

Appunti di Reti di Computer

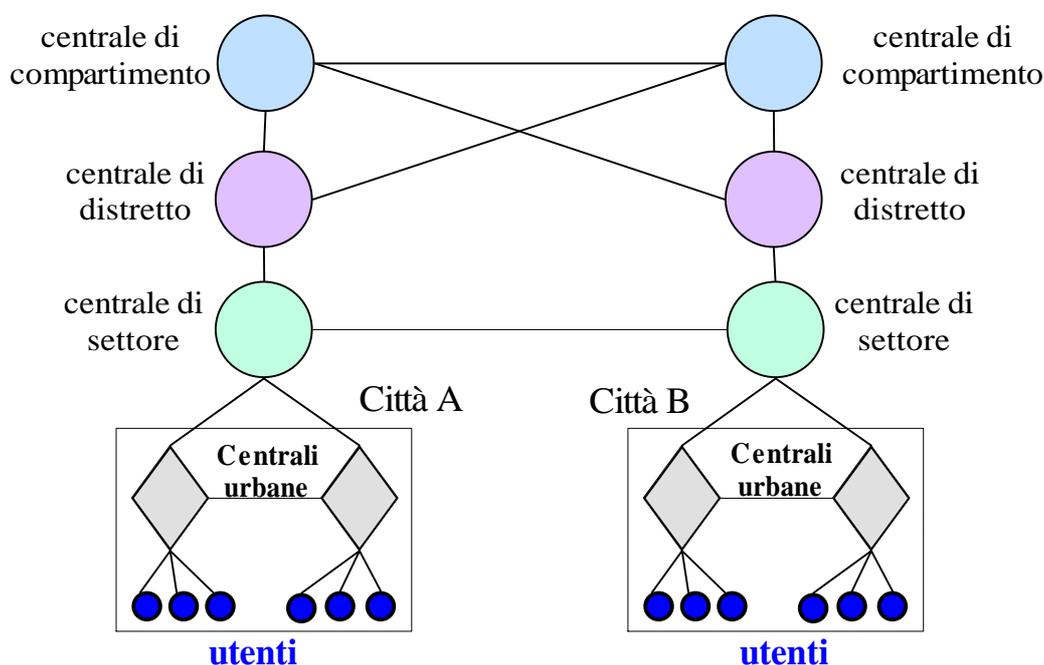
Capitolo 7 - Commutazione e instradamento

La commutazione.....	1
Commutazione di circuito	2
Cenni al funzionamento di una centrale telefonica urbana	3
Il message switching.....	5
Commutazione di pacchetto (packet switching).....	6
L'instradamento	8
Introduzione	8
Definizioni varie	11
Instradamento a circuito virtuale	11
Tecnica del "Packet Flooding"	13
Tecnica del "Random Routing"	14
Instradamento adattativo Arpanet	14
Problemi tipici delle reti a commutazione di pacchetto.....	16

La commutazione

COMMUTAZIONE DI CIRCUITO

La necessità della commutazione risulta evidente se pensiamo che, senza di essa, ogni utente della **rete telefonica pubblica** dovrebbe avere tante linee punto-a-punto permanenti quanti sono gli utenti della rete con cui esso vuole parlare (al limite tutti gli utenti della rete). Al contrario, essendo le centrali telefoniche pubbliche dotate della **funzione di commutazione**, la cosa è più semplice: l'apparecchio telefonico di ogni utente è collegato, tramite una *linea dedicata*, solo alla **centrale urbana** più vicina:



All'atto di una *chiamata*, la centrale interpreta il numero dell'utente chiamato e provvede a realizzare un **circuito fisico** che collega l'apparecchio di chi chiama a quello di chi risponde.

Questa è la tecnica cosiddetta a **commutazione di circuito** ed è valida non solo per le conversazioni telefoniche, ma anche per la trasmissione dati: il terminale dell'utente è collegato alla centrale telefonica (o a qualche altra apparecchiatura con funzioni simili, come ad esempio un PABX), la quale provvede a stabilire un circuito fisico tra il DTE chiamante ed il DTE remoto che si vuol contattare.

L'alternativa a questo modo di procedere è la cosiddetta tecnica a **commutazione di pacchetto**, che sarà descritta più avanti.

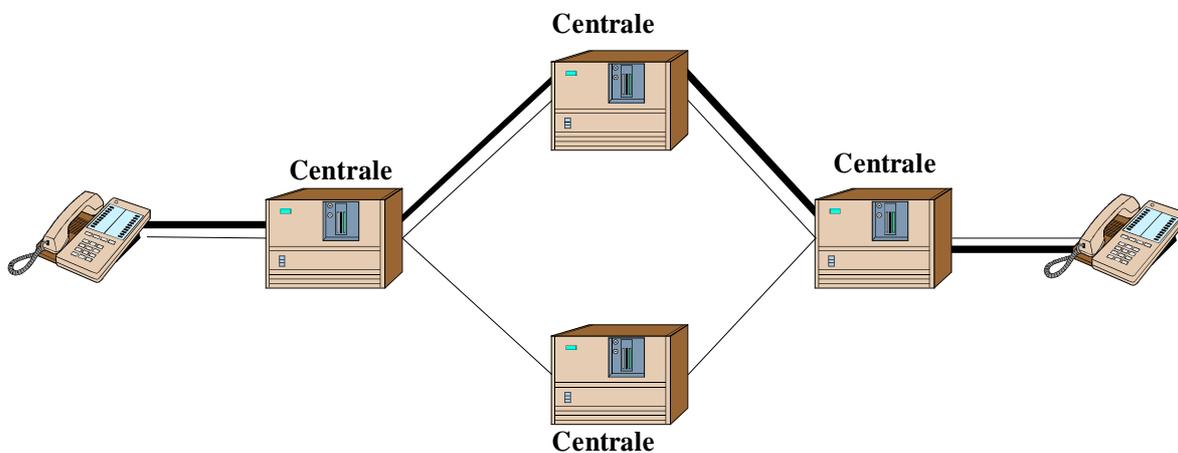
*La più antica forma di commutazione è proprio quella tramite centrale telefonica pubblica: c'è una fase iniziale di chiamata, necessaria affinché il numero dell'utente chiamato arrivi alla prima **centrale urbana** e questa inoltri il segnale su una linea di uscita diretta verso un'altra centrale; il procedimento segue fino a raggiungere la centrale cui è connesso l'utente chiamato; in questo modo, le varie centrali (*urbana*, di *settore*, di *distretto*, di *compartimento*) mettono in serie segmenti di cavo trasmissivo (o, in alternativa, il loro equivalente su un cavo a banda larga, nel quale quindi ad ogni conversazione viene riservato un canale, ossia una*

certa banda di frequenze) fino a formare il circuito completo tra utente chiamante e utente chiamato. Una volta realizzato tale circuito, la comunicazione può avere luogo. La tariffa di tale circuito può essere applicata in vari modi; in generale, essa cresce proporzionalmente alla distanza ed al tempo di attivazione del circuito.

Da notare che le centrali hanno un ruolo puramente passivo: ad esempio, nel caso della trasmissione dati, esse non memorizzano i messaggi trasmessi né, salvo funzioni di valore aggiunto espressamente richiamate dall'utente, convertono il protocollo di linea utilizzato. Se, a causa di un sovraccarico di lavoro, non dispongono di linee di uscita libere, esse bloccano il collegamento direttamente in fase di chiamata¹.

CENNI AL FUNZIONAMENTO DI UNA CENTRALE TELEFONICA URBANA

Possiamo dare dei cenni sulla logica con cui una centrale urbana controlla le linee degli utenti locali per determinare e realizzare il circuito fisico tra utente chiamante e utente chiamato.



Si definisce **off-hook** l'azione con cui si avverte la centrale che si intende fare una chiamata: se si tratta di una conversazione telefonica, per cui si usa l'apparecchio telefonico, l'*off-hook* consiste semplicemente nel sollevare la cornetta, il che lascia partire un impulso elettrico di avviso verso la centrale; nel caso, invece, di una trasmissione dati, per la quale si usa tipicamente un modem collegato alla linea telefonica, la funzione di *off-hook* è svolta dal modem stesso.

In centrale è presente un **registro** in cui viene memorizzato il numero dell'utente chiamato: se c'è un registro disponibile, significa che la centrale è in grado di effettuare la chiamata, per cui la centrale restituisce all'utente un segnale di **linea pronta**, indicante appunto la possibilità di procedere. Se invece non c'è alcun registro disponibile, bisogna riprovare.

Una volta ottenuta la linea libera, si compone il numero dell'utente con cui si vuol comunicare e tale numero viene registrato nell'apposito registro selezionato nella fase precedente. Segue l'importante fase per la **determinazione del percorso**, che assume una particolare rilevanza nel caso in cui la chiamata è interurbana, il che significa che saranno coinvolte almeno due centrali

¹ Facciamo osservare che questo è un criterio assolutamente generale, valido per qualsiasi rete di telecomunicazioni: quando una rete riceve una richiesta di servizio da parte di un utente, deve per prima cosa rendersi conto che sono disponibili le risorse necessarie a soddisfare la richiesta con la qualità richiesta; in caso affermativa, la rete impegna tali risorse e soddisfa la richiesta; in caso contrario, la rete deve necessariamente respingere in partenza la richiesta. Ovviamente, se al momento della richiesta di servizio dovessero risultare disponibili le risorse richieste, non è comunque escluso che, una volta accettata la richiesta, tali risorse vengano a mancare, per periodi di tempo più o meno lunghi: in queste situazioni, la rete deve necessariamente garantire il proseguimento del servizio e di questo bisogna assicurarsi in sede progetto.

urbane. Per fare un esempio concreto, supponiamo che tra la centrale X dell'utente chiamante e la centrale Y dell'utente chiamato si possano scegliere due strade: quella che passa per le centrali intermedie M ed N e quella che passa per le centrali intermedie P, Q ed R. A prescindere dalla distanza fisica tra le centrali, è ovvio che la seconda possibilità comporta sia un allungamento del tempo complessivo della fase di chiamata, dato che sono coinvolte 3 centrali anziché 2, sia anche un maggior onere della rete, dato che il circuito fisico da realizzare impegna parte delle risorse di 5 centrali anziché di 4.

Questo esempio aiuta a capire la criticità degli **algoritmi di instradamento** delle chiamate: un buon algoritmo di instradamento dovrebbe cercare sempre di individuare il percorso con il minor numero di nodi intermedi e ricorrere agli altri solo quando la prima scelta non risultasse agibile (ad esempio a causa del malfunzionamento di una o più centrali). Il cosiddetto **segnale di occupato** viene inviato all'utente chiamante solo quando nessuna strada in uscita risulta disponibile.

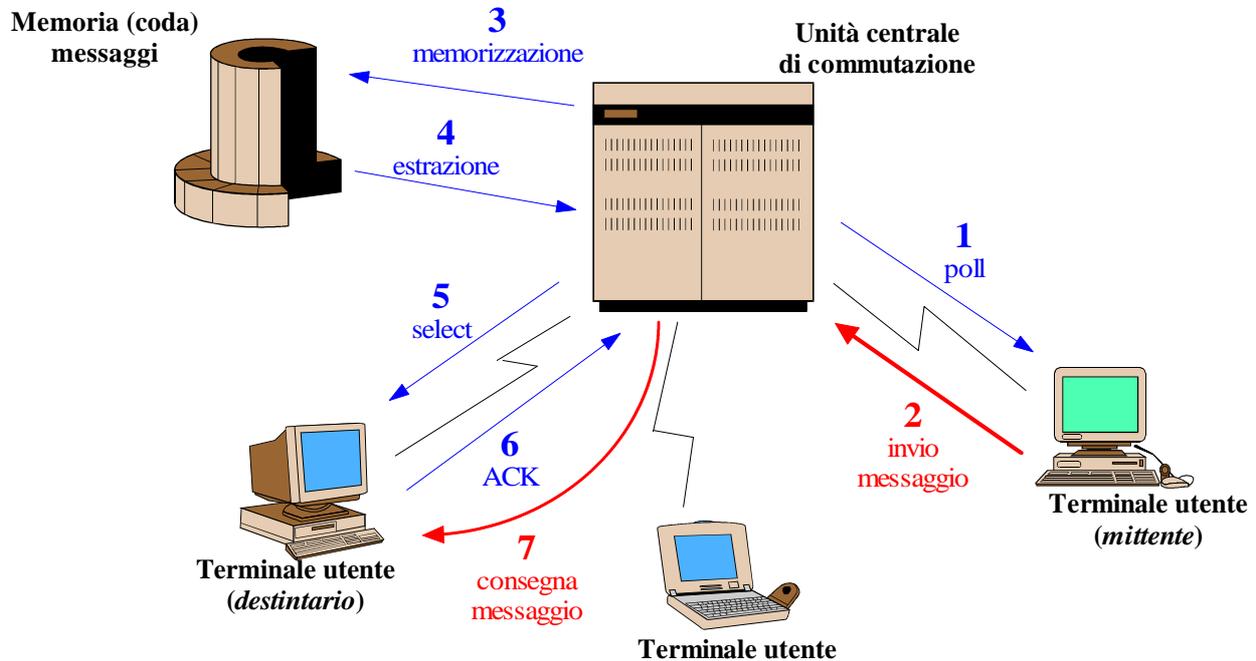
Una volta riscontrata la disponibilità di una linea tra l'utente chiamante e la centrale dell'utente chiamato, quest'ultima riceve il numero dalla centrale dell'utente chiamante; questo consente di individuare direttamente l'utente chiamato. A questo punto, segue una ulteriore verifica: bisogna vedere se la linea dell'utente chiamato è libera. In caso negativo, il chiamante riceve i toni di **occupato**. Se, invece, la linea è libera, viene inviato al chiamato un segnale che faccia suonare il telefono. Non appena l'utente chiamato alza la cornetta, la connessione viene istantaneamente stabilita e quindi può partire la conversazione.

Come è noto, *le centrali di commutazione della rete telefonica pubblica sono state per lungo tempo di tipo elettromeccanico, mentre solo recentemente sono state introdotte prima quelle elettroniche, ma sempre analogiche, e finalmente quelle numeriche, che al giorno d'oggi sono la quasi totalità.* Nelle centrali elettromeccaniche, tutto si basava sul movimento di organi elettromeccanici: in particolare, a muoversi erano i **connettori** che collegavano la linea in ingresso a quella di uscita. I principali inconvenienti di questa soluzione erano nel tempo non trascurabile necessario per effettuare la connessione e nel fatto che il movimento di organi elettromeccanici causa disturbi sulle linee. Tali disturbi possono anche essere tollerati in una conversazione telefonica, ma non certamente in una trasmissione dati. Con le centrali elettroniche, invece, e soprattutto con quelle numeriche, i problemi di rumore sono scomparsi e si sono inoltre ottenuti altri grandi vantaggi: oltre alle elevate velocità di trasmissione ed agli elevati volumi di traffico gestibili, ottenibili grazie al fatto che il collegamento tra centrali numeriche viene effettuato tramite linee ad altissime velocità ed a larga banda (tipicamente fibre ottiche), sono possibili funzioni aggiuntive come il *log* (registro) degli errori, la contabilizzazione, i servizi aggiuntivi (voice mailing,...) e l'assistenza agli utenti (numero chiamato disabilitato, cambiato, ...). Non solo, ma risultano anche più agevoli gli interventi di gestione e manutenzione.

IL MESSAGE SWITCHING

Negli anni '60 e '70, il metodo più comune per commutare dati tra diversi utenti è stato il cosiddetto **message switching**, tra le cui principali applicazioni c'era la *posta elettronica*.

Il meccanismo è illustrato nella figura seguente:



Descrizione del meccanismo con cui avviene la consegna di un messaggio: 1) poll del terminale 2) invio messaggio 3) memorizzazione del messaggio in coda 4) estrazione del messaggio dalla coda 5) select del terminale destinatario 6) ACK da parte del terminale destinatario 7) consegna del messaggio

L'**unità di commutazione** è un computer, che riceve i messaggi dai terminali collegati (tramite linea dedicata o commutata), esamina l'indirizzo del destinatario indicato nella testata (*header*) del messaggio ed instrada infine il messaggio verso il destinatario.

A differenza delle centrali che realizzano la commutazione di circuito (come le centrali telefoniche prima esaminate), i sistemi basati su *message switching* sono di tipo **store and forward**: i messaggi in arrivo vengono memorizzati in code² su disco, che possono essere più di una se vi sono diversi livelli di *priorità*. Non si tratta dunque di sistemi che lavorano in tempo reale, anche se in certi casi è possibile evitare l'accodamento di certi messaggi: infatti, i messaggi accodati ad alta priorità subiscono un tempo di coda inferiore a quelli a bassa priorità.

La memorizzazione in code su disco permette di gestire ordinatamente il traffico, dato che i messaggi a bassa priorità giunti durante un periodo di intenso traffico non vengono subito inviati, ma semplicemente memorizzati: solo quando il traffico lo consente, questi messaggi vengono estratti dalla coda e inviati.

Si tratta inoltre di sistemi di tipo **master/slave**: l'unità di commutazione periodicamente interroga i terminali (azione di *poll*), i quali, solo quando interrogati, possono inviare i propri messaggi; questi vengono accodati e, quando arriva il loro turno, vengono spediti (dopo aver fatto una operazione di

² Ricordiamo che una **code** è una struttura logica secondo la quale vengono memorizzati dati (di qualsiasi tipo) in una memoria fisica: l'accesso ai dati conservati in una code è di tipo FIFO (*First In First Out*), il che significa che i dati possono essere estratti solo nell'ordine con cui sono stati memorizzati, ossia cominciando dal primo memorizzato e passando via via agli altri fino all'ultimo memorizzato.

select sul terminale destinatario, cioè dopo che si è verificato se tale terminale può ricevere oppure no).

Oltre a commutare i messaggi, il programma di controllo permette anche l'accesso dei singoli terminali ad *applicazioni conversazionali in tempo reale*. E' ovvio, però, che, in questo caso, le transazioni interessanti evitano l'accodamento.

I limiti principali del message switching sono essenzialmente i seguenti:

- l'affidabilità dell'intero sistema dipende da quella del computer centrale (come del resto in tutti i sistemi costituiti da una stazione centrale cui fanno riferimento tutte le altre): per evitare che una caduta del computer centrale provochi l'interruzione del servizio, si provvedeva talvolta ad installare al centro un sistema duplicato (*mirror*), pronto a rilevare le funzioni del sistema principale qualora questo cadesse;
- in caso di carico trasmissivo alto, il sistema centrale può aumentare notevolmente il tempo impiegato dai messaggi per attraversare la rete;
- le linee di connessione ai terminali sono poco utilizzate.

In definitiva, sistemi di questo tipo presentano generalmente costi abbastanza alti rispetto al servizio fornito.

COMMUTAZIONE DI PACCHETTO (PACKET SWITCHING)

Negli anni '70 fu introdotta la tecnica della **commutazione di pacchetto (packet switching)**, che aveva i seguenti obiettivi fondamentali:

- moltiplicazione del traffico su più linee;
- bilanciamento del traffico;
- tempi ottimali di attraversamento della rete;
- alta affidabilità dei collegamenti (tramite l'uso di strade alternative);
- distribuzione del rischio mediante l'uso di svariati nodi intermedi;
- condivisione delle risorse.

Vediamo di spiegare bene questi aspetti.

In primo luogo, la tecnica a commutazione di pacchetto prevede l'uso di diversi **nodi intermedi**, il che diminuisce l'incidenza della eventuale caduta di un singolo componente sull'efficienza dell'intera rete.

Il termine **pacchetto** significa semplicemente che la lunghezza massima dei messaggi è prefissata. Il pacchetto contiene, in una propria testata (*header*) di livello 3 (*livello network* della **pila ISO-OSI**), l'indirizzo del destinatario. Il generico nodo che riceve il pacchetto dall'utente mittente decide verso quale altro nodo intermedio inoltrare il pacchetto, a seconda dell'indirizzo del destinatario, degli **algoritmi di instradamento** di cui è dotato e delle condizioni di traffico della rete.

In queste reti sono dunque possibili percorsi alternativi, il che aumenta l'affidabilità dell'intera struttura: infatti, nel caso un determinato pacchetto debba passare per un determinato nodo e quest'ultimo, però, vada fuori servizio, è sempre possibile trovare per il pacchetto una strada alternativa che lo farà comunque arrivare al destinatario.

Inoltre, una qualsiasi linea tra due nodi, dopo essere stata utilizzata per l'inoltro di un pacchetto di un utente, viene successivamente utilizzata per trasmettere pacchetti di altri utenti: il traffico sfrutta quindi in modo condiviso le risorse di rete, massimizzando il rendimento delle linee. Sulle linee che collegano vari nodi viene resa possibile la trasmissione di pacchetti di utenti diversi, secondo una tecnica che, in pratica, risulta molto simile al Time Division Multiplexing (TDM).

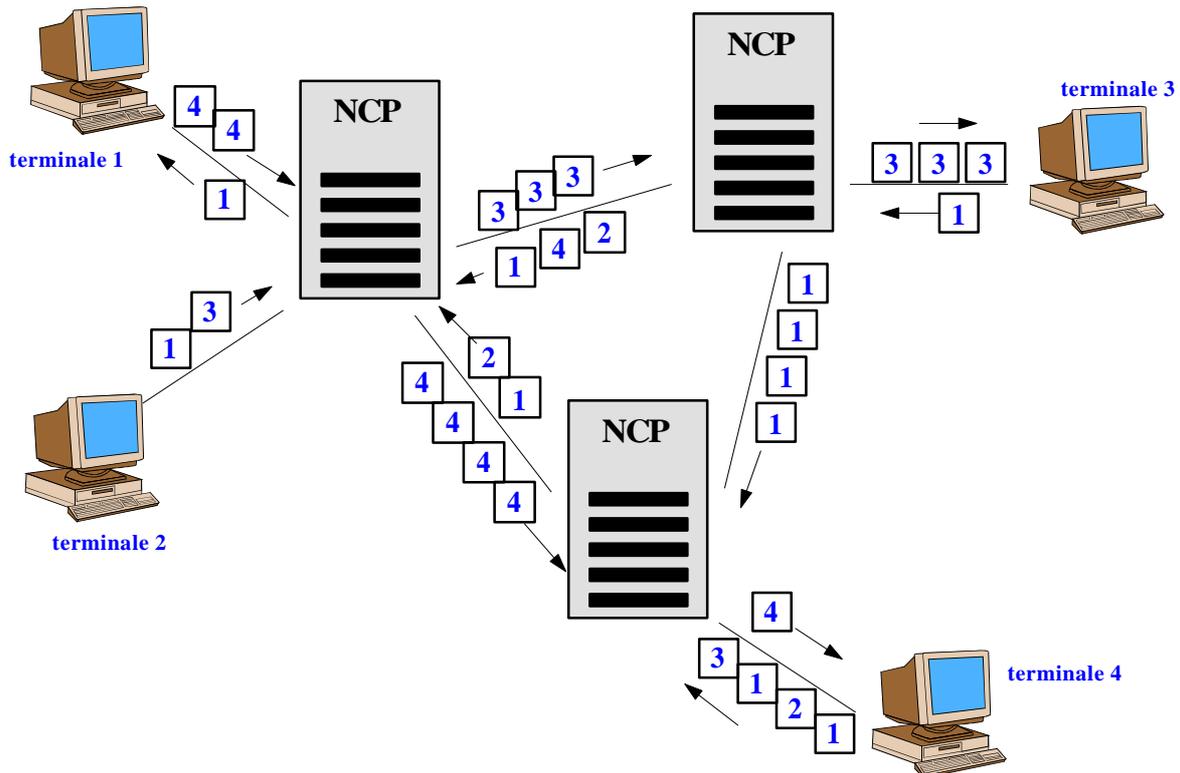
C'è anche un altro vantaggio, che è quello per cui i terminali possono essere collegati in modo permanente al nodo più vicino, eliminando, per quanto li riguarda, il tempo della *fase di chiamata*, che invece abbiamo visto essere una delle principali caratteristiche delle tecniche a commutazione di circuito.

Grazie all'**instradamento dinamico** (detto anche **adattativo**), il traffico complessivo risulta quindi bilanciato sulle varie risorse di rete, premessa anche per un buon tempo di attraversamento della rete.

L'utente ha la certezza che la spedizione dei suoi pacchetti verso un altro utente della rete venga effettuata, anche se non viene dedicato ai due alcun circuito reale: poiché la capacità di instradamento dei nodi assicura la consegna, nelle reti a commutazione di pacchetto si parla di connessioni tra due utenti tramite **circuito virtuale**.

Parecchie aziende ricorrono oggi alla **rete pubblica a commutazione di pacchetto** anziché a quella a commutazione di circuito, poiché tale rete è più affidabile della rete telefonica commutata e sono inoltre maggiori le velocità raggiungibili. Tra l'altro, nel caso di collegamenti su distanze elevate, si ha spesso un significativo vantaggio economico, in quanto sulle reti pubbliche a commutazione di pacchetto il costo del traffico è proporzionale al numero di pacchetti trasmessi e non alla durata della connessione né alla distanza tra i due utenti connessi dal circuito virtuale. Ci sono poi aziende che, sempre per motivi di convenienza, hanno ritenuto utile impiantare una rete propria a commutazione di pacchetto.

Per comprendere i vantaggi della commutazione di pacchetto, principalmente in termini di utilizzazione delle linee, possiamo considerare la figura seguente:



Abbiamo qui un certo numero di terminali, numerati semplicemente con 1,2,3 e così via, connessi ad un certo numero di nodi di rete, detti **NCP** (*Nodi a Commutazione di Pacchetto*). Ciascun terminale è connesso ad un NCP tramite una linea dedicata permanente³. Attraverso questa linea

³ Si noti che ogni NCP è connesso, in generale, a più stazioni d'utente, mentre non è vero il contrario: ogni terminale d'utente è connesso ad un solo NCP.

transitano i vari pacchetti (rappresentati da "mattoncini", ognuno contraddistinto dal numero corrispondente all'indirizzo del terminale destinatario), sia quelli spediti dal terminale sia quelli diretti al terminale.

I nodi di rete ricevono via via i pacchetti sia delle **linee d'utente** sia dalle **linee internodali**; memorizzano questi pacchetti e, dopo aver controllato la correttezza della trasmissione, li inoltrano sulle varie linee di uscita, a seconda dell'indirizzo del destinatario e degli algoritmi di instradamento. Un caso particolare si ha quando un pacchetto appena ricevuto da un nodo è destinato ad una stazione d'utente collegata allo stesso nodo: in questo caso, non c'è alcun instradamento, se non quello sulla linea dell'utente destinatario.

L'instradamento

INTRODUZIONE

L'instradamento in rete consiste essenzialmente nell'utilizzo delle risorse (software, hardware o microcodice), da parte dei nodi di rete, per trasmettere i pacchetti attraverso la rete, fino alla stazione destinataria.

Un concetto generale dal quale partire è il seguente: una rete di telecomunicazioni, che mantiene connesse un certo numero di stazioni e utilizza per questo un certo numero di risorse, possiede una propria **capacità di smaltimento del traffico**, dove per *traffico* intendiamo sostanzialmente la quantità di pacchetti che le stazioni immettono nella rete perché siano recapitati ai rispettivi destinatari, mentre per *smaltimento* intendiamo l'effettiva consegna dei pacchetti ai destinatari. E' allora ovvio che il traffico in ingresso alla rete dovrà essere sempre inferiore alla capacità di smaltimento della rete stessa. Viceversa, quanto più il traffico in ingresso si avvicina alla capacità di smaltimento della rete, tanto più la rete è "in difficoltà", in quanto si creano problemi di **congestione**; maggiore è la congestione e peggiori sono le prestazioni (in quanto aumentano notevolmente i tempi di ritardo con cui i pacchetti giungono a destinazione ed aumentano anche le probabilità di perdita dei pacchetti stessi). I problemi di congestione di una rete (o di un singolo nodo) possono essere affrontati in vari modi; per quanto riguarda la prevenzione dalla congestione, un ruolo sicuramente importante è svolto proprio dall'instradamento: quanto più efficienti sono gli algoritmi di instradamento, tanto meno probabile risulta il verificarsi di un congestionamento.

L'instradamento viene effettuato seguendo 3 criteri primari:

- assicurare sia il minor tempo di attraversamento sia il massimo rendimento della rete (**throughput**);
- ridurre al minimo il **costo** dell'attraversamento, cioè impegnare il minor numero di risorse;
- garantire un accettabile livello di sicurezza e di affidabilità.

Ci sono due possibili modi di effettuare l'instradamento: nell'**instradamento centralizzato**, esiste un unico centro (**NNC**, *Network Control Center*) che determina l'instradamento all'interno

dell'intera rete⁴; nell'**instradamento distribuito**, invece, ogni nodo di rete assume le decisioni sulle strade da seguire.

La soluzione centralizzata è quella più semplice da implementare, ma ha un punto debole ancora una volta nel fatto che la caduta della stazione principale (l'NNC appunto) comporta l'inutilizzabilità di tutta la rete. Una possibile soluzione a questo problema è quella di avere due NNC identici, di cui uno in funzione e l'altro di riserva. Se l'NNC attivo dovesse andare fuori-servizio, interviene l'altro a soppiantarlo. Lo svantaggio di questa soluzione è però che tutte le informazioni di controllo provenienti dai singoli nodi devono essere inviate ad entrambi gli NNC e devono essere memorizzate da entrambi gli NNC, il che costituisce un appesantimento della rete ed un raddoppio dei costi per gli NNC.

La soluzione distribuita è sicuramente più complessa di quella centralizzata (si pensi, ad esempio, ai problemi di coerenza tra le decisioni sull'instradamento prese dai diversi nodi), ma ha una maggiore affidabilità, proprio perché il tutto non dipende più da un'unica stazione centrale.

Dobbiamo ora capire con quali criteri, sia nel caso centralizzato sia nel caso distribuito, si possa determinare l'instradamento dei pacchetti. L'insieme dei parametri sui quali vengono decisi gli instradamenti è contenuto in apposite *tabelle*, che possono essere di due tipi:

- nelle **tabelle statiche**, le informazioni sui percorsi e sulle risorse vengono fissate al momento della generazione del software di sistema;
- nelle **tabelle dinamiche**, invece, ci sono aggiornamenti periodici delle informazioni.

E' ovvio che la soluzione preferibile è quella dinamica.

Nel caso del calcolo centralizzato dell'instradamento, tutti i nodi devono periodicamente mandare alla stazione centrale (NNC) le informazioni sul loro stato e sullo stato delle linee cui sono connessi. L'NNC, sulla base di queste informazioni, calcola il conseguente instradamento. Nel caso, invece, del calcolo distribuito dell'instradamento, la cosa è più complessa, perché non è proponibile che ciascun nodo debba conoscere lo stato dell'intera rete; infatti:

- in primo luogo, ogni nodo dovrebbe avere grosse risorse di memoria (le stesse che, nella soluzione centralizzata, ha l'NNC), il che può risultare poco conveniente da un punto di vista economico;
- in secondo luogo, il traffico delle informazioni sullo stato della rete diventerebbe un onere troppo pesante per la rete stessa.

Di conseguenza, si fa in modo che ciascun nodo riceva solo le informazioni relative ai nodi più vicini, in modo che l'instradamento venga calcolato solo con riferimento ai nodi più vicini.

A proposito delle informazioni sullo stato della rete, abbiamo poco fa osservato che esse costituiscono in ogni caso un traffico addizionale che la rete deve gestire. Si tratta anche di un traffico molto "importante", nel senso che, senza queste informazioni, non sarebbe possibile indirizzare i pacchetti all'interno della rete. Si può allora pensare ad almeno due soluzioni per gestire questo traffico di informazioni di controllo:

- la soluzione più banale, ma che può anche risultare meno efficiente, è quella di inviare le informazioni di controllo sugli stessi mezzi trasmissivi usati dai dati veri e propri; in questo caso, però, bisogna necessariamente dare una sorta di "precedenza" a queste informazioni rispetto ai dati: si tratta sostanzialmente di attribuire una **priorità di trasmissione**, in base

⁴ All'NNC, dotato di opportune risorse elaborative e collegato alla rete dei nodi, vengono in genere affidate anche altre funzioni, quali il monitoring globale della rete stessa, le decisioni in merito ad eventuali riconfigurazioni della rete e simili.

alla quale, se un nodo deve trasmettere sia dati sia informazioni di controllo, queste ultime saranno le prime ad essere inviate;

- una soluzione più efficiente, ma spesso meno economica, è invece quella di usare dei canali appositi (**canali di segnalazione**) per la trasmissione delle informazioni di controllo: ad esempio, se una data linea di comunicazione è gestita con la tecnica FDM (moltiplicazione a divisione di frequenza), uno o più canali saranno destinati alle informazioni di controllo e gli altri canali ai pacchetti veri e propri. In modo ancora più pratico, si può pensare ad una **rete dedicata**, che cioè trasmetta solo le informazioni di controllo: si tratterà, ovviamente, di una rete ad alta velocità (tipicamente in fibra ottica), ed è la soluzione adottata, per esempio, nella **rete GSM**, dove le informazioni di segnalazione viaggiano su una rete (cablata) completamente indipendente dalla rete su cui invece vengono trasmesse le conversazioni vere e proprie.

Un altro aspetto da considerare riguarda il criterio con cui viene scelto l'instradamento per un messaggio composto da più pacchetti:

- nella maggior parte dei casi, il percorso tra due stazioni utente viene scelto nel momento della definizione di una sessione tra di esse e mantenuto, salvo modifiche in seguito a problemi di rete, per tutta la durata della connessione; si parla in questo caso di **instradamento a circuito virtuale**;
- in altri casi (ad esempio in **Internet**, dove si usa lo schema **TCP/IP**), il percorso può essere determinato per ogni singolo pacchetto, per cui è possibile che i pacchetti di uno stesso messaggio seguano strade diverse per giungere alla stessa stazione destinataria (la quale, ovviamente, dovrà essere in grado di ricostruire la sequenza esatta, in quanto è possibile che un pacchetto trasmesso dopo arrivi invece prima). Si parla in questo caso di **instradamento a datagramma**: il concetto è quello di scegliere per ogni pacchetto l'instradamento ottimale, dato che lo stato della rete cambia in continuazione e quindi la scelta fatta per un pacchetto potrebbe non essere più quella ottimale per il pacchetto successivo.

Non è detto che l'instradamento, anche per il singolo pacchetto, sia sempre ottimale. Infatti, bisogna tener conto che le informazioni sulla base delle quali l'instradamento viene calcolato non sempre sono aggiornate. Per rendercene conto, possiamo considerare un esempio molto semplice: mettiamoci nel contesto di un calcolo distribuito dell'instradamento, nel quale cioè sono i nodi a eseguire gli algoritmi di instradamento; supponiamo allora che il nodo A, dovendo inviare un certo pacchetto, vada a leggere le informazioni contenute nelle proprie tabelle e calcoli di conseguenza l'instradamento; anche se le tabelle sono state aggiornate un attimo prima, non è detto che lo stato dei nodi vicini sia rimasto nel frattempo invariato; per esempio, supponiamo che un nodo B vicino sia andato nel frattempo fuori servizio e abbia trasmesso la segnalazione di fuori-uso; questa segnalazione impiega un certo tempo per raggiungere A, per cui potrebbe arrivare dopo che A stesso ha letto le proprie tabelle; succede, perciò, che A calcoli l'instradamento "credendo" che B funzioni perfettamente, quando invece non è così. Lo stesso esempio si potrebbe anche fare nel contesto di un calcolo centralizzato dell'instradamento, dove anzi il problema è ancora maggiore, in quanto le informazioni di un nodo molto lontano dall'NNC possono impiegare parecchio tempo per arrivare fino all'NNC.

DEFINIZIONI VARIE

Prima di esaminare le diverse tecniche di instradamento, è necessario definire alcuni termini che, a seconda della tecnica di instradamento adottata, ne designano alcune conseguenze oppure specificano certi provvedimenti che si rendono necessarie:

- **traffic multiplication effect (TME)**: è l'effetto che si verifica quando un singolo pacchetto, nell'attraversare la rete, genera altri pacchetti identici a se stesso;
- **packet kill**: è l'eliminazione dei pacchetti duplicati, per evitare, in tutto o in parte, le conseguenze del TME. Infatti, il TME solo talvolta è utile, mentre nella maggior parte dei casi incide negativamente sulle prestazioni complessive della rete, perché l'inutile trasporto di pacchetti duplicati appesantisce il carico trasmissivo, aumentando il traffico complessivo ed i tempi di attraversamento della rete;
- **node route around**: si tratta dei provvedimenti da prendere quando bisogna evitare che un pacchetto passi per un determinato nodo, sovraccarico o malfunzionante;
- **loop (o ping-pong)**: è il fenomeno per cui un pacchetto, una volta entrato in rete, continua ad essere scambiato tra i nodi senza mai arrivare a destinazione. Quasi sempre questo fenomeno nasce da incoerenti informazioni per l'instradamento. Un modo di eliminare l'inconveniente è quello di scartare un pacchetto quando effettua un numero prestabilito di passaggi sullo stesso nodo.

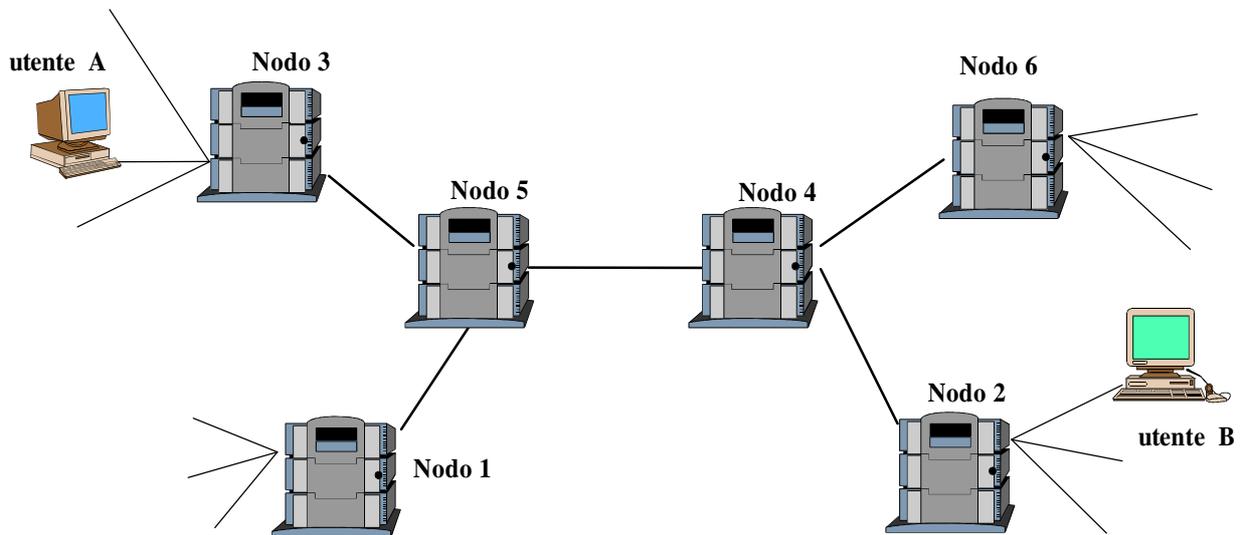
Osserviamo che, a volte, per motivi di sicurezza e protezione delle informazioni, si prendono provvedimenti esattamente contrari al packet-kill: per rendere più difficoltosa l'eventuale interpretazione abusiva di quello che transita sulla rete, vengono generati, insieme ai pacchetti reali, un certo numero di *pacchetti fasulli*; questi pacchetti viaggiano sulla rete senza essere mai eliminati (o comunque non prima di un certo numero di passaggi per lo stesso nodo).

INSTRADAMENTO A CIRCUITO VIRTUALE

Abbiamo osservato che, con l'instradamento a circuito virtuale (tipico delle **reti X25** o della recente **rete ATM**), il percorso tra due stazioni utente viene scelto nel momento in cui deve essere attivata una sessione tra di esse e tale percorso viene mantenuto, salvo problemi di rete, per tutta la durata della connessione. In pratica, quindi, si ha qualcosa di molto simile alla commutazione di circuito: in quel caso, infatti, il circuito reale che collega i due utenti viene fissato prima della trasmissione, viene mantenuto e gestito durante la trasmissione e viene infine rimosso dopo la fine della trasmissione. La differenza fondamentale con l'instradamento a circuito virtuale è che, in quest'ultimo, non vengono riservate risorse: viene solo stabilito attraverso quali nodi passeranno i pacchetti, senza però che venga vincolato l'uso di determinate risorse; in corrispondenza di ogni nodo si provvede a destinare le opportune risorse (unità di memoria, linee di comunicazioni e così via) per il passaggio al nodo successivo.

E' ovvio che il criterio con cui viene scelto il percorso dei dati si basa sulle stesse tecniche valide per il *datagramma*, dove però la scelta viene fatta pacchetto per pacchetto.

Ogni nodo della rete conosce tutti i circuiti virtuali che fanno capo al nodo stesso, più altre informazioni. Per fare un esempio, consideriamo una rete composta da soli 6 nodi ed un certo numero di utenti, disposti come nella figura seguente:



Supponiamo che l'utente A voglia inviare un pacchetto all'utente B. L'utente A invia tale pacchetto al nodo cui è connesso, ossia il nodo 3. Questo legge l'indirizzo del destinatario e va successivamente a leggere il contenuto della propria **tabella per l'instradamento**; in questa tabella è contenuta l'informazione (calcolata sulla base delle informazioni di controllo e degli algoritmi di instradamento) circa il prossimo nodo cui il pacchetto deve essere inviato.

Tipicamente, ogni riga della tabella corrisponde ad un circuito virtuale e può avere la seguente struttura:

3,5	13	5,4	7
nodi collegati	nome circuito virtuale	coppia di nodi successivi	nome successivo circuito virtuale

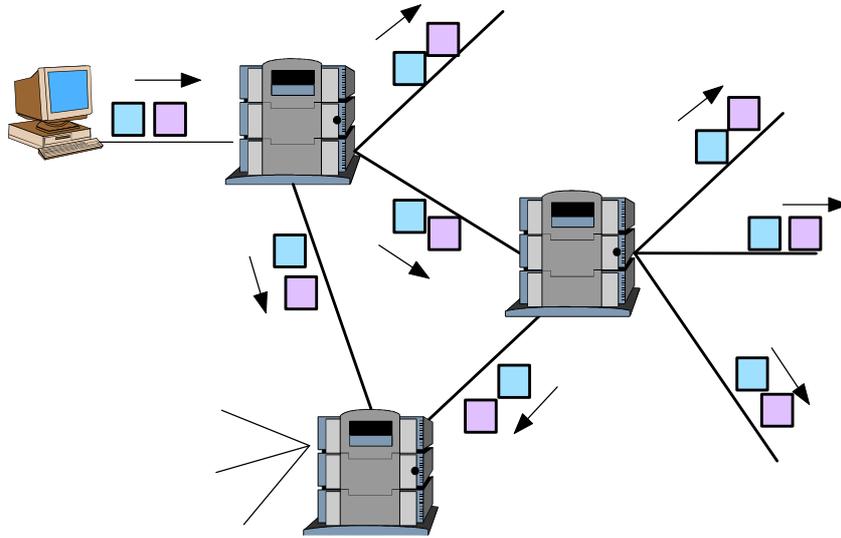
Nel primo campo sono contenuti il nodo in questione (3) ed il nodo successivo (5) ed il campo seguente fornisce il nome del corrispondente circuito virtuale. In un campo ulteriore potrebbe già essere indicato anche il successivo percorso (dal nodo 5 al nodo 4 secondo il circuito virtuale numerato con 7).

Con questo meccanismo, ogni nodo sa perfettamente a quale nodo successivo inviare il pacchetto e, come nell'esempio appena considerato, potrebbe anche sapere quale sarà il percorso ancora successivo.

Queste tabelle di instradamento, come detto, vengono periodicamente aggiornate in base alle informazioni che ogni nodo percepisce circa lo stato della rete ed in base agli algoritmi di decisione predisposti all'atto dell'installazione della rete.

TECNICA DEL “PACKET FLOODING”

La tecnica del **Packet Flooding** consiste nel ritrasmettere su tutti i canali in uscita dal nodo ogni pacchetto ricevuto:



E' ovvio che il grande inconveniente è l'altissimo effetto TME ed il conseguente alto carico della rete. Ci sono però anche diversi vantaggi:

- la rete ha una altissima affidabilità: la caduta di una qualsiasi risorsa non impedisce la comunicazione tra i due utenti, almeno finché rimane utilizzabile anche una sola strada. Ad esempio, si nota, nella figura precedente, che il nodo più in basso riceve gli stessi pacchetti da due distinte linee di ingresso, per cui, in caso di malfunzionamento di una delle due (o dei corrispondenti nodi), c'è probabilmente l'altra funzionante. Questa affidabilità è il motivo per cui alcune reti utilizzate in ambito militare (dove l'aspetto economico è sempre secondario) adottano questa tecnica. Viceversa, essa non viene per nulla utilizzata nelle reti a carattere commerciale, a causa del notevole sovradimensionamento delle risorse di sistema e delle linee trasmissive che richiede.
- il tempo di attraversamento della rete è comunque buono: tra i vari pacchetti inviati, ce ne sarà almeno uno che, trovata la strada ottimale, arriva al destinatario in un tempo minimo di attraversamento della rete; è ovvio che il destinatario, una volta ricevuto un dato pacchetto, scarterà tutti gli eventuali duplicati arrivati successivamente. C'è però da osservare che il tempo di attraversamento della rete, per quanto buono, non è comunque quello ottimale: quest'ultimo, infatti, si avrebbe solo se i nodi e le linee non fossero appesantiti, nei loro compiti, dal dover servire anche pacchetti duplicati.

Un modo per ridurre al minimo l'effetto della moltiplicazione dei pacchetti (TME) è quello di attribuire ad ogni nodo la facoltà di eliminare pacchetti in arrivo che siano identici ad altri già spediti. In questo modo si è certi che l'ultimo nodo, quello direttamente collegato al destinatario, manderà al destinatario stesso un'unica copia del pacchetto.

TECNICA DEL "RANDOM ROUTING"

La tecnica del **Random Routing** (*instradamento casuale*) consiste banalmente nel fatto che ogni nodo sceglie a caso il canale di uscita su cui trasmettere il pacchetto appena ricevuto. In questo modo, il traffico risulta bilanciato sulle varie risorse, il che rappresenta un grosso vantaggio (oltre alla semplicità) per le prestazioni complessive della rete.

Il problema è che la strada percorsa da un pacchetto risulta mediamente più lunga, talvolta anche di molto, di quella ottimale, il che si traduce in tempi di attraversamento della rete tutt'altro che ottimizzati. A questo si aggiunge anche un maggiore carico sulla rete, in quanto il pacchetto usa un numero di risorse mediamente superiore a quello minimo necessario.

Il maggiore inconveniente, però, è nel rischio che un pacchetto immesso in rete entri in loop e non pervenga più al destinatario. Questo comporta non solo che il destinatario non riceva il pacchetto, ma anche un inutile spreco di risorse della rete. Quest'ultimo problema può essere parzialmente risolto, come detto in precedenza, scartando il pacchetto al superamento di un certo numero di nodi attraversati. Molto più serio è invece il problema della mancata ricezione: infatti, anche se si usa un protocollo (di livello 3) in base al quale la stazione mittente aspetta di ricevere una conferma di avvenuta ricezione da parte del destinatario, è possibile che anche tale conferma si "perda" nella rete, con conseguenti ritrasmissioni e quindi ulteriore appesantimento della rete.

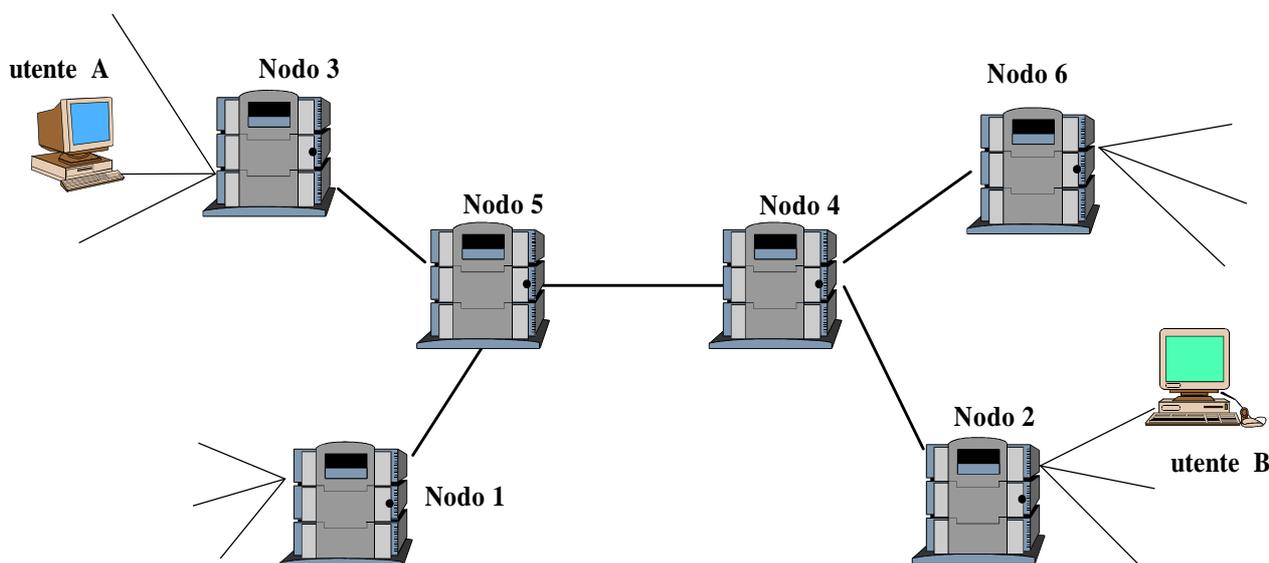
In generale, possiamo dire che i vantaggi di questa tecnica non sono tali da bilanciare gli svantaggi, per cui l'instradamento casuale non viene di fatto più usato.

INSTRADAMENTO ADATTATIVO ARPANET

Arpanet è la rete sviluppata agli inizi degli anni '70 dal ministero della difesa degli Stati Uniti in collaborazione con enti di ricerca e università. Si tratta di una pietra miliare nella storia delle reti di computer: da essa è stato sviluppato il concetto di **commutazione di pacchetto** ed è ora incorporata nella più nota **rete Internet**. Arpanet ha visto la nascita della struttura TCP/IP ed è anche un interessante esempio di **instradamento distribuito dinamico** (o *adattativo*), dove le responsabilità sono cioè distribuite tra i nodi ed ogni nodo conosce, con un certo grado di aggiornamento, la situazione almeno dei nodi più vicini.

L'elemento base per le scelte è ancora una volta una **tabella**, che ogni nodo possiede e mantiene periodicamente aggiornata sulla base delle informazioni scambiate con gli altri nodi e di opportuni calcoli.

Consideriamo la stessa rete vista in uno dei precedenti paragrafi:



Preso un qualsiasi nodo, ad esempio il 5, la sua tabella sarà del tipo seguente:

Destinatario	Prossimo nodo	Ritardo
1	1	2
2	4	8
3	3	1
4	4	1
6	4	11
...

Per ogni destinatario, la tabella contiene, tra le altre informazioni, l'indicazione del nodo successivo ed il costo globale (inteso come tempo o come qualche altro parametro) per l'attraversamento. Per esempio, si nota che i pacchetti destinati ai nodi 3 e 4 vanno inoltrati agli stessi nodi 3 e 4, visto che essi sono adiacenti al nodo 5. Viceversa, un pacchetto destinato al nodo 6 andrà inoltrato al nodo 4, il quale disporrà di una ulteriore tabella e quindi di ulteriori indicazioni per l'inoltro verso il nodo 6 (che in questo caso è adiacente al 4).

Il generico nodo, tutte le volte che invia un pacchetto verso i suoi nodi adiacenti, calcola il **tempo di invio**, che può essere valutato come l'intervallo tra tempo di spedizione del pacchetto e tempo di ricezione di ACK. Questi dati vengono memorizzati e periodicamente (ad esempio ogni 10 secondi), ne viene fatta una media, per ciascuna linea. Tale media è quella che viene inserita nella tabella.

Se, in un certo momento, viene rilevato un tempo di invio molto maggiore del valore medio, l'informazione viene comunicata (ad esempio tramite il packet-flooding, ossia con invio su tutte le linee di uscite) a tutti i nodi vicini. In questo modo, anche gli altri nodi vengono a conoscere i dati sperimentali delle varie tratte e sono quindi in grado di modificare le proprie tabelle. Questo è il motivo per cui si parla di **instradamento dinamico** o adattativo: le scelte possono cambiare ad ogni ricalcolo, secondo le attuali condizioni della rete.

Gli obiettivi principali dell'instradamento dinamico sono sostanzialmente due:

- tenere conto delle mutevoli condizioni della rete;
- predisporre una tecnica (quanto più veloce possibile) che eviti di utilizzare un nodo che manifesti dei problemi: se un nodo dovesse risultare malfunzionante o congestionato, i tempi di attraversamento delle linee ad esso collegate vengono automaticamente impostati al valore massimo possibile, in modo che il nodo non venga più usato fin quando non torna in condizioni normali.

Questi discorsi mostrano una volta di più la necessità di avere dei fitti scambi di informazioni di controllo tra i nodi, con tutti i problemi che ne derivano e che sono stati già accennati in precedenza.

Arpanet ha però anche dei problemi:

- la tecnica adattativa è sicuramente quella più complessa;
- il carico dovuto al packet-flooding appesantisce i nodi;
- ogni tanto qualche pacchetto entra in loop o viene addirittura perso, perché transitato per nodi con tabella non aggiornate;

- pacchetti destinati allo stesso utente, ma viaggianti su percorsi diversi, possono arrivare in sequenza alterata rispetto a quella dell'immissione.

Quest'ultimo problema può essere affrontato sia a livello direttamente dell'utente, al quale viene demandato l'onere di ricostruire la sequenza, sia a livello dell'ultimo nodo: in questo caso, il nodo deve poter memorizzare tutti i pacchetti, al fine di ristabilire la sequenza corretta, dopo di che potrà inviare il messaggio all'utente. E' ovvio che la soluzione di lasciare il riordinamento all'utente consente di alleggerire i nodi.

Per quanto riguarda, invece, l'appesantimento dovuto al packet-flooding, si può pensare di rallentare i ritmi di calcolo delle strade: questo, però, se da un lato riduce il sovraccarico della rete e dei nodi per l'aggiornamento reciproco, ha l'ovvio inconveniente che i nodi risultano, in ogni istante, non perfettamente aggiornati sullo stato attuale della rete.

PROBLEMI TIPICI DELLE RETI A COMMUTAZIONE DI PACCHETTO

Esaminiamo rapidamente i problemi principali di una rete a commutazione di pacchetto, alcuni dei quali già esaminati in precedenza:

- probabilità di perdita di pacchetti: la perdita di un pacchetto può derivare sia dal fenomeno del **loop** sia anche da altre situazioni. Una di queste è la seguente: immaginiamo un pacchetto spedito dal nodo A verso il nodo B; il nodo B lo riceve ed invia il corrispondente ACK, ma, allo stesso tempo, non riesce ad inviare il pacchetto verso il nodo C, a causa di una interruzione della linea; in questa situazione, il nodo A non viene avvisato dei problemi riscontrati tra B e C, ma il pacchetto si ferma al nodo B. La soluzione più semplice a questo problema è che B invii l'ACK ad A solo dopo aver ricevuto l'ACK dal nodo destinatario. Un'altra possibilità è invece quella di prevedere un pacchetto specifico tra i due nodi finali, che indichi al mittente quali eventuali pacchetti di una sequenza non sono stati ricevuti;
- probabilità di duplicazione dei pacchetti: supponiamo che il nodo A abbia inviato a B un pacchetto e che B abbia risposto con il corrispondente ACK; supponiamo inoltre che, mentre B provvede all'inoltro del pacchetto verso C, l'ACK inviato ad A venga perso; di conseguenza, il nodo A, non ricevendo alcuna conferma, deduce che il nodo B ha dei problemi, per cui ritrasmette lo stesso pacchetto su altre strade. Il risultato è che nella rete ci sono almeno 2 copie dello stesso pacchetto. Ancora una volta, si può aggirare l'inconveniente tramite una numerazione dei pacchetti: quando la stazione destinataria riceve due o più pacchetti uguali, ne considera uno solo e scarta gli altri;
- pericoli di sovraccarico dei nodi: dei problemi legati al controllo della congestione parleremo diffusamente più avanti; per il momento, limitiamoci a dire che, per motivi vari, un dato nodo può avere un carico di lavoro eccessivo o troppe code sulle linee di uscita: in altre parole, esso sta ricevendo pacchetti in ingresso ad un ritmo superiore rispetto alle proprie possibilità di output. Questo comporta che la memoria del nodo arrivi a saturarsi: quando risulta piena, ogni ulteriore pacchetto in arrivo risulta perso. Per evitare la congestione (cioè appunto la saturazione di tutti i buffer di Input/Output), il nodo invia uno speciale pacchetto (detto **choke**) col quale invita i nodi vicini a rallentare l'emissione dei pacchetti verso di lui. Il problema è che questo rallentamento può provocare, per identici motivi, la congestione dei nodi vicini. Essi cercano quindi di rimediare allo stesso modo: la congestione si allarga a macchia d'olio e la rete rischia di non operare più. Si intuisce che l'unico rimedio realmente valido contro la congestione sia la prevenzione: bisogna tener continuamente sotto controllo la situazione e avvertire i nodi adiacenti in anticipo, pur continuando ad operare.

- *problemi legati ad operazioni di reset*: quando un nodo sperimenta una situazione di errore non rimediabile, esso emette un comando di **reset** verso gli altri nodi, ossia li invita ad interrompere e reinizializzare tutte le sessioni d'utente. In casi come questi, i pacchetti che sono ancora in rete e che appartengono a sessioni resettate vengono inevitabilmente perduti.

Autore: **SANDRO PETRIZZELLI**
e-mail: sandry@iol.it
sito personale: <http://users.iol.it/sandry>
succursale: <http://digilander.iol.it/sandry1>