5.1.4 Appello del 18/09/2007

ESERCIZIO 1: Consideriamo due host H_1 ed H_2 collegati mediante una linea punto-punto esente da errori ($BER = 0$) e di rate R bps. Supponiamo che i due host utilizzino il TCP/IP e che H_1 voglia trasferire un file ad H_2 di dimensione pari ad F bits (multiplo intero di segmenti di dimensione *MSS*). Dopo aver iniziato la procedura di apertura della connessione di trasporto, H_1 inizia il trasferimento del file. Al fine di semplificare i calcoli richiesti supponiamo che:

(i) tutti i segmenti dati abbiano la stessa dimensione $m = MSS$ bits e tempo di trasmissione $T = m/R$ msec;

(ii) la latenza dei segmenti di controllo relativi all'apertura della connessione di trasporto tra H_1 ed H_2 sia T msec mentre quella relativa ai segmenti della k.ma congestion window sia kT msec;

(iii) i segmenti di *ACK* e *SYN* abbiano tempi di trasmissione trascurabili;

(iv) la latenza degli *ACK* sia nulla;

(v) la congestion window venga aggiornata soltanto quando H_1 riceve un *ACK* cumulativo da H_2, relativo all'intero gruppo di segmenti compresi nella congestion window;

(vi) la finestra di flow control del TCP sia infinita ed il timeout sia infinito.

Indichiamo con Λ il tempo di trasferimento del file definito come intervallo di tempo che intercorre tra l'istante in cui H_1 inizia la fase di apertura della connessione di trasporto e quello in cui H_2 riceve l'ACK dell'ultimo segmento del file. Supponiamo che il TCP utilizzi il meccanismo di *slow start* per controllare la congestione. Il candidato:

1. illustri la dinamica del sistema;

2. illustri in una tabella la corrispondenza tra la k.ma $\{k = 1, 2, 3, \dots\}$ congestion window, il numero di segmenti in eassa compresi $(CW(k))$ e l'intervallo di tempo $(t(k))$ che intercorre tra la trasmissione del primo bit del primo segmento della k.ma congestion window e l'istante in cui viene ricevuto l'ACK cumulativo relativo ai segmenti della congestion window medesima;

3. calcoli il numero N di finestre di congestione necessarie per trasmettere il file;

4. calcoli Λ supponendo che il file abbia dimensione tale da riempire completamente anche l'ultima congestion window;

5. sotto l'ipotesi del punto precedente calcoli il throughput γ, commentando cosa accade per quest'ultimo quando la dimensione del file tende all'infinito. Nel calcolo di γ si ignori il sovraccarico (overhead) introdotto dalle intestazioni dei vari protocolli (data-link layer, IP, TCP);
6. risponda alle domande precedenti nel caso in cui il meccanismo di controllo della congestione sia *Additive Increase Multiplicative Decrease* (*AIMD*)

RISOLUZIONE

1. La Figura 1.1 illustra la dinamica del sistema. Come illustrato nella stessa figura, H_1 apre la connessione di trasporto e quindi inizia con una congestion window di un solo segmento che invia a H_2 nel *"terzo ramo"* del three-way handshake. Dopo un tempo pari a $2T$ H_1 è perciò in grado di inviare il primo segmento.

2. La Figura 1.2 illustra la corrispondenza richiesta. Dalla Figura 1.2 si deduce che la k.ma $\{k = 1, 2, 3, \ldots\}$ congestion window contiene un numero di segmanti pari a $CW(k) = 2^{k-1}$ mentre $t(k) = (2^{k-1} + k)T$.

3. Il numero di segmenti di cui si compone il file risulta F/m. Quindi, l'ultimo segmento del file sarà compreso nella N.ma congestion window, con

$$N = min\left\{n: 2^0 + 2^1 + 2^2 + \ldots + 2^{n-1} \geq \frac{F}{m}\right\}$$

$$= min\left\{n: \sum_{h=0}^{n-1} 2^h \geq \frac{F}{m}\right\} = min\left\{n: \frac{1-2^n}{1-2} \geq \frac{F}{m}\right\} = min\left\{n: 2^n - 1 \geq \frac{F}{m}\right\}$$

$$= min\left\{n: n \geq log_2\left(\frac{F}{m} + 1\right)\right\}$$

Quindi

$$N = \left\lceil log_2\left(\frac{F}{m} + 1\right)\right\rceil$$

4. Per calcolare Λ facciamo riferimento alla Figura 1.1 e alla Figura 1.2. Tenendo presente che servono $2T$ msec prima di inviare il primo segmento si ha:

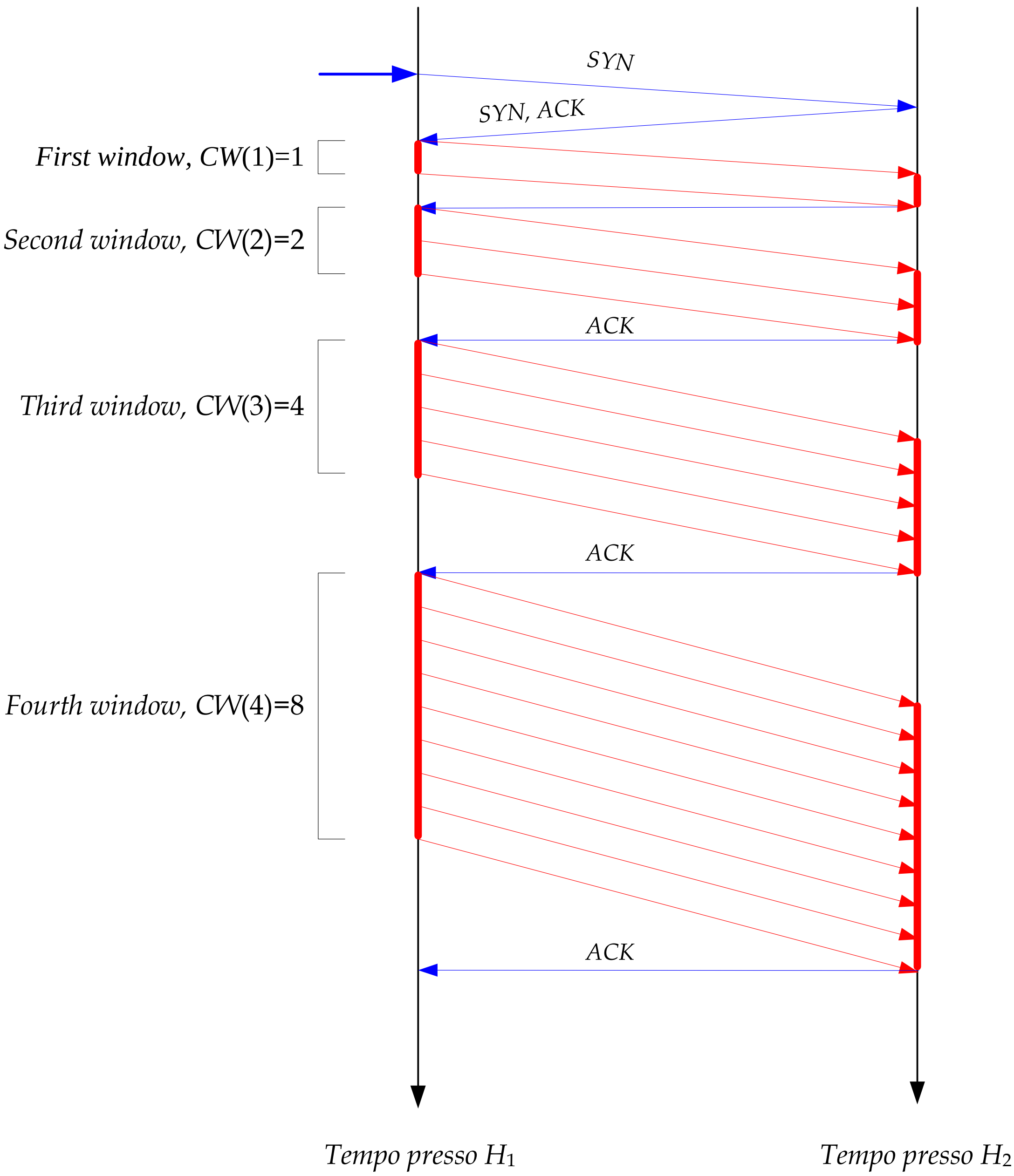

Figura 1.1: Dinamica del sistema nel caso dello *Slow Start*

$$\Lambda = 2T + \sum_{k=1}^{N} t(k) = 2T + \sum_{k=1}^{N} (2^{k-1} + k)T$$

$$= 2T + \sum_{k=1}^{N} (2^{k-1} + k)T = 2T + \sum_{h=0}^{N-1} 2^{h}T + \sum_{k=1}^{N} kT$$

k	$CW(k)$	$t(k)$
k=1	$CW(1)=1=2^0$	$t(1)=T+T=2^0T+1T$
k=2	$CW(2)=2=2^1$	$t(2)=2T+2T=2^1T+2T$
k=3	$CW(3)=4=2^2$	$t(3)=4T+3T=2^2T+3T$
k=4	$CW(4)=8=2^3$	$t(4)=8T+4T=2^3T+4T$
k=5	$CW(5)=16=2^4$	$t(5)=16T+5T=2^4T+5T$
⋮	⋮	⋮

Figura 1.2: Relazione tra k, $CW(k)$ e $t(k)$ nel caso di *Slow Start*

$$= \; 2T + \left(\frac{1-2^N}{1-2}\right)T + \left[\frac{N(N+1)}{2}\right]T$$

Dopo alcuni passaggi algebrici

$$\Lambda \;=\; \left(2^N + 1 + \frac{N(N+1)}{2}\right)T$$

5. Per quanto riguarda il throughput γ si ha

$$\gamma \;=\; \frac{F}{\Lambda} = \frac{F}{\left(2^N + 1 + \frac{N(N+1)}{2}\right)T} \tag{1.1}$$

Poichè $F \;=\; m(2^N - 1)$

$$\gamma = \frac{m(2^N - 1)}{\left(2^N + 1 + \frac{N(N+1)}{2}\right)T}$$

Se la dimensione del file tende ad infinito anche N tende ad infinito. Dividendo numeratore e denominatore per 2^N è facile dedurre che

$$\lim_{N \to \infty} \gamma = \frac{m}{T} = R$$

Questo risultato era atteso in quanto all'aumentare di N la dimensione delle finestra di congestione aumenta esponenzialmente mentre il tempo di idle della linea aumenta come $O(N^2)$. In altri termini, al limite, la linea risulta permanentemente occupata (spariscono gli intervalli di idle!) per cui il throughput tende al rate di trasmissione della linea.

6. La dinamica del sistema è riportata in Figura 1.3. La tabella riportata in

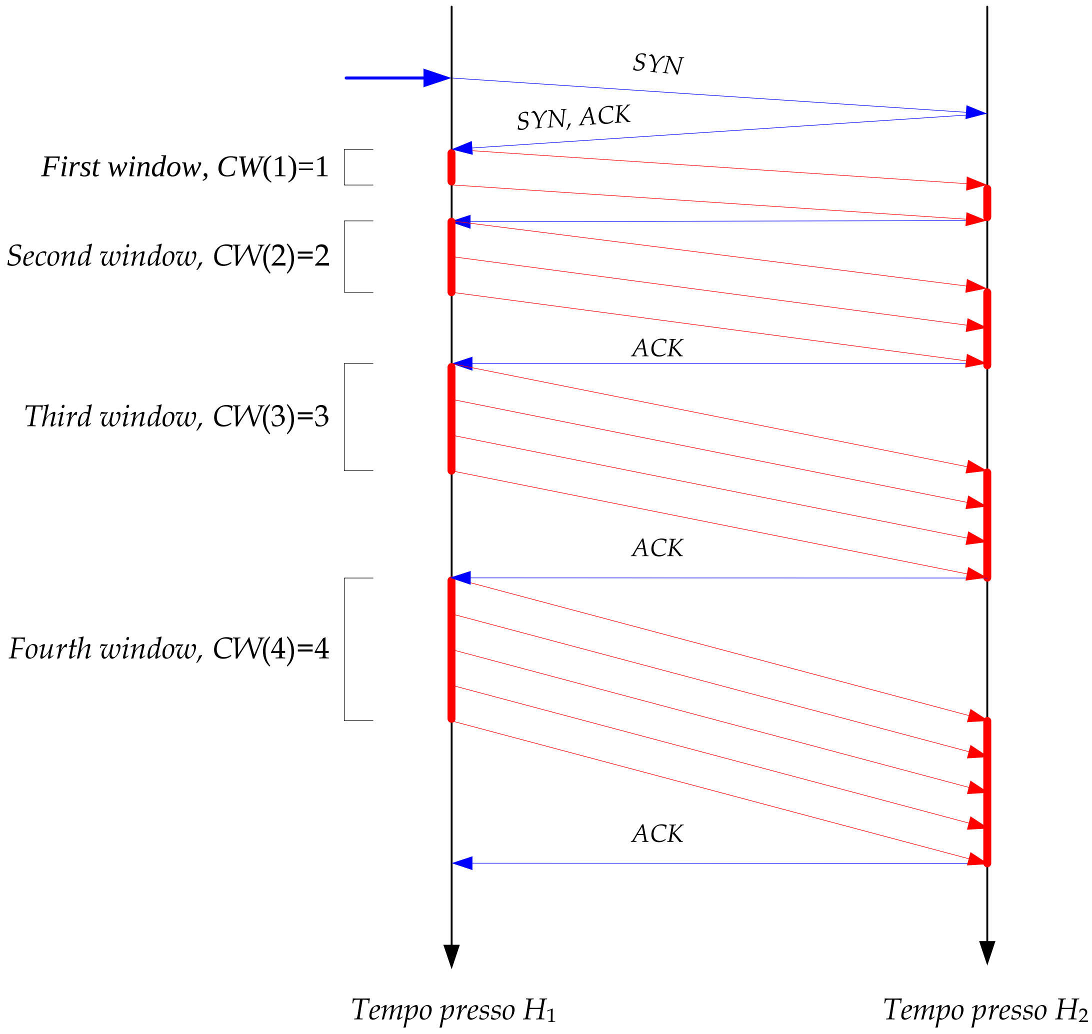

Figura 1.3: Dinamica del sistema nel caso AIMD

Figura 1.2 (valida per lo *Slow Start*) nel caso AIMD viene sostituita dalla tabella di Figura 1.4 dalla quale si deduce che la k.ma $\{k = 1, 2, 3, \ldots\}$ congestion window contiene $CW(k) = k$ segmenti. Inoltre, sempre relativamente alla k.ma congestion window si ha $t(k) = 2kT$. Indichiamo con $\tilde{N}$,

k	$CW(k)$	$t(k)$
k=1	$CW(1)$=1	$t(1)=T+T$
k=2	$CW(2)$=2	$t(2)=2T+2T=2(T+T)$
k=3	$CW(3)$=3	$t(3)=3T+3T=3(T+T)$
k=4	$CW(4)$=4	$t(4)=4T+4T=4(T+T)$
k=5	$CW(5)$=5	$t(5)=5T+5T=5(T+T)$
⋮	⋮	⋮

Figura 1.4: Relazione tra k, $CW(k)$ e $t(k)$ nel caso di AIMD

$\tilde{\Lambda}$ e $\tilde{\gamma}$ gli indici di prestazione AIMD corrispondenti a N, Λ e γ dello *Slow Start*. Quindi

$$\tilde{N} = min\left\{n: 1 + 2 + 3 + \ldots + n \geq \frac{F}{m}\right\}$$

$$= min\left\{n: \sum_{h=1}^{n} h \geq \frac{F}{m}\right\} = min\left\{n: \frac{n(n+1)}{2} \geq \frac{F}{m}\right\}$$

Perciò

$$\tilde{N} = \left\lceil \frac{1}{2}\left(-1 + \sqrt{1 + \frac{8F}{m}}\right) \right\rceil$$

Inoltre

$$\tilde{\Lambda} = 2T + \sum_{k=1}^{N} t(k) = 2T + \sum_{k=1}^{N} 2kT = 2T\left(1 + \frac{N(N+1)}{2}\right)$$

Quindi

$$\tilde{\Lambda} = T[2 + N(N+1)]$$

Infine

$$\gamma = \frac{F}{\Lambda} = \frac{F}{T[2 + N(N+1)]}$$

Poichè

$$F = \frac{N(N+1)}{2}m$$

si ha

$$\tilde{\gamma} = \frac{N(N+1)m}{2T[2 + N(N+1)]}$$

Di conseguenza

$$\lim_{N \to \infty} \tilde{\gamma} = \frac{m}{2T} = \frac{R}{2}$$

risultato che può essere così interpretato: all'aumentare di N la dimensione delle finestra di congestione e quindi la quantità di bit trasmessi in tale finestra aumentano linearmente come km mentre il tempo di idle della linea aumenta come kT. Questo significa che dopo aver trasmesso i segmenti contenuti in una qualunque congestion window la linea rimane idle per una durata pari al tempo di trasmissione di tali pacchetti. Di conseguenza il rate R della linea viene sfruttato al 50%.