

Reti di calcolatori e internet

Appunti

Alessandro Ferro

Introduzione

Questi appunti, presi dal libro "Reti di Calcolatori e Internet - un approccio Top Down" (Kurose, Ross, 7th edition) sono parziali in quanto una parte sono presi a penna. La numerazione dei capitoli, sezioni e paragrafi segue quella del libro.

Capitolo 1

Reti di calcolatori e Internet

[Le sezioni 1.1 e 1.2 sono scritte a penna sul quaderno]

1.3 Il nucleo della rete

1.3.2 Commutazione di circuito

Per spostare i dati in una rete esistono due approcci: commutazione di pacchetto e commutazione di circuito.

Nelle reti a commutazione di circuito, le risorse necessarie sono allocate per l'intera durata della comunicazione tra i sistemi periferici. Nelle reti a commutazione di pacchetto, invece, tali risorse non sono riservate e sono allocate in funzione della necessità della rete.

Le reti telefoniche sono esempi di rete a commutazione di circuito in quanto il canale resta aperto anche in assenza di comunicazione.

Quando la rete stabilisce un circuito, viene riservata anche una certa larghezza di banda, pertanto, il mittente può trasferire i dati a una velocità costante garantita.

D'inverso, quando un host invia un pacchetto a un altro host su una rete a commutazione di pacchetto, esso viene immesso nella rete senza che vengano ad esso riservate risorse.

Multiplexing nelle reti a commutazione di circuito

In una rete a commutazione di circuito può essere implementato il multiplexing, ovvero l'utilizzo dello stesso mezzo trasmissivo da più utenti.

Il multiplexing viene implementato tramite:

- **Suddivisione di frequenza:** lo spettro di frequenza di un collegamento viene suddiviso tra le connessioni stabilite. Ciò, ad esempio, è il caso di walkie-talkie che comunicano su un circuito a una frequenza diversa da altri walkie-talkie che comunicano sullo stesso circuito.
- **Suddivisione di tempo:** quando la rete stabilisce una connessione a commutazione di circuito, assegna degli slot temporali per ogni sessione.

Confronto tra commutazione di pacchetto e commutazione di circuito

I denigratori della commutazione di pacchetto sostengono che quest'ultimo mal si adatti alle applicazioni in tempo reale a causa dei suoi ritardi variabili e non deterministici. I suoi sostenitori, invece, affermano che la commutazione di pacchetto non soltanto offra una migliore condivisione della larghezza di banda, ma è anche più semplice, efficiente ed economica.

La commutazione di pacchetto risulta essere più efficiente perché la commutazione di circuito pre-allocava l'uso del collegamento trasmissivo indipendentemente dalla richiesta necessaria, con collegamenti garantiti ma potenzialmente non necessari. D'altro canto la commutazione di pacchetto alloca le risorse su richiesta.

1.3.3 Una rete di reti

Gli ISP si distinguono per la copertura geografica: gli ISP di accesso, che forniscono Internet direttamente all'utente finale, sono connessi agli ISP regionali che a loro volta sono connessi agli ISP di primo livello (cosiddetti ISP globali). Tale configurazione permette a tutti i nodi della rete di essere in comunicazione.

Per ridurre i costi una coppia di ISP vicini e di pari livello gerarchico può fare uso di peering, cioè connettere direttamente le loro reti in modo tale che il traffico passi direttamente tra di loro anziché passare per un intermediario.

1.4 Ritardi, perdite e throughput nelle reti a commutazione di pacchetto

Idealmente, vorremmo che i servizi Internet siano in grado di spostare dati tra due sistemi periferici istantaneamente e senza nessuna perdita dati. Ciò, ovviamente, non è fattibile nella realtà: il throughput è necessariamente limitato, ci possono essere ritardi e si potrebbero perdere pacchetti.

1.4.1 Panoramica del ritardo nelle reti a commutazione di pacchetto

Un pacchetto parte da un host, passa una serie di router e conclude il viaggio in un altro host. In ciascun nodo lungo il percorso (inclusi sorgente e destinazione) il pacchetto subisce vari tipi di ritardo. I principali ritardi sono dovuti al ritardo di elaborazione, ritardo di accodamento, ritardo di trasmissione e ritardo di propagazione che complessivamente formano il ritardo totale di nodo.

La figura 1.16 mostra un ottimo schema che riassume i tipi di ritardo.

Ritardo di elaborazione

Il tempo richiesto per esaminare l'intestazione del pacchetto e per determinare dove dirigerlo, nonché il tempo richiesto per controllare eventuali errori al suo interno fa parte del ritardo di elaborazione.

Ritardo di accodamento

Una volta in coda, il pacchetto subisce un ritardo di accodamento mentre attende la trasmissione sul collegamento. Ne abbiamo già parlato in 1.3.1 "ritardi di accodamento e perdita di pacchetti"

Ritardo di trasmissione

Sia L la lunghezza del pacchetto in bit e R la velocità di trasmissione in bps dal router A al router B, il ritardo di trasmissione è $\frac{L}{R}$ secondi e risulta essere il tempo richiesto per trasmettere tutti i bit del pacchetto sul collegamento.

Ritardo di propagazione

Una volta immesso sul collegamento, il pacchetto deve propagarsi fino al successivo nodo della rete. Questo tempo si chiama ritardo di propagazione. Il ritardo di propagazione è dato da d/v dove d è la distanza tra i due router e v è la velocità di propagazione nel collegamento (in m/s). Questo ritardo è puramente fisico: indica quanto tempo impiega un singolo bit a "viaggiare" dal mittente al destinatario una volta che è stato trasmesso. Questo ritardo solitamente è trascurabile.

Confronto tra ritardi di trasmissione e di propagazione

Il ritardo di trasmissione è la quantità di tempo impiegata dal router per trasmettere in uscita il pacchetto ed è funzione della lunghezza del pacchetto e della velocità di trasmissione sul collegamento, ma non ha niente a che fare con la distanza tra i due router. Come detto poc'anzi, è il tempo richiesto per trasmettere tutti i bit del pacchetto sul collegamento.

Il ritardo di propagazione, invece, è il tempo richiesto per la propagazione di un solo bit da un router a quello successivo ed è funzione della distanza tra i due router e della velocità di propagazione nel mezzo trasmissivo ma non ha niente a che fare con la dimensione del pacchetto o la velocità di trasmissione.

Nota come velocità di trasmissione \neq velocità di propagazione.

Nome (simbolo)	Significato	Unità tipica
1 Velocità di propagazione (v)	Velocità con cui un segnale elettrico/ottico si propaga fisicamente lungo il mezzo (rame, fibra, aria, ecc.).	metri/secondo (m/s)
2 Velocità di trasmissione (o data rate) (R)	Velocità con cui i bit vengono trasmessi sul collegamento, cioè quanti bit al secondo vengono "iniettati" nel canale.	bit/secondo (bps)

Table 1.1: Confronto tra velocità di propagazione e velocità di trasmissione.

Formula

Siano d_{elab} , d_{acc} , d_{tra} , d_{prop} , i ritardi, rispettivamente di elaborazione, accodamento, trasmissione e propagazione. Allora, il ritardo totale di un nodo è dato da:

$$d_{elab} + d_{acc} + d_{tra} + d_{prop}$$

1.4.2 Ritardo di accodamento e perdita di pacchetti

A differenza degli altri tre ritardi, quello di accodamento può variare da pacchetto a pacchetto. Ciò vuol dire che mentre elaborazione, trasmissione e propagazione sono costanti per ogni pacchetto che viaggia su quella specifica tratta, il ritardo di accodamento è variabile. Per tale motivo, nel caratterizzare il ritardo di accodamento si fa uso di misure statistiche quali il ritardo di accodamento medio e la sua varianza.

Denotiamo con a la velocità di arrivo di pacchetti nella coda espressa in pacchetti al secondo, R la velocità di trasmissione alla quale i bit vengono trasmessi in uscita alla coda espressa in bit al secondo e supponiamo che tutti i pacchetti abbiano dimensione L bit. Pertanto arrivano al nodo La bit/s.

Con La/R si denota l'intensità di traffico e il risultato è un numero adimensionale (es. $\frac{500bps}{250bps} = 2$)

- (1) Se $La/R > 1$ vuol dire che stanno entrando più pacchetti di quanti il nodo può trasferirne in uscita e la coda cresce verso ∞ bit.

Questo perché $\frac{La}{R} > 1$ implica che $La > R$

- (2) Se $La/R \leq 1$ succede quanto segue: Innanzitutto nota che se ogni pacchetto ha dimensione L vuol dire che il nodo impiega $\frac{L}{R}$ secondi per smaltirlo.

- (2A) Se $a = 1$ vuol dire che $La = L$, ovvero arriva un pacchetto al secondo, ognuno di L bit. Poiché $La/R \leq 1$ allora vuol dire che $L/R \leq 1$ ovvero un pacchetto viene elaborato in al più 1 secondo. Poiché $a = 1$ vuol dire che arriva un pacchetto al secondo e quindi ogni pacchetto trova sempre la coda vuota.

Ciò varrebbe anche nel caso in cui $L/R = 1$ perché all'istante 1s il primo pacchetto ha completamente lasciato la coda, e allo stesso istante, il secondo trova la coda vuota.

- (2B) Adesso analizziamo il caso in cui $a > 1$. Nota che se $\frac{L}{R}$ sta ad indicare i secondi necessari per smaltire un pacchetto, $\frac{R}{L}$ sta ad indicare quanti pacchetti sono smaltiti al secondo.

Assodato ciò, notiamo che $\frac{La}{R} \leq 1 \rightarrow \frac{L}{R} \leq \frac{1}{a} \rightarrow \frac{R}{L} \geq a$

Notiamo che i pacchetti smaltibili ogni secondo sono più dei pacchetti che arrivano ogni secondo (o in egual misura).

Una dimostrazione equivalente è notare che se $La/R \leq 1$ allora $R \geq La$ che indica che i bit smaltibili al secondo sono più dei bit in ingresso al secondo (o in egual misura).

Ciò implica che:

(I) Ad ogni secondo, il primo degli a pacchetti troverà sempre la coda vuota, come avviene nel caso 2A (di cui 2B ne è una generalizzazione) e

(II) Alcuni degli $a - 1$ pacchetti potrebbero non trovare la coda vuota e quindi aspettare. Abbiamo detto che $\frac{L}{R}$ sono i secondi per smaltire un pacchetto. Se gli a pacchetti arrivano con un ritardo temporale di almeno $\frac{L}{R}$ l'uno dall'altro, allora ogni pacchetto troverà la coda vuota, ma se uno o più pacchetti arrivano più velocemente (cosiddetto "burst"), allora sono costretti ad aspettare che il pacchetto (o i pacchetti) precedenti escano dalla coda. Il caso limite è in cui gli a pacchetti arrivino contemporaneamente. In questo caso il primo pacchetto viene immediatamente elaborato, il secondo pacchetto deve aspettare L/R secondi, il terzo deve aspettare $2 * \frac{L}{R}$, il quarto deve aspettare $3 * \frac{L}{R}$ secondi e così via. Più in generale, il pacchetto N° deve aspettare $(N - 1) * \frac{L}{R}$.

Per rendere più chiaro il concetto facciamo un esempio. Si supponga che arrivino 10 pacchetti all'istante 0 ognuno grande 50bit con una velocità di trasmissione R pari a 1000bit/s. Il primo pacchetto viene elaborato subito, ma il primo bit del secondo pacchetto viene elaborato a tempo $\frac{L}{R} = \frac{50}{1000} = 0.05s$.

Ciò dimostra che sebbene la condizione $La/R \leq 1$ è necessaria per stabilità a lungo termine (la coda non diverge verso ∞), non garantisce assenza di attese.

Perdita di pacchetti

La quantità di pacchetti perduti aumenta in proporzione all'intensità di traffico in quanto la coda potrebbe non essere abbastanza capiente per soddisfare la richiesta. Pertanto, le prestazioni di un nodo sono misurate non solo in termini di ritardo ma anche dalla probabilità di perdita di pacchetti.

1.4.3 Ritardo end to end

Nella sottosezione 1.3.1 (paragrafo "trasmissione store e forward") abbiamo detto che, supponendo di avere $N-1$ nodi tra sorgente e destinazione (e quindi N collegamenti) il ritardo end-to-end è

$$d_{e2e} = N(d_{trasmissione}) = N\left(\frac{L}{R}\right)$$

ma prendevamo in considerazione solo il ritardo di trasmissione. La formula generale è

$$d_{e2e} = N(d_{elab} + d_{accodamento} + d_{trasmissione} + d_{prop})$$

Traceroute

Traceroute è un programma eseguibile su qualsiasi host di Internet. Quando l'utente specifica il nome di un host di destinazione, il programma invia pacchetti speciali verso di essa, che, durante il loro percorso, passano attraverso una serie di router. Quando uno di questi router riceve questo pacchetto speciale, invia un messaggio che torna alla destinazione che contiene il nome e l'indirizzo del router.

Il funzionamento è il seguente: Supponiamo di avere $N-1$ router dalla sorgente alla destinazione. La sorgente invia N pacchetti ognuno dei quali contiene l'indirizzo della destinazione. Quando l' n -esimo router riceve il pacchetto marcato come n , anziché indirizzarlo verso la destinazione risponde al mittente come poc'anzi descritto. L' N -esimo pacchetto, che raggiunge la destinazione, invita anche quest'ultima a rispondere.

In questo modo l'origine può ricostruire il percorso intrapreso dai pacchetti ed è inoltre in grado di determinare i ritardi per ogni nodo.

Sistemi periferici, applicazioni e altri ritardi

Oltre ai ritardi descritti nelle sezioni precedenti, si potrebbero manifestare ulteriori ritardi nei sistemi periferici. A titolo esemplificativo, un sistema periferico può volontariamente ritardare la sua trasmissione in quanto condivide il mezzo trasmissivo con altri sistemi periferici.

Un altro esempio è nel VoIP: In VoIP il mittente deve prima di tutto riempire il pacchetto con conversazione digitalizzata prima di inviarlo su Internet. Questo tempo per riempire un pacchetto è detto ritardo di pacchettizzazione.

1.4.4 Throughput nelle reti di calcolatori

Un'altra misura critica delle prestazioni in una rete di calcolatori è il throughput end-to-end.

Consideriamo un trasferimento di file da A a B. Il throughput istantaneo in ogni istante di tempo è la velocità di bit/s alla quale B sta ricevendo il file. È misurato in B e non in A in quanto è l'unico modo per individuare la velocità effettiva di trasferimento dopo tutti i ritardi e colli di bottiglia che intercorrono nel percorso.

Se il file consiste di F bit e il trasferimento richiede T secondi affinché la destinazione riceva tutti i bit, allora il throughput medio è F/T bit/s.

Supponiamo che A e B si stiano trasferendo un file e che tra di loro ci sia un router C. Supponiamo R_1 essere il throughput da A a C e R_2 il throughput da C a B. Se $R_1 \leq R_2$ allora la velocità di collegamento da A a B (throughput end to end) è R_1 ma se $R_1 > R_2$ la velocità di collegamento è R_2 in quanto

il router fa da collo di bottiglia. Quindi il throughput end to end è $MIN(R_1, R_2)$ o più in generale $MIN(R_1, \dots, R_{n+1})$ se ci sono n router.

1.5 Livelli dei protocolli e loro modelli di servizio

1.5.1 Architettura a livelli

Un'architettura a livelli consente di discutere e analizzare una parte specifica e ben definita di un sistema complesso. Ciò permette di introdurre un ulteriore vantaggio: la modularità, che rende molto più facile cambiare l'implementazione di un servizio fornito da un determinato livello. Fino a quando il livello fornisce lo stesso servizio allo strato superiore e utilizza gli stessi servizi dello strato inferiore, la parte rimanente del sistema (ovvero gli altri livelli) rimane invariata al variare dell'implementazione del livello.

Stratificazione dei protocolli

I protocolli, nonché l'hardware e il software che li implementano, sono organizzati in livelli. Ciascun protocollo appartiene a uno dei livelli. Ogni livello fornisce il suo servizio (1) effettuando determinate azioni all'interno del livello stesso e (2) utilizzando i servizi del livello immediatamente inferiore (se c'è).

Un livello può essere implementato via hardware, software o con una combinazione di essi.

L'insieme dei protocolli di tutti i livelli viene chiamato pila di protocolli. Internet è formato da una pila a 5 livelli. Questi livelli sono: fisico, collegamento, rete, trasporto e applicazione. In ambito accademico si studia anche la pila a 7 livelli denominata ISO/OSI.

Un protocollo, per sua natura, serve per scambiare messaggi. Pertanto, un protocollo "vive" su più sistemi di rete.

Degli svantaggi dell'architettura a strati sono la possibilità che un livello duplichi le funzionalità di un livello inferiore e che un livello possa prelevare informazioni da altri livelli bypassandone i servizi esposti. Ciò viola lo scopo della separazione tra livelli.

Livello di applicazione

Il livello di applicazione è la sede dei protocolli usati dalle applicazioni di rete. Alcuni dei protocolli a questo livello sono HTTP, SMTP e FTP. I pacchetti di informazione del livello di rete vengono denominati messaggi.

Livello di trasporto

Il livello di trasporto trasferisce i messaggi del livello di applicazione. Vi sono due protocolli di trasporto: TCP e UDP.

- TCP: fornisce alle applicazioni un servizio orientato alla connessione (cioè l'utente deve stabilire una connessione, usarla e quindi rilasciarla), garantisce la consegna dei messaggi di applicazione e il loro ordine. Gestisce il controllo di flusso (ovvero la corrispondenza tra le velocità del mittente e destinatario) e ha inoltre un controllo di congestione della rete.
- UDP: questo protocollo è molto semplice. Non è orientato alla connessione e non garantisce affidabilità, controllo di flusso e controllo della congestione.

I pacchetti a livello di trasporto sono denominati segmenti.

Livello di rete

Il livello di rete si occupa di trasferire i pacchetti a livello di rete, detti datagrammi, da un host a un altro.

Il livello di trasporto passa al livello di rete il proprio segmento e un indirizzo di destinazione.

Il livello di rete comprende il protocollo IP.

Livello di collegamento

Per trasferire un pacchetto da un nodo a quello successivo sul percorso, il livello di rete si affida ai servizi del livello di collegamento. Esempi di protocolli a livello di collegamento includono Ethernet e WiFi.

Un datagramma potrebbe essere gestito da differenti protocolli a livello di collegamento lungo le diverse tratte che costituiscono il suo percorso.

Chiameremo frame i pacchetti a livello di collegamento.

Livello fisico

Mentre il compito del livello di collegamento è spostare frame tra nodi adiacenti, il ruolo del livello fisico è spostare i singoli bit. I protocolli di questo livello dipendono dall'effettivo mezzo trasmissivo.

1.5.2 Incapsulamento

Dalla sorgente, i dati scendono lungo la pila dei protocolli e vi risalgono nella destinazione. Nei commutatori, i dati scendono e salgono fino a un determinato livello: per i commutatori a livello di collegamento fino al livello 2, per i router fino al livello 3. Gli host implementano tutti e cinque i livelli.

In altre parole, presso un host mittente, un messaggio a livello di applicazione viene passato al livello di trasporto. Questo livello prende il messaggio e gli concatena informazioni aggiuntive (le informazioni di intestazioni) che verranno utilizzate dal protocollo di trasporto nell'host ricevente. Messaggio livello applicazione + intestazioni del protocollo di trasporto costituiscono il segmento.

Il protocollo di trasporto passa il segmento al livello di rete, il quale gli concatena le proprie intestazioni, formando il datagramma e così via scendendo nella pila.

Quindi a ciascun livello il pacchetto ha due tipi di campi: quello di intestazione e quello di payload (il carico utile). Il payload è tipicamente un pacchetto proveniente dal livello superiore.

Capitolo 2

Livello di applicazione