

Programmazione di Reti

Formulario

Luca Casadei - Francesco Pazzaglia - Martin Tomassi

18 maggio 2024

Indice

1	Ritardi di trasferimento	2
1.1	Tempo di trasmissione	2
1.2	Tempo di propagazione	2
1.3	Tempo totale	2
1.4	Quantità di bit presenti sul canale	3
1.5	Scenari Cut-Through e Store & Forward	3
1.5.1	Esempio con Store & Forward	3
1.5.2	Esempio con Cut-Through	3
2	Round-Trip-Time (RTT)	4
2.1	Capacità di propagazione	4
2.2	RTT	4
3	Efficienza protocolli "ARQ"	5
3.1	Go-back-N	5
3.2	Stop & Wait	5

Capitolo 1

Ritardi di trasferimento

1.1 Tempo di trasmissione

- $T_{trasmissione}$ = Tempo di trasmissione (s)
- L = Lunghezza del pacchetto (bit)
- R = Frequenza (capacità) di trasmissione (bit-rate) ($\frac{bit}{s}$)

Trasferimento di un pacchetto da router **A** a router **B**

$$T_{trasmissione} = \frac{L}{R}$$

1.2 Tempo di propagazione

- D = Lunghezza del collegamento (m)
- v = Velocità (ritardo) di propagazione ($\frac{m}{s}$)
- τ = Tempo di propagazione (s)

Si può ricavare il tempo di propagazione:

$$\tau = \frac{D}{v}$$

Nel caso della suddivisione del canale in n sotto-canali e considerando la lunghezza del canale totale D :

$$\tau_n = \frac{\tau}{n}$$

1.3 Tempo totale

Si ricava da:

$$T_{tot} = \tau + T_{trasmissione} + T_{accodamento} + T_{elaborazione}$$

1.4 Quantità di bit presenti sul canale

Si ricava attraverso:

$$L = R * T_{propagazione}$$

1.5 Scenari Cut-Through e Store & Forward

1.5.1 Esempio con Store & Forward

Consideriamo n elementi trasmissivi, avremmo n tempi di trasmissione:

$$n * T_{trasmissione}$$

Consideriamo ora k elementi che introducono latenza per accodamento e ritrasmissione, otteniamo:

$$k * T_{accodamento}$$

Con tempo di propagazione fisico τ

$$T_{totale} = n * T_{trasmissione} + k * T_{accodamento} + \tau$$

1.5.2 Esempio con Cut-Through

In questo caso si considera il tempo di accodamento del solo header e non di tutto il pacchetto, sapendo che per trasmettere un pacchetto trascurando eventuali tempi di elaborazione è: $T_H + (T - T_H)$, da questo si ottiene che con header H :

$$T_{totale} = n * T + k * T_H + \tau$$

Capitolo 2

Round-Trip-Time (RTT)

2.1 Capacità di propagazione

- $C = R$ come nella sezione precedente 1.1, è la capacità o frequenza di trasmissione.

2.2 RTT

- T_{ack} = Tempo acknowledge.

Equivale al $T_{trasmissione}$ più tutti i tempi di elaborazione e di accodamento $T_{accodamento}$, e anche dei tempi di propagazione τ , come nella sezione precedente 1.1, ma considerando tutta la tratta da percorrere sia per l'andata che per il ritorno (in genere si ha 2τ perché va considerata la propagazione di invio e ricezione.).

$$RTT = T_{trasmissione} + 2\tau + T_{ack}$$

Capitolo 3

Efficienza protocolli "ARQ"

- MSS = Maximum segment size (equivalente a L) 1.1.
- SSTRESH = Segment size threshold (soglia).
- CWND = Congestion window.
- RCWND = Receiver congestion window.
- $T_{trasmissione} = \frac{MSS}{R}$

Dimensione del file trasferibile nella window (unità di misura MSS) = $\frac{L}{CWND}$, considerando che la conversione in MSS si ottiene facendo: $\frac{RCWND}{CWND}$ e la conversione in MSS del threshold: $\frac{SSTHRESH}{CWND}$, in generale, si ha $\frac{L(\text{byte})}{CWND(\text{byte})} = MSS$.

3.1 Go-back-N

- W Numero di pacchetti della window (dimensione).
- T Tempo per pacchetto.

Dipende dal rapporto tra RTT e la lunghezza della finestra NT , consideriamo il tempo di trasmissione $T_{trasmissione}$ e il tempo di propagazione τ :

Dobbiamo avere $WT \geq RTT \implies$ sviluppando i calcoli si ottiene $W > x$ dove x è il numero di pacchetti da mandare prima dell'acknowledge per avere un'efficienza 1.

La trasmissione è continua quando $WT \geq RTT$
(discontinua quando $WT < RTT$)

3.2 Stop & Wait

- Efficienza $\eta = \frac{T_{trasmissione}}{T_{trasmissione} + T_{propagazione}}$