{ Gbit/s}$.

Il protocollo che controlla la trasmissione su questo collegamento sia così caratterizzato:

- dimensione fissa dei riscontri: $L_a = 8 \text{ byte}$,
- dimensione variabile delle UI informative, che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{f\max} = 80 \text{ byte}$ dei quali 8 byte rappresentano l'overhead,
- tempo di elaborazione nelle stazioni A e B di una UI o di un riscontro trascurabile

Si consideri il trasferimento da A a B di un segmento di dati di lunghezza 392 byte, imponendo che le UI utilizzate abbiano lunghezza massima ad eccezione eventuale dell'ultima.

Si calcoli quanti pacchetti vengono trasmessi da A a B, indicando le dimensioni di ciascun pacchetto, nonché i tempi di trasmissione di ciascun pacchetto e di ciascun riscontro.

- $L_{f\max} = (80 \cdot 8) \text{ bit} = 640 \text{ bit}$
- $B = 392 \text{ byte} = (5 \cdot 72 + 1 \cdot 32) \text{ byte} \Rightarrow 5 \text{ UI } L_{f\max} + 1 \text{ UI } L_{flast}$
- $L_{flast} = (32 + 8) \text{ byte} = 320 \text{ bit}$

A invia a B 6 pacchetti in totale, di cui 5 di dimensione $L_{f\max} = 640 \text{ bit}$ ed 1 di dimensione $L_{flast} = 320 \text{ bit}$.

I tempi di trasmissione dei vari pacchetti e dei riscontri sono ottenuti come:

- $T_{f\max} = L_{f\max} / C = 250 \text{ ns}; T_{flast} = L_{flast} / C = 125 \text{ ns};$
- $T_a = L_a / C = 25 \text{ ns}$

Esercizio 4.14

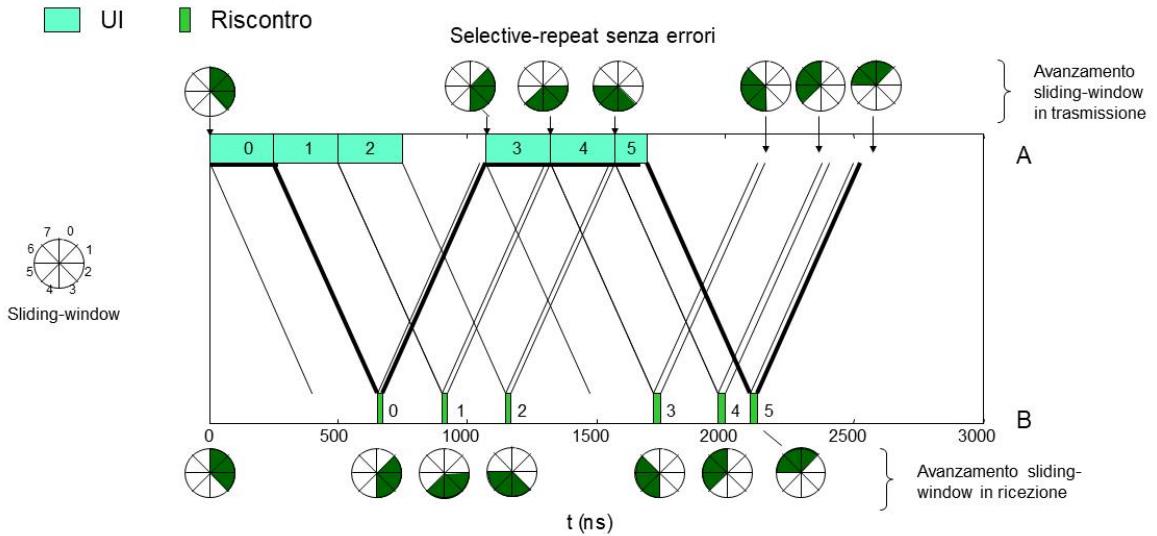
Con riferimento allo scambio di dati dell'esercizio precedente, si consideri il protocollo di tipo *selective-repeat*, con dimensione della finestra di trasmissione $W_s = 3$ UI e finestra di ricezione, W_r , di ampiezza pari a 3, modulo di numerazione delle UI pari a $N=8$, timeout $T_O = 2 \mu s$. Dopo aver rappresentato lo scambio di UI su di un diagramma temporale, indicando per ogni UI o riscontro la relativa numerazione, si calcolino, sempre in assenza di errori sul collegamento, i seguenti parametri:

- il tempo di trasferimento T_{SR} del segmento di dati (dall'inizio della trasmissione della prima trama I al termine della ricezione dell'ultimo ACK);
- il throughput dati effettivo THR_{SR} della connessione, misurato in [bit/s], e l'efficienza η_{SR} del collegamento A → B.

Sempre con protocollo di tipo *selective-repeat*, il canale A → B sia ora soggetto ad errori. In particolare, si consideri il caso in cui la quarta UI trasmessa da A vada perduta e non venga ricevuta da B. Rappresentare lo scambio di UI su di un diagramma temporale, indicando per ogni UI e per ogni riscontro la relativa numerazione. Si calcoli

- il tempo di trasferimento T'_{SR} del segmento di dati (dall'inizio della trasmissione della prima UI I al termine della ricezione dell'ultimo ACK);
 - il throughput THR'_{SR} conseguito in bit/s e l'efficienza η'_{SR} sul collegamento rispetto alla capacità disponibile.
-

- Ritardo di propagazione unitario in fibra = $\tau_U \cong 5 \mu s/km$
- $l = 80 m = 0.08 km$
- $\tau = l \cdot \tau_{pul_f} = 0.4 \mu s = 400 ns$
- $L_{fmax} = (80 \cdot 8) bit = 640 bit$
- $B = 392 byte = (5 \cdot 72 + 1 \cdot 32) byte \Rightarrow 5 UI L_{fmax} + 1 UI L_{flast}$
- $L_{flast} = (32 + 8) byte = 320 bit$
- $L_a = 64 bit$
- $C = 2.56 Gbit/s$
- $T_{fmax} = L_{fmax} / C = 250 ns; T_{flast} = L_{flast} / C = 125 ns;$
- $T_a = L_a / C = 25 ns$



Selective-repeat $\Rightarrow W_r = W_s = 3$

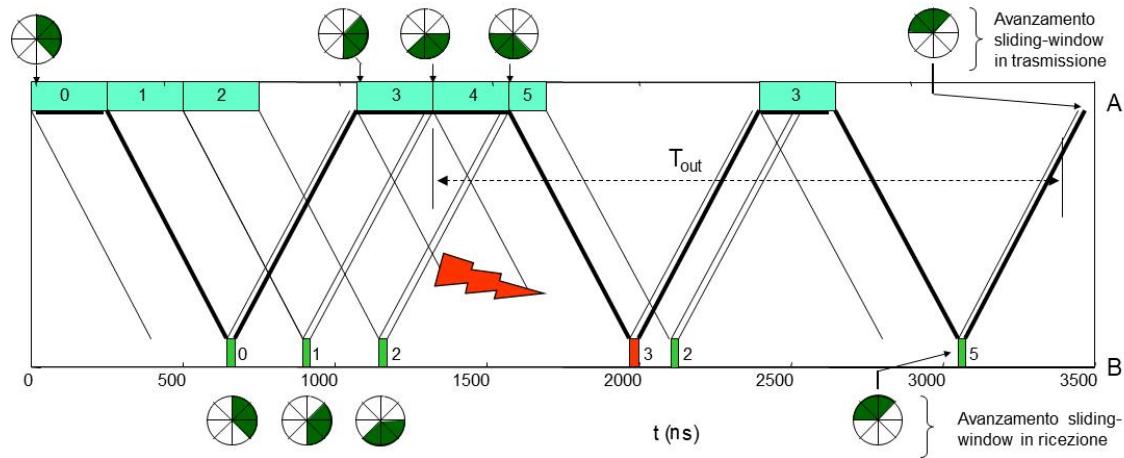
- $W_s \cdot T_{fmax} = 750 \text{ ns} < T_{fmax} + T_a + 2\tau = 1075 \text{ ns}$

Selective-repeat senza errori

- $T_{sr} = T_{fmax} + \tau + T_a + \tau + 2T_{fmax} + T_{flast} + \tau + T_a + \tau = 3T_{fmax} + 2T_a + 4\tau + T_{flast} = 2525 \text{ ns}$
- $THR_{SR} = B / T_{SR} = (392 \cdot 8 / 2525) \text{ Gbit/s} = 1.242 \text{ Gbit/s}$
- $\eta_{SR} = THR_{SR} / C = 1.242 / 2.560 = 0.4852 \Rightarrow \eta_{SR\%} = 48.52 \%$

Selective-repeat con errori

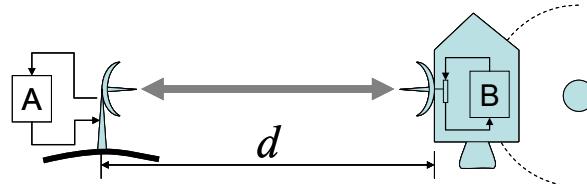
- Errore sulla quarta UI (UI numero 3)



- $T'_{SR} = T_{fmax} + \tau + T_a + \tau + 2 T_{fmax} + \tau + T_a + \tau + T_{fmax} + \tau + T_a + \tau = 4T_{fmax} + 3T_a + 6\tau = 3475$ ns
- $THR'_{SR} = B / T'_{sr} = (392 \cdot 8 / 3475) \text{ Gbit/s} = 902.4 \text{ Mbit/s}$
- $\eta'_{SR} = THR'_{sr} / C = 902.4 / 2560 = 0.3525 \Rightarrow \eta_{SR\%} = 35.25 \%$

Esercizio 4.15

Il modulo orbitale di una missione di esplorazione lunare B si trova in orbita intorno al nostro satellite ad una distanza $d = 1.2$ secondi-luce dalla superficie terrestre. La stazione di controllo missione sulla Terra A trasferisce un blocco di dati a B utilizzando un protocollo di tipo ARQ. Si supponga che durante il trasferimento del blocco B sia fermo rispetto A.



Lo scambio di dati tra A e B è caratterizzato dai seguenti parametri:

- capacità del collegamento A → B: $C_u = 1.2$ kbit/s
- capacità del collegamento B → A: $C_d = 320$ bit/s

Le caratteristiche del protocollo ARQ utilizzato sono le seguenti:

- dimensione fissa dei riscontri: $L_a = 8$ byte,
- dimensione variabile delle UI informative, che dipende della dimensione dei dati trasportati, fino ad una lunghezza massima di UI $L_{fmax} = 60$ byte dei quali 8 byte rappresentano l'overhead
- tempo di elaborazione nelle stazioni A e B di una UI o di un riscontro trascurabile

Si supponga che il blocco di dati di utente da trasferire sia di lunghezza 328 byte, e che le UI informative utilizzate abbiano tutte lunghezza massima ad eccezione eventualmente dell'ultima.

Si calcoli quanti pacchetti vengono trasmessi da A a B, indicando le dimensioni di ciascun pacchetto, nonché i tempi di trasmissione di ciascun pacchetto e di ciascun riscontro

-
- $L_{fmax} = (60 \cdot 8) \text{ bit} = 480 \text{ bit}$
 - $B = 328 \text{ byte} = (6 \cdot 52 + 1 \cdot 16) \text{ byte} \Rightarrow 6 \text{ UI } L_{fmax} + 1 \text{ UI } L_{flast}$
 - $L_{flast} = (16 + 8) \text{ byte} = 192 \text{ bit}$

A invia a B 7 pacchetti in totale, di cui 6 di dimensione $L_{fmax} = 480$ bit ed 1 di dimensione $L_{flast} = 192$ bit.

I tempi di trasmissione dei vari pacchetti e dei riscontri sono ottenuti come:

- $T_{fmax} = L_{fmax} / C_u = 400 \text{ ms}; T_{flast} = L_{flast} / C_u = 160 \text{ ms};$
- $T_a = L_a / C_d = 200 \text{ ms}$

Esercizio 4.16

Con riferimento allo scambio di dati dell'esercizio precedente, si consideri il protocollo di tipo *selective repeat*, con dimensione della finestra di ricezione $W_r = 4$ UI, finestra di trasmissione, W_s , di ampiezza pari a 4, modulo di numerazione delle UI pari a $N=8$, timeout $T_O = 3$ s. Dopo aver rappresentato lo scambio di UI su di un diagramma temporale, indicando per ogni UI e riscontro la relativa numerazione, si calcolino, sempre in assenza di errori sul collegamento, i seguenti parametri:

- il tempo di trasferimento T_{SR} del segmento di dati (dall'inizio della trasmissione della prima UI al termine della ricezione dell'ultimo ACK);
- il *throughput* dati effettivo THR_{SR} della connessione, misurato in [bit/s], e l'efficienza η del collegamento A → B.

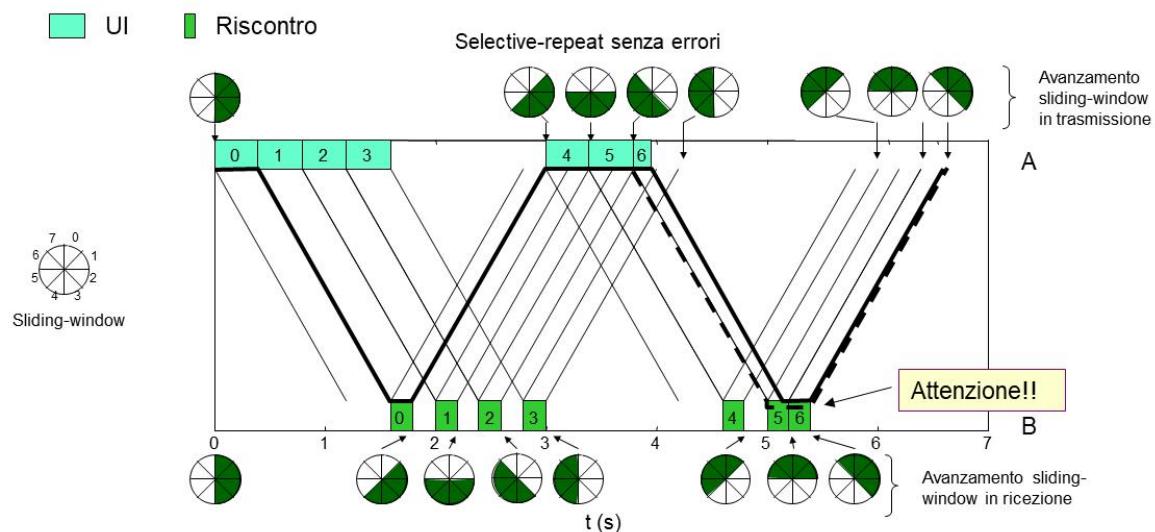
Sempre con protocollo di tipo *selective repeat*, il canale A → B sia ora soggetto ad errori. In particolare, si consideri il caso in cui la quarta UI trasmessa da A vada persa e non venga ricevuta da B. Rappresentare lo scambio di UI su di un diagramma temporale, indicando per ogni UI o riscontro la relativa numerazione. Si calcoli

- il tempo di trasferimento T'_{SR} del segmento di dati (dall'inizio della trasmissione della prima trama I al termine della ricezione dell'ultimo ACK).

- Ritardo di propagazione A-B: $\tau_u = \tau_d = \tau = 1.2$ s
- $L_{f\max} = (60 \cdot 8)$ bit = 480 bit
- $B = 328$ byte = $(6 \cdot 52 + 1 \cdot 16)$ byte ⇒ 6 UI $L_{f\max}$ + 1 UI L_{flast}
- $L_{flast} = (16 + 8)$ byte = 192 bit
- $L_a = 64$ bit
- $C_u = 1.2$ kbit/s; $C_d = 0.32$ kbit/s
- $T_{f\max} = L_{f\max} / C_u = 400$ ms; $T_{flast} = L_{flast} / C_u = 160$ ms;
- $T_a = L_a / C_d = 200$ ms

Selective-repeat ⇒

- $W_s \cdot T_{f\max} = 1600$ ms < $T_{f\max} + T_a + 2\tau = 3000$ ms

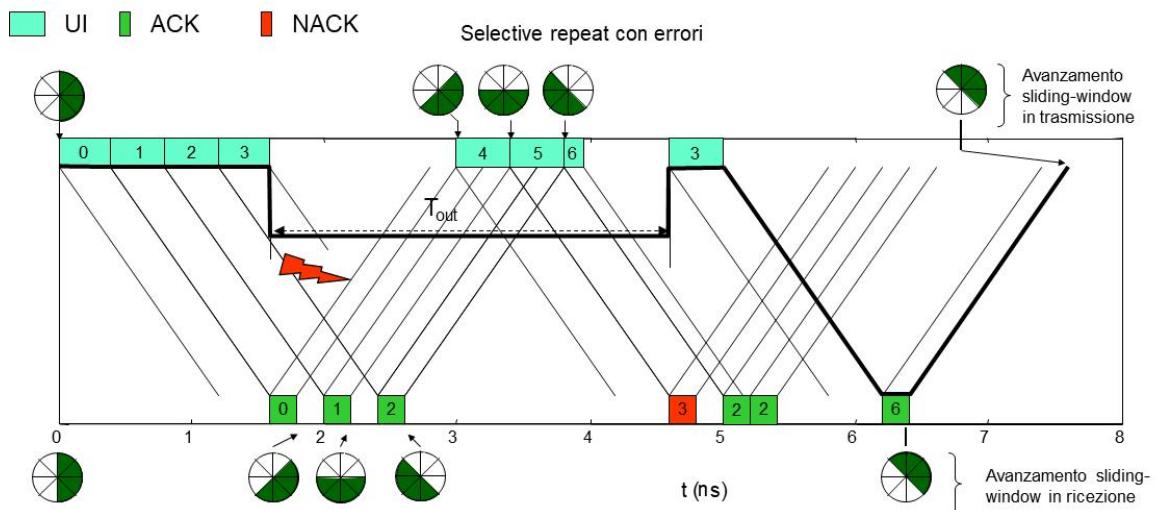
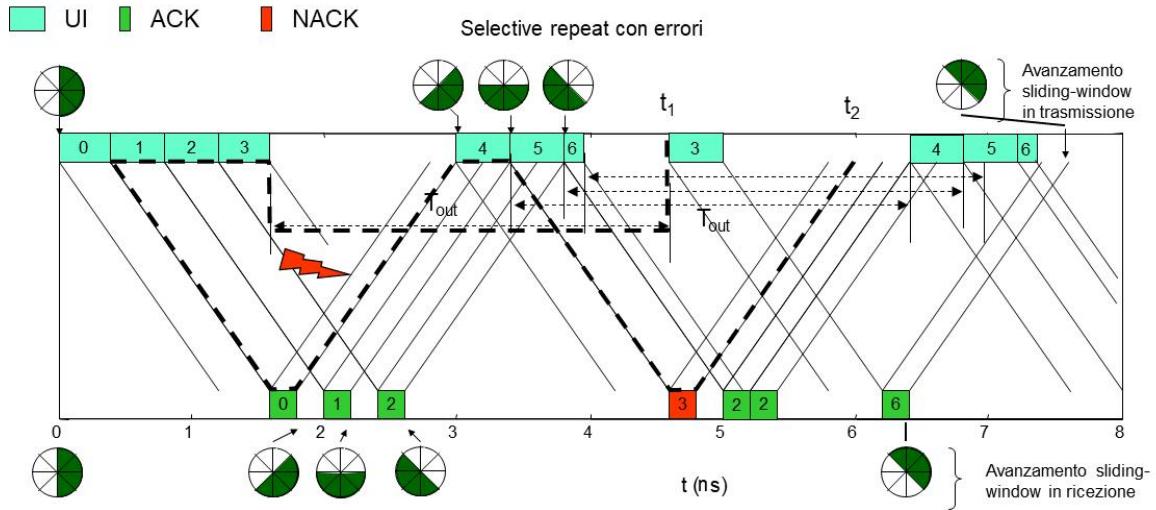


Selective-repeat senza errori

- $T_{SR} = T_{fmax} + \tau + T_a + \tau + 2T_{fmax} + \tau + 2T_a + \tau = 3T_{fmax} + 3T_a + 4\tau = 6600 \text{ ms} = 6.6 \text{ s}$
- $THR_{SR} = B / T_{SR} = (328 \cdot 8 / 6.6) \text{ bit/s} = 398 \text{ bit/s}$
- $\eta_{SR} = THR_{SR} / C_u = 398 / 1200 = 0.3313 \Rightarrow \eta_{SR\%} = 33.13 \%$
- La UI numero 6 viene ricevuta interamente da B prima che B abbia terminato la trasmissione del riscontro sulla UI numero 5. Quindi il riscontro della 6 non viene trasmesso immediatamente, ma solo al termine della trasmissione del riscontro precedente. Per facilitare il calcolo del ritardo totale di trasferimento conviene seguire allora la linea spezzata tratteggiata anzichè quella a tratto continuo.

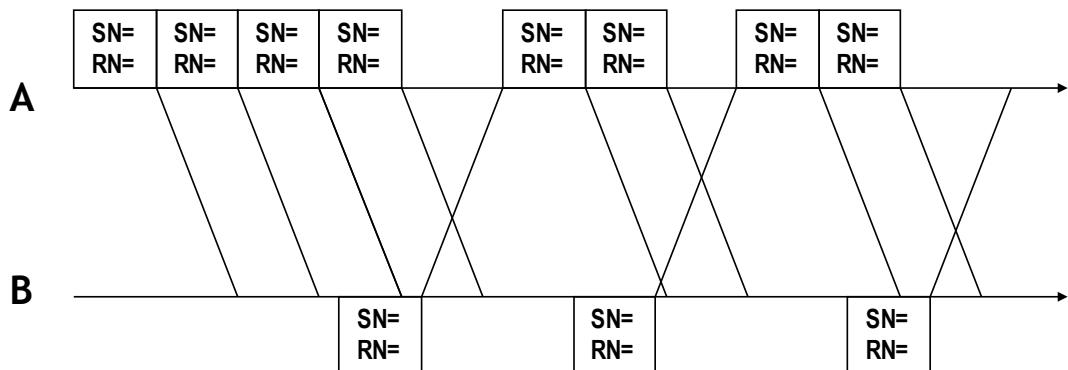
Selective-repeat con errori

- Errore sulla quarta UI

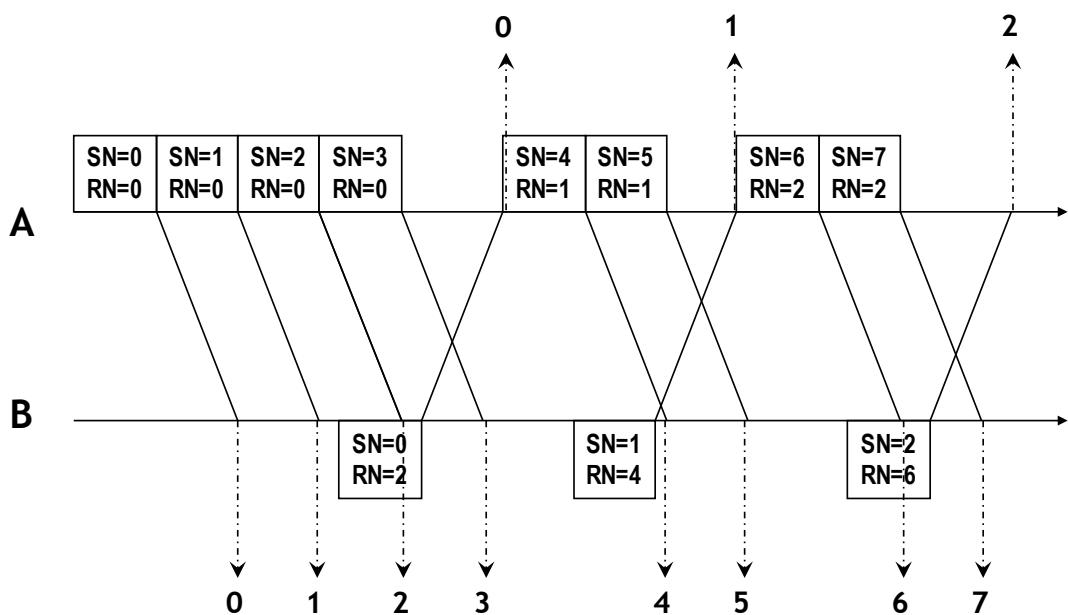


Esercizio 4.17

Si consideri il protocollo Go-BACK-N con $N = 4$ e time-out pari a 8 volte il tempo di trasmissione di un pacchetto (con inizio il primo bit del primo pacchetto della finestra). Si completi la figura in accordo alle regole del protocollo

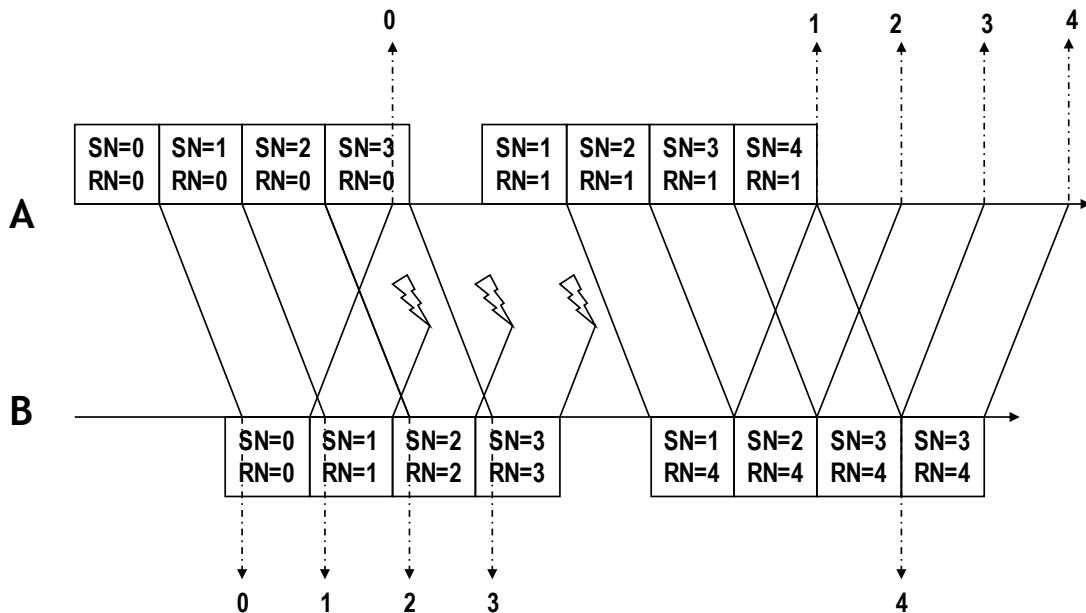


Soluzione

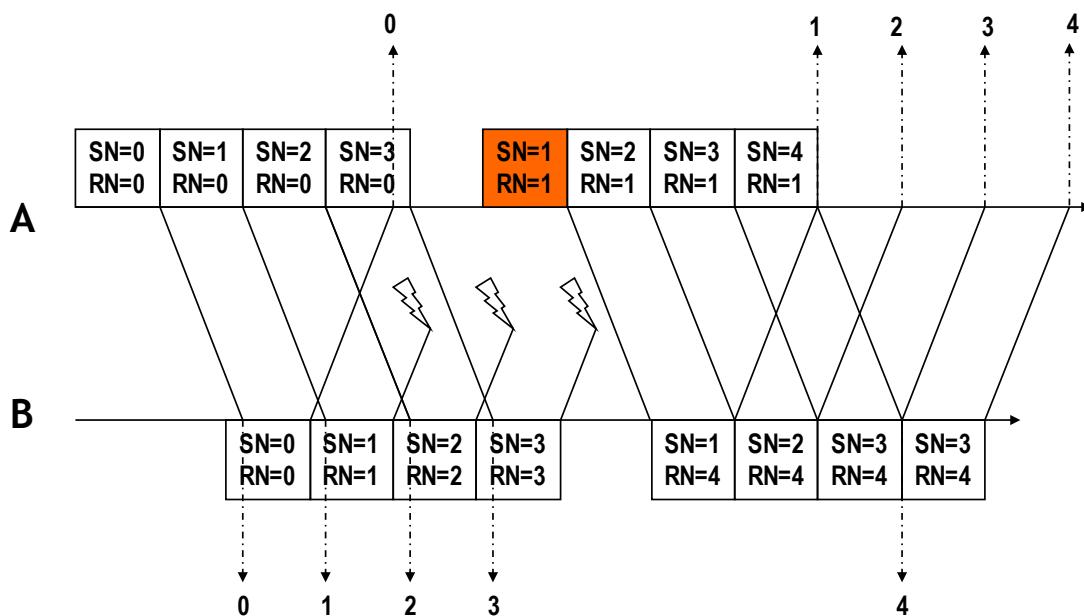


Esercizio 4.19

Si consideri il protocollo Go-BACK-N con $N = 4$. L'esempio in figura è corretto? Se no, perché?

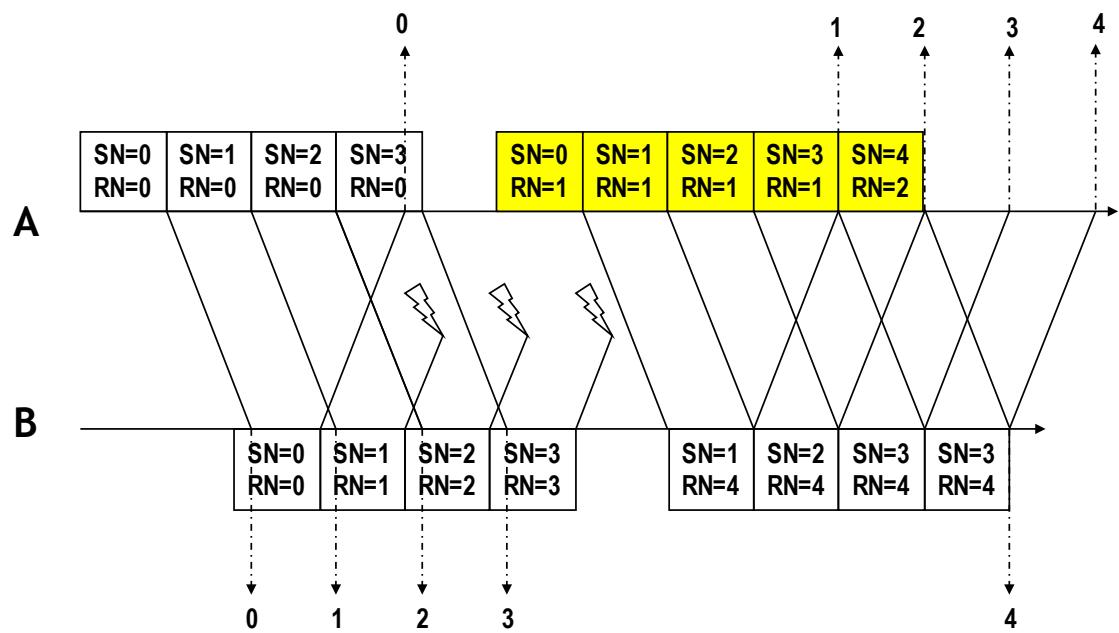


Soluzione



L'esempio è errato l'SN = 0 (A->B) non è stato riscontrato, infatti l'unico ACK arrivato ha RN = 0 (il ricevitore si aspetta di ricevere SN = 0). Infatti, SN = 0 (A->B) è stato finito di ricevere dopo l'inizio della trasmissione dell'ACK RN = 0 (B->A).

L'esempio corretto è:



Esercizio 4.20

Si calcoli il checksum secondo la modalità del protocollo UDP della seguente sequenza di bit:

**1 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 0 1 0 1 0 1 0 1 1 0 0 0 0 1 0 0 0 1
0 0 0 1 0 1**

Il primo passo consiste nello spezzare la sequenza in blocchi da 16 bit

**1 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
1 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1**

Sommiamo i primi due blocchi

**1 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 +
1 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 =
1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1**

Il riporto viene aggiunto al risultato come bit meno significativo

**1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 +
1 =
1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 0**

Sommiamo il terzo blocco

**1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 0 +
1 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 1 =
1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1**

Il riporto viene aggiunto al risultato come bit meno significativo

**0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 +
1 =
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0**

Eseguiamo il complemento a 1 e troviamo il valore del checksum

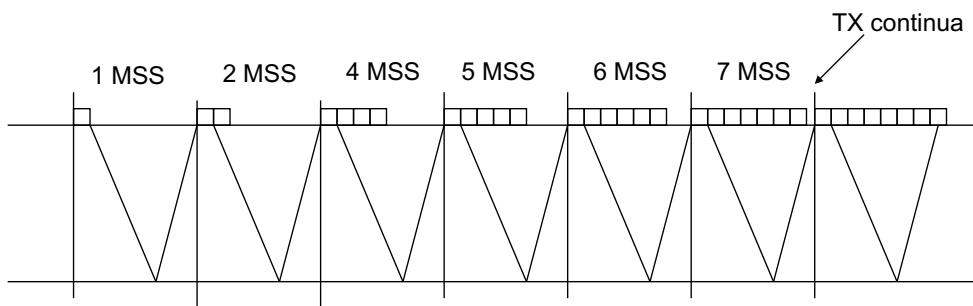
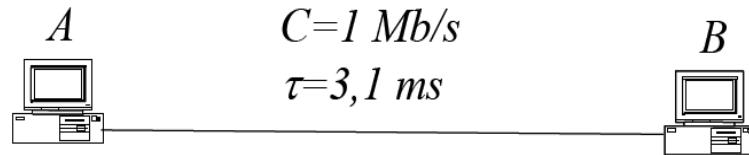
1 0 1 1 1 1 1 1 1 1 0 1

4.2 Esercizi sul LIVELLO di TRASPORTO – parte II

Esercizio 4.21

Si consideri il collegamento in figura tra i due host A e B. A deve trasferire una sequenza di 100 segmenti di lunghezza massima usando TCP. Si calcoli il tempo necessario supponendo:

- $MSS = 1000 \text{ [bit]}$
- lunghezza degli header di tutti i livelli trascurabile
- la connessione venga aperta da A e la lunghezza dei segmenti di apertura della connessione sia trascurabile
- la lunghezza degli ACK sia trascurabile
- STHRESH sia pari a 5 MSS



Il tempo di trasmissione $T = 1000 \text{ [bit]} / 1 \text{ [Mb/s]} = 1 \text{ [ms]}$, mentre $RTT = 6.2 \text{ [ms]}$ + $T = 7.2 \text{ [ms]}$

La trasmissione è dunque discontinua fino a che $WT < RTT$, cioè fino a che $W = 8$.

Il tempo totale di trasferimento è pari a:

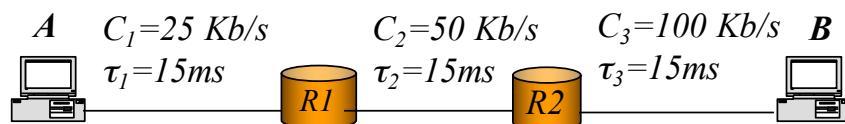
$$2 \tau(\text{setup connessione}) + 6 RTT (\text{Primi } 25 \text{ MSS}) + 75 T (\text{75 MSS in trasmissione continua}) + 2 \tau(\text{ritorno ACK dell'ultimo MSS}) = 130.6 \text{ [ms]}$$

Esercizio 4.22

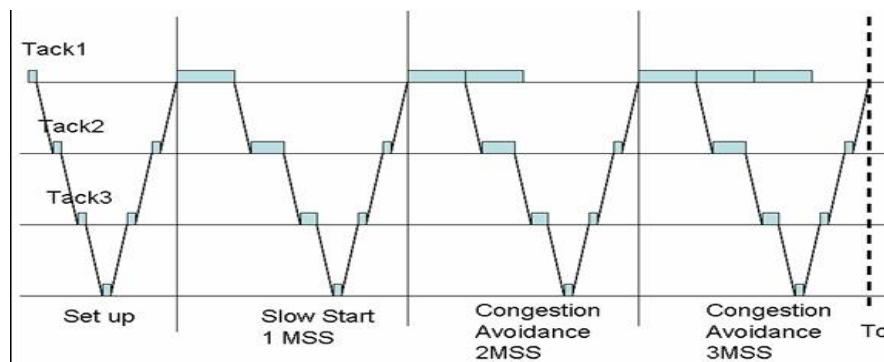
All'istante 0 viene attivata una connessione TCP tra l'host A e l'host B. Si calcoli l'istante di tempo oltre il quale la trasmissione sul link 1 risulta continua, supponendo

- header trascurabili
- link bidirezionali e simmetrici
- STHRESH = 400 [byte]
- dimensione segmenti MSS = 200 [byte]
- dimensione ACK = dimensione segmenti per apertura della connessione = 20 [byte]
- connessione aperta dal terminale A

Quanto tempo occorre per trasferire un file da 2 [kbyte] (dall'istante di attivazione della connessione all'istante di ricezione dell'ACK dell'ultimo segmento) considerando il meccanismo di invio dei riscontri applicato end-to-end?



Soluzione



Cominciamo calcolando i tempi di trasmissione sui vari link, il RTT end-to-end ed il tempo di setup:

$$T_1 = 200 \cdot 8 [\text{bit}] / 25 [\text{kb/s}] = 64 \text{ ms}$$

$$T_2 = \frac{1}{2} T_1 = 32 \text{ ms}$$

$$T_3 = \frac{1}{2} T_2 = 16 \text{ ms}$$

$$RTT = T_1 + T_2 + T_3 + 2(\tau_2 + \tau_1 + \tau_3) + (Tack_1 + Tack_2 + Tack_3) = 213.2 \text{ [ms]}$$

$$T_{\text{setup}} = 2(Tack_1 + Tack_2 + Tack_3) + 2(\tau_2 + \tau_1 + \tau_3) = 112.4 \text{ [ms]}$$

Il link più lento è il link 1, che sarà il collo di bottiglia, cioè il primo a saturarsi. Dunque la trasmissione è continua sul link 1 quando: $WT_1 > RTT$. Vale a dire $W > RTT/T_1 = 3.3$

L'istante in cui la trasmissione diventa continua è quando la finestra vale 4 MSS, cioè

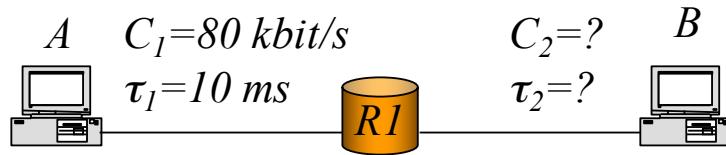
$$Tc = T_{\text{setup}} + 3 RTT = 112.4 \text{ [ms]} + 649.6 \text{ [ms]} = 752 \text{ [ms]}$$

Il file da trasferire è di 2 kbyte, equivalenti a 10 MSS. Il tempo per trasferire 10 MSS è:

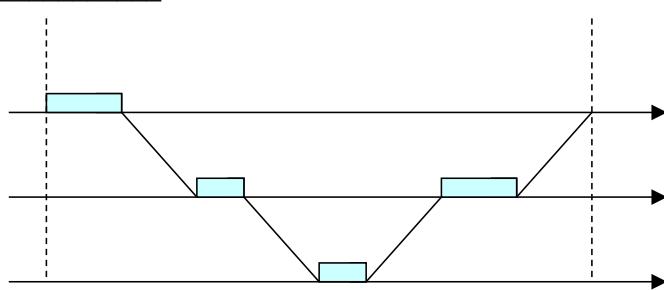
$$T_{\text{tot}} = T_{\text{setup}} + 4 RTT + 3 T_1 = 1.15 \text{ [s]}$$

Esercizio 4.23

Si consideri il collegamento in figura



A vuole conoscere la capacità e il ritardo di propagazione del link 2 e allo scopo invia a B due messaggi di echo: M_1 di lunghezza $m_1 = 1000 \text{ byte}$, ed M_2 di lunghezza $m_2 = 1500 \text{ byte}$; per ognuno di essi misura un Round-Trip-Time (RTT) pari a 780 ms e 1130 ms , rispettivamente. Nella risposta, B utilizza messaggi con le stesse lunghezze. Calcolare C_2 e τ_2 nell'ipotesi che le lunghezze degli header siano trascurabili.



Secondo lo scambio in figura, impostiamo un sistema di due equazioni (una per pacchetto) in due incognite (C_2 e τ_2)

$$RTT_1 = 2 \left(\frac{m_1}{C_1} + \tau_1 + \frac{m_1}{C_2} + \tau_2 \right)$$

$$RTT_2 = 2 \left(\frac{m_2}{C_1} + \tau_1 + \frac{m_2}{C_2} + \tau_2 \right)$$

Inserendo i valori numerici abbiamo

$$780 = 2 \left(\frac{8000}{80} + 10 + \frac{8000}{C_2} + \tau_2 \right) = 220 + \frac{16000}{C_2} + 2\tau_2$$

$$1130 = 2 \left(\frac{12000}{80} + 10 + \frac{12000}{C_2} + \tau_2 \right) = 320 + \frac{24000}{C_2} + 2\tau_2$$

E risolvendo

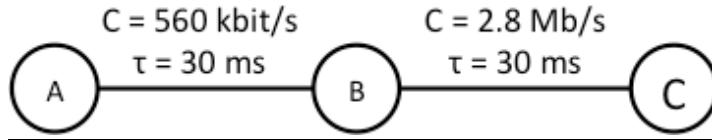
$$\tau_2 = 280 - \frac{8000}{C_2}$$

$$810 = \frac{24000}{C_2} + 2 \left(280 - \frac{8000}{C_2} \right) = \frac{8000}{C_2} + 560;$$

$$C_2 = \frac{8000}{250} = 32 \text{ kbit/s}$$

$$\tau_2 = 280 - \frac{8000}{32} = 30 \text{ ms}$$

Esercizio 4.24



Attraverso il collegamento in figura si vuole aprire una connessione TCP e trasferire un file da 60850 [byte]. La connessione è caratterizzata dai seguenti parametri:

- $MSS = 1200 \text{ byte}$
- Lunghezza header TCP, $H_{TCP} = 40 \text{ byte}$
- Lunghezza totale header inferiori a TCP, $H_{INF} = 160 \text{ byte}$
- Lunghezza ACK, $L_A = 350 \text{ byte}$ (inclusi tutti gli header)
- $SSTHRESH = 60000 \text{ [byte]}$

Si chiede di indicare:

1. Se la trasmissione diventerà mai continua. Se sì, a partire da quale istante?
2. Il tempo totale di trasferimento del file in assenza di errori (fino alla ricezione dell'ultimo ACK alla sorgente).

Punto 1

Il file è composto da $50 \text{ MSS} + 850 \text{ byte}$, dunque i segmenti di lunghezza massima sono lunghi

$$L = 1200 + 40 + 160 = 1400 \text{ [byte]} = 11200 \text{ bit}$$

mentre l'ultimo segmento è lungo

$$L' = 850 + 40 + 160 = 1050 \text{ [byte]} = 8400 \text{ bit}$$

I tempi di trasmissione sono

$$T_1 = \frac{L}{C_1} = 20 \text{ [ms]}$$

$$T_2 = \frac{L}{C_2} = 4 \text{ [ms]}$$

$$TACK_1 = \frac{ACK}{C_1} = 5 \text{ ms}$$

$$TACK_2 = \frac{ACK}{C_2} = 1 \text{ [ms]}$$

$$T'_1 = \frac{L'}{C_1} = 15 \text{ [ms]}$$

$$T'_2 = \frac{L'}{C_2} = 3 \text{ [ms]}$$

Abbiamo dunque

$$RTT = T_1 + T_2 + TACK_1 + TACK_2 + 4\tau = 150 \text{ [ms]}$$

La trasmissione è continua quando

$$W_c \geq \frac{RTT}{T_1} = 7.5 ==> 8 MSS \quad (\text{W=1, W=2, W=4, continua})$$

Considerando il tempo di setup

$$T_{open} = 2 TACK_1 + 2 TACK_2 + 4\tau = 132 [ms]$$

L'istante in cui la trasmissione è continua

$$T_{continua} = T_{open} + 3RTT = 582 [ms]$$

Punto 2

Vengono inviati 7 MSS prima che la trasmissione diventi continua, dunque abbiamo $50 - 7 = 43$ MSS in trasmissione continua + ultimo segmento più corto

Il RTT del segmento più corto è diverso dagli altri

$$RTT' = T'_1 + T'_2 + TACK_1 + TACK_2 + 4\tau = 144 [ms]$$

Dunque il tempo totale è

$$T_{tot} = T_{continua} + 43 T_1 + RTT' = 1586 [ms]$$

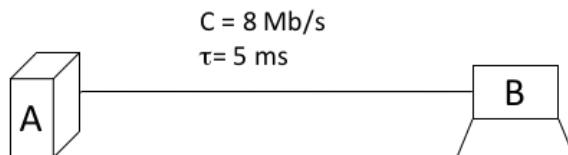
Esercizio 4.25

Una connessione TCP tra l'*host A* e l'*host B* è caratterizzata dai seguenti parametri:

- Lunghezze di *header* e *ack* trascurabili;
- *link* bidirezionali;
- SSTHRESH = 8 MSS
- MSS = 1 kbyte
- Ritardo di propagazione, $\tau = 5 \text{ ms}$
- Valore iniziale del *Time-Out*= 3 s

Si risponda ai seguenti quesiti:

- a) Dire se la trasmissione sul link diventa mai continua; in caso affermativo, trovare il tempo oltre cui la trasmissione sul link diventa continua;
- b) Trovare il tempo di trasferimento di un file di 80 kbyte da A a B;
- c) Ipotizzando che l'ultimo segmento in trasmissione venga perso, ripetere il punto b)



$$T = \frac{MSS}{C} = \frac{8 \cdot 10^3 [\text{bit}]}{8 \cdot 10^6 [\text{b/s}]} = 1 \text{ ms}$$

$$RTT = T + 2\tau = 11 \text{ ms}$$

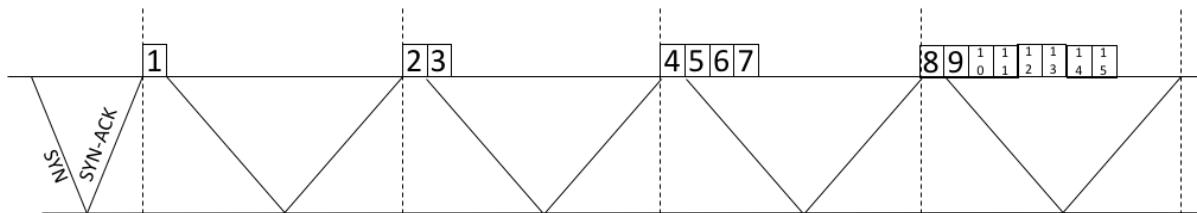
$$\text{Finestra tx continua: } W_c = \frac{RTT}{T} = 11 \text{ MSS}$$

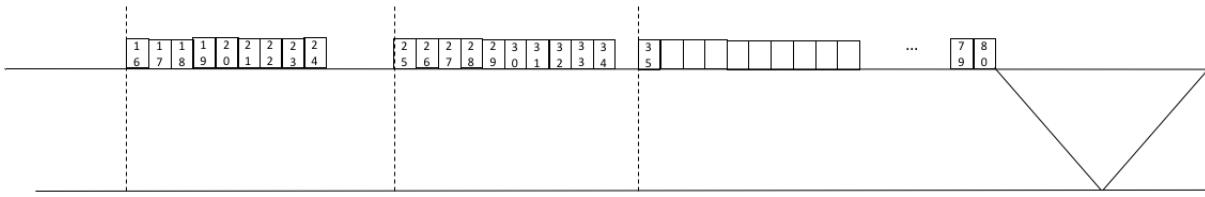
a)

- Apertura connessione: 2τ
- Slow start: (1) – (2) – (4) – (8): $4RTT$
- Congestion avoidance: (9) – (10): $2RTT$

Poi trasmissione continua

$$T_{\text{continua}} = 2\tau + 6RTT = 76 \text{ [ms]}$$





b)

File = 80 MSS

- Apertura connessione: 2τ
- Slow start: (1) – (2) – (4) – (8): $4RTT$
- Congestion avoidance: (9) – (10): $2RTT$

In totale 34 segmenti

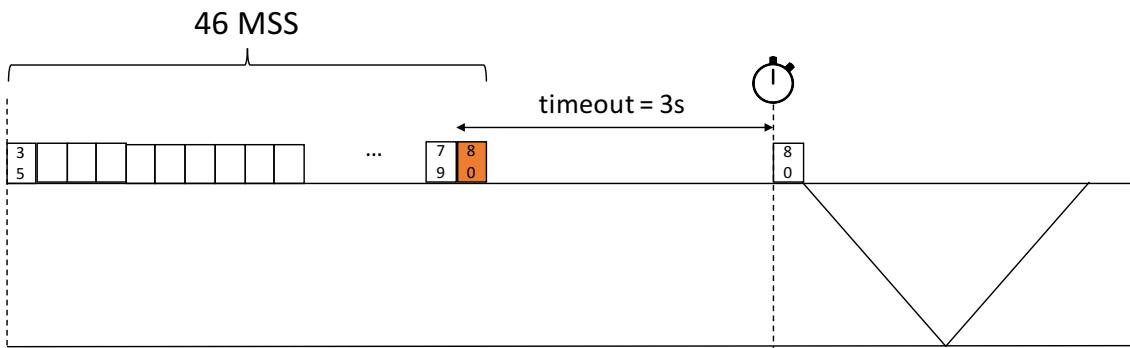
- Trasmissione continua dei restanti 46 segmenti: $46T + 2\tau$

Tempo totale:

$$T_{tot} = 2\tau + 6RTT + 46T + 2\tau = 132 \text{ ms}$$

c)

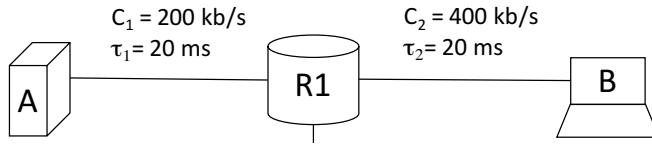
L'ultimo segmento viene ritrasmesso dopo lo scadere del timeout.



e quindi

$$T_{tot} = 2\tau + 6RTT + 45T + T_{out} + T + 2\tau = 3.132 \text{ s}$$

Esercizio 4.26



Una connessione TCP tra l'host A e l'host B nella rete in figura è caratterizzata dai seguenti parametri:

- Link bidirezionali e simmetrici
- $\text{MSS} = 200 \text{ byte}$
- Lunghezza header complessivo (tutti i livelli), $H = 50 \text{ byte}$
- Lunghezza ACK e segmenti di apertura, $L_{ACK} = 250 \text{ byte}$
- $\text{RCWND} = 1000 \text{ byte}$, $\text{SSTHRESH} = 1600 \text{ byte}$

a) Si calcoli il tempo necessario a trasferire un file di dimensione $F = 5 \text{ [kbyte]}$ (dall'apertura della connessione alla ricezione dell'ultimo ACK)

b) Si indichi il rate medio di trasferimento del file da A a B

$$\text{RCWND} = 1000 \text{ [byte]} / 200 \text{ [byte]} = 5 \text{ MSS}$$

$$\text{SSTHRESH} = 1600 \text{ [byte]} / 200 \text{ [byte]} = 8 \text{ MSS}$$

$$\text{File} = 5000 \text{ [byte]} / 200 \text{ [byte]} = 25 \text{ MSS}$$

$$L = \text{MSS} + H = 250 \text{ byte}$$

$$T_1 = \frac{L}{C_1} = \frac{250}{200} \cdot 8 / 200000 = 10 \text{ ms}$$

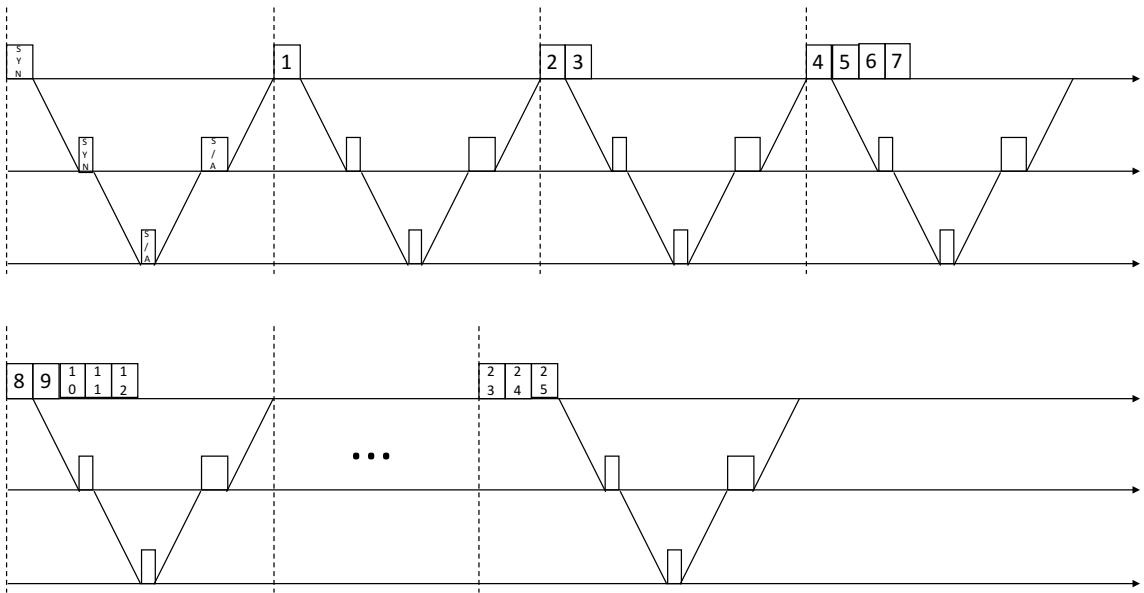
$$T_2 = \frac{L}{C_2} = \frac{250}{400} \cdot 8 / 400000 = 5 \text{ ms}$$

$$T_1^{ACK} = T_1; T_2^{ACK} = T_2$$

$$\text{RTT} = 2(T_1 + \tau_1 + T_2 + \tau_2) = 110 \text{ ms}$$

$$W_c = \text{RTT}/T_1 = 11 \text{ MSS}$$

$$T_{\text{setup}} = \text{RTT} = 110 \text{ ms}$$



Dopo i segmenti di apertura della connessione, il TCP parte in modalità Slow Start. Tuttavia, prima di raggiungere la Ssthresh, la finestra è limitata dalla RCWND a 5MSS. Quindi, una volta raggiunto tale valore, la finestra non aumenterà. Inoltre, dato che $W_c > RCWND$, la trasmissione non sarà mai continua.

Il tempo totale di trasferimento è dato da:

$$T_{tot} = T_{setup} + 6RTT + 2T_1 + RTT = 8RTT + 2T_1 = 900 [ms]$$

Il rate medio di trasferimento è dato da:

$$R_{medio} = \frac{F}{T_{tot}} = 5000 * \frac{8}{0.9} = 44,44 [kb/s]$$

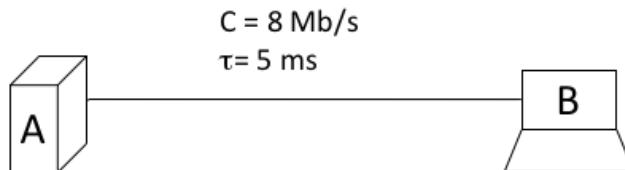
Esercizio 4.27

Una connessione TCP tra l'*host A* e l'*host B* è caratterizzata dai seguenti parametri:

- Lunghezze di *header* e *ack* trascurabili;
- *link* bidirezionali;
- $RCWND = 9 \text{ MSS}$
- $SSTHRESH = 8 \text{ MSS}$
- $MSS = 1 \text{ [kbyte]}$
- Ritardo di propagazione, $\tau = 5 \text{ [ms]}$
- Valore del *Time-Out* = $2RTT$ calcolato a partire dal primo bit di ogni UI inviata

Si risponda ai seguenti quesiti:

- Dire se la trasmissione sul link diventa mai continua; in caso affermativo, trovare il tempo oltre cui la trasmissione sul link diventa continua;
- Trovare il tempo di trasferimento di un file di 60 [kbyte] da A a B;
- Si consideri che il penultimo segmento in trasmissione venga perso. Ripetere il punto b) nel caso in cui il protocollo TCP accetti anche i pacchetti fuori sequenza.



a)

$$RTT = T + 2\tau = \frac{8000 \text{ [bit]}}{8 \left[\frac{\text{Mb}}{\text{s}} \right]} + 10 \text{ [ms]} = 11 \text{ [ms]}$$

$$W_{cont} = \frac{RTT}{T} = \frac{11}{1} = 11 \text{ MSS}$$

Ma abbiamo una *RCWND* di 9 MSS, dunque la trasmissione non sarà mai continua.

b)

$$T_{setup} = 2\tau = 10 \text{ [ms]}$$

$$N_{MSS} = \frac{60 \text{ [kB]}}{1 \text{ [kB]}} = 60$$

$$T_{tot} = T_{setup} + 9 RTT(1 - 2 - 4 - 8 - 9 - 9 - 9 - 9 - 9) + 8 T = 117 \text{ [ms]}$$

Tempo dall'apertura della connessione alla ricezione dell'ultimo ACK

c)

All'invio del primo bit del 59° pacchetto, comincia il conteggio del timeout. Dopo 2RTT, poiché non è arrivato il riscontro del 59° pacchetto, scatta il timeout. Se il TCP accetta anche i segmenti fuori sequenza, solo il penultimo pacchetto viene ritrasmesso.

$$T_{tot} = T_{setup} + 8 RTT + 7 T + \text{timeout} + RTT = T_{setup} + 12 RTT + 7 T = 138 \text{ [ms]}$$

(per ipotesi il timeout viene avviato all'inizio della trasmissione di ogni UI).

Esercizio 4.28

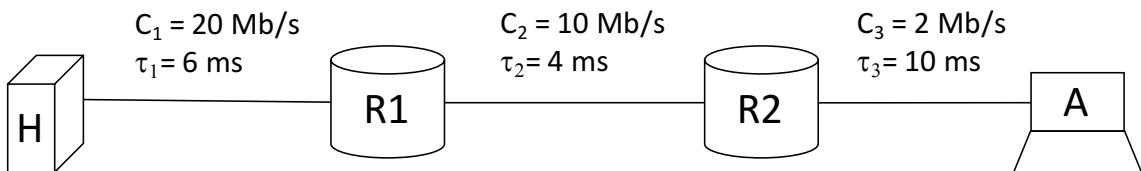
La connessione TCP tra l'host A e l'host H (riportata in figura) è caratterizzata dai seguenti parametri:

- lunghezze di header, ack e segmenti di apertura trascurabili;
- link bidirezionali simmetrici;
- MSS = 1250 Byte,;
- STHRESH = 10000 Byte.

a) Tenendo conto delle diverse velocità di trasmissione dei link in figura, si calcoli la lunghezza della finestra Wc (calcolata in numero di MSS) che permette la trasmissione continua.

b) Si calcoli il tempo necessario (dall'istante t=0 di inizio dell'apertura della connessione alla ricezione dell'ultimo ack da parte di A) per trasferire un file di 50 KByte da A a H, rappresentando chiaramente il trasferimento del file in un diagramma spazio/temporale (anche in modo sintetico).

c) Si ripeta il calcolo del tempo necessario per trasferire il file da A a H, assumendo che tutti i segmenti della quarta finestra vadano persi e il timeout corrispondente sia Ttimeout=100 ms (si assuma che il conteggio del timeout inizi all'istante di invio del primo bit di ogni segmento e che i pacchetti fuori sequenza non vengano memorizzati).



Soluzione

$$MSS = 1250 \text{ Byte} = 10.000 \text{ bit} \quad STHRESH = 1000 \text{ Byte} = 8 \text{ MSS}$$

$$\text{File } F = 50 \text{ kByte} = 40 \text{ MSS}$$

$$T_l = 0,5 \text{ ms} \quad T_2 = 1 \text{ ms} \quad T_3 = 5 \text{ ms.} \quad T_{timeout} = 100 \text{ ms}$$

$$RTT = 2T_l + 2T_2 + 2T_3 + 2\tau_l + 2\tau_2 + 2\tau_3 = 46,5 \text{ ms}$$

Il collo di bottiglia è rappresentato dal tratto più lento R2->A

$$\text{a)} \quad W_c = 10 \text{ poiché } RTT / T_3 = 9,3 \text{ trasmissione continua}$$

$$\text{b)} \quad T_{open} = 2\tau_l + 2\tau_2 + 2\tau_3 = 40 \text{ ms}$$

$$T_{tot} = T_{open} + 5 RTT(1 - 2 - 4 - 8 - 9) + 15 T_3 + RTT = 395 \text{ ms}$$

Infatti File $F = 40 \text{ MSS}$

Apertura connessione: T_{open}

Slow start: (1) - (2) - (4) - (8): $4RTT$

Congestion avoidance: (9) + 16 MSS finali (continui) e riscontro -> quindi $RTT + 15T_3 + RTT$

- d) Si perdono i pacchetti della quarta finestra, cioè 8,9,10,11,12,13,14 e 15. Finita la trasmissione degli 8 pacchetti della quarta finestra si attende l'ACK e trascorre tutto il time-

out. Scaduto il time-out si dimezza SSTHRESH -> nuovo SSTHRESH = 8MSS/2 = 4MSS e si riparte con la finestra dimensionata a 1MSS

Si hanno quindi 6RTT (per la trasmissione dei pacchetti 8-14 in modalità slow-start e dei pacchetti 15-32 in modalità congestion-avoidance) + 7T3 + RTT (per la trasmissione degli ultimi 8 pacchetti 33-40 + il riscontro)

$$T_{\text{tot}} =$$

$$= T_{\text{open}} + 3 \text{ RTT}(1 - 2 - 4) + T_{\text{timeout}} + 6 \text{ RTT}(1 - 2 - 4 - 5 - 6 - 7) + 7 \text{ T3} + \text{RTT} = \\ = 640 \text{ ms}$$

5. Esercizi sul Livello di RETE

Esercizio 5.1

Nel sistema di indirizzamento IP, si consideri l'indirizzo della rete 141.12.0.0/16.

Quante sottoreti /26 si possono ricavare dalla rete base (supponendo che anche gli indirizzi di sottorete con tutti 0 e tutti 1 siano associabili a sottoreti effettive)?

Scrivere in formato decimale puntato la maschera (netmask) delle sottoreti /26

- Numero sottoreti /26: $2^{26-16} = 2^{10} = 1024$
- Ultimo ottetto con i primi due bit a 1: 11111111.11111111.11111111.11000000 → 255.255.255.192

Esercizio 5.2

Nel sistema di indirizzamento IP, si consideri l'indirizzo della rete 129.16.0.0/16.

- a) Quante sottoreti /22 possono essere ricavate dalla rete base, assumendo che un identificatore di subnet può anche essere costituito da tutti 0 o tutti 1?
- b) Si partizioni ulteriormente la sottorete 129.16.248.0/22 in N sottoreti $/n$ che permettano di indirizzare almeno 64 host ognuna (si considerino anche 2 indirizzi riservati per sottorete).
 - Qual è la lunghezza del prefisso di sottorete n ? Quante sottoreti N_n con prefisso $/n$ è possibile creare?
 - Si scriva in formato decimale la maschera (netmask) delle sottoreti $/n$

a) $N_{22} = 2^{22-16} = 2^6 = 64$

b)

64 hostID +2 = 66 indirizzi: $2^6 = 64 < 66$ NO /. $2^7 = 128 > 66 \rightarrow$ occorrono almeno 7 bit per hostID

- Qual è la lunghezza del prefisso di sottorete n ? Quante sottoreti N_n con prefisso $/n$ è possibile creare?

$$n = 25 = 32 - 7$$

$$N_n = 2^{25-22} = 2^3 = 8$$

- Netmask: 255.255.255.10000000 \rightarrow 255.255.255.128

Esercizio 5.3

Nel sistema di indirizzamento IP, si consideri l'indirizzo della rete 105.0.0.0/8.

- a) Quante sottoreti /11 possono essere ricavate dalla rete base, assumendo che un identificatore di subnet può anche essere costituito da tutti 0 o tutti 1?
- b) Si partizioni ulteriormente la sottorete 105.160.0.0/11 in N sottoreti $/n$ che permettano di indirizzare almeno 512 host ognuna (si considerino anche 2 indirizzi riservati per sottorete).
 - Qual è la lunghezza del prefisso di sottorete n ? Quante sottoreti N_n con prefisso $/n$ è possibile creare?
 - Si scriva in formato decimale la maschera (netmask) delle sottoreti $/n$

a) $N_{11} = 2^{11-8} = 2^3 = 8$

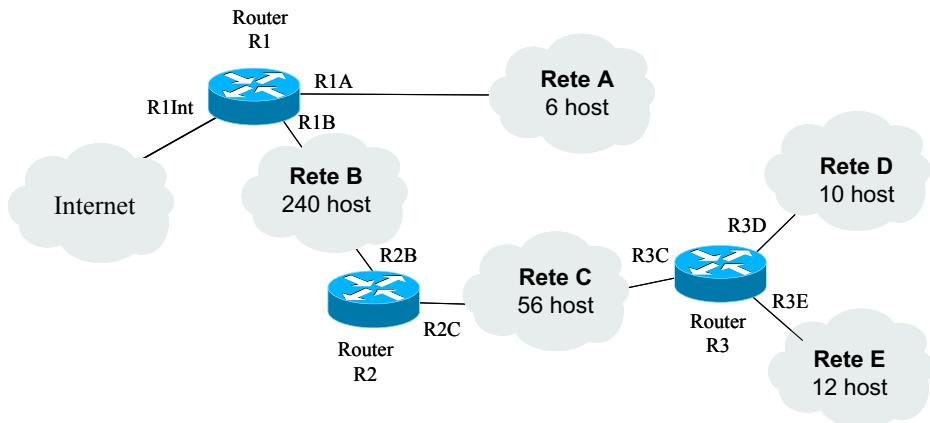
b)

512 HostID + 2 = 514 indirizzi $2^9 = 512 < 514$ NO /. $2^{10} = 1024 > 514 \Rightarrow$ occorrono almeno 10 bit per host-ID

- $n = 22 = 32 - 10$ $N_n = 2^{22-11} = 2^{11} = 2048$
- Netmask: 255.255.11111100.0 \Rightarrow 255.255.252.0

Esercizio 5.4

Si consideri la seguente configurazione di rete di una società Alfa che vuole connettersi alla rete Internet.



La società Alfa riceve la disponibilità di indirizzi IP da un Internet Service Provider (ISP) che dispone del blocco di indirizzi 87.10.0.0/16. Di questi indirizzi l'ISP ne ha già assegnati i primi 2048 ad altri clienti. Si chiede di

- Assegnare gli indirizzi alle 5 reti da realizzare (A, B, C, D, E) minimizzando gli indirizzi che risulteranno inutilizzati alla fine dell'assegnazione degli indirizzi richiesti. I vincoli da rispettare sono i seguenti:
 - A partire dall'indirizzo IP libero più basso nel blocco a disposizione, assegnare ordinatamente blocchi di indirizzi IP crescenti e adiacenti a reti con identificatore crescente, a parità di maschera di rete (netmask, /n) adottata,

- Cerchiamo la soluzione migliore: blocchi di indirizzi tutti adiacenti tra loro.
- Il blocco di indirizzi dell'ISP 87.10.0.0/16 ha netmask 255.255.0.0
- Per rappresentare i primi 2048 indirizzi sono necessari 11 bit ($2048=2^{11}$) e $32-11=21$. Pertanto a questo primo gruppo di clienti è assegnato l'indirizzo 87.10.0.0/21. Possiamo vedere meglio questo fatto osservando la netmask: 11111111.11111111.11111 000.00000000.
- A questo punto dobbiamo assegnare un indirizzo ad ogni rete da A ad E: per ogni rete è necessario aggiungere 2 indirizzi riservati e 1 indirizzo per ogni interfaccia router.

Rete	Indirizzi necessari	Potenza di 2 superiore più vicina
A	$6+3=9$	16
B	$240+4=244$	256
C	$56+4=60$	64
D	$10+3=13$	16
E	$12+3=15$	16

Rete	Indirizzo decimale	Indirizzo binario	/n
A	87.10.9.64	01010111.00001010.00001001.01 <u>0000000</u>	/28
B	87.10.8.0	01010111.00001010.00001000. <u>00000000</u>	/24
C	87.10.9.0	01010111.00001010.00001001.00 <u>000000</u>	/26
D	87.10.9.80	01010111.00001010.00001001.01 <u>010000</u>	/28
E	87.10.9.96	01010111.00001010.00001001.01 <u>100000</u>	/28

Esercizio 5.5

Ad un'organizzazione viene assegnato lo spazio di indirizzamento 131.175.0.0/21. Tale organizzazione ha la necessità di definire le seguenti sottoreti: 1 sottorette con almeno 1000 *host*, 3 sottoreti con almeno 220 *host* ciascuna, 3 sottoreti con almeno 56 *host* ciascuna, 4 sottoreti con esattamente 2 *host*.

Definire un piano di partizionamento dello spazio di indirizzamento congruente con le specifiche sopra indicando per ogni sottorette l'indirizzo IP di rete.

Lo spazio di indirizzamento originale comprende 11 bit nella parte di *host*. La sottorette più grande che deve essere definita è quella con 1000 *host*. Per supportare 1000 *host* servono 10 bit nel campo di *host* ($2^{10}=1024$). Si può quindi allungare la *netmask* originale di 1 bit (/22) definendo così spazio per due sottoreti ciascuna in grado di supportare 1022 *host* (1024 meno i due indirizzi riservati).

Uno dei due spazi di indirizzamento così definiti può essere assegnato alla sottorette con 1000 *host*:

131.175.0.0/22 rete con almeno 1000 *host*

L'altro spazio di indirizzamento 131.175.4.0/22 può essere ulteriormente suddiviso.

Le sottoreti più grandi a questo punto sono quelle con 220 *host*. Per supportare 220 *host* servono 8 bit nel campo di *host* ($2^8=256$). Si può quindi allungare la *netmask* originale di 2 bit (/24) definendo così spazio per quattro sottoreti ciascuna in grado di supportare 254 *host* (256 meno i due indirizzi riservati).

Tre dei quattro spazi di indirizzamento così definiti possono essere assegnati alle sottoreti con 220 *host*:

131.175.4.0/24 rete con almeno 220 *host*

131.175.5.0/24 rete con almeno 220 *host*

131.175.6.0/24 rete con almeno 220 *host*

L'altro spazio di indirizzamento 131.175.7.0/24 può essere ulteriormente suddiviso.

Le sottoreti più grandi a questo punto sono quelle con 56 *host*. Per supportare 56 *host* servono 8 bit nel campo di *host* ($2^6=64$). Si può quindi allungare la *netmask* originale di 2 bit (/26) definendo così spazio per quattro sottoreti ciascuna in grado di supportare 62 *host* (64 meno i due indirizzi riservati).

Tre dei quattro spazi di indirizzamento così definiti possono essere assegnati alle sottoreti con 56 *host*:

131.175.7.0/26 rete con almeno 56 *host*

131.175.7.64/26 rete con almeno 56 *host*

131.175.7.128/26 rete con almeno 56 *host*

L'altro spazio di indirizzamento 131.175.7.192/26 può essere ulteriormente suddiviso.

Rimangono a questo punto solo le sottoreti con 2 *host*. Per supportare 2 *host* servono 2 bit nel campo di *host* ($2^2=4$). Si può quindi allungare la *netmask* originale di 4 bit (/30) definendo così spazio per sedici sottoreti ciascuna in grado di supportare 2 *host* (4 meno i due indirizzi riservati).

Quattro dei sedici spazi di indirizzamento così definiti possono essere assegnati alle sottoreti con 2 *host*:

131.175.7.192/30 rete con esattamente 2 *host*

131.175.7.196/30 rete con esattamente 2 *host*

131.175.7.200/30 rete con esattamente 2 *host*

131.175.7.204/30 rete con esattamente 2 *host*

Esercizio 5.6

Ad un'organizzazione è assegnato lo spazio d'indirizzamento 195.123.224.0/21. Da questo gruppo d'indirizzi occorre ricavare le seguenti sottoreti:

- 1 sottorete con almeno 500 indirizzi di *host* disponibili
- 1 sottorete con almeno 210 indirizzi di *host* disponibili
- 3 sottoreti con almeno 30 indirizzi di *host* disponibili
- 4 sottoreti con almeno due indirizzi di *host* disponibili.

Pianificare il partizionamento dello spazio d'indirizzamento dato specificando per ciascuna delle sottoreti sopra elencate:

- indirizzo in formato decimale e *netmask*
 - numero di utenti indirizzabili
-

Lo spazio di indirizzamento originale comprende 11 bit nella parte di *host*. La sottorete più grande che deve essere definita è quella con 500 *host*. Per supportare 500 *host* servono 9 bit nel campo di *host* ($2^9=512$). Si può quindi allungare la *netmask* originale di 2 bit (/23) definendo così spazio per quattro sottoreti ciascuna in grado di supportare 510 *host* (510 meno i due indirizzi speciali).

Uno degli spazi di indirizzamento così definiti può essere assegnato alla sottorete con 500 *host*:

195.123.224.0/23, rete con 510 *Host* massimo

Gli altri tre spazi di indirizzamento 195.123.226.0/23, 195.123.228.0/23, 195.123.230.0/23 possono essere ulteriormente suddivisi.

La sottorete più grande a questo punto è quella con 210 *host*. Per supportare 210 *host* servono 8 bit nel campo di *host* ($2^8=256$). Si può quindi allungare la *netmask* originale di 1 bit (/24) definendo così spazio per due sottoreti ciascuna in grado di supportare 254 *host* (256 meno i due indirizzi speciali).

Uno dei due spazi di indirizzamento così definiti può essere assegnato alla sottorete con 210 *host*

195.123.226.0/24, rete con 254 *host* massimo

L'altro spazio di indirizzamento 195.123.227.0/24 può essere ulteriormente suddiviso.

Le sottoreti più grandi a questo punto sono quelle con 30 *host*. Per supportare 30 *host* servono 5 bit nel campo di *host* ($2^5=32$). Si può quindi allungare la *netmask* originale di 3 bit (/27) definendo così spazio per otto sottoreti ciascuna in grado di supportare 30 *host* (32 meno i due indirizzi speciali).

Tre degli otto spazi di indirizzamento così definiti possono essere assegnati alle sottoreti con 30 *host*:

195.123.227.0/27, rete con 30 *host*
195.123.227.32/27, rete con 30 *host*
195.123.227.64/27, rete con 30 *host*

Gli altri cinque spazi di indirizzamento 195.123.227.128/27, 195.123.227.96/27, 195.123.227.192/27, 195.123.227.160/27, 195.123.227.224/27 possono essere ulteriormente suddivisi.

Rimangono a questo punto solo le sottoreti con 2 *host*. Per supportare 2 *host* servono 2 bit nel campo di *host* ($2^2=4$). Si può quindi considerare uno degli spazi di indirizzamento sopra definiti ed allungare la *netmask* originale di 3 bit (/30) definendo così spazio per otto sottoreti ciascuna in grado di supportare 2 *host* (4 meno i due indirizzi speciali).

Quattro degli otto spazi di indirizzamento così definiti possono essere assegnati alle sottoreti con 2 *host*:

195.123.227.128/30, rete con 2 *host*
195.123.227.132/30, rete con 2 *host*
195.123.227.136/30, rete con 2 *host*
195.123.227.140/30, rete con 2 *host*

La soluzione proposta non è l'unica, essendo il numero di indirizzi disponibile molto maggiore rispetto alle dimensioni delle sottoreti IP da definire.

Esercizio 5.7

Un *router* ha la seguente tabella di *routing*. E' possibile ridurre la dimensione della tabella di *routing*? Se sì, come?

Destinazione	Netmask	Next Hop
131.175.132.0	255.255.255.0	131.123.124.125
131.175.21.0	255.255.255.0	131.124.123.121
131.175.20.0	255.255.255.0	131.124.123.121
131.175.133.0	255.255.255.0	131.123.124.125
131.175.135.0	255.255.255.0	131.123.124.128
0.0.0.0	0.0.0.0	131.123.124.126

La seconda e la terza sottorete in tabella hanno indirizzi IP contigui (differiscono per l'ultimo *bit* del terzo *byte*) e hanno *next hop* in comune.

La prima, la quarta e la quinta sottorete hanno indirizzi IP uguali fino al terzultimo *bit* del terzo *byte*; la prima e la quarta sottorete hanno anche *next hop* comune. E' possibile ridurre la tabella di *routing* come segue.

Destinazione	Netmask	Next Hop
131.175.20.0	255.255.254.0	131.124.123.121
131.175.132.0	255.255.254.0	131.123.124.125
131.175.135.0	255.255.255.0	131.123.124.128
0.0.0.0	0.0.0.0	131.123.124.126

In generale le regole da seguire per l'aggregazione sono:

1. Si possono aggregare gruppi di reti contigue che hanno lo stesso next-hop. Il gruppo è sostituito da un'unica riga che contiene l'aggregato, ovvero la *supernet*, ed è ottenuto accorciando la *netmask*.
2. Si possono eliminare tutte le reti con next-hop pari alla rotta di default.

Esercizio 5.8

Un *router* ha la seguente tabella di *routing*. E' possibile ridurre la dimensione della tabella di *routing*? Se sì, come?

<i>Destinazione</i>	<i>Netmask</i>	<i>Next Hop</i>
131.175.132.0	255.255.255.0	131.123.124.125
131.175.21.0	255.255.255.0	131.124.123.121
131.175.20.0	255.255.255.0	131.124.123.121
131.175.133.0	255.255.255.0	131.123.124.125
131.175.134.0	255.255.255.0	131.123.124.130
131.175.135.0	255.255.255.0	131.123.124.125
131.175.50.0	255.255.254.0	131.123.124.126
0.0.0.0	0.0.0.0	131.123.124.126

La seconda e la terza sottorete in tabella hanno indirizzi IP contigui (differiscono per l'ultimo *bit* del terzo *byte*) e hanno *next hop* in comune. Possono dunque essere aggregate:

131.175.20.0 255.255.254.0 131.124.123.121

La prima, la quarta, la quinta e la sesta sottorete hanno indirizzi IP uguali fino al terzultimo *bit* del terzo *byte*; la prima, la quarta e la sesta sottorete hanno anche *next hop* comune.

131.175.132.0 255.255.252.0 131.123.124.125
131.175.134.0 255.255.255.0 131.123.124.130

La settima sottorete ha lo stesso next-hop della route di default e quindi può essere eliminata.

E' dunque possibile ridurre la tabella di *routing* come segue:

<i>Destinazione</i>	<i>Netmask</i>	<i>Next Hop</i>
131.175.20.0	255.255.254.0	131.124.123.121
131.175.132.0	255.255.252.0	131.123.124.125

131.175.134.0	255.255.255.0	131.123.124.130
0.0.0.0	0.0.0.0	131.123.124.126

In generale le regole da seguire per l'aggregazione sono:

3. Si possono aggregare gruppi di reti contigue che hanno lo stesso next-hop. Il gruppo è sostituito da un'unica riga che contiene l'aggregato, ovvero la *supernet*, ed è ottenuto accorciando la *netmask*.
4. Si possono eliminare tutte le reti con next-hop pari alla rotta di default.

Esercizio 5.9

Un router ha la seguente tabella di *routing* e la seguente configurazione delle interfacce.

eth0: 192.170.123.4, 255.255.255.0

eth1: 192.170.124.4, 255.255.255.0

Network	Netmask	Next Hop
191.138.112.0	255.255.240.0	192.170.123.1
191.138.96.0	255.255.224.0	192.170.124.1
191.138.0.0	255.255.0.0	192.170.123.2
191.138.160.0	255.255.224.0	192.170.124.2
0.0.0.0	0.0.0.0	192.170.123.3

Il router riceve dei pacchetti i cui indirizzi di destinazione sono:

191.138.163.13

191.138.113.32

131.175.123.244

255.255.255.255

192.170.123.255 proveniente dall'interfaccia *eth1*

Indicare come avviene l'inoltro di ciascuno dei pacchetti sopra specificando il tipo di inoltro (diretto o indiretto), il *Next hop* e la riga della tabella di *routing* corrispondente in caso di inoltro indiretto e l'interfaccia d'uscita in caso di inoltro diretto.

Il procedimento generale è il seguente:

- I. Il router controlla se il pacchetto IP è destinato al router stesso (IP di destinazione uguale ad uno degli IP del router). Se sì, il pacchetto IP è passato ai livelli superiori.
- II. Altrimenti (pacchetto IP non destinato al router), il router controlla se l'indirizzo IP di destinazione appartiene ad una delle sottoreti a cui il router è direttamente collegato; per fare questo, il router effettua un AND bit a bit tra l'indirizzo IP di destinazione del pacchetto e la *netmask* di ognuna delle sue interfacce locali; se il risultato dell'AND bit a bit è uguale all'indirizzo di una sottorete locale a cui il router è collegato, il pacchetto IP viene inoltrato in modo diretto attraverso l'interfaccia corrispondente.
- III. Se il confronto precedente tra indirizzo IP di destinazione ed indirizzi delle sottoreti a cui il router è direttamente collegato da esito negativo, il pacchetto IP è da inoltrare in modo indiretto (inviarlo ad un altro router). Per fare questo, il router effettua l'AND bit a bit tra indirizzo IP di destinazione nel pacchetto e gli indirizzi di sottorete memorizzati nella tabella di *routing*; per tutte le sottoreti (righe della tabella di *routing*) per cui il confronto è positivo, il router inoltra il pacchetto IP verso il *next hop* che corrisponde nella tabella di *routing* alla sottorete di destinazione con la *netmask* più lunga.

Applicando questo procedimento nel caso dell'esercizio si ha:

Pacchetto 1: 191.138.163.13

E' facile verificare che il pacchetto non è indirizzato ad un *host* presente nelle sottoreti direttamente collegate al router. Il router, quindi, deve procedere all'inoltro indiretto attraverso la tabella di *routing*.

191.138.163.13 AND

1. /20 = 191.138.160.0
2. /19 = 191.138.160.0
3. /16 = 191.138.0.0 ————— OK
4. /19 = 191.138.160.0 ————— OK
5. /0 = 0.0.0.0

	Network	Netmask	Next Hop
1	191.138.112.0	255.255.240.0	192.170.123.1
2	191.138.96.0	255.255.224.0	192.170.124.1
3*	191.138.0.0	255.255.0.0	192.170.123.2
4	191.138.160.0	255.255.224.0	192.170.124.2
5	0.0.0.0	0.0.0.0	192.170.123.3

Inoltro indiretto a
192.170.124.2

Interfaccia uscita eth1

Pacchetto 2: 191.138.113.32

E' facile verificare che il pacchetto non è indirizzato ad un *host* presente nelle sottoreti direttamente collegate al router. Il router, quindi, deve procedere all'inoltro indiretto attraverso la tabella di *routing*.

191.138.113.32 AND

1. /20 = 191.138.112.0 ————— OK
2. /19 = 191.138.96.0 ————— OK
3. /16 = 191.138.0.0 ————— OK
4. /19 = 191.138.112.0 ————— OK
5. /0 = 0.0.0.0

	Network	Netmask	Next Hop
1	191.138.112.0	255.255.240.0	192.170.123.1
2	191.138.96.0	255.255.224.0	192.170.124.1
3	191.138.0.0	255.255.0.0	192.170.123.2
4	191.138.160.0	255.255.224.0	192.170.124.2
5	0.0.0.0	0.0.0.0	192.170.123.3

Inoltro indiretto verso
192.170.123.1

Interfaccia uscita eth0

Pacchetto 3: 131.175.123.244

E' facile verificare che il pacchetto non è indirizzato ad un *host* presente nelle sottoreti direttamente collegate al router. Il router, quindi, deve procedere all'inoltro indiretto attraverso la tabella di *routing*.

131.175.123.244 AND

1. /20 No match
2. /19 No match
3. /16 No match
4. /19 No match
5. /0 = 0.0.0.0 ————— OK

	Network	Netmask	Next Hop
1	191.138.112.0	255.255.240.0	192.170.123.1
2	191.138.96.0	255.255.224.0	192.170.124.1
3	191.138.0.0	255.255.0.0	192.170.123.2
4	191.138.160.0	255.255.224.0	192.170.124.2
5*	0.0.0.0	0.0.0.0	192.170.123.3

Inoltro indiretto a
192.170.123.3

Interfaccia uscita eth0

Pacchetto 4: 192.170.123.255 proveniente dall'interfaccia eth1

L'indirizzo di destinazione appartiene alla sottorete IP a cui il router è collegato tramite *eth0*. Il router procede quindi all'inoltro diretto attraverso *eth0*.

Esercizio 5.10

Un router ha la seguente tabella di *routing* e la seguente configurazione delle interfacce. Dire come avviene l'inoltro per pacchetti con indirizzo di destinazione:

131.17.123.88

131.56.78.4

190.78.90.2

network	netmask	first hop
131.175.21.0	255.255.255.0	131.17.123.254
131.175.16.0	255.255.255.0	131.17.78.254
131.56.0.0	255.255.0.0	131.17.15.254
131.155.0.0	255.255.0.0	131.17.15.254
0.0.0.0	0.0.0.0	131.17.123.254

interface eth0

IP address	131.17.123.1
netmask	255.255.255.0

interface eth1

IP address	131.17.78.1
netmask	255.255.255.0

interface eth2

IP address	131.17.15.12
netmask	255.255.255.0

131.17.123.88: viene inoltrato sull'interfaccia *eth0*

131.56.78.4: viene inoltrato al *next hop* 131.17.15.254

190.78.90.2: viene inoltrato al *next hop* 131.17.123.254

Esercizio 5.11

Un router ha la seguente configurazione delle interfacce e la seguente tabella di *routing*. Il livello 3 del router riceve gli 5 pacchetti riportati di seguito, per ciascuno dei quali vengono riportati l'indirizzo IP di destinazione e l'interfaccia attraverso cui il router riceve il pacchetto. Si chiede di indicare il comportamento del router per ciascuno dei pacchetti specificando l'indirizzo IP del *next hop* e se l'inoltro è di tipo diretto o indiretto.

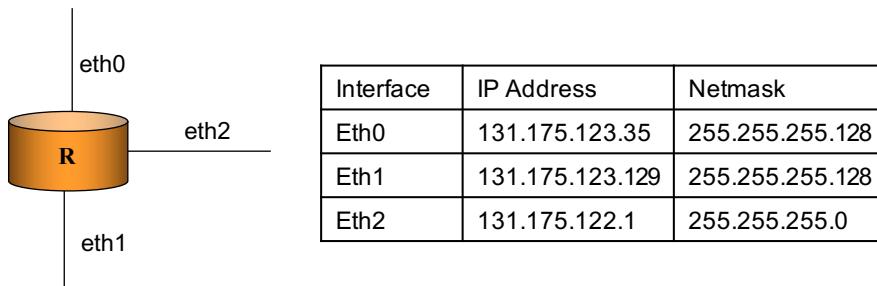


Tabella di routing

Network	Netmask	Next Hop
130.170.0.0	255.255.0.0	131.175.123.1
130.171.0.0	255.255.0.0	131.175.123.132
130.171.4.0	255.255.252.0	131.175.122.2
130.170.10.0	255.255.254.0	131.175.122.3
0.0.0.0	0.0.0.0	131.175.123.3

Pacchetti ricevuti

- A. 131.175.123.64 da eth2
- B. 131.175.123.255 da eth0
- C. 130.170.132.240 da eth1
- D. 130.170.11.64 da eth1
- E. 130.171.5.125 da eth1

-
- A. 131.175.123.64 da eth2, Inoltro diretto attraverso *eth0*
 - B. 131.175.123.132 da *eth0*, Inoltro diretto attraverso *eth1*
 - C. 130.170.132.240 da *eth1*, Inoltro indiretto, prima linea *routing table*, *NextHop*: 131.175.123.1, Interfaccia uscita: *Eth0*
 - D. 130.170.11.64 da *eth1*, Inoltro indiretto, quarta linea *routing table*, *NextHop*: 131.175.122.3, Interfaccia uscita: Eth2
 - E. 130.171.5.125 da *eth1*, Inoltro indiretto, terza linea *routing table*, *NextHop*: 131.175.122.2, Interfaccia uscita: Eth2

Esercizio 5.12

Un router ha le seguenti interfacce e tabella di *routing*. Il router riceve i pacchetti con destinazione e dimensioni indicati sotto. Si dica per ciascuno di essi come si comporta il router: inoltro diretto o indiretto, interfaccia di uscita, riga della tabella.

Eth0 - Address: 131.175.21.254 – *Netmask*: 255.255.255.128

Eth1: Address: 131.175.20.126 – *Netmask*: 255.255.255.128

Network	<i>Netmask</i>	<i>Next-hop</i>
131.175.70.0	255.255.254.0	131.175.21.133
131.175.71.128	255.255.255.128	131.175.21.145
131.175.72.0	255.255.254.0	131.175.20.5
0.0.0.0	0.0.0.0	131.175.20.1

131.175.71.72

131.175.73.12

131.175.20.12 da *Eth0*

131.175.71.72

Inoltro indiretto riga 1

131.175.73.12

Inoltro indiretto riga 3

131.175.20.12

Inoltro diretto su *Eth1*

Esercizio 5.13

Un router è caratterizzato dalla seguente configurazione delle interfacce locali e della seguente tabella di *routing*. Il router riceve i seguenti pacchetti. Per ciascuno dei pacchetti indicati di seguito (caratterizzati da interfaccia di provenienza, indirizzo di destinazione) dire come si comporta il router specificando se procede con inoltro diretto o indiretto. Indicare chiaramente l'interfaccia di inoltro e la riga della tabella di *routing* “scelta” per l'inoltro indiretto.

Eth0: 131.175.192.1, 255.255.192.0

Eth1: 131.175.128.1, 255.255.192.0

WiFi0: 128.10.10.1, 255.255.255.0

Destinazione	Netmask	Next Hop
131.175.32.0	255.255.224.0	128.10.10.123
131.175.64.0	255.255.192.0	131.175.220.14
131.175.144.0	255.255.240	128.10.10.123
131.0.0.0	255.0.0.0	128.10.10.123
0.0.0.0	0.0.0.0	131.175.145.13

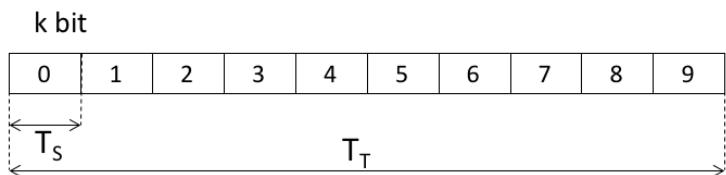
- 1) 175.123.12.123 da WiFi0
- 2) 131.175.64.12 da *eth1*
- 3) 131.175.65.120 da *eth0*
- 4) 131.175.228.13 da *eth1*
- 5) 131.175.191.255 da *eth0*

-
- 1. Inoltro indiretto, ultima riga tabella di *routing*, attraverso *eth1*
 - 2. Inoltro indiretto, seconda riga tabella di *routing*, attraverso *eth0*
 - 3. Inoltro indiretto, seconda riga tabella di *routing*, attraverso *eth0*
 - 4. Inoltro diretto attraverso *eth0*
 - 5. Inoltro diretto attraverso *eth1*

6. Esercizi sul Livello di LINEA

Esercizio 6.1

Un sistema di multiplazione TDM presenta una trama di $N=10$ slot; in ciascuno slot vengono trasmessi $k=128$ bit. Se il sistema è usato per multiplare 10 canali ciascuno a $V=64$ kbit/s, si dica quale è la velocità W , la durata T_T della trama di multiplazione e quella T_s dello slot.



La durata della trama si può calcolare imponendo che se si assegna uno slot per trama ad un tributario, il rate equivalente del canale così definito sia uguale al rate di ingresso del tributario V .

$$T_T = \frac{k}{V} = \frac{128}{64 \cdot 10^3} = 2 \text{ ms}$$

Il sistema di multiplo deve avere velocità (rate) sufficiente per supportare tutti gli N tributari, quindi:

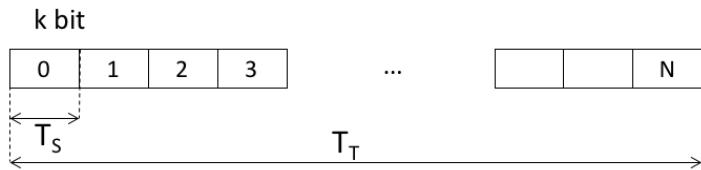
$$W = V \cdot N = 64 \cdot 10^3 \cdot 10 = 640 \text{ kb/s}$$

Il tempo di slot è definito come il tempo necessario per inviare tutti i bit che compongono lo slot, k , ad una velocità pari a W ; quindi:

$$T_s = \frac{k}{W} = \frac{128}{640 \cdot 10^3} = 200 \mu\text{s}$$

Esercizio 6.2

Un sistema di multiplazione TDM utilizza una capacità del canale in uscita dal multiplatore $W=2,048$ Mbit/s e $k=8$ bit per slot. Assumendo una velocità di ciascun canale pari a $V=64$ kbit/s, calcolare il numero di canali N , la lunghezza di trama T_T , e la lunghezza dello slot T_S .



Il Sistema di multiplo ha un rate complessivo di W e deve supportare tributari con rate V . Il numero di tributari supportabili è quindi:

$$N = \frac{W}{V} = \frac{2048}{64} = 32$$

La durata della trama si può calcolare imponendo che se si assegna uno slot per trama ad un tributario, il rate equivalente del canale così definito sia uguale al rate di ingresso del tributario V .

$$T_T = \frac{k}{V} = \frac{8}{64 \cdot 10^3} = 125 \mu s$$

Il tempo di slot può essere calcolato come tempo di trama diviso per numero di slot nella trama, ovvero come tempo necessario per inviare k bit ad una velocità di W ; quindi:

$$T_S = \frac{T_T}{N} = \frac{k}{W} = \frac{125 \cdot 10^{-6}}{32} \cong 3,90 \mu s$$

Esercizio 6.3

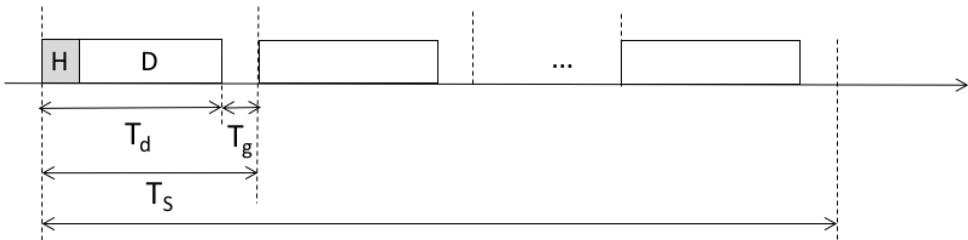
Un sistema di multiplazione a divisione di tempo è caratterizzato da un grado di interallacciamento $k=8[\text{bit}]$ e deve servire flussi in ingresso (tributari) con rate $r=128[\text{kbit/s}]$. Trovare la durata della trama di multiplo, T_T . Sapendo poi che il singolo slot nella trama di multiplo ha durata $T_s=3,125[\mu\text{s}]$, trovare il rate trasmissivo a valle del multiplatore, W , ed il numero massimo di flussi in ingresso che possono essere serviti, N .

La durata della trama si può trovare imponendo: $r = k/T_T$, ossia che il numero di bit che il singolo tributario può inviare in un tempo di trama definisca un canale equivalente con capacità uguale a quella con cui il tributario “porta” i bit al multiplatore; da cui si ha: $T_T=62,5[\mu\text{s}]$.

La capacità W è definita come: $W=k/T_s=2,56[\text{Mbit/s}]$. Il numero massimo di tributari è pari alla capacità complessiva del multiplatore diviso per la capacità del singolo tributario: $N= W/r=20$.

Esercizio 6.4

Un sistema di accesso multiplo TDMA utilizza $N=10$ time slot, un tempo di guardia $T_g=200 \mu s$, pacchetti dati composti da $D=180$ bit dati e $H=20$ bit di *header*, e un tempo di trama T_T pari a 10 ms. Calcolare la velocità di portante (multiplex) W e la velocità netta (dati) V di ciascun canale.



La dimensione di uno slot (in bit) è:

$$k = H + D = 200$$

Il tempo di slot è dato dal tempo di trama diviso per il numero di slot nella trama:

$$T_s = \frac{T_T}{N} = \frac{10 \cdot 10^{-3}}{10} = 1 \text{ ms}$$

Il tempo di trasmissione della parte dati dello slot è dato dal tempo di slot meno il tempo di guardia:

$$T_d = T_s - T_g = 0,8 \text{ ms}$$

La velocità del flusso multiplato è data dalla dimensione (in bit) della parte dati dello slot divisa per il tempo necessario per il suo invio:

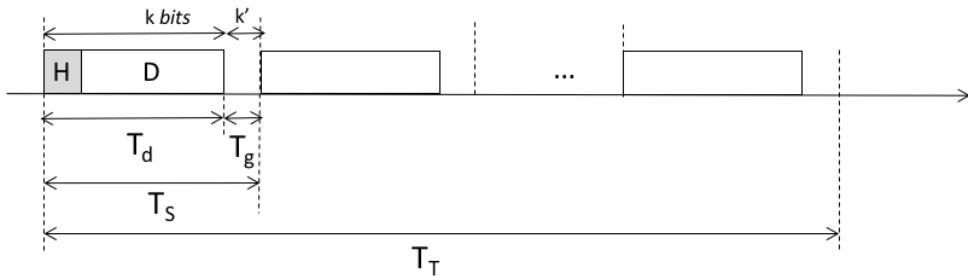
$$W = \frac{k}{T_d} = \frac{200}{0,8 \cdot 10^{-3}} = 250 \text{ kbit/s}$$

La velocità netta del canale definito come “uno slot per trama” è uguale alla quantità di informazione inviata nello slot diviso per la durata della trama

$$V = \frac{D}{T_T} = \frac{180}{10 \cdot 10^{-3}} = 18 \text{ kbit/s}$$

Esercizio 6.5

Il sistema di accesso multiplo TDMA del sistema cellulare GSM utilizza $N=8$ time slot, un tempo di guardia pari a $k' = 8,25$ tempi di bit, pacchetti dati composti da $D=114$ bit dati e $H=34$ bit di *overhead*, e un tempo di trama T_T pari a 4,615 ms. Calcolare la velocità di multiplex W e la velocità netta (dati) V di ciascun canale.



Il numero di bit di *overhead* e di dati contenuto in uno slot è:

$$k = H + D = 148 \text{ bit}$$

Il numero di bit complessivo (*overhead* + dati + guardia) è:

$$k_{TOT} = k + k' = 156,25$$

La durata del singolo slot è pari alla durata della trama divisa per il numero di slot:

$$T_S = \frac{T_T}{N} = \frac{4,615 \cdot 10^{-3}}{8} = 577 \mu\text{s}$$

La velocità dalla portante può essere calcolata osservando che in un tempo di slot devono essere trasmessi k_{tot} bit, quindi:

$$W = \frac{k_{TOT}}{T_S} = \frac{156,25}{0,577 \cdot 10^{-6}} = 270,8 \text{ kbit/s}$$

La velocità netta corrispondente del canale definito come “uno slot per trama” è:

$$V = \frac{D}{T_T} = \frac{114[\text{bit}]}{4,615 \cdot 10^{-3}} = 24,70 \text{ kbit/s}$$

Si noti che nel calcolo sono stati considerati 114[bit] al numeratore (senza i bit di guardia) perché questo è lo “spazio” che può essere usato per “ospitare” informazione (*overhead* + dati).

Esercizio 6.6

Un sistema di accesso multiplo centralizzato a divisione di tempo (TDMA) è caratterizzato da una trama con slot di durata $T_S = 10 \text{ ms}$, con un tempo di guardia minimo $T_G = 2 \mu\text{s}$. Il sistema serve 8 utenti e ha un *rate* trasmissivo del segnale multiplato di $C = 1 \text{ Mbit/s}$.

Si chiede di:

- 1) indicare il numero di bit di ciascun tributario trasmessi in ogni slot, n
- 2) indicare il massimo rate possibile per ciascun tributario in ingresso, r

$$1) n = (10 \cdot 10^{-6} - 2 \cdot 10^{-6}) 1 \cdot 10^6 = 8$$

$$2) r = \frac{8}{8 \cdot 10 \cdot 10^{-6}} = 100 \text{ kbit/s}$$

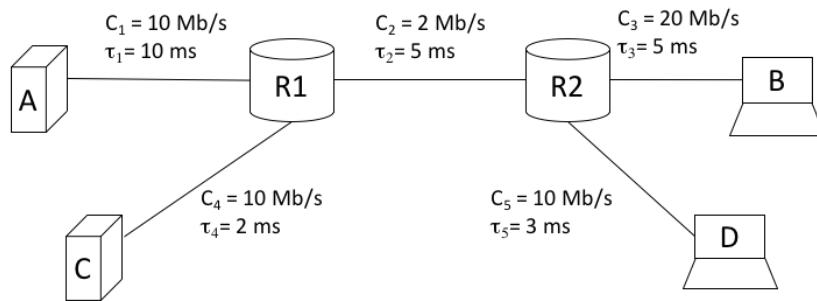
7. Esercizi riassuntivi

7.1 Esercizio

Una connessione TCP tra l'*host A* e l'*host B* nelle reti in figura è caratterizzata dai seguenti parametri: lunghezze di *header* e *ack* trascurabili, *link* bidirezionali simmetrici, $MSS = 1250 \text{ B}$, $SSTHRESH = 8 \text{ MSS}$.

Si calcoli il tempo necessario a trasferire un file di 30 kB (dall'istante di tempo di attivazione della connessione all'istante di ricezione dell'ACK dell'ultimo segmento).

(*nota la soluzione è la stessa se si scambiano di posto il link 1 con il link 2*)



Soluzione

$$MSS = 1250 \cdot 8 \text{ bit} = 10.000 \text{ bit}$$

tempi di trasmissione da A, R1 e R2: $T_1 = 1 \text{ ms}$, $T_2 = 5 \text{ ms}$, $T_3 = 0.5 \text{ ms}$

$$RTT = T_1 + T_2 + T_3 + 2\tau_1 + 2\tau_2 + 2\tau_3 = 46.5 \text{ ms}$$

Considerando che il link più lento è il link 2

$$\text{numero di pacchetti della finestra per cui la trasmissione è continua } W_{cont} = \left\lceil \frac{RTT}{T_2} \right\rceil = \left\lceil \frac{46.5}{5} \right\rceil = 10$$

$$\text{dimensione del file da trasferire } F = \frac{30}{1.25} = 24 \text{ MSS}$$

Modalità slow start: (1) – (2) – (4) – (8)

Modalità congestion avoidance: (9)

$$T_{tot} = T_{open} + 4 RTT + RTT + 8T_2 = 40 + 5 \cdot 46.6 = 312.5 \text{ ms}$$

T_{open} è tempo di setup della connessione

4 RTT x il trasferimento in modalità slow start (in totale 15 pacchetti)

8T2 + RTT per gli ultimi 9 pacchetti rimanenti in modalità Cong. Avoidance (domina il link più lento e quindi si considera T2. RTT qui include il tempo di trasferimento dell'ultimo pacchetto e il tempo di arrivo del suo riscontro ad A).

7.2 Esercizio

Si assuma che A sia un server http e B un client http. Occorre trasferire un documento di 100 kB base e 10 immagini di 5 MB. Si consideri che il messaggio usato per aprire una connessione sia trascurabile e il tempo di setup della connessione sia 40 ms. Si calcoli il tempo necessario assumendo il RTT di 46,5 ms e un ritmo medio di trasmissione di 1 Mb/s nel caso di connessione http persistente e non persistente (con trasmissione in parallelo delle immagini).

Soluzione

Caso di connessione persistente:

$$R_{html} = R_{obj} = 1 \text{ Mbps}$$

$$T_{tot} = T_{open} + \left(RTT + \frac{L_{html}}{R_{html}} \right) + 10 \left(RTT + \frac{L_{obj}}{R_{obj}} \right) = 401,3515 \text{ s}$$

Caso di connessione non-persistente:

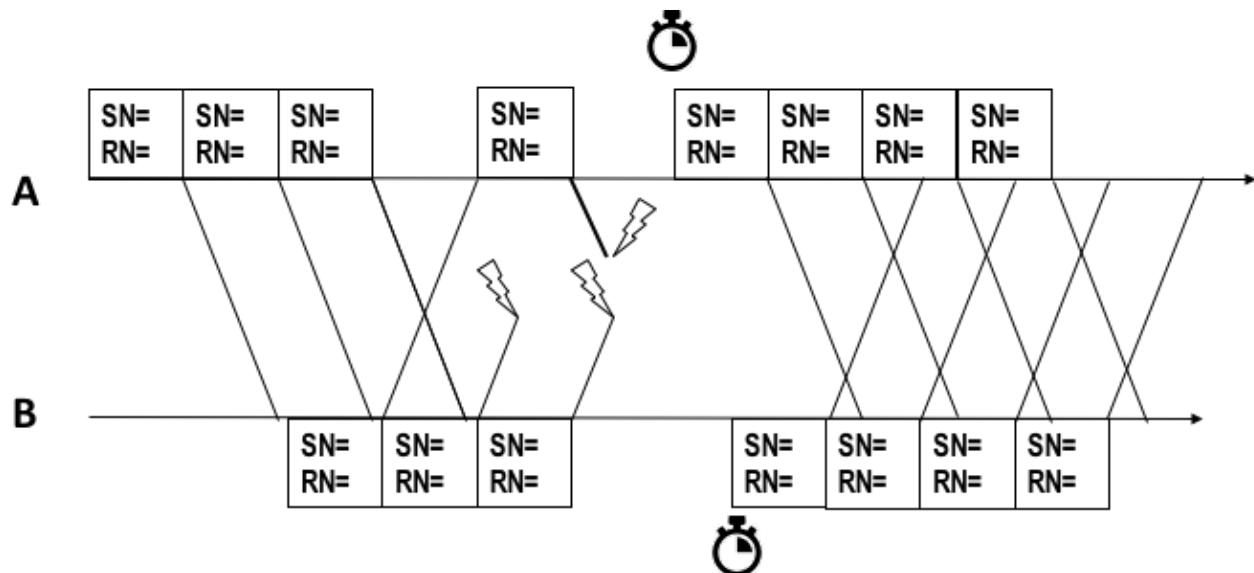
$$R_{html} = 1 \text{ Mbps}$$

$$R_{obj} = \frac{C}{10} = 0.1 \text{ Mbps}$$

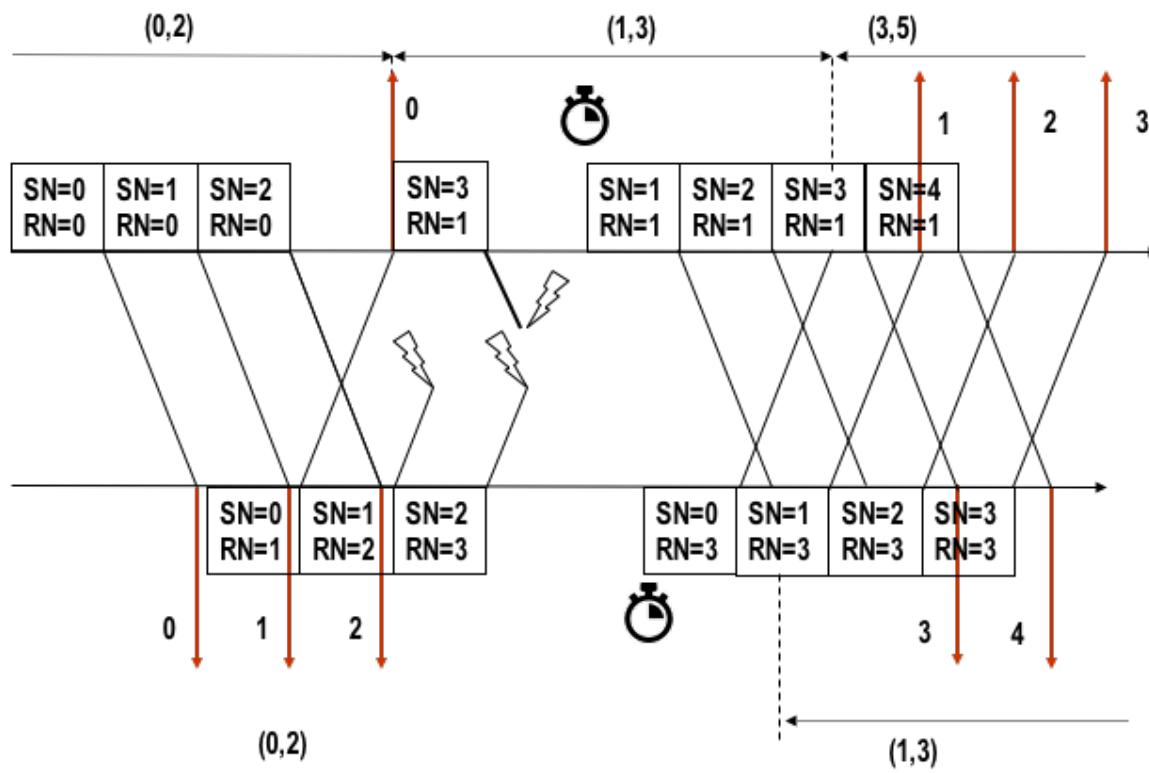
$$T_{tot} = \left(T_{open} + RTT + \frac{L_{html}}{R_{html}} \right) + \left(T_{open} + RTT + \frac{L_{obj}}{R_{obj}} \right) = 400,973 \text{ s}$$

7.3 Esercizio

Si completi la figura in accordo alle regole del protocollo Go-back-N con N=3. Si inseriscano i valori di SN ed RN, si indichino gli istati di accettazione delle trame corrette e in sequenza.



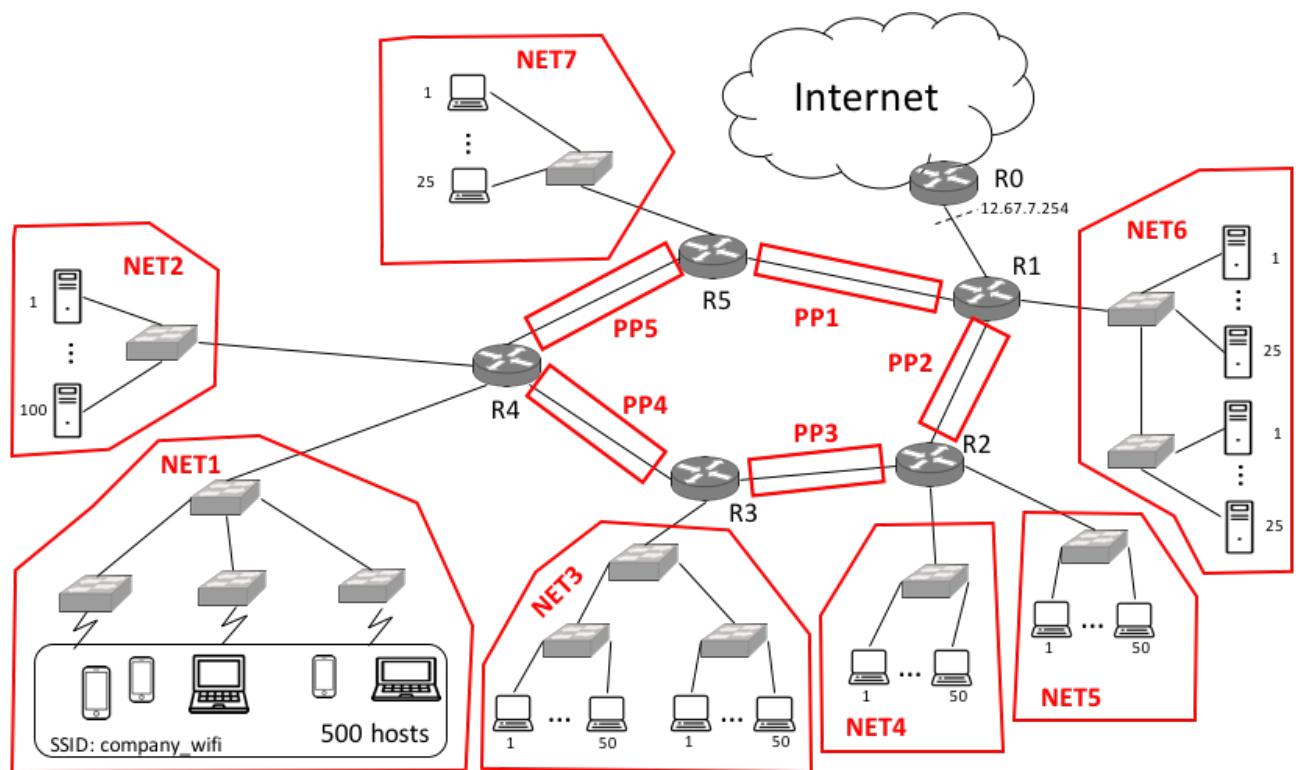
Soluzione



7.4 Esercizio

Un ISP possiede il seguente spazio di indirizzamento IP: 29.88.192.0/22 La rete complessiva dell'ISP è rappresentata in figura.

Indicare le sotto-rete IP graficamente nella figura, mettendone in evidenza i confini.



7.5 Esercizio

Un router ha le seguenti interfacce e tabella di routing. Riceve i pacchetti con destinazione indicati sotto. Si dica per ciascuno di essi come si comporta il router: inoltro diretto o indiretto, interfaccia di uscita, riga della tabella.

Eth0 - Address: 131.175.21.254 – Netmask: 255.255.255.128

Eth1: Address: 131.175.20.126 – Netmask: 255.255.255.128

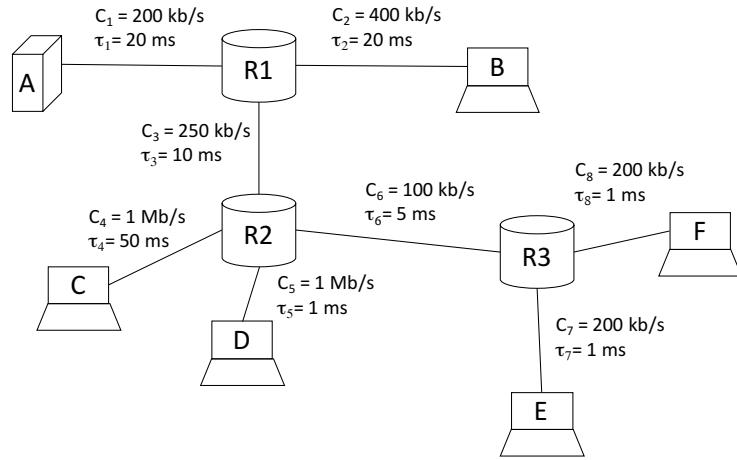
Network	Netmask	Next-hop
131.175.70.0	255.255.254.0	131.175.21.133
131.175.71.128	255.255.255.128	131.175.21.145
131.175.72.0	255.255.254.0	131.175.20.5
0.0.0.0	0.0.0.0	131.175.20.1

131.175.71.72 131.175.20.12 da Eth0	131.175.73.12
--	----------------------

Soluzione

131.175.71.72 Inoltro indiretto riga 1 131.175.20.12 da Eth0 Inoltro diretto su Eth1	131.175.73.12 Inoltro indiretto riga 3
---	--

7.6 Esercizio



Nella rete a commutazione di pacchetto in figura, al tempo $t=0$ sono presenti 6 pacchetti in A diretti rispettivamente alle seguenti destinazioni: C, D, E, E, F, E. Calcolare l'istante di fine ricezione di ciascuno dei pacchetti a destinazione assumendo che i pacchetti abbiano le seguenti dimensioni: pacchetti verso C, $L_C = 375$ byte; pacchetti verso D, $L_D = 250$ byte; pacchetti verso E, $L_E = 375$ byte; pacchetti verso F, $L_F = 125$ byte.

Soluzione

$$T_1^C = \frac{L_C}{C_1} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_3^C = \frac{L_C}{C_3} = \frac{375 * 8 \text{ bit}}{250 \text{ kbps}} = 12 \text{ ms}$$

$$T_4^C = \frac{L_C}{C_4} = \frac{375 * 8 \text{ bit}}{1 \text{ Mbps}} = 3 \text{ ms}$$

$$T_1^D = \frac{L_D}{C_1} = \frac{250 * 8 \text{ bit}}{200 \text{ kbps}} = 10 \text{ ms}$$

$$T_3^D = \frac{L_D}{C_3} = \frac{250 * 8 \text{ bit}}{250 \text{ kbps}} = 8 \text{ ms}$$

$$T_5^D = \frac{L_D}{C_5} = \frac{250 * 8 \text{ bit}}{1 \text{ Mbps}} = 2 \text{ ms}$$

$$T_1^E = \frac{L_E}{C_1} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_3^E = \frac{L_E}{C_3} = \frac{375 * 8 \text{ bit}}{250 \text{ kbps}} = 12 \text{ ms}$$

$$T_6^E = \frac{L_E}{C_6} = \frac{375 * 8 \text{ bit}}{100 \text{ kbps}} = 30 \text{ ms}$$

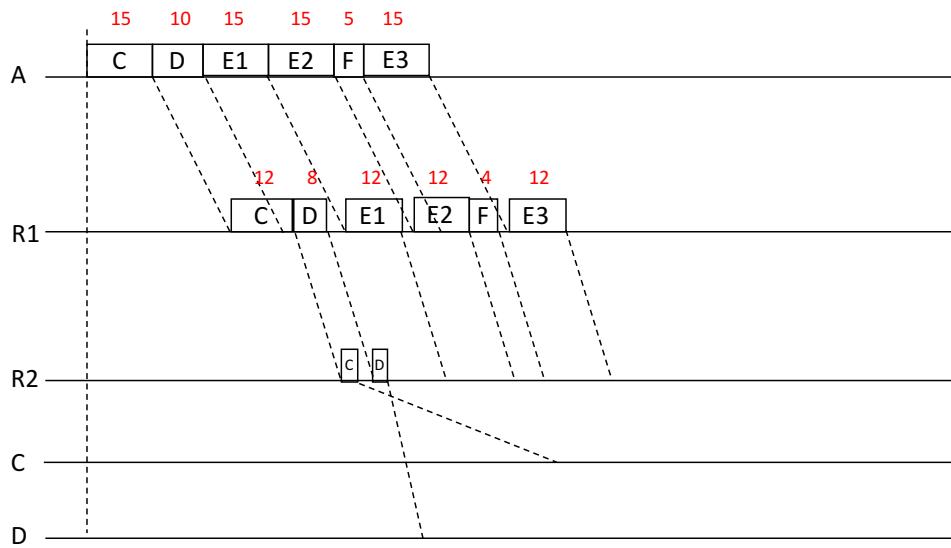
$$T_7^E = \frac{L_E}{C_7} = \frac{375 * 8 \text{ bit}}{200 \text{ kbps}} = 15 \text{ ms}$$

$$T_1^F = \frac{L_F}{C_1} = \frac{125 * 8 \text{ bit}}{200 \text{ kbps}} = 5 \text{ ms}$$

$$T_3^F = \frac{L_F}{C_3} = \frac{125 * 8 \text{ bit}}{250 \text{ kbps}} = 4 \text{ ms}$$

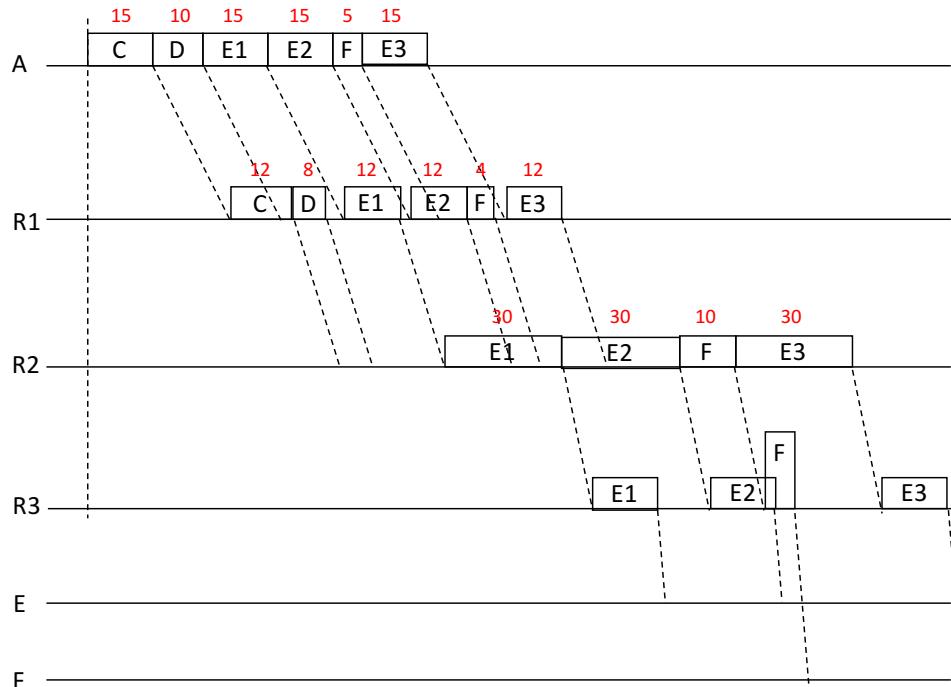
$$T_6^F = \frac{L_F}{C_6} = \frac{125 * 8 \text{ bit}}{100 \text{ kbps}} = 10 \text{ ms}$$

$$T_8^F = \frac{L_F}{C_8} = \frac{125 * 8 \text{ bit}}{200 \text{ kbps}} = 5 \text{ ms}$$



$$T_C = T_1^C + \tau_1 + T_3^C + \tau_3 + T_4^C + \tau_4 = 15 + 20 + 12 + 10 + 3 + 50 = 110 \text{ ms}$$

$$T_D = T_1^D + \tau_1 + T_3^D + \tau_3 + T_5^D + \tau_5 = 15 + 20 + 12 + 8 + 10 + 2 + 1 = 68 \text{ ms}$$



$$\begin{aligned} T_{E1} &= T_1^E + T_1^F + T_1^E + \tau_1 + T_3^E + \tau_3 + T_6^E + \tau_6 + T_7^E + \tau_7 \\ &= 15 + 10 + 15 + 20 + 12 + 10 + 30 + 5 + 15 + 1 = 133 \text{ ms} \end{aligned}$$

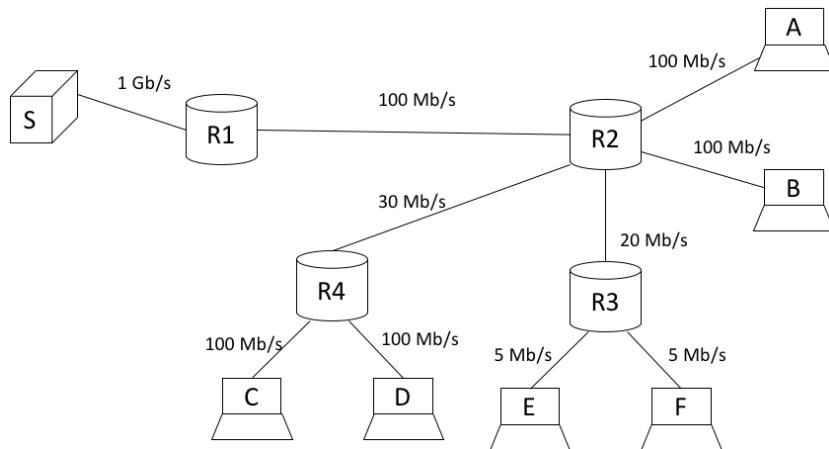
$$T_{E2} = T_{E1} + T_6^E = 133 + 30 = 163 \text{ ms}$$

$$\begin{aligned}T_F &= T_1^C + T_1^D + T_1^E + \tau_1 + T_3^E + \tau_3 + 2T_6^E + T_6^F + \tau_6 + T_7^F + \tau_7 \\&= 15 + 10 + 15 + 20 + 12 + 10 + 60 + 10 + 5 + 5 + 1 = 163 \text{ ms}\end{aligned}$$

$$T_{E3} = T_{E2} + T_6^E + T_6^F = 163 + 30 + 10 = 203 \text{ ms}$$

7.7 Esercizio

Si consideri la rete sotto e si assuma siano presenti contemporaneamente 2 trasferimenti file con protocollo TCP da ciascun host (A, B, C, D, E, F) verso il server S. Assumendo di avere raggiunto attraverso TCP una condivisione equa delle risorse, qual è la velocità di ciascuna connessione?



Soluzione

- Per la connessione S-E il link collo di bottiglia è quello tra il nodo R3 ed E. Il testo ci dice che abbiamo 2 flussi (trasferimenti di file) contemporanei per ciascun host verso il server. Quindi significa che ciascun flusso (ciascuna connessione) da E a R3 può avere al massimo una velocità di:

$$5/2=2.5 \text{ Mb/s} \rightarrow \text{S-E: } 2.5 \text{ Mb/s}$$

Lo stesso per la connessione S-F \rightarrow S-F: 2.5 Mb/s

- Per le connessioni S-C e S-D il link collo di bottiglia è quello tra R4 e R2, dove ci sono 4 flussi contemporanei (2 da C e 2 da D). Quindi ciascuna connessione può avere al massimo una velocità di $30/4 = \text{S-C: } 7.5 \text{ Mb/s} \rightarrow \text{S-C: } 7.5 \text{ Mb/s e S-D: } 7.5 \text{ Mb/s}$

- Le connessioni S-A e S-B condividono con tutte le altre connessioni il link R1-R2. Quindi per le 4 connessioni da E e F e per le 4 connessioni da C e D avremo: $4(2.5 \text{ Mb/s}) + 4(7.5 \text{ Mb/s}) = 40 \text{ Mb/s}$

Nel link R1-R2 abbiamo $100 \text{ Mb/s} - 40 \text{ Mb/s} = 60 \text{ Mb/s}$ da dividere per 4. Di conseguenza:

S-A: 15 Mb/s

S-B: 15 Mb/s

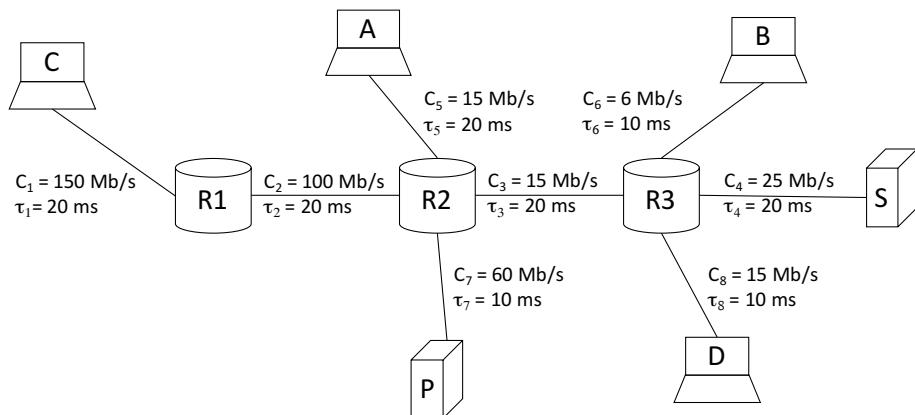
7.8 Esercizio

Nella rete in figura sono rappresentati 4 client (A, B, C e D), 3 router (R1, R2 e R3), un server HTTP (S) e un proxy HTTP (P). Il client C vuole trasferire un documento formato da una pagina HTML di dimensione $L_{HTML} = 30$ kbyte che richiama 11 oggetti di dimensione $LOGG = 75$ kbyte. Nella rete sono presenti anche 4 flussi interferenti di lunga durata: 2 da A a D, da A a B. Si supponga che: i) i messaggi di apertura della connessione e di richiesta HTTP siano di lunghezza trascurabile, ii) la connessione TCP tra proxy HTTP e server HTTP sia sempre aperta e non occorra mandare una richiesta di apertura. Assumiamo di avere raggiunto attraverso TCP una condivisione equa delle risorse in tutta la rete.

Si calcoli:

- il tempo di trasferimento del documento nel caso in cui il client C senza proxy configurato apra in parallelo in modalità non-persistente tutte le connessioni TCP necessarie.
- il tempo di trasferimento del documento nel caso in cui il client C con proxy configurato apra in serie un'unica connessione TCP persistente per scaricare tutti gli oggetti, ipotizzando che solo i primi 6 degli 11 oggetti siano presenti nella cache del proxy
- nel caso b), il numero minimo di oggetti che occorre trovare nella cache del proxy per avere un tempo di trasferimento minore di 2 s.

Per semplicità nel calcolo del tempo di trasferimento non si consideri il tempo di ritrasmissione di ciascuno dei router R.



Soluzione

Punto a

Il client manda/riceve messaggi HTTP direttamente al/dal server.

Durante il trasferimento della pagina HTML (1 flusso tra C e S) sul link R2-R3 ci sono 5 flussi che condividono 15 Mb/s (un flusso tra C e S + 4 flussi interferenti indicati nel testo). Il link è collo di bottiglia per tutti i flussi che ottengono 3 Mb/s ciascuno di velocità. Quindi la capacità vista dal trasferimento C-S sarà $C_{eff-5} = 3Mb/s$.

Il tempo di trasferimento della pagina HTML è: $T_{HTML} = \frac{L_{HTML}}{C_{eff-5}} = \frac{30*8 \text{ kbit}}{3 \text{ Mbps}} = 80 \text{ ms}$ (si considera solo la capacità calcolata per il link collo di bottiglia).

Durante il trasferimento degli oggetti (11 flussi in parallelo tra C e D) sul link R2-R3 ci sono 15 flussi che condividono 15 Mb/s (11 flussi in parallelo tra C e D + i 4 flussi interferenti indicati nel testo). Il link è collo di bottiglia per tutti i flussi che ottengono 1 Mb/s ciascuno. Quindi la capacità vista dal trasferimento C-S sarà $C_{eff-15} = 1 \text{ Mb/s}$.

Il tempo di trasferimento di un oggetto è: $T_{OGG} = \frac{L_{OGG}}{C_{eff-15}} = \frac{75*8 \text{ kbit}}{1 \text{ Mbps}} = 600 \text{ ms}$

Il RTT tra C e S è: $RTT_{C-S} = 2(\tau_1 + \tau_2 + \tau_3 + \tau_4) = 2 * 80 = 160 \text{ ms}$

Il tempo totale di trasferimento (considerando una sola volta il tempo di trasmissione della pagina HTML e di ciascun oggetto JPEG, senza contare quello dovuto ai router, come da ipotesi nel testo) sarà:

$$T_{tot} = RTT + RTT + T_{HTML} + RTT + RTT + T_{OGG} = 4RTT + T_{HTML} + T_{OGG} = 1320 \text{ ms}$$

Punto b

Il client manda/riceve messaggi HTTP al/dal proxy. Per i primi 6, riceve l'oggetto direttamente dal proxy, per gli ultimi 5, il proxy scarica l'oggetto dal server e poi lo invia al client.

La capacità del trasferimento tra client e proxy è determinata dal link P-R2 e dunque pari a $C_{eff} = 60 \text{ Mb/s}$.

Il tempo di trasferimento della pagina HTML (diretta al proxy) è: $T_{HTML} = \frac{L_{HTML}}{C_{eff}} = \frac{30*8 \text{ kbit}}{60 \text{ Mbps}} = 4 \text{ ms}$

Il tempo di trasferimento di un oggetto (in cache) è: $T_{OGG-cache} = \frac{L_{OGG}}{C_{eff}} = \frac{75*8 \text{ kbit}}{60 \text{ Mbps}} = 10 \text{ ms}$.

La capacità del trasferimento tra proxy e server è determinata dal link R2-R3, su cui transitano 5 flussi, dunque pari a $C_{eff-5} = 3 \text{ Mb/s}$.

Il tempo di trasferimento di un oggetto (nel server) è: $T_{OGG-nocache} = \frac{L_{OGG}}{C_{eff-5}} = \frac{75*8 \text{ kbit}}{3 \text{ Mbps}} = 200 \text{ ms}$.

Il RTT tra C e P è: $RTT_{C-P} = 2(\tau_1 + \tau_2 + \tau_7) = 2 * 50 = 100 \text{ ms}$, mentre il RTT tra P e S è: $RTT_{P-S} = 2(\tau_7 + \tau_3 + \tau_4) = 2 * 50 = 100 \text{ ms}$

Il tempo totale di trasferimento (considerando una sola volta il tempo di trasmissione della pagina HTML e di ciascun oggetto JPEG, senza contare quello dovuto ai router, come da ipotesi nel testo) è:

$$\begin{aligned}T_{tot} &= RTT_{C-P} + RTT_{C-P} + T_{HTML} + 6(RTT_{C-P} + T_{OGG-cache}) \\&\quad + 5(RTT_{P-S} + T_{OGG-nocache} + RTT_{C-P} + T_{OGG-cache}) \\&= 100 + 100 + 4 + 6(100 + 10) + 5(100 + 200 + 100 + 10) \\&= 204 + 660 + 2050 = 2914 \text{ ms}\end{aligned}$$

Punto c

Dall'ultima espressione del punto b) si può scrivere

$$T_{tot} = 204 + x(110) + (11 - x)(410)$$

dove x è il numero di oggetti trovati in cache.

Quindi da $T_{tot} = 204 + x(110) + (11 - x)(410) < 2000 \text{ ms}$, si ottiene $x > 9.046$.
Occorre trovare in cache almeno 10 oggetti.

7.9 Esercizio

Nella rete in figura sono rappresentati 4 router (R1, R2, R3 e R4), un client A, un HTTP proxy P e un HTTP server S. Accanto ad ogni collegamento è indicata la propria capacità, mentre il tempo di propagazione è pari a 10 ms su ciascun collegamento.

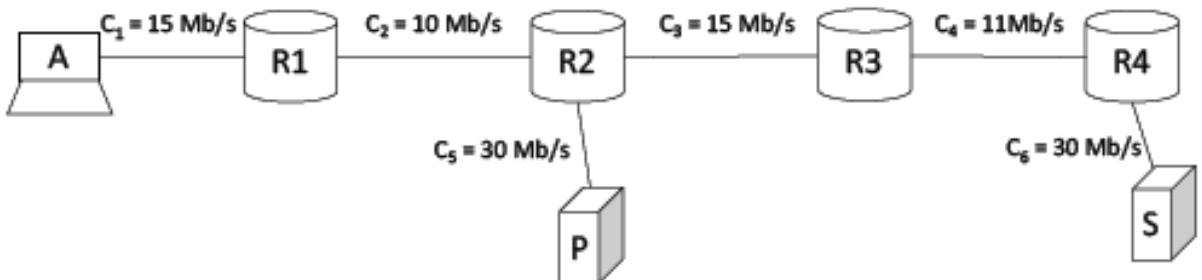
Il client vuole scaricare dal server un sito web composto da 1 pagina HTML di dimensione LHTML=80 [kbyte] e 6 oggetti JPEG richiamati nella pagina HTML di dimensione 500 kbyte. Nella rete sono presenti flussi interferenti di lunga durata: 4 tra R1 e R2, 10 tra R3 e R4. Assumiamo di avere raggiunto attraverso TCP una condivisione equa delle risorse in tutta la rete.

Si chiede di calcolare il tempo di trasferimento del sito web a livello applicativo nei seguenti casi:

- il client A non ha proxy configurato, apre connessioni non-persistenti in parallelo (quando possibile e nel massimo numero possibile)
- il client A utilizza il proxy P, apre al massimo una connessione alla volta in modalità non-persistent.

Solo la pagina HTML ed i primi 2 oggetti JPEG sono presenti nella cache del proxy.

Per semplicità nel calcolo del tempo di trasferimento non si consideri il tempo di ritrasmissione di ciascuno dei router R.



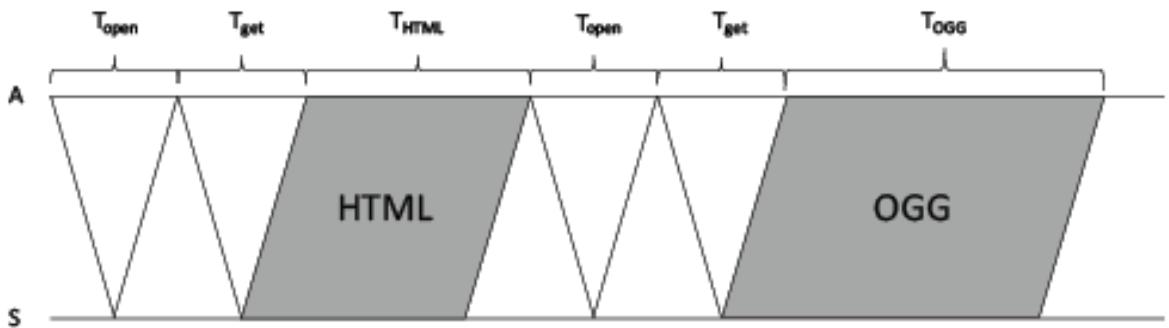
Soluzione

Il client A colloquia direttamente con il server S.

Assumiamo di avere raggiunto una condivisione equa delle risorse in tutta la rete, per la pagina HTML, i collegamenti attraversati hanno le seguenti capacità effettive: A-R1 15 Mb/s, R1-R2 2 Mb/s (4+1 flussi), R2-R3 15Mb/s, R3-R4 1Mb/s (10+1 flussi). Dunque il trasferimento è governato dal collo di bottiglia R3-R4 a 1 Mb/s. Il tempo di trasferimento della pagina HTML è pari a $T_{HTML} = 8*80 [kbit] / 1 [Mb/s] = 640 ms$

Per gli oggetti JPEG, i collegamenti attraversati sono i medesimi, dunque il collo di bottiglia sarà il link R3-R4 con una capacità effettiva di 687.5 kb/s (10+6 flussi). Il tempo di trasferimento di ogni oggetto JPEG è pari a $T_{OGG} = 8*500 [kbit] / 687.5 [kb/s] = 5.82 s$.

Il tempo totale di trasferimento (considerando una sola volta il tempo di trasmissione della pagina HTML e di ciascun oggetto JPEG, senza contare quello dovuto ai router, come da ipotesi nel testo) è pari a $T = T_{\text{open}} + T_{\text{get}} + T_{\text{HTML}} + T_{\text{open}} + T_{\text{get}} + T_{\text{OGG}}$ in accordo con la figura sotto. Considerando 10 ms di tempo di propagazione per ciascun collegamento e tenendo conto di 5 collegamenti in andata e 5 in ritorno, $T_{\text{open}} = T_{\text{get}} = 100 \text{ [ms]}$ $\rightarrow T = 6.86 \text{ s}$.



Punto b)

Il client A colloquia con il proxy P. Il colloquio client-proxy è definito dal collo di bottiglia R1-R2 con una capacità di 2 Mb/s (4+1 flussi), mentre il colloquio proxy-server ha R3-R4 come collo di bottiglia, dunque una capacità di 1 Mb/s (10+1 flussi). I tempi di trasferimento sono:

$$T_{\text{HTML}}^{\text{AP}} = (640 \text{ kb} / 2 \text{ Mb/s}) = 320 \text{ ms}$$

$$T_{\text{OGG}}^{\text{AP}} = (4 \text{ Mb} / 2 \text{ Mb/s}) = 2 \text{ s}$$

$$T_{\text{OGG}}^{\text{PS}} = (4 \text{ Mb} / 1 \text{ Mb/s}) = 4 \text{ s}$$

rispettivamente per il trasferimento della pagina HTML dal proxy al client, di un oggetto JPEG dal proxy al client e di un oggetto JPEG dal server al proxy.

I tempi di propagazione sono $T_{\text{open}}^{\text{CP}} = T_{\text{get}}^{\text{CP}} = 60 \text{ ms}$ per la connessione client-proxy e sono $T_{\text{open}}^{\text{PS}} = T_{\text{get}}^{\text{PS}} = 80 \text{ ms}$ per la connessione proxy-client.

Il tempo totale di trasferimento (considerando una sola volta il tempo di trasmissione della pagina HTML e di ciascun oggetto JPEG, senza contare quello dovuto ai router, come da ipotesi nel testo) è pari a

$$\begin{aligned} T = & T_{\text{open}}^{\text{AP}} + T_{\text{get}}^{\text{AP}} + T_{\text{HTML}}^{\text{AP}} + 2(T_{\text{open}}^{\text{AP}} + T_{\text{get}}^{\text{AP}} + T_{\text{OGG}}^{\text{AP}}) + 4(T_{\text{open}}^{\text{AP}} + T_{\text{get}}^{\text{AP}} + T_{\text{OGG}}^{\text{AP}} \\ & + T_{\text{open}}^{\text{PS}} + T_{\text{get}}^{\text{PS}} + T_{\text{OGG}}^{\text{PS}}) = 29.8 \text{ s} \end{aligned}$$

7.10 Esercizio

Sia dato il pool di indirizzi 131.175.17.0/25. Da questo pool occorre ricavare 2 sottoreti per due reti LAN aziendali L1 e L2 di almeno 50 postazioni ciascuna.

Si indichi l'indirizzo di rete (netid di ciascuna delle 2 reti con il corrispondente netmask).

Soluzione.

L1: 131.175.17.0/26 (netmask 255.255.255.192).

L2: 131.175.17.64/26 (netmask 255.255.255.192).

7.11 Esercizio

Un sistema cellulare GSM sfrutta l'accesso multiplo di tipo TDMA con 32 time slot. I pacchetti inviati sono composti da 256 bit di dati con l'aggiunta di 24 bit di header. Viene considerato un tempo di guardia tra i pacchetti $T_g=20 \mu s$. Il tempo di trama complessivo è pari a 9,6 ms.

- a) Calcolare la velocità di multiplex W del sistema e la velocità netta (solo dei dati) di ciascun canale di accesso.
- b) Si spieghi l'utilità del tempo di guardia nell'accesso multiplo di tipo TDMA.
- c) Cosa cambierebbe nelle velocità calcolate nel punto a) se non si considerasse il tempo di guardia? Si commentino i risultati ottenuti.

Soluzione.

$$a) H=24 \text{ bit} \quad D=256 \text{ bit} \quad K=H+D=280 \text{ bit}$$

$$T_t = 9,6 \text{ ms} \quad T_s = T_t/N = 9,6/32 = 300 \text{ us}$$

$$T_d = T_s - T_g = 300 - 20 = 280 \text{ us}$$

$$W = K/T_d = 1 \text{ Mb/s}$$

$$V_{\text{netto}} = D/T_t = 27 \text{ kb/s}$$

$$c). T_d = T_s = 300 \text{ us}$$

$W=0,93 \text{ Mb/s}$ è più lento

$V=27 \text{ kb/s}$ è uguale al caso a)

7.12 Esercizio

Si consideri la rete in figura. Una trasmissione TCP tra l'host S e l'host C nelle reti è caratterizzata dai seguenti parametri:

Lunghezze di header e ack trascurabili, lunghezza dei messaggi di apertura della connessione trascurabili, link bidirezionali simmetrici, MSS=1500 Byte, STHRESH=12 kByte.

- Si calcoli il tempo necessario per trasferire un file di 0,24 MByte (dall'istante di tempo in cui parte la richiesta di apertura della connessione da parte di S all'istante di ricezione dell'ACK dell'ultimo segmento da parte di S), considerando il meccanismo di invio dei riscontri applicato end-to-end.
- Si ripeta il calcolo assumendo che il 25° pacchetto vada perso e il timeout corrispondente sia $T_{out}=200$ ms (si assuma che il timeout parta all'istante di trasmissione del primo bit di ogni pacchetto).
- Cosa accade se i pacchetti fuori sequenza vengono comunque memorizzati? Si ripete il calcolo del caso b) in questo caso.

Non è necessario disegnare la rappresentazione spazio-temporale. Ma può essere d'aiuto per la risoluzione dell'esercizio.

