

Università degli Studi di Roma "Tor Vergata"
Laurea in Informatica

Sistemi Operativi e Reti
(modulo Reti)
a.a. 2024/2025

Livello di collegamento (parte2)

dr. Manuel Fiorelli

manuel.fiorelli@uniroma2.it

<https://art.uniroma2.it/fiorelli>

Basate sulle slide del libro di testo:

https://gaia.cs.umass.edu/kurose_ross/ppt.php

Livello di collegamento e LAN: tabella di marcia

- introduzione
- rilevazione e correzione degli errori
- protocolli di accesso multiplo
- LAN
 - indirizzamento, ARP
 - Ethernet
 - switch
 - VLAN
- canali virtuali: MPLS
- Reti dei data center



- un giorno nella vita di una richiesta web

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**: un canale broadcast **condiviso** tra più nodi trasmittenti e riceventi, ciascun frame viene ricevuto da tutti i nodi
 - Ethernet "vecchia scuola" (cavo condiviso)
 - 802.11 wireless LAN, 4G/5G, satellite



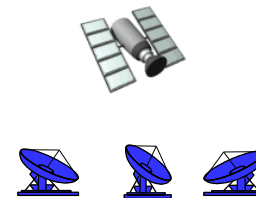
cavo condiviso (es.,
Ethernet cablata)



spettro radio
condiviso: 4G/5G



spettro radio
condiviso: WiFi



spettro radio
condiviso : satellite



umani a una festa
(aria condivisa, acustico)

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

Un protocollo 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

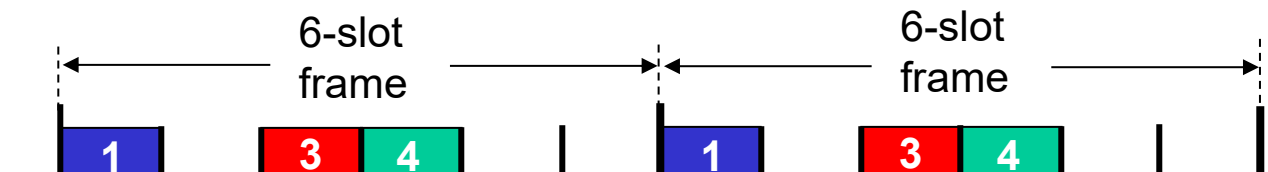
Tre ampie classi:

- **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

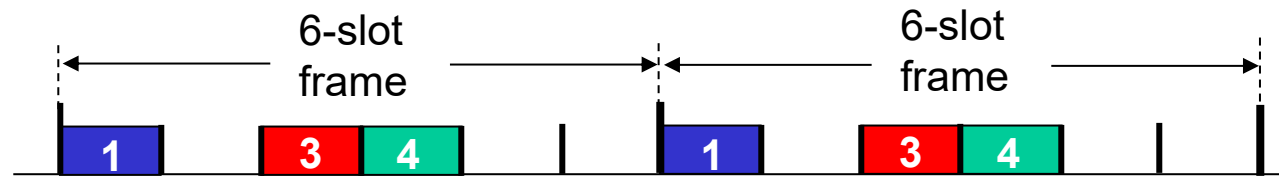
TDMA: time division multiple access (accesso multiplo a divisione di tempo)

- 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



Protocolli a suddivisione del canale: TDMA

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

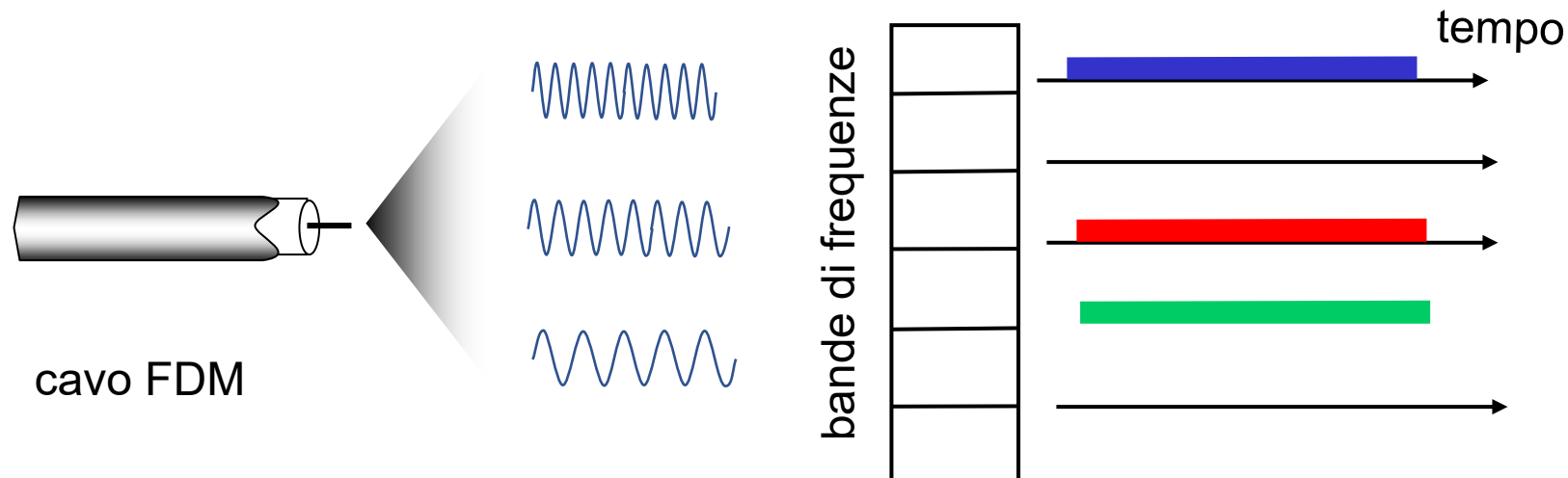


- un nodo che ha dati da trasmettere deve attendere il proprio turno (cioè lo slot assegnatogli)
- nel proprio turno, un nodo trasmette a R bps, ma potendo farlo solo in $1/N$ della durata dell'intervallo temporale, la sua velocità media è R/N ... a prescindere dal fatto che ci siano altri nodi che vogliono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

Protocolli a suddivisione del canale: FDMA

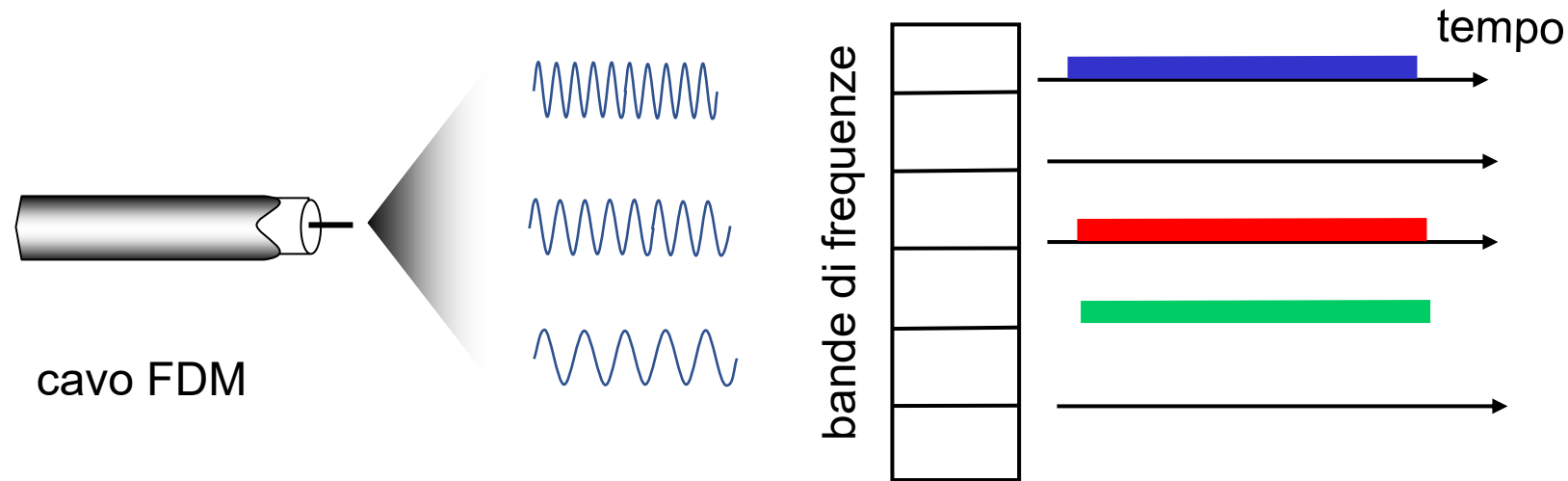
FDMA: frequency division multiple access (accesso multiplo a divisione di frequenza)

- 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*)



Protocolli a suddivisione del canale: FDMA

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

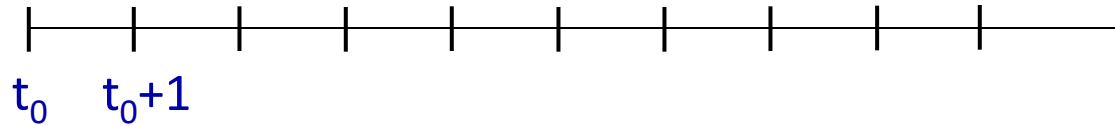


- un nodo può trasmettere nella propria banda di frequenze appena ha dati da inviare, senza dover attendere turni
- Trasmettendo alla velocità massima consentita dalla sua banda di frequenze ridotta, la velocità è R/N a prescindere che altri vogliano trasmettere

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 access 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:

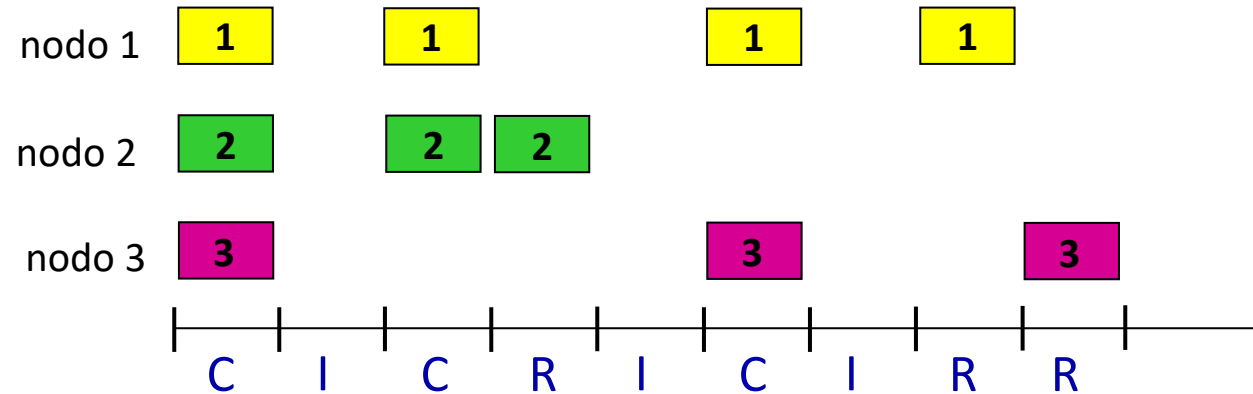
- tutti i frame hanno la stessa dimensione
- tempo suddiviso in slot temporali uguali (equivalenti al tempo per trasmettere in frame)
- i nodi cominciano la trasmissione soltanto all'inizio degli slot
- i nodi sono sincronizzati
- se 2 o più nodi trasmettono nello stesso slot, tutti i nodi rilevano la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

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?

Slotted ALOHA



C: slot di collisione
R: slot riuscito
I: slot inutilizzato

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 : il numero di nodi che trasmettono in uno slot è una variabile aleatoria binomiale $B(N, p)$
 - 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!



Slotted ALOHA: efficienza

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

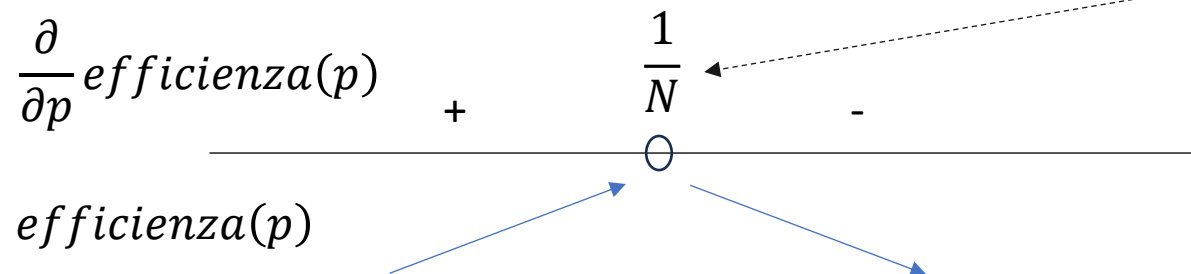
$$\begin{aligned}\frac{\partial}{\partial p} efficienza(p) &= \frac{\partial}{\partial p} Np(1 - p)^{N-1} = N((1 - p)^{N-1} - p(N - 1)(1 - p)^{N-2}) \\ &= N(1 - p)^{N-2}((1 - p) - p(N - 1)) \geq 0\end{aligned}$$

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*

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

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

Slotted ALOHA: efficienza

Efficienza massima per un *gran numero di nodi*

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

visto che:

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

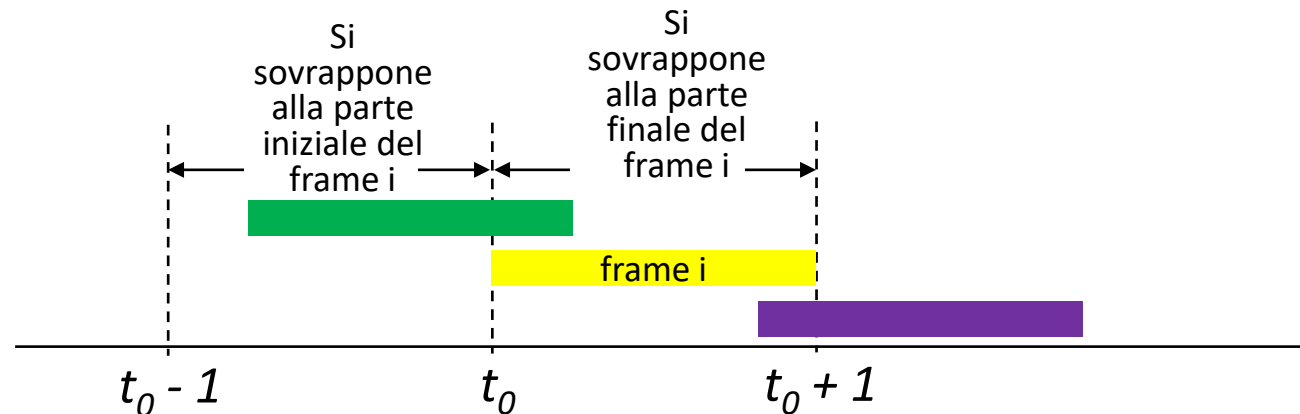
Dato il limite notevole $\lim_{x \rightarrow \pm\infty} \left(1 + \frac{a}{x}\right)^{bx} = e^{ab}$ https://it.wikipedia.org/wiki/Limite_notevole

- $\lim_{N \rightarrow \infty} 1 - \frac{1}{N} = 1$

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 (lo rileva per esempio a causa dell'assenza di ACK): 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:

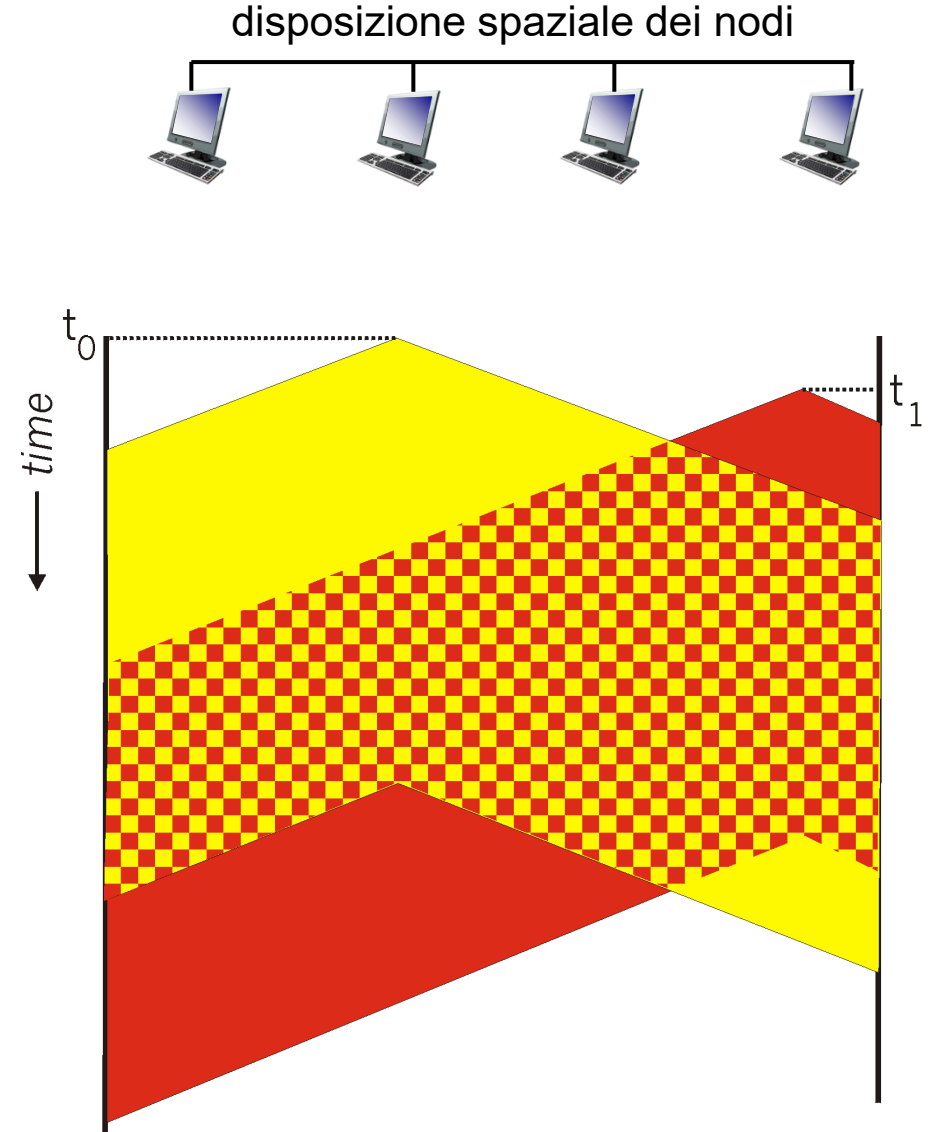
- se percepisce il canale inattivo: trasmette l'intero frame
 - se percepisce il canale occupato: differisce la trasmissione
- analogia umana: non interrompere gli altri!

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

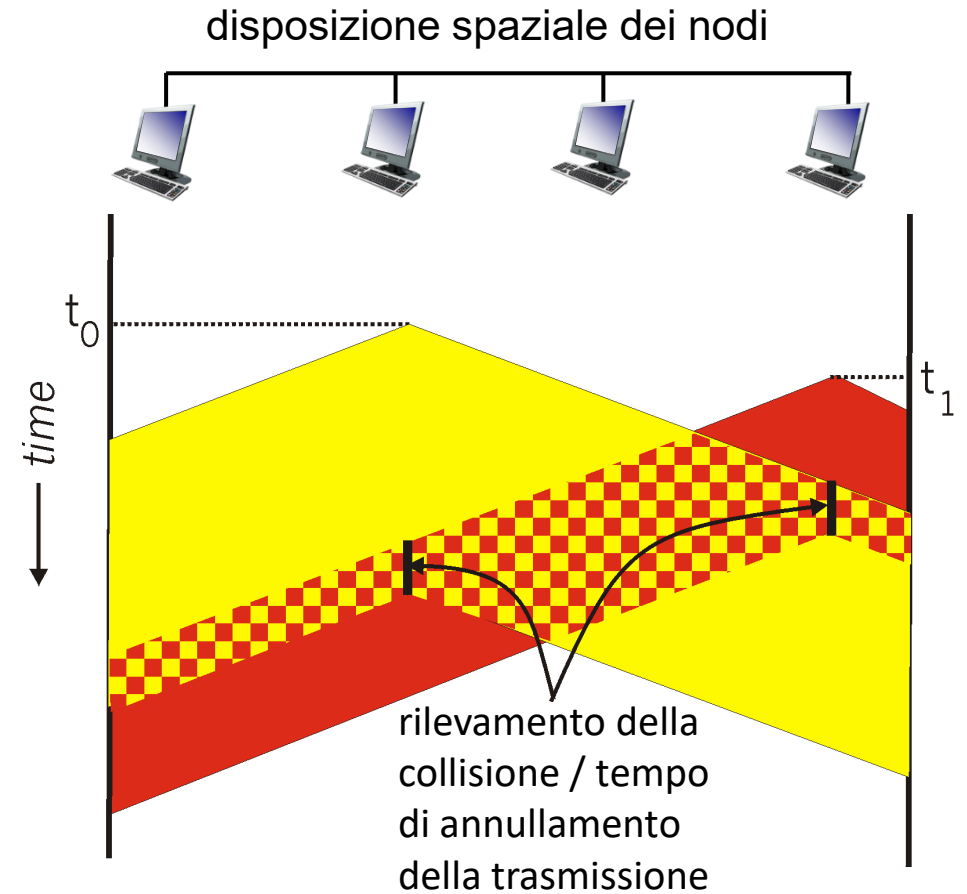
CSMA: collisioni

- 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

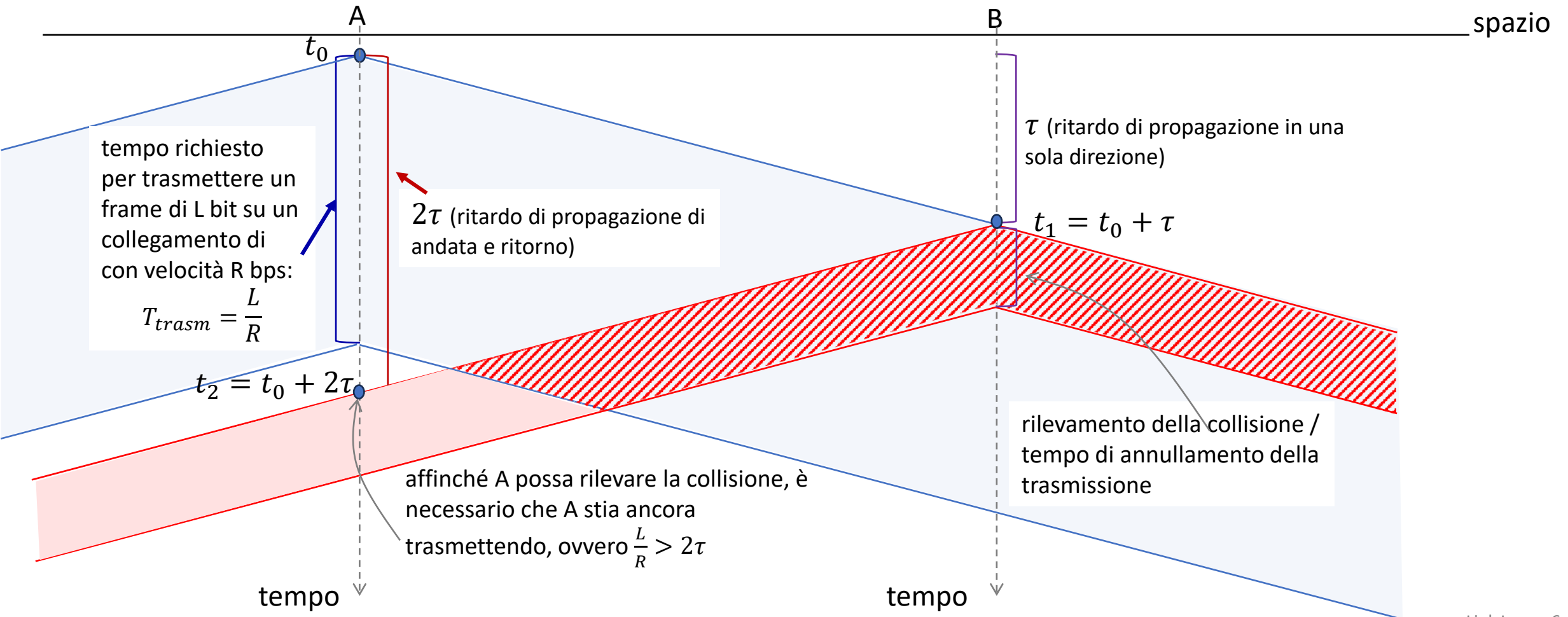


Algoritmo CSMA/CD di Ethernet

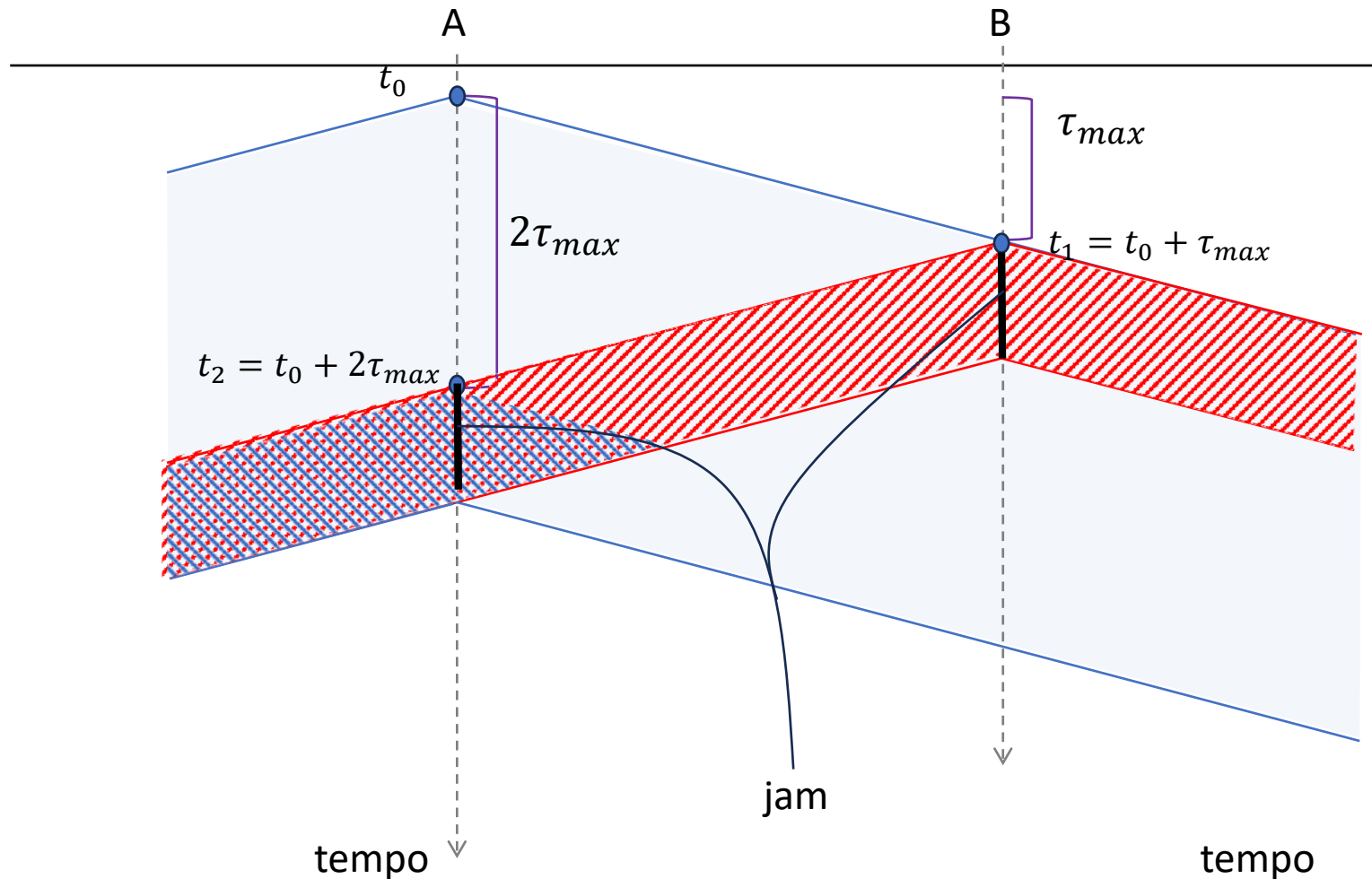
1. Ethernet riceve un datagramma dal livello di rete, crea un frame
2. se Ethernet ascolta il canale:
 - inutilizzato:** avvia la trasmissione del frame.
 - occupato:** aspetta finché il canale è libero, poi trasmette
3. se l'intero frame viene trasmesso senza collisioni - fatto!
4. se durante l'invio 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)

CSMA/CD: Vincoli (1)

B inizia a trasmettere un attimo prima che possa rilevare il segnale di **A**. Rilevata la collisione (all'istante $t_1 = t_0 + \tau$), **B** interrompe la trasmissione del frame e invia il disturbo (*jam*). Il segnale di **B** può essere rilevato da **A** all'istante $t_2 = t_0 + 2\tau$. Se **A** avesse già finito la trasmissione del suo frame, non rileverebbe la collisione.



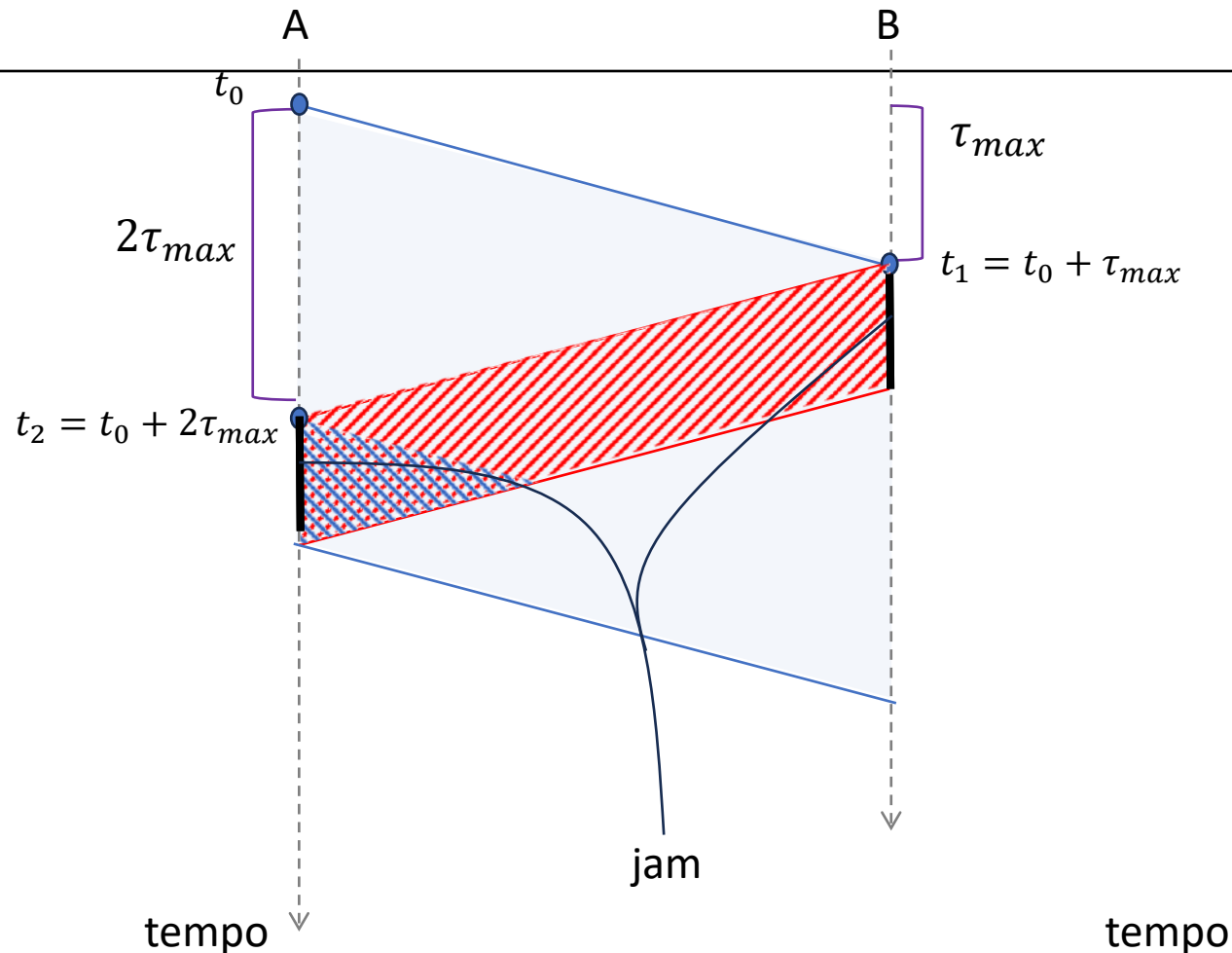
CSMA/CD: Vincoli (2)



Viene stabilito un ritardo massimo τ_{max} , che include il ritardo di propagazione attraverso il mezzo così come il contributo di altri componenti e processi, più un margine di sicurezza.

τ_{max} è spesso espresso in bit e va interpretato come il tempo impiegato per trasmettere quel numero di bit: ne consegue che maggiore è la velocità di trasmissione, minore è il ritardo ammesso in secondi. Ne deriva, quindi, che anche il limite di distanza fisica (il diametro fisico del dominio di collisione) diminuisce al crescere della velocità di trasmissione.

CSMA/CD: Vincoli (3)



Supponiamo che **B** sia proprio alla massima distanza da **A**. Affinché CSMA/CD funzioni, cioè che **A** possa rilevare sempre la collisione, dobbiamo garantire che **A** stia ancora trasmettendo quando arriva il segnale da **B**. A tale scopo si richiede che un frame abbia una lunghezza minima tale per cui la sua trasmissione richieda almeno uno *slot time*, così definito:

$slot\ time = 2\tau_{max} + massima_durata_jam$
(uguale a 512 bit in Ethernet a 10 e 100 Mbps, cioè 64 byte)

Si noti che nel caso peggiore **B** inizia a trasmettere un *epsilon* prima di poter rilevare la trasmissione di **A**: quest'ultimo rileverà la collisione entro lo *slot time*. Trascorso uno slot time, **A** è sicuro che non rileverà più alcuna collisione, perché ciò richiederebbe che **B** abbia iniziato a trasmettere in un istante $t'_1 > t_0 + \tau_{max}$, cosa impossibile perché a quel punto **B** avrebbe potuto rilevare la trasmissione di **A** (carrier sense). Si dice che **A** ha acquisito il canale.

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), $1/N$ della larghezza di banda allocata anche se solo 1 nodo attivo!

protocolli ad accesso casuale

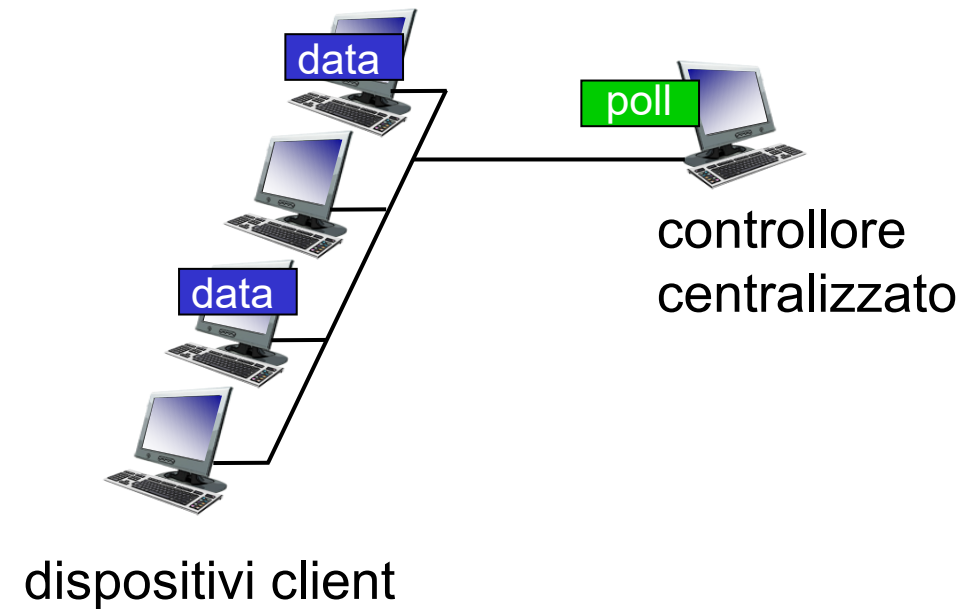
- efficiente a basso carico: un singolo nodo può usare il canale completamente
- alto carico: *overhead* di collisione

protocolli a rotazione

- cercate il meglio dei due mondi!

Protocolli a rotazione polling:

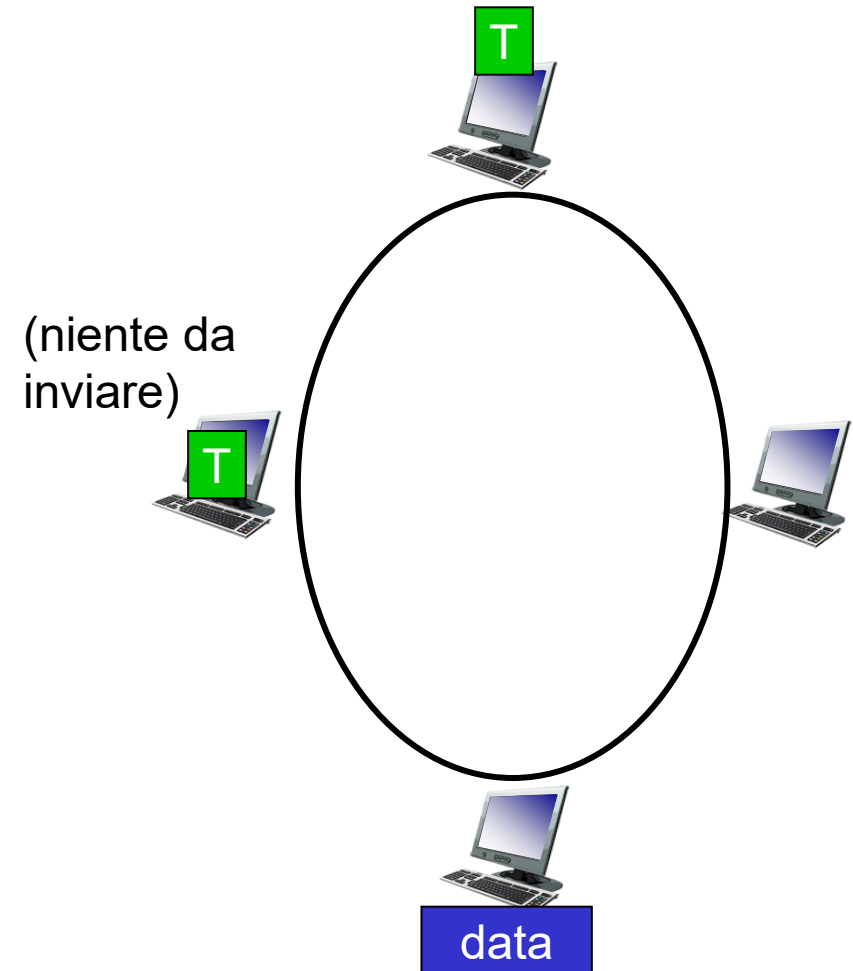
- il controllore centralizzato “invita” gli altri nodi a trasmettere a loro volta (fino a un massimo di frame per turno)
 - il controllore determina che il client ha finito osservando la mancanza di segnale
 - elimina collisioni e slot inutilizzati
- problemi:
 - **overhead del polling**
 - **ritardo 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



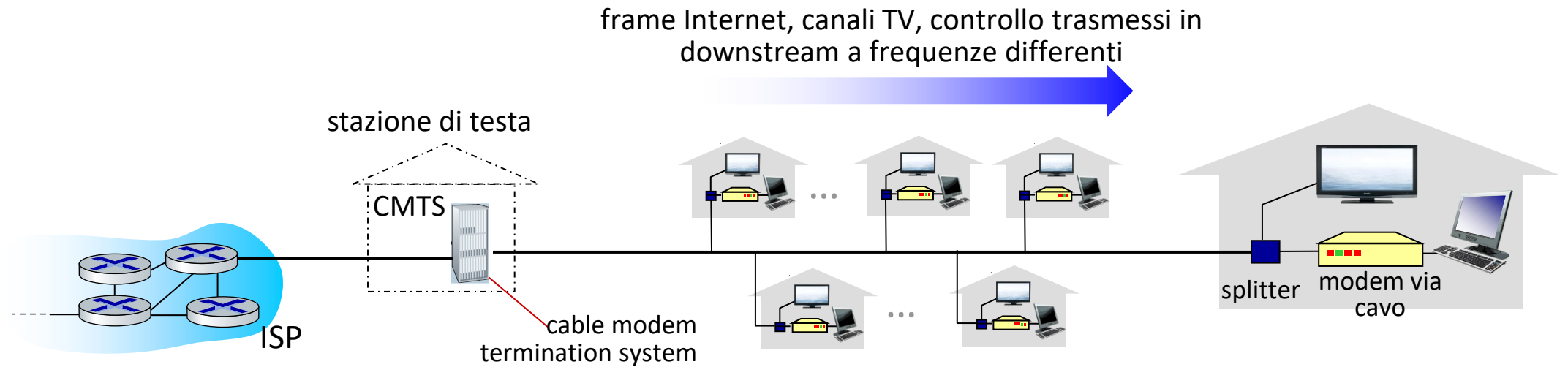
Protocolli a rotazione

token passing:

- Messaggio di controllo detto *token* (**gettone**) passato esplicitamente da un nodo al successivo, sequenzialmente
 - trasmette mentre possiede il token (entro un massimo accordato)
- problemi:
 - overhead associato al token
 - latenza
 - singolo punto di rottura (token)
- Usato in: FDDI e token ring (IEEE 802.5)

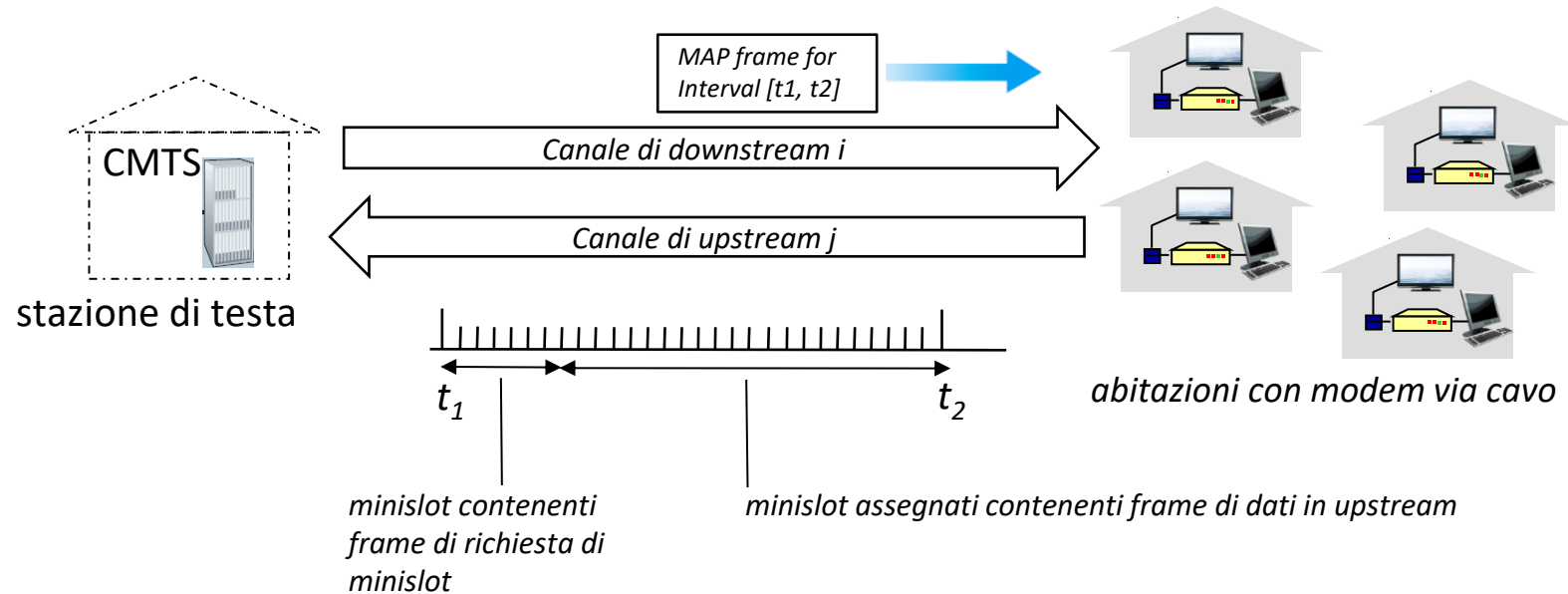


Rete di accesso via cavo: FDM, TDM, allocazione centralizzata e accesso casuale!



- **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

Rete di accesso via cavo;



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 passing
 - Bluetooth, FDDI, token ring