

Reti di Calcolatori

Appunti delle Lezioni di Reti di Calcolatori

Anno Accademico: 2024/25

Giacomo Sturm

*Dipartimento di Ingegneria Civile, Informatica e delle Tecnologie Aeronautiche
Università degli Studi “Roma Tre”*

Sorgente del file LaTeX disponibile al seguente link:

<https://github.com/00Darxk/Reti-di-Calcolatori/>

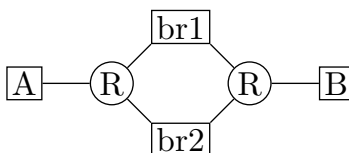
Indice

1	Esercitazione 23/10/24	1
1.1	Rete Bridge con Cicli	1
1.2	Effetto dello Store & Forward sulla Latenza	3
1.3	Learning nei Bridge	5

1 Esercitazione 23/10/24

1.1 Rete Bridge con Cicli

Si considera una rete locale IEEE 802.3 di topologia seguente:



Sono presenti due repeater R, e due computer A e B. Le connessioni avvengono solo su cavi utp, "Unshielded Twisted Pair", si suppone che questi bridge numerati 1 e 2, non siano in grado di realizzare lo spanning tree. Quindi tutta la rete, compreso il ciclo, è attiva. Si suppone che tutte le porte dei bridge siano pienamente attive. Si suppone che i bridge siano appena stati accesi, quindi i loro filtering database siano vuoti.

Il primo evento in questa rete locale è l'invio di un pacchetto da A e B.

Domanda 1

Determinare quanti pacchetti circolano nella rete dopo l'invio del singolo pacchetto:

Questo pacchetto (A->B) arriva ad entrambi i bridge **bridge 1** e **bridge 2**, che hanno un filtering database vuoto, per cui lo mandano su tutte le porte attive. Poiché il repeater può inviare ricevere un pacchetto alla volta si considera che il pacchetto del primo bridge arrivi per primo al repeater, nella contesa del dominio di collisione. Si considerano i due pacchetti copiati dai due bridge (A->B) ', il primo, e (A->B) ', il secondo. Ma il repeater invia il pacchetto ricevuto su tutte le direzioni, quindi oltre ad arrivare a B, arriva anche alla porta destra dell'altro bridge, che invia la sua copia del pacchetto dalla porta sinistra. Questo pacchetto incontra il repeater della sotto-rete A e viene ripetuto su tutte le porte del repeater. Quindi questo pacchetto oltre a raggiungere A viene inviato sulla porta sinistra dell'altro bridge, e si ripete la sequenza, per entrambi i bridge.

Quindi in questa situazione le copie del pacchetto (A->B) ' viene inviato continuamente in un ciclo antiorario, mentre (A->B) ' ' segue il percorso orario.

Domanda 2

Determinare cosa succede al filtering database di **bridge 1** e **bridge 2**:

Ogni volta che il pacchetto copiato raggiunge un bridge, considera la porta da cui è arrivato la nuova posizione della stazione A, così per entrambi i bridge. Quindi la sua posizione nel filtering database cambia continuamente.

Domanda 3

Determinare il comportamento dei bridge all'invio di un pacchetto (B->A):

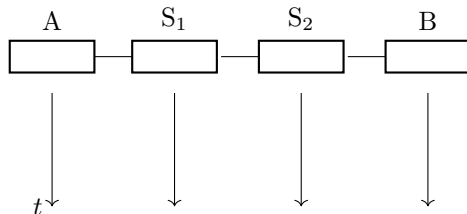
Se il pacchetto arriva ad entrambi i bridge quando la stazione A si trova nel filtering database sulla porta destra, il pacchetto non viene inviato ad A. Se solo uno presenta la stazione A alla porta sinistra, una singola copia raggiunge A. Se entrambi presentano la stazione A alla porta sinistra, questa riceve due volte il pacchetto.

Dati questi problemi non si accetta di avere un ciclo in una rete neanche per un tempo transitorio, poiché può saturare la rete ed impedire ai pacchetti necessari a passare. Gli algoritmi per determinare lo spanning tree sono molto conservativi, e prima di effettuare l'algoritmo interrompono il transito di tutti i pacchetti.

1.2 Effetto dello Store & Forward sulla Latenza

Si considera una rete mostrata nella figura seguente, dove A e B sono computer e S1 e S2 sono switch. Dove i cavi ethernet sono utp, ed il ritardo di propagazione è trascurabile. Si suppone che le schede di rete utilizzate sono tutte a 10 Mb/s.

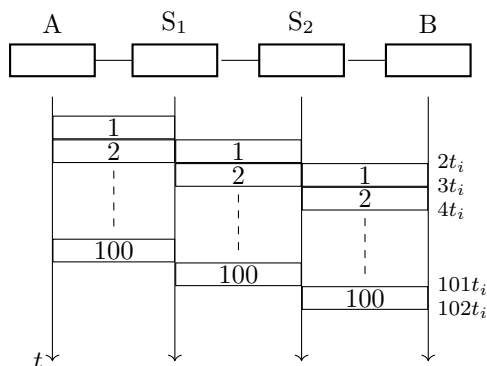
Questo disegno mostra cosa avviene sulla rete mentre è in funzione, rispetto al tempo:



Il computer deve spedire un file di 100000 bit a B, suddividendolo in 100 pacchetti. Si suppone che il trasferimento inizi all'istante $t = 0$, e si suppone di trascurare gli header e l'IPG. Si suppone che la rete sia a completa disposizione del trasferimento dei file, che gli switch non operino in modalità "cut-through". Si suppone che non siano utilizzati riscontri di alcun tipo.

Domanda 1

Completa il diagramma temporale riportato al di sotto della figura con la rete, mostrando la sequenza dell'invio dei pacchetti dai vari componenti:



Ad ogni istante lo switch ha un pacchetto in ricezione su una porta, ed un pacchetto in trasmissione sulla seconda porta.

Per inviare un singolo pacchetto sulla rete si impiega un tempo t_i :

$$t_i = \frac{100000 \text{ bit}/100 \text{ pk}}{10^7 \text{ bit/s}} = 10^{-4} \text{ s/pk} \quad (1.2.1)$$

In caso il ritardo non sia trascurabile, ogni segmento viene traslato verso il basso di questo ritardo.

Domanda 2

In quale istante tutti i bit sono arrivati a B:

Il primo pacchetto viene ricevuto completamente ad un tempo $3t_i$, poiché deve passare attraverso tre spazi intermedi, dopo il quale i pacchetti arrivano ogni t_i , e rimanendo 99 pacchetti, l'ultimo arriverà al tempo $102t_i$.

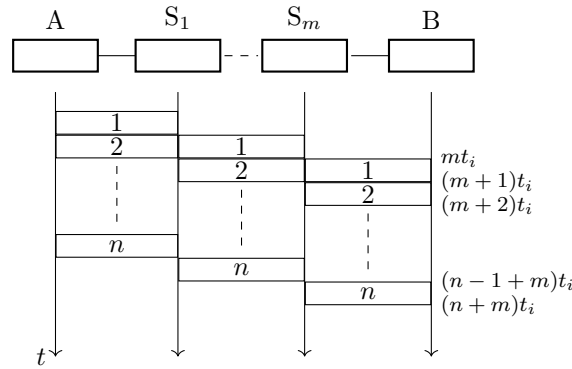
$$T = 102 t_i \quad (1.2.2)$$

Domanda 3

Determinare una formula generale per risolvere ogni problema di questo tipo con n pacchetti ed m switch:

Ogni switch aggiunge un altro stadio alla trasmissione, quindi aumenta il tempo di un fattore t_i , da aggiungere al numero di pacchetti totali:

$$T = (n + m) \cdot t_i \quad (1.2.3)$$



Domanda 4

Determinare una formula generale che tenga conto di ritardi sul primo, sul secondo, o sul terzo filo:

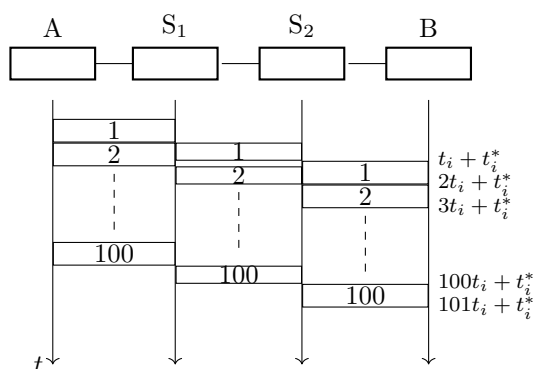
In caso di un ritardo su un certo filo, questo rappresenta il collo di bottiglia del sistema, e quindi il tempo totale ne risentirà considerevolmente. Il tempo in cui il primo bit del primo pacchetto arriva T_1 è dato dal tempo di attraversamento di ogni filo, quindi è $T_1 = 3t_i + t_i^r$, dove t_i^r rappresenta il tempo di ritardo su di un filo. Tutti gli altri $n - 1$ pacchetti arriveranno ad intervalli di $t_i + t_i^r$, quindi il tempo di arrivo dell'ultimo di n pacchetti trasmessi è:

$$T = T_1 + (n - 1) \cdot (t_i + t_i^r) = (n + 2) \cdot t_i + nt_i^r \quad (1.2.4)$$

Domanda 5

Si suppone di poter aumentare la connessione ad una banda di 100 Mb/s, ma questo aumento è possibile solo su uno dei tre fili, utilizzando una doppia scheda su di un filo. Anche se il tempo di trasmissione di un pacchetto in uno dei fili diventa il decimo: $t_i^* = t_i/10$, il resto dei fili rimane alla stessa banda, quindi rappresenta il collo di bottiglia del sistema. Quindi l'aumento di banda su un singolo filo non influisce sulla diminuzione del tempo necessario per tutti i pacchetti, ma solo per il primo $T_1 = 2t_i + t_i^*$. Tutti gli altri 99 pacchetti arrivano alla destinazione ogni t_i , come per il problema originario:

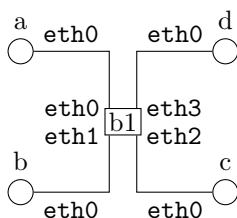
$$T = T_1 + 99t_i = 101t_i + t_i^* \quad (1.2.5)$$



1.3 Learning nei Bridge

Domanda 1

Si considera un bridge b1 con quattro porte **eth0**, **eth1**, **eth2** e **eth3**, a cui sono collegati quattro computer a, b, c, d:



Nella rete transitano cinque pacchetti in sequenza, inviati dopo che il precedente è giunto a destinazione:

- t_1 : (a->b);

- $t_2: (b \rightarrow a);$
- $t_3: (c \rightarrow d);$
- $t_4: (c \rightarrow d);$
- $t_5: (d \rightarrow a).$

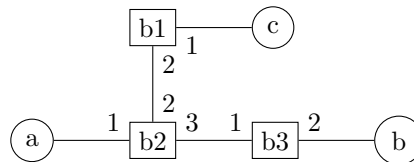
Descrivere il contenuto del filtering database di B1, dopo ciascun pacchetto, e specificare le porzioni di rete in cui i diversi pacchetti sono visibili.

Il pacchetto t_1 viene ricevuto dalla porta **eth0**, quindi viene assegnata ad A nel filtering database, e viene inviato su tutte le altre le porte. Il pacchetto t_2 viene ricevuto dalla porta **eth1**, questa porta viene associata al computer B, e sapendo dove si trova A, lo invia solamente sulla porta **eth0**. I pacchetti t_3 e t_4 effettuano un procedimento analogo a t_1 e t_2 , sulle porte **eth2** e **eth3**. All'arrivo del pacchetto t_5 , il bridge conosce l'intera topologia della rete.

Ma il filtering database non è statico, e viene dimenticato in intervalli di tempo prestabiliti, oppure se uno dei computer viene connesso ad un'altra porta e ne riceve un pacchetto. Quindi se uno di questi pacchetti arriva dopo che è stato resettato il filtering database, ricomincia il processo di learning.

Domanda 2

Si considera la seguente topologia di rete ad albero, con tre bridge b1, b2, e b3, e tre computer a, b e c:



Nella rete vengono trasmessi tre pacchetti, inviati in tre istanti di tempo successivi:

- $t_1: (a \rightarrow c);$
- $t_2: (c \rightarrow a);$
- $t_3: (b \rightarrow a).$

Descrivere il filtering database per all'invio di ciascun pacchetto, e specificare in quali porzioni di rete i pacchetti sono visibili:

All'invio del primo pacchetto tutti i filtering database sono vuoti, quindi il pacchetto t_1 viene inviato su ogni porzione della rete. Dopo il suo invio, il bridge b2 ha sulla porta 2 la stazione a, il bridge b3 ha sulla porta 1 la stazione a, ed il bridge b1 la ha sulla porta 2. All'invio del pacchetto t_2 , sia il bridge uno che due sanno dove si trova la stazione a, quindi il pacchetto viene prima inviato dalla stazione c a b1, poi da b1 all'altro bridge b2, poi da b2 alla stazione di arrivo a. Alla fine della

trasmissione i bridge uno e due hanno la stazione c rispettivamente sulla porta 1, e 2. All'invio dell'ultimo pacchetto t_3 , entrambi i bridge b3 e b2 hanno la stazione a nel loro filtering database, quindi effettua un percorso da b a b3, a b2 fino ad a. Il bridge b2 e b3 conoscono la posizione della stazione b rispettivamente sulle porte 3 e 2.

Domanda 3

Considerando la stessa rete dell'esercizio seguente, come cambia il filtering database se i pacchetti precedenti vengono inviati nell'ordine:

- $t_1:(c \rightarrow a);$
- $t_2:(b \rightarrow a);$
- $t_3:(a \rightarrow c).$

I primi due pacchetti vengono trasmessi su tutta la rete, mentre solo l'ultimo viene trasmesso solo sulle porte necessarie, dato che i bridge conoscono la posizione del destinatario. Dopo il primo pacchetto i bridge b1, b2 e b3, conoscono la posizione della stazione c, rispettivamente sulla porta 1, 2, 1. All'invio del secondo pacchetto analogamente conoscono la posizione della stazione b sulle porte, 2, 2, 2. Mentre all'invio dell'ultimo pacchetto i bridge b1 e b2 conoscono la posizione della stazione a sulle porte 1 e 2. Rispetto alla domanda precedente, il bridge b3 ora conosce la posizione della stazione c, ma il bridge b3 non conosce la posizione della stazione a.