

Lezione 21 - Livello di Collegamento

Checksum Internet

Obiettivo: rilevare gli "errori" (bit alternati) nel segmento trasmesso

mittente:

- tratta il contenuto del segmento come una sequenza di interi a 16 bit (inclusi i campi dell'intestazione UDP e gli indirizzi IP)
- checksum: complemento a 1 della somma (in complemento a 1) della sequenza di interi a 16 bit
- pone il valore del checksum nel campo checksum del segmento UDP

ricevente:

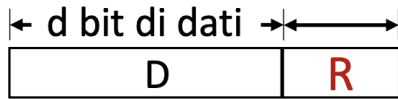
- calcola la somma in complemento a 1 allo stesso modo del mittente, includendo però il checksum ricevuto
- il risultato è costituito da tutti bit 1 (-0 nell'aritmetica del complemento a 1)?
 - Sì - nessun errore rilevato
 - No - errore rilevato
- oppure, si esegue il complemento a 1 finale: si è calcolato il checksum di tutti i dati ricevuti (incluso il checksum stesso) e si verifica che sia formata da soli 0

Codice di controllo a ridondanza ciclica

- codifica di rilevamento degli errori più potente
- D: dati da trasmettere (d bit)
- G: sequenza di (r + 1) bit concordata, detta generatore (definito nello standard CRC)

essere 1

r bit CRC



bit da inviare

$\langle D, R \rangle = D * 2^r \text{ XOR } R$ formula per questi bit

mittente: calcola r bit CRC, R , tali che $\langle D, R \rangle$ si divide esattamente da $G \pmod{2}$

- il ricevente conosce G , divide $\langle D, R \rangle$ per G . Se il resto è diverso da zero: errore rilevato!
- può rilevare tutti gli errori a burst di lunghezza inferiore a $r+1$ bit (ovvero, errori di non più di r bit consecutivi)
- la frazione di burst più lunghi che può rilevare è approssimativamente $1 - 2^{-r}$
- largamente usato in pratica (Ethernet, 802.11 WiFi)

tutti i calcoli di CRC sono eseguiti in aritmetica modulo 2 senza riporti nelle addizioni e prestiti nelle sottrazioni

- addizione e sottrazione sono la stessa operazione, corrispondente all'or esclusivo (exclusive or, XOR) bit a bit

$$\begin{array}{r} 1011 + \\ 1101 = \\ \hline 0110 \end{array}$$

$$\begin{array}{r} 1011 - \\ 1101 = \\ \hline 0110 \end{array}$$

$$1011 \text{ XOR } 1101 = 1011 \oplus 1101 = 0110$$

- la moltiplicazione e la divisione sono calcolate come al solito, usando queste definizioni di addizione e sottrazione

$$\begin{array}{r}
 1011 \times \\
 101 = \\
 \hline
 1011 + \\
 0 + \\
 \hline
 1011 = \\
 \hline
 100111
 \end{array}$$

$$\begin{array}{r}
 1011 \times \longrightarrow x^3 + x + 1 \\
 101 = \longrightarrow x^2 + 1 \\
 \hline
 1011 + \\
 0 + \\
 \hline
 1011 = \\
 100111 \longrightarrow x^5 + x^2 + x + 1
 \end{array}$$

assumendo che il primo bit sia 1, allora il grado del polinomio è uguale al numero di bit - 1

$$(x^3 + x + 1) \cdot (x^2 + 1) = (x^5 + x^3 + x^2) + (x^3 + x + 1) = x^5 + \underbrace{(1 + 1)}_{\text{somma modulo 2}} x^3 + x^2 + x + 1 = x^5 + x^2 + x + 1$$

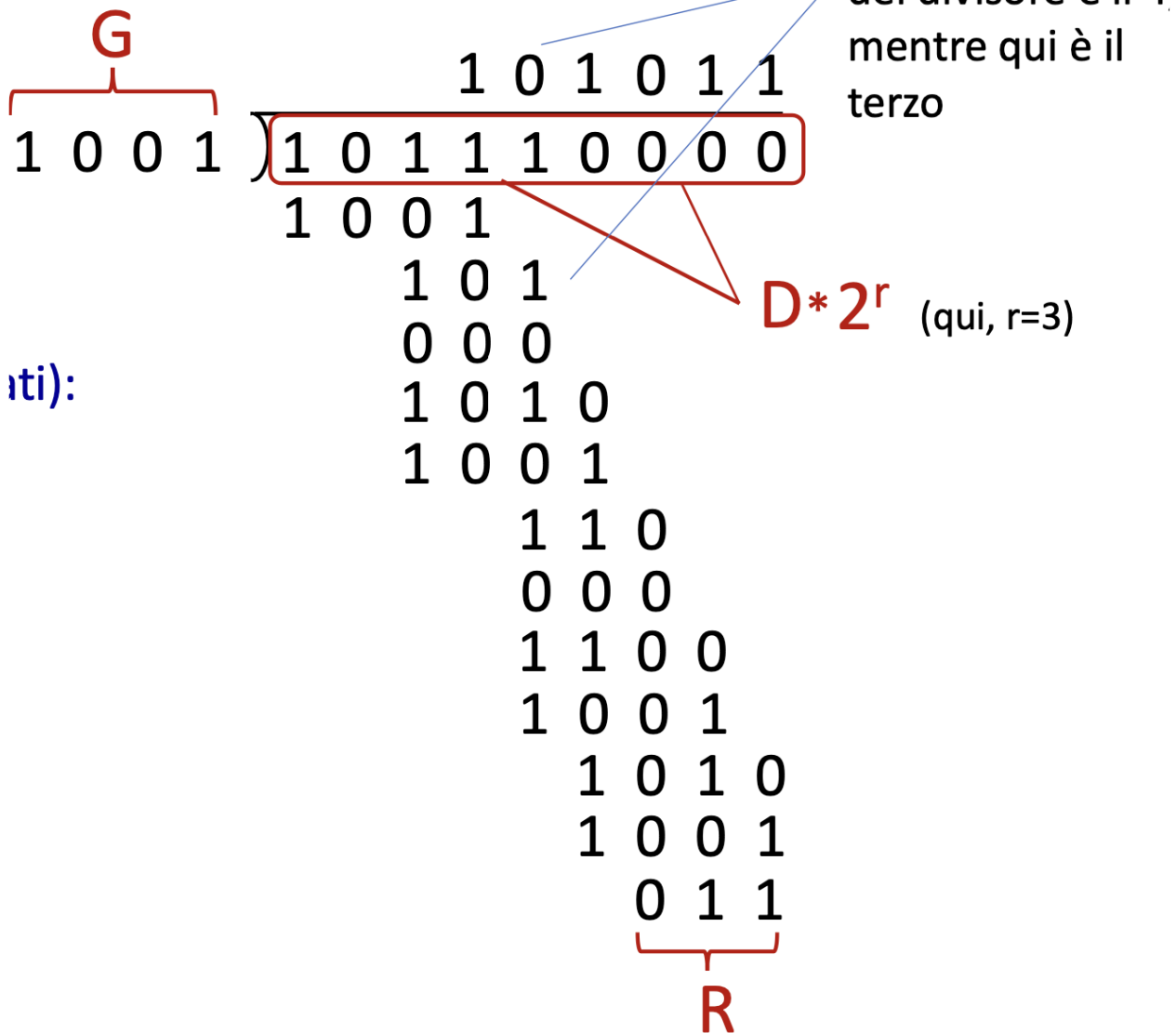
somma modulo 2

Esempio

Il mittente vuole calcolare R tale che: $D * 2^r \text{ XOR } R = nG \dots$ o
 equivalentemente (XOR R in entrambi i lati): $D * 2^r = nG * \text{XOR} * R$
 ... che ci dice che: se dividiamo $D * 2^r$ per G il resto è precisamente R

$$R = \text{resto} \left[\frac{D \cdot 2^r}{G} \right]$$

*algoritmo per
calcolare R*



CRC: polinomio di errore e scelta del generatore

il mittente invia $T = D * 2^r * XOR * R$

il ricevente riceve $T' = T XOR E = T + E$ (usando l'addizione CRC)

dove E è un numero (ovvero un polinomio), i cui bit a 1 indicano dove si è verificato un errore.

Siccome T è divisibile per G , T' sarà divisibile per G se e solo se E è divisibile per G .

Quindi, G deve essere scelto in modo tale che NON divida i polinomi

di errore.

In pratica, si cerca di definire G in modo che rilevi diversi tipi di errore.

Se G ha un numero pari di bit a 1, allora è in grado di rilevare qualsiasi numero dispari di errori.

Perché?

La moltiplicazione CRC equivale a fare lo XOR di copie shiftate dello stesso numero. Se questo numero ha parità pari, lo XOR con un altro valore (es. il risultato parziale) produce un risultato che ha la parità di quest'ultimo. Quindi i multipli di G devono avere parità pari.

È anche possibile definire G in modo che sia multiplo di 11 (cioè $x + 1$).

Se G ha almeno due bit a 1, esso è in grado di rilevare qualunque errore singolo.

Perché?

Un errore singolo è espresso dal polinomio x^k che è divisibile solo dai polinomi x^i per $i \leq k$.

Collegamenti e protocolli di accesso multiplo

Due tipi di collegamenti:

- *punto a punto* (point-to-point): un trasmittente a un'estremità del collegamento e un unico ricevente all'altra estremità
 - collegamento punto a punto tra host e switch Ethernet
 - protocollo PPP per accesso dial-up
- *broadcast* (cavo o mezzo condiviso): più nodi, ciascun frame viene ricevuto da tutti i nodi
 - Ethernet "vecchia scuola" (cavo condiviso)
 - canale upstream reti di accesso HFC
 - 802.11 wireless LAN, 4G/4G, satellite

Protocolli di accesso multiplo

- singolo canale broadcast condiviso

- due o più trasmissioni simultanee dai nodi: interferenza
 - *collisione* se un nodo riceve due o più segnali nello stesso istante

Protocollo di accesso multiplo:

- algoritmo distribuito che determina come i nodi condividono il canale, determina quando i nodi possono trasmettere
- la comunicazione sulla condivisione del canale deve utilizzare il canale stesso!
 - nessun canale fuori banda per il coordinamento

Protocolli di accesso multiplo *ideale*

dato: un canale ad accesso multiplo (multiple access channel, MAC) con velocità di R bps

desiderata:

1. quando un solo nodo vuole trasmettere, può inviare a velocità R .
2. quando M nodi vogliono trasmettere, ciascun può inviare a una velocità media R/M .
3. totalmente decentralizzato:
 - nessun nodo speciale che coordina le trasmissioni (il cui fallimento potrebbe bloccare il sistema);
 - nessuna sincronizzazione degli orologi, slot temporali, etc.
4. semplice.

Protocolli di accesso multiplo *tassonomia*

Tre ampie classi:

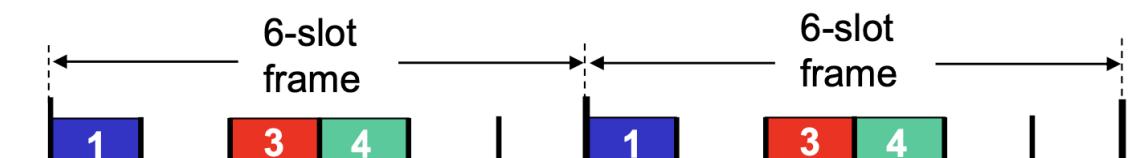
- **a suddivisione del canale (channel partitioning):**

- divide il canale in "pezzi" più piccoli (slot temporali, bande di frequenza, codici)
- assegna un pezzo a un nodo per uso esclusivo
- **ad accesso casuale (random access):**
 - canale non diviso, permette le collisioni
 - recupera ("recover") dalle collisioni (attraverso ritrasmissioni)
- **a rotazione ("taking turns"):**
 - i nodi si avvicendano a turno, ma i nodi con una quantità maggiore di materiale da inviare possono fare turni più lunghi

Protocolli a suddivisione del canale: TDMA

TDMA: time division multiple access (accesso multiplo a divisione di tempo)

- accesso al canale in "intervalli di tempo" (time frame)
- ciascun intervallo è ulteriormente suddiviso in N slot temporali (time slot), ciascun assegnato a uno degli N nodi
- la durata di uno slot temporale è in genere tale da consentire la trasmissione di un pacchetto a livello di collegamento
- gli slot inutilizzati rimangono inutilizzati (idle)
- esempio: LAN con 6 nodi, 1,3,4 hanno pacchetti da inviare, gli slot 2,5,6 sono inattivi

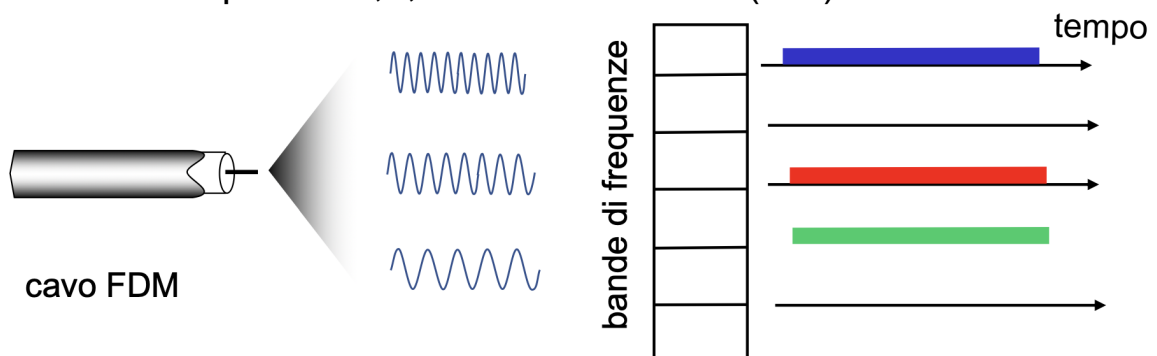


- un nodo che ha dati da trasmettere deve attendere il proprio turno (cioè lo slot assegnatogli)
- nel proprio turno, un nodo trasmette a R bps, ma potendo farlo solo in $1/N$ della durata dell'intervallo temporale, la sua velocità media è R/N ... a prescindere dal fatto che ci siano altri nodi che vogliono trasmettere sul canale

Protocolli a suddivisione del canale: FDMA

FDMA: frequency division multiple access (accesso multiplo a divisione di frequenza)

- lo spettro del canale diviso in bande di frequenza
- a ciascun nodo viene assegnata una banda di frequenze fisse
- il tempo di trasmissione non utilizzato nelle bande di frequenza resta inutilizzato
- esempio: LAN con 6 nodi, 1,3,4 hanno pacchetti da inviare, le bande di frequenza 2,5,6 sono inutilizzate (idle)



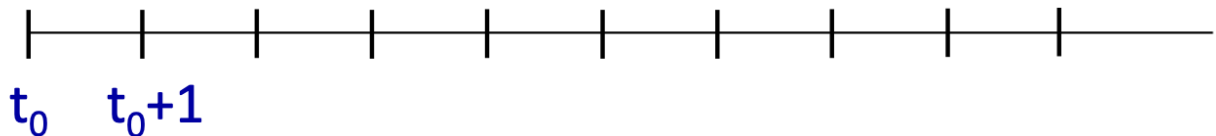
- un nodo può trasmettere nella propria banda di frequenze appena ha dati da inviare, senza dover attendere turni
- Trasmettendo alla velocità massima consentita dalla sua banda di frequenze ridotta, la velocità è R/N a prescindere che altri vogliano trasmettere

Protocolli ad accesso casuale

- quando un nodo ha un pacchetto da inviare
 - trasmette alla massima velocità consentita dal canale, cioè R bps
 - nessun coordinamento a priori tra i nodi
- due o più nodi stanno trasmettendo nello stesso momento: "collisione"
- un **protocollo ad accesso casuale** specifica:
 - come rilevare le collisioni

- come recuperare dalle collisioni (es. attraverso ritrasmissioni con ritardo casuale)
- Esempi di protocolli ad accesso casuale:
 - ALOHA, slotted ALOHA
 - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA



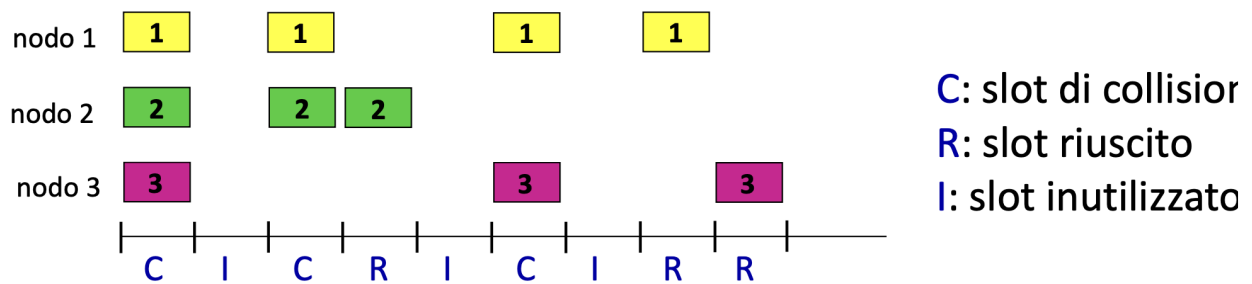
assunzioni:

- tutti i frame hanno la stessa dimensione
- tempo suddiviso in slot temporali uguali (equivalenti al tempo per trasmettere in frame)
- i nodi cominciano la trasmissione soltanto all'inizio degli slot
- i nodi sono sincronizzati
- se 2 o più nodi trasmettono nello stesso slot, tutti i nodi rilevano la collisione prima del termine dello slot

operazioni:

- quando un nodo ha un nuovo frame da spedire, lo trasmette all'inizio dello slot successivo
 - **se non si verifica una collisione:** il nodo può inviare un nuovo frame nello slot successivo
 - **se si verifica una collisione:** il nodo ritrasmette il frame nello slot successivo con probabilità
 - p finché non ha successo

randomizzazione – perché?



Pro:

- un singolo nodo attivo può trasmettere continuamente alla massima velocità del canale
- altamente decentralizzato: solo gli slot nei nodi devono essere sincronizzati
- semplice

Contro:

- collisioni, spreco di slot
- slot inutilizzati (a causa della politica di trasmissione probabilistica)
- i nodi potrebbero essere in grado di rilevare la collisione in meno del tempo necessario per trasmettere il pacchetto
- sincronizzazione degli orologi

Slotted ALOHA: efficienza

efficienza: frazione a lungo termine di slot riusciti (molti nodi, tutti con molti frame da inviare)

- si supponga: N nodi con molti frame da inviare, ciascuno trasmette nello slot con probabilità p
 - probabilità che un dato nodo ha successo in uno slot = $p(1 - p)^{N-1}$
 - probabilità che un nodo qualunque abbia successo = $Np(1 - p)^{N-1}$
 - efficienza massima: trovare p^* che massimizza $Np(1 - p)^{N-1}$

- per molti nodi, calcolare il limite di $Np^*(1 - p^*)^{N-1}$ per N che tende all'infinito, si ottiene:

efficienza massima = $1/e = .37$

- al massimo: canale usato per trasmissione utile solo per il 37% del tempo!

Data la formula dell'efficienza

$$efficienza(p) = Np(1 - p)^{N-1}$$

Ne calcolo la derivata parziale rispetto a p , e ne studio il segno

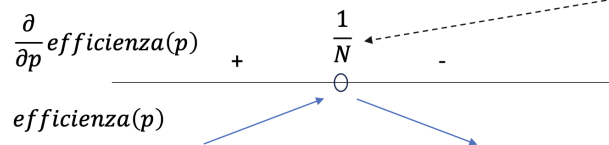
$$\begin{aligned} \frac{\partial}{\partial p} efficienza(p) &= \frac{\partial}{\partial p} Np(1 - p)^{N-1} = N((1 - p)^{N-1} - p(N - 1)(1 - p)^{N-2}) \\ &= N(1 - p)^{N-2}((1 - p) - p(N - 1)) \geq 0 \end{aligned}$$

I primi due fattori sono sempre positivi (assumendo $p \in (0,1)$), pertanto mi concentro sul terzo fattore.

$$(1 - p) - p(N - 1) \geq 0$$

$$1 - p - pN + p \geq 0$$

$$p \leq \frac{1}{N}$$



Massimo della funzione *efficienza*

$$p^* = \frac{1}{N}$$

$$eff(p^*) = eff\left(\frac{1}{N}\right) = \left(1 - \frac{1}{N}\right)^{N-1}$$

Link Layer 4

Efficienza massima per un *gran numero di nodi*

$$\lim_{N \rightarrow \infty} efficienza(p^*) = \lim_{N \rightarrow \infty} \left(1 - \frac{1}{N}\right)^{N-1} = \lim_{N \rightarrow \infty} \frac{\left(1 - \frac{1}{N}\right)^N}{1 - \frac{1}{N}} = \frac{1}{e} = 37\%$$

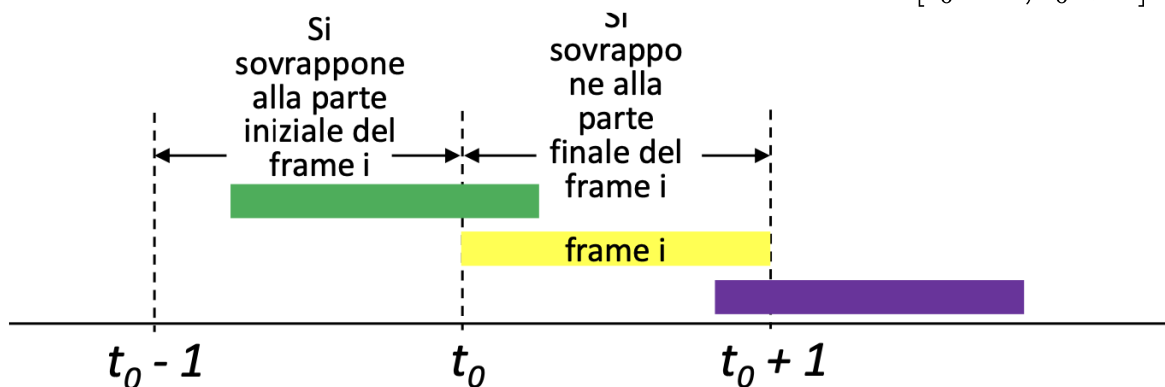
$$\text{visto che } \lim_{N \rightarrow \infty} \left(1 - \frac{1}{N}\right)^N = \frac{1}{e} \text{ e } \lim_{N \rightarrow \infty} 1 - \frac{1}{N} = 1$$

Solo il 37% degli slot svolge un lavoro utile. Pertanto, la velocità di trasmissione effettiva del canale non è R bps, ma solo $0.37 R$ bps! Un'analisi simile mostra anche che il 37% degli slot va a vuoto e il 26% degli slot subisce collisioni.

ALOHA puro

- "unslotted" ALOHA: più semplice, nessuna sincronizzazione
- appena arriva un nuovo frame: lo trasmette immediatamente e integralmente
- se la trasmissione va in collisione: ritrasmette il frame immediatamente (dopo averne completato la trasmissione) con probabilità p , altrimenti attende il tempo di trasmissione di un frame e ripete il processo di attesa casuale, finché non ha successo

- la probabilità di collisione aumenta in assenza di sincronizzazione:
 - il frame inviato a t_0 collide con altri frame inviati in $[t_0 - 1, t_0 + 1]$



- Efficienza massima del protocollo ALOHA puro: 18% !

Accesso multiplo con rilevamento della portante (carrier sense multiple access, CSMA)

CSMA: ascolta prima di trasmettere

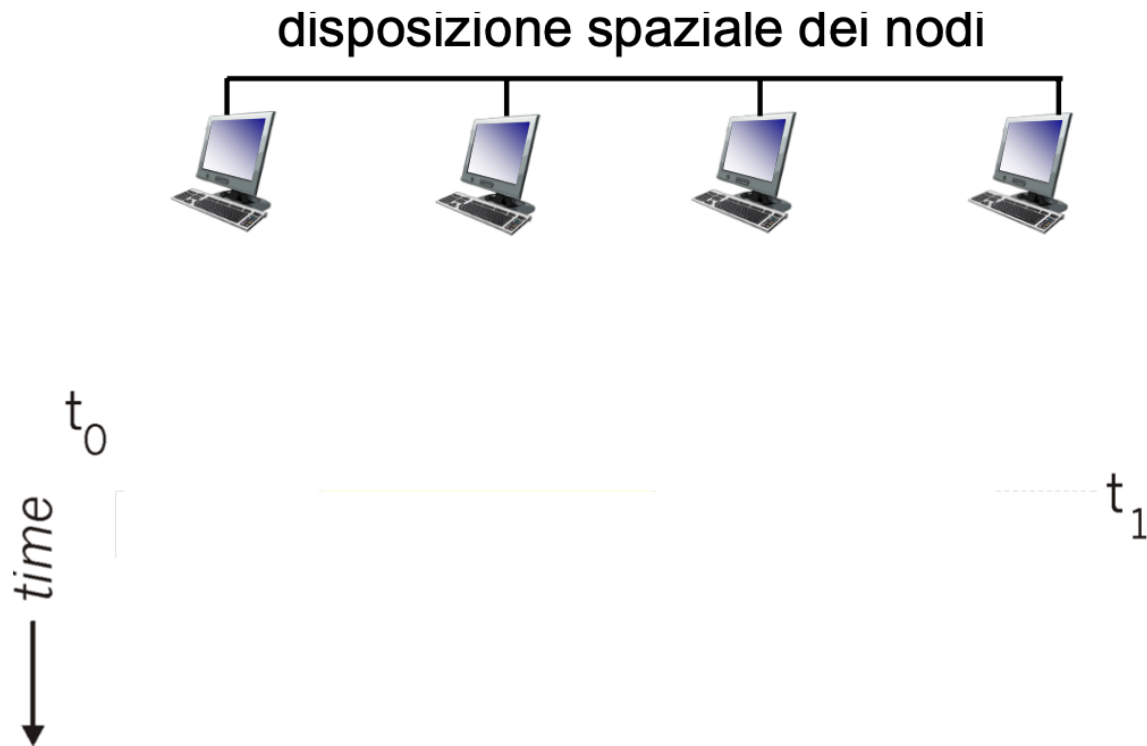
- se percepisce il canale inattivo: trasmette l'intero frame
- se percepisce il canale occupato: differisce la trasmissione
- analogia umana: non interrompere gli altri!

CSMA/CD: CSMA con *rilevamento della collisione* (collision detection)

- collisioni rilevate in breve tempo
- le trasmissioni in collisione vengono interrotte, riducendo gli sprechi di canale
- rilevamento delle collisioni facile con il cavo, difficile con il wireless
- analogia umana: se qualcun altro comincia a parlare insieme a voi, smettere di parlare

CSMA collisioni

- le collisioni possono ancora verificarsi con il rilevamento della portante:
 - il ritardo di propagazione significa che due nodi potrebbero non sentire la trasmissione appena avviata dell'altro
- collisione: spreco dell'intero tempo di trasmissione dei pacchetti
 - la distanza e il ritardo di propagazione giocano un ruolo importante nel determinare la probabilità di collisione



CSMA/CD

- CSMA/CD riduce la quantità di tempo sprecato nelle collisioni
 - trasmissione interrotta su rilevamento di collisione



Algoritmo CSMA/CD di Ethernet

1. Ethernet riceve un datagramma dal livello di rete, crea un frame
2. se Ethernet percepisce il canale:
 - inutilizzato*: avvia la trasmissione del frame.
 - occupato*: aspetta finché il canale è libero, poi trasmette
3. se l'intero frame viene trasmesso senza collisioni - fatto!
4. se durante l'invio viene rilevata un'altra trasmissione:
interrompere, inviare il segnale di disturbo (jam)
5. dopo aver interrotto, entra nella binary exponential backoff:
 - dopo la m -esima collisione, scegli K casualmente tra $\{0, 1, 2, \dots, 2^m - 1\}$. Ethernet aspetta il tempo di trasmissione di $K \cdot 512$ bit, ritorna allo Step 2
 - più collisioni: maggiore intervallo di backoff (ma m viene limitato a 10)

Efficienza CSMA/CD

- d_{prop} = massimo ritardo di propagazione tra due schede di rete

- d_{trasm} = tempo necessario per trasmettere un frame di dimensione massima

$$efficienza = \frac{1}{1 + 5d_{prop}/d_{trasm}}$$

- l'efficienza tende a 1
 - se t_{prop} tende a 0 (perché la trasmissione viene interrotta subito in presenza di collisioni, evitando sprechi)
 - se t_{trasm} tende a infinito (perché un frame, appropriatosi del canale, lo impegna a lungo)
- prestazioni migliori di ALOHA: e semplice, economico, decentralizzato!

Protocolli a rotazione

protocolli a suddivisione del canale:

- condividere il canale in modo efficiente ed equo con un carico elevato
- inefficiente a basso carico: ritardo nell'accesso al canale (nel TDMA), larghezza di banda $1/N$ allocata anche se solo 1 nodo attivo!

protocolli ad accesso casuale:

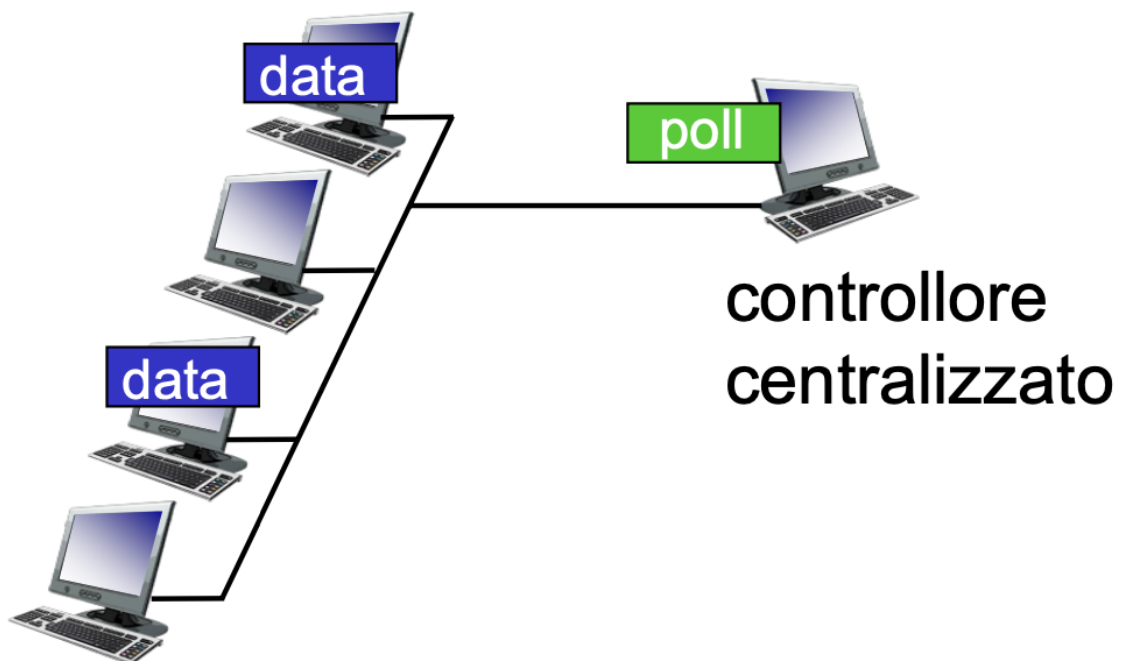
- efficiente a basso carico: un singolo nodo può usare il canale completamente
- alto carico: overhead di collisione

protocolli a rotazione:

- cercate il meglio dei due modi

Polling:

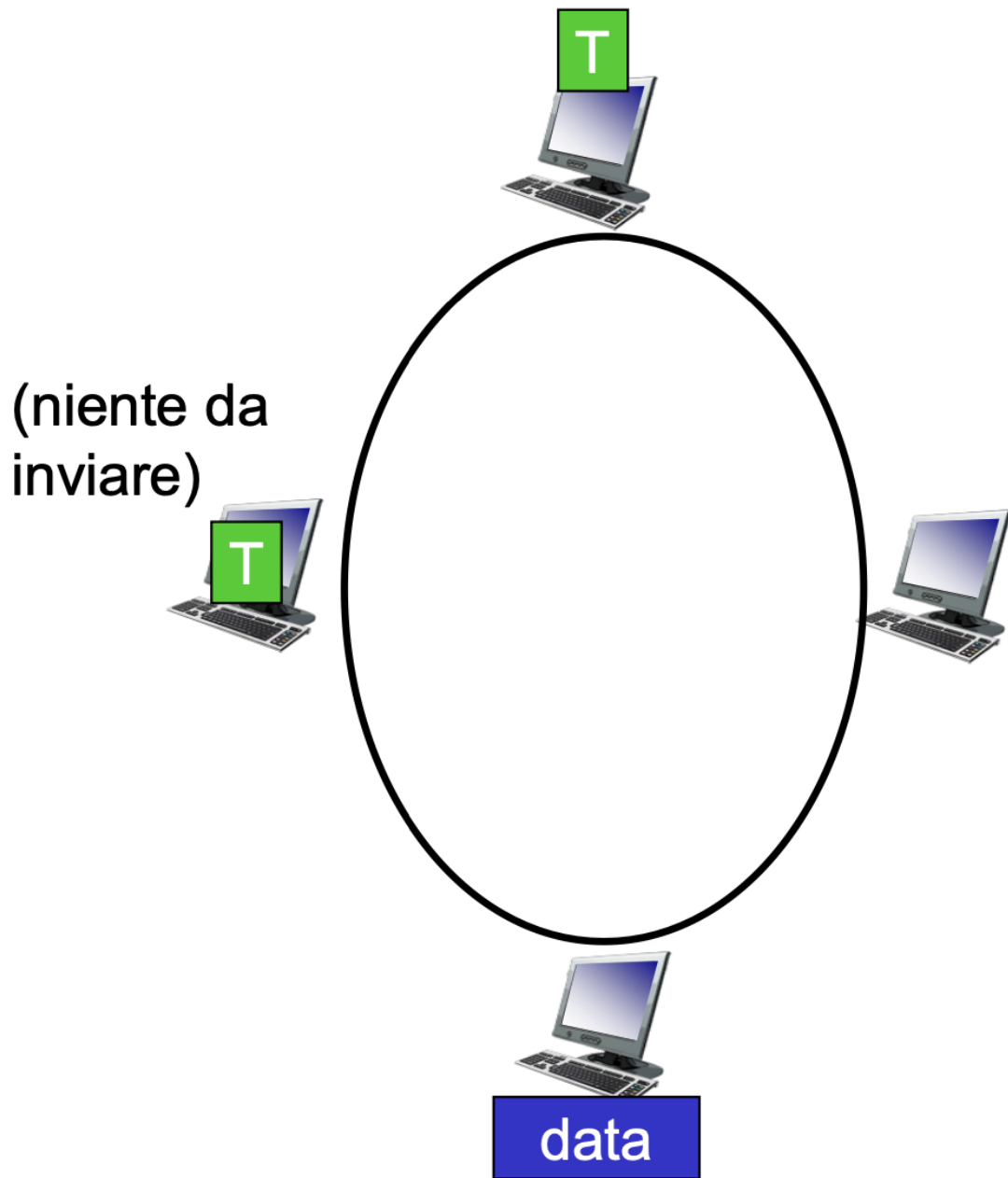
- il controllore centralizzato “invita” gli altri nodi a trasmettere a loro volta (fino a un massimo di frame)
 - il controllore determina che il client ha finito osservando la mancanza di segnale
- *problemi:*
 - **Overhead del polling**
 - **ritardo di polling:** il tempo impiegato per notificare a un nodo il permesso di trasmettere -> anche in presenza di un solo nodo attivo, il controllore deve contattare periodicamente tutti gli altri nodi, determinando un throughput effettivo minore di R
 - **singolo punto di rottura** (master)
 - Il Bluetooth usa il polling



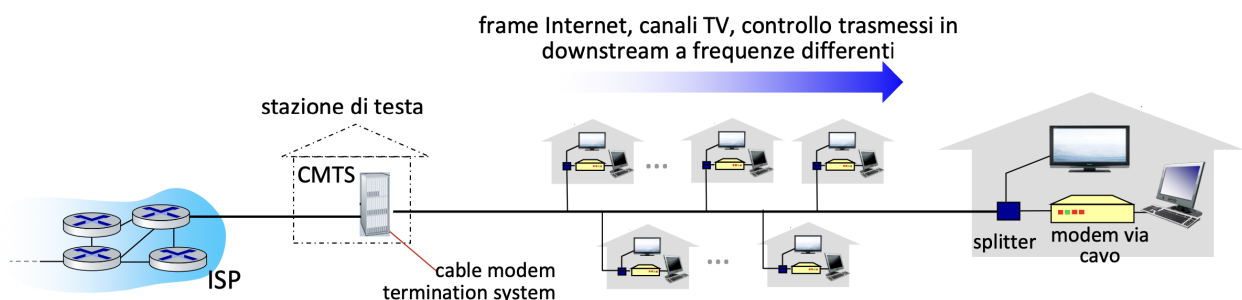
Token passing:

- Messaggio di controllo detto token (gettone) passato esplicitamente da un nodo al successivo, sequenzialmente
- trasmette mentre possiede il token
- *problemi:*
 - overhead associato al token

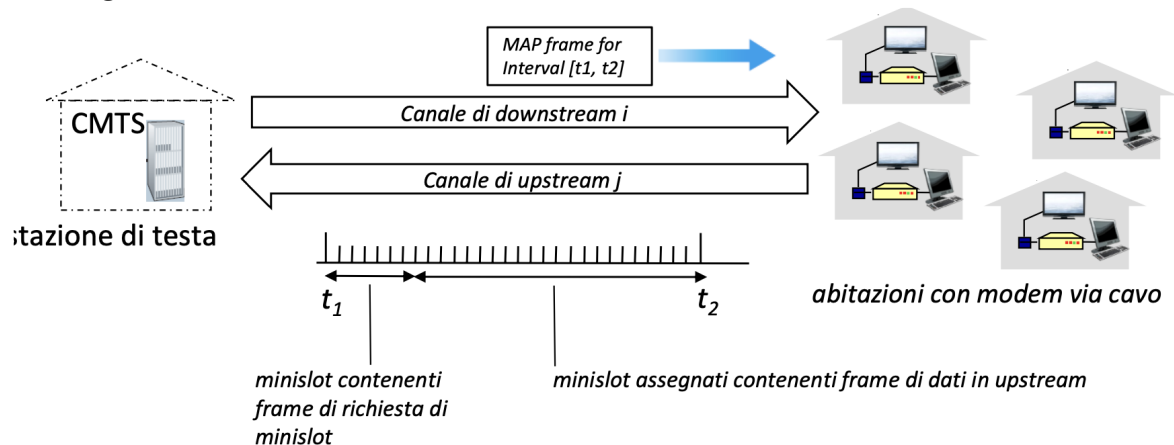
- latenza
- singolo punto di rottura (token)
- Usato in: FDDI e token ring (IEEE 802.5)



Rete di accesso via cavo: FDM, TDM, allocazione centralizzata e accesso casuale!



- *molteplici* canali FDM downstream (broadcast): fino a 1.6 Gbps/canale
 - un solo CMTS trasmette nei canali -> nessun problema di accesso multiplo
- *molteplici* canali upstream (fino a 1 Gbps/canale)
 - *accesso multiplo*: tutti gli utenti si contendono (accesso casuale) determinati slot temporali del canale upstream; agli altri vengono assegnati TDM



DOCSIS: specifiche di interfaccia del servizio dati via cavo

- FDM su canali di frequenze upstream e downstream
- TDM upstream: alcuni slot assegnati, alcuni sono contesi
 - Frame MAP in downstream: assegna i minislot in upstream
 - Richieste di frame in upstream (e dati) trasmessi con accesso casuale (binary backoff) in slot selezionati

Riassunto dei protocolli di accesso multiplo

- **suddivisione del canale**, per tempo, frequenza o codice
 - Time Division, Frequency Division
- **accesso casuale** (dinamico),
 - ALOHA, S-ALOHA, CSMA, CSMA/CD
 - rilevamento della portante: facile in alcune tecnologie (cablate), difficile in altre (wireless)
 - CSMA/CD usato in Ethernet
 - CSMA/CA usato in 802.11
- **a rotazione**

- polling da un sito centrale, token