

1. Commutazione (Switching)

La commutazione è il processo mediante il quale i dati vengono trasferiti da un nodo all'altro in una rete. Esistono diverse tecniche di commutazione, ognuna con specifiche applicazioni e vantaggi.

1.1 Commutazione di pacchetto

- **Definizione:** I dati vengono suddivisi in pacchetti più piccoli per la trasmissione
- **Caratteristiche:**
 - Ogni pacchetto contiene intestazione (header) con informazioni di routing
 - I pacchetti possono seguire percorsi diversi nella rete
 - Riassemblaggio a destinazione
- **Vantaggi:**
 - Utilizzo efficiente della larghezza di banda
 - Resilienza ai guasti (se un percorso fallisce, i pacchetti possono seguirne un altro)
 - Condivisione del canale tra più comunicazioni
- **Svantaggi:**
 - Possibile jitter (variazione del ritardo)
 - Ritardi variabili
 - Pacchetti possono arrivare fuori ordine o essere persi
- **Applicazioni:** Internet e moderne reti di dati

1.2 Commutazione di circuito

- **Definizione:** Viene stabilito un canale dedicato per l'intera durata della comunicazione
- **Caratteristiche:**
 - La larghezza di banda è riservata anche quando non utilizzata
 - Connessione permanente tra mittente e destinatario
- **Vantaggi:**
 - Prestazioni costanti e garantite
 - Nessun jitter
 - Latenza fissa
- **Svantaggi:**
 - Utilizzo inefficiente delle risorse
 - Tempo di configurazione iniziale
 - Vulnerabilità ai guasti
- **Esempio classico:** Rete telefonica tradizionale (PSTN)

1.3 Commutazione di messaggio

- **Definizione:** L'intero messaggio viene trasmesso da un nodo all'altro
- **Caratteristiche:**
 - Ogni nodo intermedio riceve, memorizza e inoltra l'intero messaggio
 - Store-and-forward
- **Vantaggi:**
 - Adatto per messaggi brevi
 - Tolleranza agli errori (controllo a ogni hop)
 - Non richiede percorsi predefiniti
- **Svantaggi:**
 - Elevata latenza
 - Richiede molta memoria nei nodi intermedi
 - Non efficiente per comunicazioni continue
- **Esempi:** Primi sistemi di posta elettronica, reti telegrafiche

2. Protocolli per LAN Wireless

Le reti wireless presentano problematiche specifiche che richiedono protocolli dedicati.

2.1 Problematiche delle reti wireless

2.1.1 Stazione esposta

- Un nodo si astiene dal trasmettere perché rileva una trasmissione in corso, ma non interferirebbe con essa
- Riduce l'efficienza della rete

2.1.2 Stazione nascosta

- Un nodo non rileva una trasmissione in corso e quindi trasmette, causando interferenze
- Problema comune nelle reti wireless con ostacoli fisici

2.1.3 Altre problematiche

- **Attenuazione del segnale:** diminuzione della potenza del segnale con la distanza
- **Interferenze:** disturbi causati da altre sorgenti di segnale
- **Sicurezza:** vulnerabilità legate alla propagazione del segnale in aria

2.2 MACA/MACAW (Multiple Access with Collision Avoidance)

- Protocollo che risolve i problemi di stazione nascosta/esposta
- Utilizza pacchetti di controllo RTS (Request To Send) e CTS (Clear To Send)

2.2.1 Funzionamento RTS/CTS

1. Il mittente invia un pacchetto RTS (inclusa durata prevista della trasmissione)
2. Il destinatario risponde con un pacchetto CTS (inclusa durata)
3. Le altre stazioni che ricevono CTS attendono per il tempo indicato
4. Il mittente invia i dati
5. Il destinatario conferma la ricezione

2.2.2 Vantaggi

- Risolve il problema della stazione nascosta
- Riduce le collisioni
- Migliora l'efficienza della rete

3. Ethernet e codifica Manchester

Ethernet è uno standard per reti locali cablate, inizialmente sviluppato da Xerox e poi standardizzato come IEEE 802.3.

3.1 Codifica Manchester

- **Definizione:** Tecnica di codifica usata in Ethernet dove ogni bit è rappresentato da una transizione
- **Funzionamento:**
 - Da basso ad alto per bit 1
 - Da alto a basso per bit 0
- **Vantaggi:**
 - Sincronizzazione di clock incorporata
 - Rilevamento di errori
 - Auto-sincronizzazione
- **Svantaggi:**
 - Richiede il doppio della larghezza di banda rispetto alla codifica binaria semplice

3.2 Algoritmo di Backoff in Ethernet

- Utilizzato quando viene rilevata una collisione
- Tempi di attesa casuali per evitare collisioni ripetute
- L'algoritmo di backoff esponenziale binario:
 1. Attendi K slot di tempo, dove K è un numero casuale tra 0 e $(2^n - 1)$
 2. n è il numero di collisioni consecutive, fino a un massimo (10)
 3. Il tempo di attesa aumenta esponenzialmente con ogni collisione

3.3 Evoluzione di Ethernet

- **10Base5:** cavo coassiale spesso, 10 Mbps, "Thicknet"

- **10Base2**: cavo coassiale sottile, 10 Mbps, "Thinnet" o "Cheapernet"
- **10Base-T**: doppino intrecciato, 10 Mbps
- **100Base-TX**: Fast Ethernet, 100 Mbps
- **1000Base-T**: Gigabit Ethernet, 1 Gbps su doppino
- **10GBase-T**: 10 Gigabit Ethernet, 10 Gbps su doppino

4. Controllo degli errori e di flusso a livello Data Link

4.1 Rilevamento degli errori

4.1.1 CRC (Cyclic Redundancy Check)

- **Definizione**: Tecnica di rilevamento errori basata su divisione polinomiale in campo binario
- **Funzionamento**:
 1. Aggiungere n bit zero ai dati (n = grado del polinomio generatore)
 2. Dividere la stringa risultante per il polinomio generatore
 3. Il resto della divisione è il CRC
 4. Trasmettere i dati originali seguiti dal CRC
- **Efficacia**: Alta capacità di rilevazione errori, usato in Ethernet
- **Standard comuni**: CRC-16, CRC-32

4.1.2 Bit di parità

- **Parità semplice**: aggiunge un bit per rendere il numero totale di bit 1 pari (parità pari) o dispari (parità dispari)
- **Parità bidimensionale**: organizza i dati in una matrice e calcola la parità per ogni riga e colonna
- **Vantaggi**: semplicità di implementazione
- **Svantaggi**: limitata capacità di rilevazione errori

4.1.3 Checksum

- Somma dei valori dei dati, eventualmente con complemento a uno
- Meno efficace del CRC, ma più semplice
- Utilizzato spesso a livello trasporto (TCP/UDP)

4.2 Correzione degli errori

4.2.1 FEC (Forward Error Correction)

- I bit di ridondanza permettono di correggere un numero limitato di errori senza ritrasmettere

- **Esempi:**
 - **Codici di Hamming:** correggono errori singoli
 - **Codici Reed-Solomon:** correggono burst di errori, usati in CD/DVD
 - **Codici convoluzionali:** usati in comunicazioni wireless
- **Applicazioni:** Collegamenti satellitari, trasmissioni broadcast

4.2.2 Hamming Distance

- Numero di bit che differiscono tra due sequenze
- Per rilevare d errori: serve una distanza minima $d+1$
- Per correggere d errori: serve una distanza minima $2d+1$

4.3 Algoritmi di controllo di flusso

Il controllo di flusso è necessario per impedire che il mittente saturi i buffer del ricevitore.

4.3.1 Stop-and-Wait

- **Funzionamento:**
 - Invia un frame
 - Attende conferma (ACK) prima di inviare il successivo
 - In caso di errore/timeout, ritrasmette
- **Efficienza:**
 - $E = T_{\text{trasmissione}} / (T_{\text{trasmissione}} + \text{RTT})$
 - Molto bassa con RTT elevati
- **Vantaggi:** semplicità di implementazione
- **Svantaggi:** bassa efficienza, specialmente su collegamenti a lunga distanza

4.3.2 Go-back-N

- **Funzionamento:**
 - Permette di inviare più frame prima di ricevere ACK (finestra di invio)
 - In caso di errore, ritrasmette tutti i frame dall'ultimo confermato
 - L'ACK k conferma tutti i frame fino a $k-1$
- **Dimensione finestra:** in genere $2^m - 1$ dove m è il numero di bit per la numerazione
- **Vantaggi:** efficienza migliore con RTT elevati
- **Svantaggi:** inefficiente in caso di errori frequenti (ritrasmissione di frame già ricevuti correttamente)

4.3.3 Selective Repeat

- **Funzionamento:**
 - Permette di inviare più frame prima di ricevere ACK

- In caso di errore, ritrasmette solo i frame persi
- Richiede buffering al ricevitore
- **Dimensione finestra massima:** $2^{(m-1)}$ per evitare ambiguità nei numeri di sequenza
- **Vantaggi:** massima efficienza, ritrasmissione selettiva
- **Svantaggi:** più complesso da implementare, buffer di ricezione più grande

4.3.4 Sliding Window

- Generalizzazione dei protocolli precedenti
- Gestione dinamica di quanti frame possono essere inviati senza ACK
- Adattabilità alle condizioni della rete

4.4 Ritrasmissione e ACK/NACK

4.4.1 Timeout

- Se non arriva ACK entro un tempo limite, il frame è ritrasmesso
- Valore di timeout critico: troppo breve → ritrasmissioni inutili, troppo lungo → ritardi
- In genere: Timeout = RTT stimato + margine

4.4.2 ACK (Acknowledgment)

- Conferma ricezione corretta
- Può essere cumulativo (conferma tutti i frame fino a un certo numero)
- Riduce l'overhead di rete

4.4.3 NACK (Negative Acknowledgment)

- Notifica un errore, sollecitando la ritrasmissione immediata
- Permette di ridurre i tempi di attesa per il timeout
- Non tutti i protocolli lo utilizzano