

## 1 Il livello Data Link e le LAN

### 1.1 Il data link layer

I dispositivi (host, router, server ecc.) che supportano il data link layer sono detti **nodi**. La comunicazione avviene tramite **canali** che connettono nodi adiacenti, questi canali vengono chiamati **link** (via cavo, wireless), un'intera rete LAN viene vista come un unico link. I pacchetti che vengono trasmessi a livello data link si chiamano **frame**.

#### Richiami dell'esame di reti

A livello IP il pacchetto trasmesso è chiamato **datagramma**, a livello trasporto TCP il pacchetto è chiamato **segmento**.

#### 1.1.1 I servizi forniti

**Incapsulamento (framing)** Nella maggior parte delle volte la parte contenente i dati del frame contiene il datagramma IP. Il livello data link fornisce come servizio quello di trasferire un datagramma da un nodo ad un altro attraverso i link.

**Accesso al link** Un protocollo di medium access control (MAC) specifica come il frame deve essere trasmesso sul link.

**Trasporto affidabile** Un protocollo di trasferimento affidabile garantisce che ogni frame raggiunga la sua destinazione senza errori. Assicura che il trasferimento sia affidabile tra ogni nodo in modo da non dover rimandare tutto il pacchetto TCP ma solo i frame (datagrammi) danneggiati.

**Individuazione e correzione degli errori** Il nodo mittente fornisce un meccanismo per individuare gli errori, che verranno poi corretti dal destinatario.

**Controllo di flusso** La velocità con cui il nodo mittente trasmette e il nodo ricevente riesce a ricevere.

**Half-duplex e full-duplex** Con half-duplex in una comunicazione tra due nodi solo uno alla volta può trasmettere mentre l'altro resta in attesa e viceversa. Con full-duplex entrambi i nodi possono trasmettere e ricevere contemporaneamente.

### 1.1.2 Implementazioni

Il data link layer deve essere implementato in ogni nodo connesso alla rete. Le funzionalità Ethernet sono integrate nella scheda madre o in un chip Ethernet. Il livello collegamento è implementato su un chip detto network adapter o NIC.

## 1.2 Individuazione e correzione degli errori

L'individuazione degli errori prevede aggiungere ai dati trasmessi dei bit/byte aggiuntivi che servono per permettere di rilevare o correggere l'errore. Potrebbero presentarsi degli errori sia sulla parte dei dati che sui bit aggiuntivi perché viaggiano su un canale non affidabile.

### 1.2.1 Controllo di parità

Ad una certa sequenza di bit aggiunto un singolo bit, il bit di parità, che viene aggiunto dal mittente seguendo certe regole che dovranno essere condivise con il ricevente. Se il numero di bit corrotti è pari il test viene passato con successo anche se è presente un errore.

#### Esempio

Aggiungo un 1 se il numero di 1 è dispari 0 se il numero di 1 è pari. Attraverso le matrici di parità calcolo i bit di parità sia per le righe che per le colonne e confronto se il numero di bit coincide con il bit di parità. Posso correggere l'errore solo se è presente **1 errore** ma segnalare l'errore su **2 bit**.



Foto/CheckParita.png

### 1.2.2 CRC (Cyclic Redundancy Check)

É uno dei più potenti rilevatori di errori ma non può correggerli.

I codici CRC sono anche detti codici polinomiali in quanto 'e possibile considerare la stringa da inviare come un polinomio i quali coefficienti sono i valori 0 e 1 della stringa di bit. I CRC possono individuare *resto  $\leq$  bit errati consecutivi*.

### Svolgimento (1)

$D$  = numero di bit per i dati;

$G$  = sequenza di bit (generatore) che deve soddisfare determinate caratteristiche. In questo caso  $r + 1$  bit.

Aggiungo in coda una sequenza di CRC lunga  $r$  bit (un bit in meno rispetto al generatore).

$$\langle D, R \rangle = D \cdot 2^r \text{ XOR } R$$

$D \cdot 2^r$  shifto il dato di  $r$  bit

$\text{XOR } R$  metto come ultimi bit significativi quelli di  $R$

Il mittente prende gli  $R$  bit dei CRC e prende  $\langle D, R \rangle$  e lo divide per  $G$ , fa il modulo 2, **se il resto della divisione è 0 il test è passato** altrimenti segnalo che è presente un errore.

Posso rilevare un numero di bit errati  $< G$ . Inoltre posso rilevare un numero di errori dispari.

### Svolgimento (2) Come calcolare R

$R$  deve essere tale per cui:

$$D \cdot 2^r \text{ XOR } R = nG$$

AB	A XOR B
00	0
01	1
10	1
11	0

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

### Esempio

$D=101110$     $G=1001$     $r=3$

```
10111000 :1001
1001 XOR  101011
  101
 000 XOR
 1010
1001 XOR
  110
 000 XOR
 1100
1001 XOR
  1010
 1001 XOR
   011 = R
```

Alla fine il **mittente** trasmetterà come sequenza 101110**011** con gli ultimi 3 bit uguali a quelli calcolati.

Il **ricevente** prende la sequenza e la divide per  $G$ , se il resto è 0 il test è passato altrimenti segnalo la presenza di un errore.

**Generatori** Esistono dei generatori  $G$  standard da 8, 12, 16 o 32 bit prestabiliti.

## 1.3 Protocolli ad accesso multiplo

Esistono due tipologie di link:

1. **Punto-Punto** = un nodo è collegato direttamente ad un altro attraverso un link (canale) dedicato;
  - eg. collegamento ethernet tra un host e uno switch.
2. **Broadcast** = il link è condiviso, potenzialmente diversi nodi possono trasmettere e/o ricevere contemporaneamente;
  - Eg. prime versioni di ethernet (tipologia a bus), comunicazione wireless.

I protocolli ad accesso multiplo stabiliscono un insieme di regole che ogni nodo deve seguire affinché la comunicazione avvenga con successo. In assenza di protocolli avremmo il problema delle **collisioni**, cioè due o più nodi che trasmettono contemporaneamente.

Consideriamo algoritmi di tipo distribuito che determinano come il canale viene condiviso, ad esempio determinano quando un determinato nodo deve trasmettere oppure cosa succede in caso di collisione. La comunicazione avverrà, sia per trasmettere dati che per regolare la trasmissione stessa, sullo stesso canale.

**Protocollo ideale** Consideriamo un canale ad accesso condiviso che è in grado di trasmettere ad un rate di  $R$  bps (bit per secondo).

Si desidera che:

1. quando un singolo nodo vuole trasmettere, può trasmettere ad una velocità di trasmissione pari ad  $R$ ;
2. quando ci sono  $M$  nodi che voglio trasmettere contemporaneamente, ogni nodo può trasmettere ad una velocità media di  $\frac{R}{M}$ ;

3. completamente decentralizzato:

- non voglio nodi specializzati che coordinino la trasmissione (single point of failure);
- non voglio la sincronizzazione dei clock dei diversi nodi o degli slot.

4. semplice.

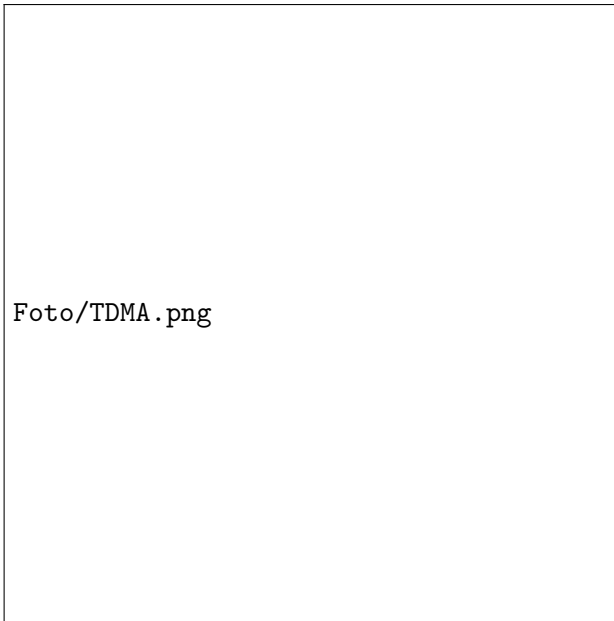
### 1.3.1 Protocolli a partizionamento del canale

Divido la capacità complessiva del canale in diverse parti che assegno ai diversi nodi.

**TDMA** Ogni nodo può accedere al canale solo in determinati intervalli di tempo. Ad ogni stazione viene assegnato un intervallo di tempo fisso in cui può trasmettere. Gli slot inutilizzati vengono persi, NON riutilizzati da altri nodi.

#### Esempio

Il time-frame viene suddiviso in un numero di slot pari al numero di nodi complessivi della rete.



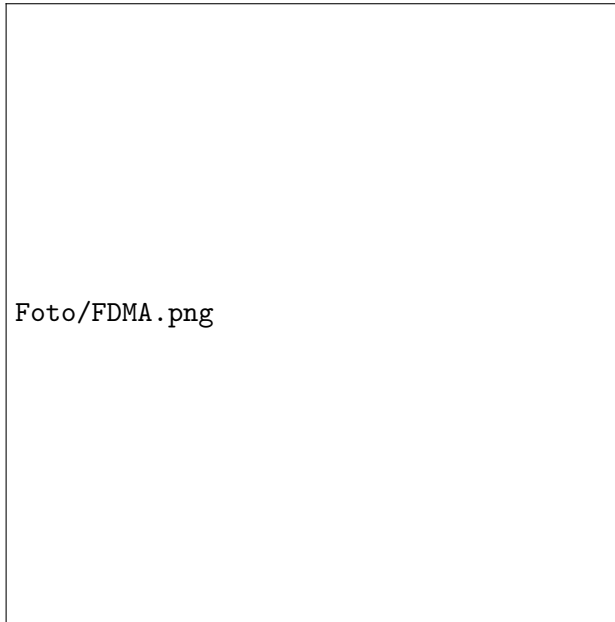
Proprietà ideali:

1. Soddisfatto;
2. Soddisfatto;
3. Non soddisfatto, è necessario mantenere gli stessi clock per sapere quando inizia uno slot;
4. Soddisfatto.

**FDMA** Viene partizionato il dominio delle frequenze. Ad ogni stazione viene assegnata una frequenza stabilita. Il tempo non utilizzato in frequenza viene sprecato. Posso non usare certe frequenze tra un intervallo ed un altro per evitare le sovrapposizioni di banda.

## Esempio

Vengono partizionate le frequenze in base al numero di nodi presenti.



Proprietà ideali:

1. Non soddisfatto;
2. Soddisfatto;
3. Soddisfatto;
4. Soddisfatto;

### 1.3.2 Protocolli ad accesso casuale

Il canale non è diviso e si permette l'occorrenza di collisioni, nel caso di collisioni devo fare in modo di risolverla. Quando un nodo ha un pacchetto da inviare lo trasmette alla massima velocità, a priori non c'è coordinamento tra i nodi.

**Slotted ALOHA** Assumiamo che tutti i frame abbiano la stessa dimensione, il tempo è diviso in slot di uguale dimensione necessaria per trasmettere un intero frame. I nodi iniziano a trasmettere solo all'inizio dello slot. I nodi sono sincronizzati. Se 2 o più nodi trasmettono simultaneamente tutti i nodi della rete sono in grado di rilevare la collisione.

Quando un nodo ottiene un nuovo frame da inviare lo trasmette nel successivo slot, dopo aver trasmetto si mette in ascolto, se c'è un picco di energia viene a conoscenza della presenza di una collisione in questo caso ritrasmetto nel successivo slot con una probabilità  $p$ .

### Esempio

All'inizio tutti i nodi vogliono trasmettere e avviene una collisione. Successivamente, dopo uno slot vuoto, i nodi 1 e 2 ritrasmettono e vanno in collisione. Allo slot successivo solo 2 trasmette ed ha successo. E così via finché tutti non hanno trasmesso.



Proprietà ideali:

1. Soddisfatta;
2. Non soddisfatta;
3. Non soddisfatta, perché è richiesta sincronizzazione;
4. Soddisfatta.

Durante le collisioni vengono sprecati degli slot. Un nodo è in grado di accorgersi subito della collisione ma deve aspettare un nuovo slot per trasmettere.

### Efficienza

Un indicatore dell'efficienza è la proporzione di slot che hanno una comunicazione con successo rispetto a tutti gli slot disponibili su un intervallo di tempo lungo.

Supponiamo che  $N$  nodi vogliono continuare a trasmettere dei frame ed ognuno trasmette con probabilità  $p$ .

- La probabilità che un dato nodo abbia successo nella trasmissione è  $= p(1 - p)^{N-1}$
- La probabilità che un qualsiasi nodo abbia successo è  $= Np(1 - p)^{N-1}$
- Per la massima efficienza devo trovare  $p^*$  che massimizzi :  $Np(1 - p)^{N-1}$
- Per molti nodi prendo il limite di  $Np^*(1 - p^*)^{N-1}$  quando  $N$  tende a infinito.
- Efficienza massima  $= \frac{1}{e} = 0.37$

Nella migliore delle ipotesi utilizzerò il canale il 37% del tempo.

**CSMA (Carrier Sense Multiple Access)** Ascolto prima di trasmettere (*carrier sense*), se il canale è libero allora trasmetto, altrimenti aspetto.

#### Esempio

Possono capitare delle collisioni anche ascoltando prima di trasmettere sul canale perché il segnale non si propaga immediatamente ovunque (*propagation delay*).

Foto/CSMA.png

**CSMA/CD** Protocollo utilizzato in ethernet. CSMA con il rilevamento della collisione (*collision detection*). Mentre trasmetto riesco a rilevare che è presente una collisione, appena riesco a farlo interrompo la trasmissione per ridurre lo spreco del canale. Facile da implementare su canali via cavo, difficile in wireless.

#### Esempio

Per velocizzare il fatto che tutti rilevino la collisione, appena qualcuno la rileva manda un segnale agli altri nodi intorno in modo tale che tutti rilevino la collisione.

Foto/CSMACD.png



### Algoritmo di Ethernet

1. Ethernet riceve il datagramma dal livello rete sovrastante e crea il frame;
2. La scheda di rete ascolta il canale:
  - se è libero: inizia la trasmissione del frame;
  - se è occupato: aspetta finché il canale non si libera e poi trasmette.
3. Se sono riuscito a trasmettere l'intero frame senza collisioni ho finito;
4. Se viene rilevata un'altra trasmissione mentre invio, smetto di trasmettere e mando un segnale di *abort* (*jamming signal*) agli altri nodi;
5. Dopo aver abortito devo ritrasmettere e utilizzo un algoritmo di **binary (exponential) backoff**:
  - dopo l' $m$ -esima collisione scelto un numero casuale  $K$  nell'intervallo  $\{0, 1, 2, \dots, 2^{m-1}\}$ . Ethernet aspetta  $k \cdot 512$  bit times (tempo necessario per trasmettere un bit) e poi ritorno al punto 2.
  - Più collisioni ci sono maggiore sarà l'attesa per ritrasmettere.

#### 1.3.3 Protocolli a turni

Ogni nodo può trasmettere in maniera dedicata ma solo in determinati turni.

**Polling** Controllore centralizzato, ciclicamente richiede ad ogni nodo se ha bisogno di trasmettere se si dedica un intervallo di tempo per trasmettere altrimenti passa al successivo.

I contro di questo protocollo sono: l'overhead del polling (pochi byte), latenza perché devo aspettare che il controllore faccia il giro e il single point of failure del controllore.

Questo protocollo è utilizzato in Bluetooth.

**Token-passing** Invece di avere un master ho un *token*. I nodi sono disposti ad anello, in modo ciclico. Il token passa al successivo nodo che conosce il suo successore. Chi ha il token può trasmettere per un intervallo di tempo fisso, costante e limitato.

I contro di questo protocollo sono: l'overhead del token, latenza per l'attesa del token e single point of failure (ciascun nodo quando ha il nodo diventa un punto di fallimento).

## 1.4 Switched Local Area Networks

Gli switch utilizzano indirizzi propri al livello collegamento per inoltrare i frame.

### 1.4.1 Indirizzo MAC

**Indirizzi IP** Indirizzi definiti a livello di rete associati alla singola interfaccia da 32 bit, utilizzati per fare il forwarding dei pacchetti.

**MAC** Le interfacce di rete degli host e dei router hanno indirizzi livello collegamento, ma non gli switch. Un indirizzo livello collegamento viene chiamato indirizzo MAC. Gli indirizzi MAC sono di solito lunghi 6 bytes (**48 bit**) e sono espressi in notazione esadecimale. Ogni interfaccia nella LAN ha un **unico indirizzo MAC** e un (localmente) unico indirizzo IP. Ogni interfaccia possiede un unico indirizzo MAC

perché questi sono controllati dalla IEEE (tipicamente si associano i primi 24 bit più significativi sono associati ad uno specifico produttore).

Quando un'interfaccia vuole inviare un frame, inserisce l'indirizzo MAC di destinazione nel frame e lo inoltra sulla LAN. Se un mittente vuole inviare un frame a tutte le interfacce presenti sulla LAN, questo inserisce un indirizzo speciale, l'indirizzo broadcast nel frame.

## 1.5 Protocollo ARP (Address Resolution Protocol)

Il protocollo ARP, tipicamente implementato su tutti gli host, serve per fare il mapping tra l'indirizzo IP e l'indirizzo MAC corrispondente all'interno di una sotto-rete.

Ciascun nodo ha una **tabella ARP** che contiene:

- Corrispondenza tra indirizzo IP e indirizzo MAC;
- TTL (Time To Live) = intervallo di tempo dopo il quale il mapping non vale più.

### Esempio protocollo ARP nella stessa LAN

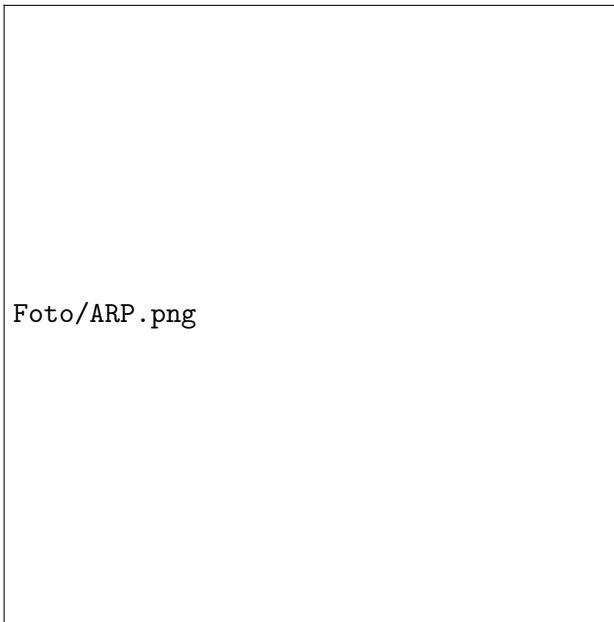
Supponendo che *A* voglia inviare un datagramma a *B* e l'indirizzo MAC di *B* non è presente nella tabella ARP di *A*.

*A* fa un **broadcast ARP**, sull'intera rete fisica attraverso l'indirizzo MAC *FF-FF-FF-FF-FF-FF*, in cui specifica l'indirizzo IP di *B* e l'indirizzo MAC sorgente. Tutti i nodi ricevono questa richiesta e la processano, solo *B* replicherà in modo diretto, unicast, specificando il suo indirizzo MAC associato al suo indirizzo IP.

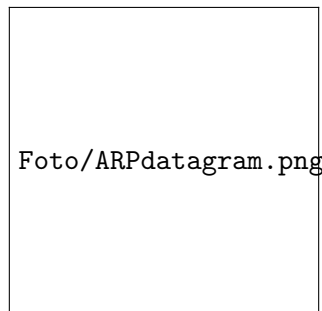
*A* salva l'informazione nella sua tabella ARP. Questa entry può essere cancellata se finisce il suo TTL oppure rinnovata se viene rinoltrata un'altra richiesta con la stessa associazione IP-ARP.

## Esempio protocollo ARP invio in sotto-reti diverse

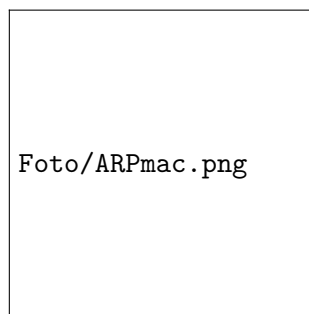
Supponiamo che  $A$  debba inviare un datagramma a  $B$ . Assumiamo che  $A$  conosca l'indirizzo IP di  $B$  e che  $A$  conosca l'indirizzo IP del primo hop router (attraverso il DHCP) e che conosca anche l'indirizzo MAC dell'interfaccia del router.



- $A$  crea il datagramma IP con sorgente l'IP di  $A$  e destinazione l'IP di  $B$ ;



- $A$  incapsula il datagramma in un frame con indirizzo MAC sorgente quello di  $A$  e **indirizzo di destinazione l'indirizzo MAC dell'interfaccia del router**;



- il frame viene inviato da  $A$  al router;
- il router processa il pacchetto, rimuove il datagramma e lo passa a livello IP;
- il router passa all'altra interfaccia il datagramma con IP sorgente e destinazione;
- il router crea un nuovo frame, gli indirizzi IP rimangono gli stessi ( $A$  e  $B$ ) invece gli indirizzi

## 1.6 Ethernet

La LAN Ethernet originariamente utilizzava un bus coassiale per connettere i nodi, e i frame erano trasmessi in broadcast. Successivamente, il bus è stato rimpiazzato con uno switch, che a differenza dei router opera esclusivamente sul livello collegamento.

### Struttura di un frame ethernet

Preambolo	Indirizzo destinazione	Indirizzo sorgente	Tipo	Campo dati	CRC
-----------	------------------------	--------------------	------	------------	-----

- **Preambolo** = utilizzato per sincronizzare mittente e destinatario e per regolare la velocità con cui invierò i bit. È formato da 7 byte con pattern *10101010* seguiti da 1 byte con il pattern *10101011*. Serve anche per capire che dopo questo pattern iniziano le informazioni;
- **Indirizzo destinazione** = indirizzo MAC del destinatario, 6 byte;
- **Indirizzo sorgente** = indirizzo MAC del mittente, 6 byte;
- **Tipo** = indica qual è il protocollo di livello superiore del payload (tipicamente IP), 4 bit;
- **Campo dati** = contiene il datagramma IP, massimo di 1500 byte, minimo 46 byte;
- **CRC** = per verificare la correttezza del frame stesso.

In ethernet non c'è handshake tra mittente e destinatario quindi è un protocollo **connectionless**.

In caso di errore ethernet lo rileva e scarta il frame quindi è un protocollo **unreliable**.

In ethernet il protocollo MAC che si usa è **CSMA/CD con binary backoff**.

**Ethernet 802.3** È uno standard che definisce sia aspetti di livello data-link che di livello fisico.

## 1.7 Switch ethernet

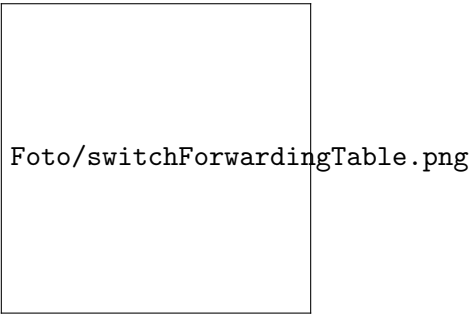
Gli switch ethernet sono dei dispositivi a livello collegamento, hanno un ruolo attivo perché memorizzano e inoltrano i frame ethernet e non solo, riescono ad esaminare gli indirizzi MAC e decidono in modo selettivo se inoltrare il frame e a quale interfaccia (porta), viene inoltrato su un determinato segmento sul quale si usa CSMA/CD per accedervi.

Gli switch sono dispositivi **trasparenti** agli host e utilizzano soluzioni di tipo **plug and play** (nessuna necessità di configurazione) e **self learning**.

Questo dispositivo permette la trasmissione simultanea di più frame, in particolare ogni host ha una connessione diretta e dedicata con lo switch (tipicamente full-duplex), inoltre bufferizzano i pacchetti (no collisioni) e si usa il protocollo ethernet su ogni singola connessione.

## Switch forwarding table

Come fanno gli switch a sapere che  $A'$  è raggiungibile attraverso l'interfaccia 5?



Ogni switch ha una tabella di inoltra che contiene:

- Indirizzo MAC;
- interfaccia per raggiungere l'host;
- time stamp = time to live.

Le entry di questa tabella sono riempite in modo diverso rispetto alle tabelle di routing. Lo switch impara la corrispondenza tra indirizzo MAC e porta a cui inoltrare il frame in modo dinamico.

**Self learning** Ipotizziamo che  $A$  voglia inoltrare un frame ad  $A'$ .  $A$  invia il frame sul cavo fino a raggiungere lo switch, in questo momento memorizza che l'host  $A$  è arrivato dalla porta 1 e lo aggiunge alla sua tabella.

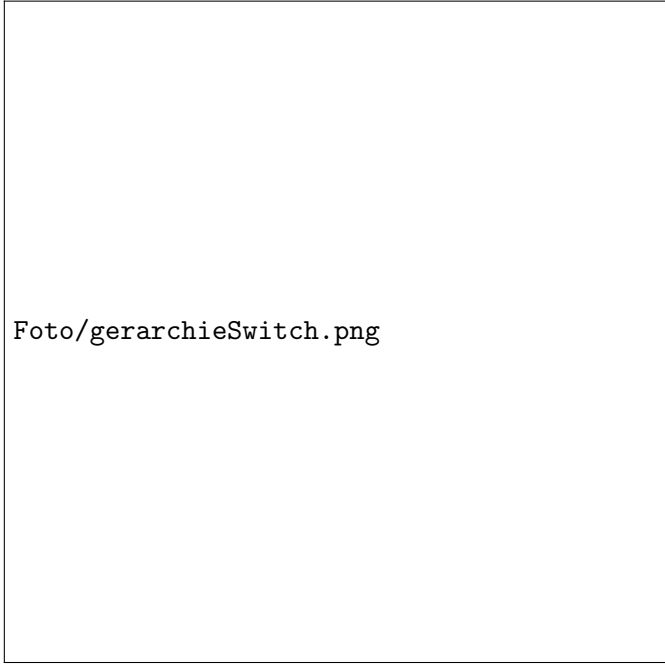
**Frame filtering/forwarding** Il **filtering** è la funzionalità degli switch che determina se un frame deve essere inoltrato o scartato. Il **forwarding** è la funzionalità che determina le interfacce alle quali il frame va inoltrato.

Quando il frame raggiunge lo switch:

1. memorizza l'indirizzo MAC dell'host mittente;
2. indicizza la tabella dello switch utilizzando l'indirizzo MAC di destinazione;
3. se viene trovata la entry del destinatario
  - allora:
    - se la destinazione è la stessa del segmento da cui il frame è arrivato lo scarta;
    - altrimenti lo inoltra all'interfaccia del destinatario.
  - altrimenti:
    - fa il **flooding**, cioè inoltra il frame su tutte le interfacce eccetto quella del mittente.

### 1.7.1 Interconnessione tra gli switch

La tipologia a stella non è sufficiente per LAN di dimensioni importanti, allora gli switch possono essere connessi tra loro in modo gerarchico. Durante il flooding l'informazione si propaga fino a raggiungere tutti gli host.



Foto/gerarchieSwitch.png

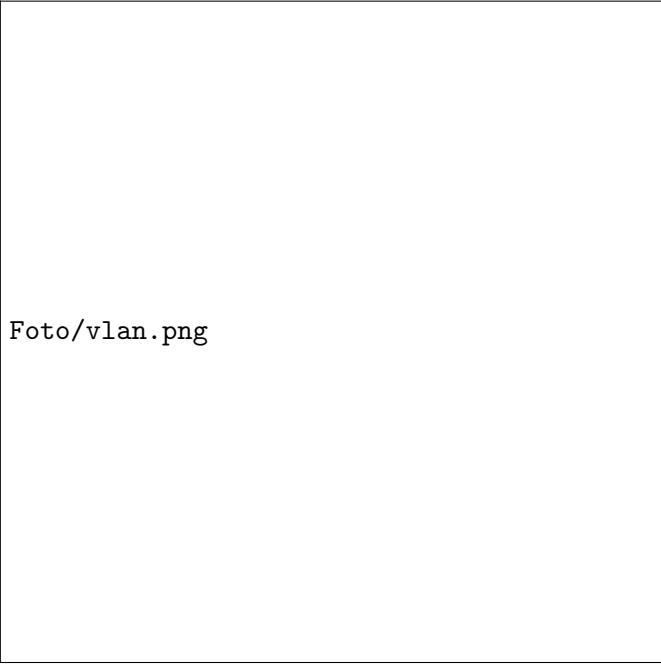
## 1.8 Switch vs Router

- Entrambi hanno un meccanismo di store-and-forward;
  - *router* = dispositivi di livello rete (esaminano solo le intestazioni a livello di rete);
  - *switch* = dispositivi di livello collegamento (esaminano solo le intestazioni a livello 2).
- Entrambi hanno delle tabelle di inoltro;
  - *router* = calcolata attraverso gli algoritmi di routing e basandosi sugli indirizzi IP;
  - *switch* = calcolata attraverso il self-learning utilizzando il flooding e imparando gli indirizzi MAC man mano.

## 1.9 VLAN (Virtual LAN)

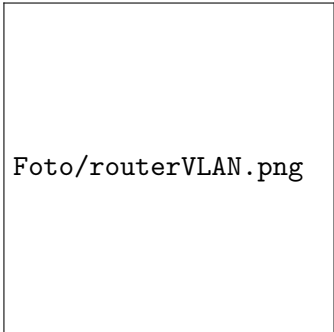
Le reti LAN tradizionali non sono facilmente scalabili o portabili, ma le VLAN sì. Uno switch che supporta VLAN permette di definire LAN virtuali su una singola LAN fisica. Gli host su una VLAN possono comunicare tra loro come se fossero solo loro connessi a quello switch.

### VLAN basata sulle porte



Foto/vlan.png

Lo switch può essere configurato per fare in modo che le porte vengano etichettate e dedicate ad una particolare VLAN (solitamente caratterizzate da un colore), le porte in ogni VLAN formano un dominio di broadcast. Per far comunicare dei dispositivi di VLAN diverse devo utilizzare un router esterno.

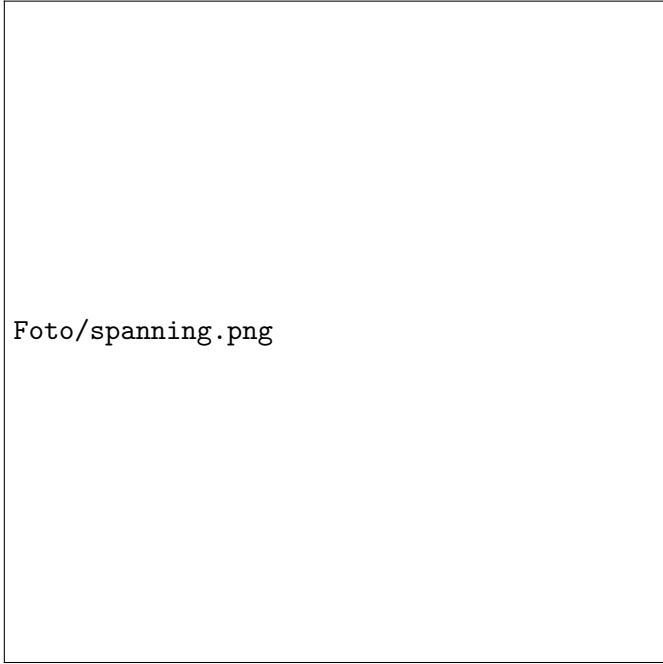


Foto/routerVLAN.png

**Vantaggi** Posso **configurare in modo dinamico** lo switch per, ad esempio, cambiare la VLAN di un dispositivo senza doverlo spostare fisicamente. Si può anche definire la VLAN basandosi sugli indirizzi MAC degli endpoint piuttosto che per le porte.

### 1.10 VLAN spanning (trunk port)

In presenza di più switch distinti posso fare in modo che host dello switch secondario possano far parte delle VLAN del principale.



Foto/spanning.png

Utilizzo una particolare **porta trunk** dello switch che permette di interconnettere LAN differenti. (Eg. se si manda un messaggio in broadcast sulla VLAN 1 viene inoltrato anche sulla porta 16, connessa fisicamente all'altro switch, e arriveranno alla porta 1 del secondo switch. Sulla porta trunk passano i frame di una qualunque VLAN, allora per identificare a quale VLAN appartiene il frame si è stabilito un nuovo **protocollo 802.1q** che aggiunge un campo aggiuntivo nell'header del protocollo ethernet, fatto solo dagli switch. Quando uno switch riceve il frame riconosce a quale VLAN deve essere indirizzato il frame e lo inoltra eliminando i dati aggiuntivi e tornando ad un protocollo normale.

Pream.	Dest.	Sorg.	Tipo di protocollo	Informazioni di controllo	Tipo	Dati	CRC
--------	-------	-------	--------------------	---------------------------	------	------	-----

- **Tipo di protocollo** = 2 byte (valore 81-00);
- **Informazioni di controllo** = 12 bit per identificare la VLAN e 3 bit di priorità.

## 1.11 Networking nei datacenter

### 1.11.1 Architetture dei data center

I data center non sono solamente connessi a Internet, ma a dei loro network interni che connettono gli host tra loro. Gli host nei data center sono chiamati **blades**, impilati in rack. In cima ad ogni rack c'è uno switch, detto **Top of Rack (TOR)**, che connette gli host nel rack tra di loro e con gli altri switch nel data center. Ogni host nel rack ha un'interfaccia di rete che si connette al proprio TOR, e ogni TOR ha delle porte che possono essere connesse ad altri switch.

I data center supportano due tipi di traffico: il **traffico tra client esterni e host interni** e **traffico tra host interni**. Per gestire il primo, i data center utilizzano dei **border router**, che connettono il data center con Internet.

### 1.11.2 Load balancing

Le richieste ricevute da un data center sono innanzitutto dirette a un load balancer che si occupa di distribuire le richieste agli host. I grossi data center possiedono molti load balancer, ognuno dedicato a specifiche applicazioni cloud. Un tale load balancer è spesso chiamato "layer-4 switch" in quanto basa



le sue decisioni sul numero di porta e sull'indirizzo IP del pacchetto. Al ricevimento di una richiesta, il load balancer la inoltra a uno degli host che gestisce l'applicazione. Il load balancer funziona anche come NAT, in quanto traduce l'IP pubblico della richiesta nell'IP interno e viceversa.

## 1.12 Il processamento di una richiesta



Foto/journeyTopology.png

### 1.12.1 Connessione ad Internet

Si connette e come prima cosa deve sapere qual è il suo indirizzo IP e l'indirizzo IP del first hop router oltre all'indirizzo del DNS server, per ottenere tutte queste informazioni utilizza il **DHCP**.

Fa una richiesta DHCP che viene incapsulata in UDP a sua volta incapsulata in IP e in fine incapsulata in 802.3 Ethernet. Questa richiesta viene inoltrata in broadcast.

DHCP risponde con un ACK specificando l'indirizzo IP del client e le altre informazioni richieste, incapsula tutte queste informazioni e lo inoltra in unicast al nodo.

Il client ora ha un indirizzo IP, sa il nome e l'indirizzo del server DNS e conosce l'indirizzo IP del first-hop router.



Foto/ConnessioneAInternet.png

### 1.12.2 Richiesta HTTP

Prima di mandare la richiesta HTTP necessita dell'indirizzo IP di *www.google.com* al DNS.

Viene creata una query DNS, incapsulata in UDP a sua volta incapsulata in IP e in fine incapsulata in ethernet. Per mandare il frame al router necessito dell'indirizzo MAC del router quindi utilizzo il protocollo ARP.

ARP fa una richiesta in broadcast, ricevuta dal router che risponde con l'indirizzo MAC dell'interfaccia.

Ora il client è a conoscenza dell'indirizzo MAC del router e posso fare la richiesta al DNS.




Il datagramma IP contenente la query DNS viene inoltrata, attraverso lo switch LAN, dal client al first hop router.

Poi il datagramma IP viene inoltrato dal *campus network* fino al *comcast network* fino al server DNS.

Il server DNS risponde con l'indirizzo IP di *www.google.com*.



Per la richiesta HTTP devo aprire una connessione TCP e fare il 3-way handshake.




Foto/TCP.png

La richiesta HTTP viene inoltrata attraverso il socket TCP.

Il datagramma IP contenente la richiesta HTTP viene inoltrato fino a *www.google.com*.

Il web server risponde con una HTTP reply (contenente la pagina web).

Il datagramma IP contenente la risposta HTTP viene inoltrato fino a raggiungere il client.



Foto/fine.png

## 2 Livello fisico

### 2.1 Segnali a larghezza di banda limitata

Qualunque forma di segnale, comprese quelle discrete (digitali con 1 e 0), può essere scomposto attraverso la serie di Fourier.

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

La scomposizione ci permette di rappresentare in modo diverso un segnale come una somma, potenzialmente illimitata (comunque finita nel caso della realtà), di approssimazioni di armoniche con una certa ampiezza (potenza). All'aumentare delle armoniche riesco ad approssimare meglio il segnale.

Foto/armoniche.png

**Larghezza di banda** La banda analogica di un mezzo trasmissivo è caratteristica del mezzo stesso ed è l'intervallo di frequenze che è in grado di trasmettere senza degradare, oltre ad un certo limite, il segnale stesso.

I segnali che vanno da 0 ad una frequenza massima sono detti **baseband**, quelli che sono spostati per occupare un range di frequenze più alto sono detti **passband**.

### 2.1.1 Data rate massimo per un canale

La velocità massima di trasmissione è limitata dalla larghezza di banda, indipendentemente da come sono codificati.

Utilizziamo l'equazione di Nyquist per derivare il data rate massimo per un canale senza rumore con banda finita (canale ideale):

$$\text{maximum data rate} = 2B \log_2 V \left[ \frac{\text{bit}}{\text{sec}} \right]$$

Se un segnale viene fatto passare attraverso un low-pass filter di banda  $B$ , il segnale filtrato può essere ricostruito con  $2B$  campioni al secondo.

In presenza di rumore il segnale degrada rapidamente.

Utilizziamo l'equazione di Shannon per derivare il data rate massimo per un canale in presenza di rumore:

$$\text{maximum number of bits/sec} = B \log_2 \left( 1 + \frac{S}{N} \right)$$

#### Definizioni

$V$  = numero di livelli discreti differenti.

$\frac{S}{N}$  = rapporto segnale-rumore (SNR) [decibel]

## 2.2 Modulazione digitale

Il processo di convertire i bit in segnali è detto modulazione digitale.

### 2.2.1 Trasmissione baseband

**NRZ** La forma di modulazione più diretta è quella di utilizzare una tensione positiva per rappresentare 1 e una negativa per lo 0, questo schema è detto **NRZ** (Non-Return-to-Zero). Per decodificare i bit, il destinatario mappa i segnali ai simboli più coerenti.

Con NRZ, il segnale si alterna tra due livelli, quindi è necessaria una banda di almeno  $\frac{B}{2}$  se il bit rate è  $B$  bit al secondo. Utilizzando quattro tensioni diverse, si possono inviare 2 bit alla volta come singolo simbolo.

**Manchester** È necessario un clock accurato per distinguere i bit (eg. molti 0 di fila). Una strategia è inviare segnale di clock separato al destinatario, mettendo in XOR il segnale di clock e quello dei dati. Una transizione da alta e bassa tensione simbolizza uno 0, da bassa ad alta un 1, questo schema è detto Manchester. Necessita di banda doppia rispetto a NRZ.

**NRZI** Si può codificare 1 come transizione e 0 come non transizione utilizzando NRZI.

**Mapping a blocchi** Una particolare sequenza di bit di dati da trasmettere, vengono trasmessi con un segnale digitale con un numero di bit maggiore in modo da mantenere un numero sufficiente di transizioni che permettono al ricevente di sincronizzarsi.

Ogni 4 bit sono mappati a una sequenza di 5 bit con una tabella di traduzione fissa. Aggiunge un overhead del 25%.

**Scrambling** Metto in XOR i dati e una sequenza pseudorandom.

**Segnali bilanciati** Segnali che hanno sia tanta tensione positiva che negativa sono detti bilanciati. Non hanno componente DC.

Un modo per costruire un codice bilanciato è utilizzare due tensioni diverse per rappresentare 1, mentre 0 è rappresentato da 0 volt, **bipolar encoding**.

Foto/baseband.png

### **2.2.2 Trasmissione passband**

Codifica di dati digitali (sequenza di bit) tramite un segnale analogico (forme d'onda).

## Onde

Foto/sin.png

$$B = \pi, D = 2\pi.$$

$$A \sin(2\pi ft + \phi)$$

$A$  = ampiezza.

$f$  = frequenza (inversa del periodo, tempo necessario per fare un ciclo), aumentando la frequenza riesco a trasmettere più forme d'onda.

$\phi$  = fase, stabilisce a che punto parte la forma d'onda.

L'idea è quella di codificare i bit in base alla modulazione dei parametri soprastanti.

Il segnale baseband è aumentato per occupare una banda da  $S$  a  $S + B$  Hz senza cambiare la quantità di dati trasmessi.

### ASK (Amplitude shift keying)

Foto/ASK.png

Per 0 associo un ampiezza  $A = 0$ , invece per 1 associo una certa ampiezza concordata tra mittente e ricevente (seguendo uno standard).

## FSK (Frequency shift keying)



Utilizzo due o più frequenze differenti. Ho una frequenza  $f$  di base che uso per lo 0 e una frequenza  $2f$  che uso per l'1.

## BPSK (Phase shift keying)

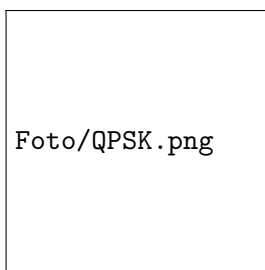


Cambiando la fase cambio quando inizia la forma d'onda.

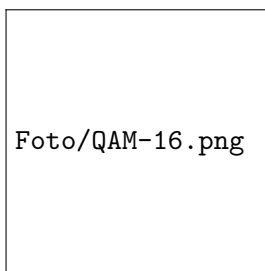


**Soluzioni ibride** Se aggiungo tante diverse configurazioni della forma d'onda che trasmetto il ricevente farà fatica a distinguerle. Possiamo utilizzare delle soluzioni dove modifichiamo più parametri della forma d'onda.

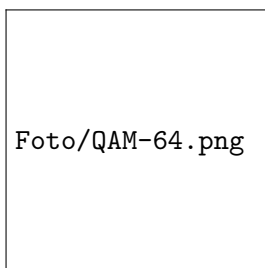
**QPSK (Quadrature Phase Shift Keying)** Utilizza quattro fasi distinte ( $0^\circ$ ,  $90^\circ$ ,  $180^\circ$ , e  $270^\circ$ ) per rappresentare i dati. Ogni simbolo trasporta due bit, perché ci sono 4 combinazioni possibili (00, 01, 10, 11).



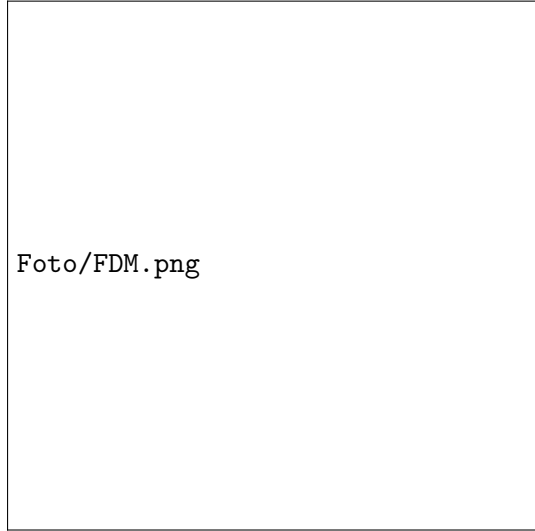
**QAM-16 (16 Quadrature Amplitude Modulation)** Una combinazione di modulazione di fase e di ampiezza. QAM-16 utilizza 16 punti distinti su una costellazione per rappresentare i dati. Ogni simbolo rappresenta 4 bit, dato che  $2^4 = 16$ .



**QAM-64 (64 Quadrature Amplitude Modulation)** Simile a QAM-16, ma con una costellazione più grande composta da 64 punti distinti. Ogni simbolo rappresenta 6 bit, dato che  $2^6 = 64$ .



**FDM (Frequency Division Multiplexing)** Se il mezzo trasmissivo permette di trasmettere, ad una determinata velocità, un certo numero di frequenze, posso usarlo per trasmettere i dati di più sorgenti. Partiziono lo spettro in bande di frequenza, separate da un eccesso detto band guard per evitare sovrapposizioni, e assegno una parte delle armoniche a diverse sorgenti che trasmettono usando solo queste componenti. Ci sono comunque delle sovrapposizioni tra canali.



Foto/FDM.png

**OFDM (Orthogonal Frequency Division Multiplexing)** In OFDM, la banda del canale è suddivisa in molti subcarrier che inviano dati indipendentemente. I subcarrier sono campionati nella loro frequenza centrale senza interferenza.

### 3 Reti wireless e mobili

#### 3.1 Elementi di una rete wireless

Nelle reti wireless si possono identificare i seguenti elementi:

- **Host wireless** = laptop, smartphone, dispositivi IoT;
  - Stazionari: laptop che si collega in università al wifi poi si ricollega a casa ad un'altra rete wifi;
  - Mobili: smartphone connesso alla rete mobile che si muove.
- **Base station** = tipicamente connesse alla rete cablata (eg. access point, antenna operatore nel caso di reti mobili). Fanno da intermediario tra il dispositivo mobile e la rete via cavo;
- **Collegamenti wireless** = un host si connette a una base station o un altro host wireless tramite un collegamento wireless.

**Modalità con infrastruttura** Modalità in cui è presente la base station e dei dispositivi che si associano alla rete in modo wireless. Un nodo che è coperto da due base station che cambia base station è detto **handoff**. Se un host fa una richiesta attraverso una base station e poi cambia l'infrastruttura deve sapere che il nodo si è spostato e deve inoltrare la risposta ad un'altra base station.

**Modalità Ad Hoc** Non è presente la base station, i nodi comunicano tra di loro. Se non vi è collegamento diretto tra gli host l'informazione dovrà viaggiare su altri nodi (passa parola) fino ad arrivare alla destinazione. La copertura cambia dinamicamente. In questa modalità potrebbe esserci un nodo che è connesso alla rete cablata che farà da tramite per tutti gli altri (eg. hotspot del cellulare).

Foto/AdHoc.png

	Single hop	Multiple hop
Con infrastruttura	Si connettono alla base station con una <b>singola connessione diretta</b> tra host nodo e base station.	Per raggiungere la base station, se non sono coperto, <b>uso altri nodi come tramiti</b> . Rete di tipo meshed.
Senza infrastruttura	Non è presente una base station, non mi collego per forza ad internet. (Eg. Bluetooth)	Nessuna base station, nessuna connessione a una rete internet più grande. Potrebbe essere necessario effettuare un inoltro per raggiungere un determinato nodo wireless. MANET, VANET.

#### 3.2 Caratteristiche collegamenti wireless

Differenze importanti dei collegamenti wireless rispetto ai collegamenti cablati:

- **Decremento della potenza di segnale** = il segnale radio si attenua man mano che si propaga nella materia (path loss);

- **Interferenza dovuta ad altre sorgenti** = se più sorgenti trasmettono negli stessi intervalli di frequenza o in intervalli di frequenza sovrapponibili, la componente dell'altra sorgente interferisce con il segnale originale, aggiungendo un'armonica che si sovrappone e cambia il segnale (anche un motore potrebbe generare interferenza);
- **Multipath propagation** = il segnale radio si riflette, quindi fa un percorso diverso rispetto ad uno in linea d'area, questo potrebbe generare una differenza nel tempo di propagazione.


**SNR (Signal-to-Noise-Ratio)** L'SNR è la misura relativa della potenza del segnale ricevuto e dal suo rumore, viene misurato in decibel. Maggiore è l'SNR più facile sarà estrarre il segnale dal rumore di sottofondo. C'è un bilanciamento tra la velocità di trasmissione, l'SNR e la probabilità che ci siano degli errori nella trasmissione (BER bit error rate). Aumentando la potenza del segnale posso incrementare l'SNR e di conseguenza decrementare il BER.

Dato un SNR scelgo un layer fisico che soddisfi il BER richiesto, ciò garantisce la massima velocità di trasmissione.

Cambiando la codifica posso decrementare la velocità ma ciò garantisce che il BER stia sotto una soglia che garantisce la comunicazione.


**Problema del terminale nascosto** Può succedere che più nodi trasmettendo dati ad uno stesso nodo ricevente, ciò causerebbe delle collisioni rendendo impossibile decifrare le informazioni al ricevente. In 802.3 una volta che viene rilevata una collisione tutti i nodi che stanno trasmettendo si fermano.

Nel caso delle trasmissioni wireless se ci sono degli ostacoli di mezzo i due mittenti non si sentono che stanno trasmettendo quindi il destinatario riceve dei dati che dovrà buttare via.



Foto/HiddenTerminal.png

Lo stesso problema avviene anche se non è presente un ostacolo ma se i nodi sono ad una distanza sufficiente, perché il decadimento della potenza del segnale avviene dopo una certa soglia.



Foto/SignalAttenuation.png

Quindi non si può usare 802.3 perché non riesco a fare la collision detection.

### 3.2.1 CDMA (Code Division Multiple Access)

Si assegna ad ogni nodo un codice, una sequenza di bit con una lunghezza fissa, che viene utilizzato da ogni nodo per trasmettere, tutti gli utenti comunicano sulla stessa frequenza ma ogni utente ha una propria chipping sequence (codice) per codificare i dati. Permette a tutti gli utenti che hanno codici diversi di trasmettere simultaneamente, ad un determinato ricevente, consentendo al ricevente di decifrare tutti i dati, solo se i codici sono scelti in modo opportuno.

**Codifica** Prodotto riga per colonna tra i dati originali e la chipping sequence.

**Decodifica** Somma del prodotto tra i dati che arrivano (codificati) e la chipping sequence.

## Esempio con 1 mittente

Foto/CDMSEg.png

Il mittente deve trasmettere il bit 1, per trasmetterlo trasmette una sequenza di bit (composta da 1 o -1), multiplico la sequenza per la chipping sequence.

$$Z_{i,m} = d_i \cdot c_m$$

Il ricevente moltiplica la sequenza, bit per bit, con la chipping sequence del mittente. Sommo il numero di 1 e divido per la lunghezza della chipping sequence  $M$  ed ottengo il dato iniziale.

$$D_i = \frac{\sum_{m=1}^M Z_{i,m} \cdot c_m}{M}$$

## Esempio con più mittenti

Foto/CDMA2.png

Con un mittente singolo il vantaggio non è percettibile. In presenza di più mittenti ognuno di essi ha una propria chipping sequence differente. Il ricevente riceve la somma dei bit trasmessi dai vari mittenti. Il ricevente decide quale mittente ascoltare e moltiplica quello che ha ricevuto per la chipping sequence del mittente desiderato ed ottengo il bit che era stato inviato.

**Chipping sequence** L'insieme delle chipping sequence deve soddisfare una certa proprietà. Tutte le coppie di chip sequence devono essere **ortogonali** tra loro. Moltiplicate le chip sequence tra loro devono fare 0.

$$S \cdot T \equiv \frac{1}{m} \sum_{i=1}^m S_i T_i = 0 \quad \text{Per tutti S,T}$$

Il prodotto di una chip sequence per se stessa è sempre = 1.

$$S \cdot S = 1$$

### Esempio

Assumiamo che A, B e C inviino i seguenti bit 1, 0, 1 rispettivamente, avremo:

$$S = A + \overline{B} + C$$

Alla ricezione la decodifica di C sarà:

$$S \cdot C = (A + \overline{B} + C) \cdot C = A \cdot C + \overline{B} \cdot C + C \cdot C = 0 + 0 + 1 = 1$$

## 3.3 Reti WLAN 802.11

Lo standard delle trasmissioni wireless è 802.11, anche detto WiFi. Tutti i protocolli 802.11 usano lo stesso protocollo di accesso al mezzo (a livello data link), CSMA/CA, e tutte hanno modalità con infrastruttura o ad hoc.

### 3.3.1 Architettura

Gli host comunicano attraverso una base station, chiamata access point (AP) di livello data link. Il blocco fondamentale dell'architettura 802.11 è il **BSS (Basic Service Set)** composto da tutti i nodi che si trovano all'interno della copertura dell'AP.

### 3.3.2 Canali e associazioni

Quando un nodo vuole connettersi alla rete deve **associarsi** ad uno specifico AP. Lo spettro è diviso in diversi intervalli di frequenza chiamati **canali**. L'AP sceglie un certo intervallo di frequenze su cui trasmettere, per cui il nodo per poter trasmettere dovrà usare lo stesso intervallo dell'AP. È possibile che ci siano interferenze se più AP utilizzano lo stesso canale.

Gli host che vogliono associarsi all'AP iniziano a fare uno scanning su tutti gli intervalli di frequenza previsti dal protocollo in ricerca dei **beacon frame** (frame inviati periodicamente dall'AP per identificare la propria rete identificata da un SSID (Service Set Identifier) e l'indirizzo MAC). Il nodo sceglie uno degli AP da cui è coperto, in questo momento potrebbe autenticarsi, se necessario. L'autenticazione può avvenire tramite indirizzo MAC o username e password, in entrambi i casi l'AP comunica con un server di autenticazione. In fine, tipicamente, utilizza il DHCP per ottenere un indirizzo IP nella subnet dell'AP.

**Scanning passivo** Il nodo rileva i segnali che vengono inviati dall'AP.

**Scanning attivo** Il nodo fa una richiesta in broadcast per rilevare quali AP sono disponibili. Gli AP inviano la risposta e in fine l'host sceglie con chi associarsi.

**Accesso multiplo al canale** La parte CSMA è la stessa di 802.3, quindi ascolto prima di trasmettere, se non rilevo nulla inizio a trasmettere. In 802.11 non c'è collision detection, quindi cerco di evitarle attraverso **CSMA/CA (Collision Avoidance)**.



## Protocollo MAC CSMA/CA

### Mittente

1. Il sender ascolta il canale, che deve essere in idle, per un tempo chiamato **DIFS**, se per questo tempo nessuno trasmette invia l'intero frame (perché tanto non sarei in grado di rilevare la collisione).
2. Se il canale è occupato faccio il backoff, come in 802.3, aspetto un tempo casuale, a seconda del numero collisioni aumento l'intervallo dei valori da cui estrarre questo tempo casuale. Se nel frattempo qualcuno trasmette sul canale fermo il timer, quando ritorna in idle riprendo il timer.

Foto/CSMA\_CA.png

### Ricevente

- Se il frame è arrivato correttamente, senza collisioni, aspetta un tempo **SIFS** prima di rispondere con un ACK al mittente, per evitare il problema del terminale nascosto.

**Spaziatura interframe** La lunghezza dei tempi di SIFS e DIFS è fatta in modo per cercare di permettere ai nodi, che hanno iniziato una comunicazione, di proseguirla e non essere interrotti da altri.

Foto/Spacing.png

### 3.3.3 Soluzione alternativa (RTS-CTS)

Esiste un altro modello di trasmissione supportato da 802.11 che si basa sulla prenotazione del canale.

Il sender fa una richiesta all'AP di avere disponibile il canale, in esclusiva, per un certo intervallo di tempo.

- Il sender prima trasmette un piccolo pacchetto RTS (Request TO Send) all'AP usando CSMA;
  - Un pacchetto RTS potrebbe comunque collidere, ma sono molto corti.
- L'AP manda un messaggio in broadcast di risposta CTS (Clear To Send),
- Il CTS viene ascoltato da tutti i nodi;
  - il sender trasmette il frame;
  - le altre stazioni posticipano le trasmissioni.

Questa modalità viene poco usata, perché lenta e non ci sono troppi frame così grandi per giustificarla. Tipicamente si configurava l'AP in modo che attivasse questa modalità solo per pacchetti di dimensioni importanti altrimenti usavano la modalità normale CSMA/CA.

### 3.3.4 Frame 802.11

2	2	6	6	6	2	6	0 - 2312	4
frame control	duration	address 1	address 2	address 3	seq control	address 4	payload	CRC

- **frame control** = preambolo, descrive il tipo di frame;

2	2	4	1	1	1	1	1	1	1
protocol version	type	subtype	to AP	from AP	more frag	retry	more data	WEP	rsvd

- **to AP** = l'indirizzo MAC destinatario è l'AP;
  - **from AP** = il mittente è l'AP.
  - **more frag** = per gestire la frammentazioni;
  - **retry** = dice se è un tentativo di ritrasmissione oppure no;
  - **power mgt** = gestione della potenza del segnale;
  - **WEP** = modalità di autenticazione;
  - **rsvd** = dati per la prenotazione del canale.
- **duration** = durata, nel caso di modalità con prenotazione;
  - **address 1** = indirizzo MAC del destinatario (nodo o AP);
  - **address 2** = indirizzo MAC del mittente (nodo o AP);
  - **address 3** = indirizzo MAC dell'interfaccia del router a cui è collegato l'AP;
  - **seq control** = sequential control necessario per la presenza dell'ACK;
  - **address 4** = indirizz MAC utilizzato solo in modalità ad hoc.

### 3.3.5 Mobilità nella stessa sotto-rete IP

Cosa succede se un nodo cambia AP nella stessa sotto-rete?

L'indirizzo IP dell'host non cambia, gli switch sono in grado, grazie al **self-learning** di mappare con quale porta possono raggiungere gli host.

Nelle vecchie implementazioni quando un host cambiava AP questa mandava in broadcast l'indirizzo IP del nuovo arrivato in modo tale da aggiornare tutte le tabelle.

### 3.3.6 Caratteristiche avanzate di 802.11

**Rate adaption** La base station può cambiare dinamicamente la codifica (quindi la velocità di trasmissione).

Alcune implementazioni di 802.11 sono in grado di selezionare adattivamente la tecnica di modulazione, variando di conseguenza il SNR. Quando un host si allontana dalla base station, il SNR diminuisce e aumenta il BER, quando il BER diventa troppo elevato, la stazione riduce il tasso di trasmissione per ridurlo.

**Power management** Un nodo può annunciare all'AP la sua entrata nello stato di sleep. L'AP bufferizza i frame fino a quando il nodo non si sveglia. Ciò permette al nodo di risparmiare energia.

## 3.4 Bluetooth 802.15

È stata pensata per una rete con range molto limitato (meno di 20 metri), queste aree vengono chiamate personal area network. Il protocollo è nato con l'idea di rimpiazzare i cavi. Utilizza una **rete ad hoc** senza infrastruttura, la modalità di comunicazione è **master/slaves** (master centrale che guida i vari slave, esistono anche i dispositivi parked, in attesa).

### 3.4.1 Topologia (piconet)

Topologia base di Bluetooth

- Massimo 8 dispositivi attivi per volta in una piconet:
  - 1 master;
  - fino a 7 slave;
  - massimo 200 dispositivi parked.
- Slave:
  - devono essere in sync con il master;
  - devono sempre chiedere il permesso al master prima di inviare.
- Master:
  - gestisce tutta la comunicazione, anche l'accesso al mezzo condiviso;
  - attiva i dispositivi parked.
- Dispositivi parked:
  - non comunicano;
  - rimangono in sync con il master.

### 3.4.2 Scatternet

La topologia di base è limitata, per aumentare la copertura si utilizzano le scatternet. Si uniscono tra di loro più piconet, poi ci penseranno i master a garantire la comunicazione tra le varie piconet.

### 3.4.3 Comunicazioni bluetooth

Utilizzano la stessa banda di 802.11,  $2.4\text{ GHz}$ , partizionata in 79 canali. Per la modulazione di frequenza utilizzano 2 frequenze, 2-FSK e cambia la frequenza su cui viene inviato il segnale periodicamente (FHSS Frequency Hopping Spread Spectrum).

**FHSS** Utilizzo FHSS per minimizzare il più possibile le interferenze perché salto da una frequenza all'altra. Gli slave devono seguire la stessa sequenza (rimanere in sync). I dispositivi cambiano canale di trasmissione 1600 volte al secondo. Il master sceglie una sequenza pseudo casuale di frequenze da cambiare, il master sceglie un seed e gli slave rimangono in sync con lui.

### 3.4.4 Medium Access Control

**TDM (Time Division Multiplexing)** Si definiscono slot da  $625\mu\text{sec}$ , negli slot pari comunica il master, in quelli dispari comunicano gli slave. La comunicazione è single slave, il master chiede allo slave se ha da trasmettere, se affermativo potrà trasmettere nello slot successivo.

### 3.4.5 Modalità

**Inquiry** Per stabilire quali altri dispositivi sono presenti nei dintorni.

**Pairing** Connessione effettiva tra i dispositivi che potranno successivamente comunicare.

#### Durante la connessione

- **Active mode** = comunicazione attiva;
- **Sniff mode** = dormienti, periodicamente si risvegliano per ascoltare il master;
- **Hold mode** = dormono per un certo periodo di tempo deciso dal master;
- **Park mode** = rimangono addormentati finché non vengono svegliati dal master. Periodicamente si mettono in sync e ascoltano il master.

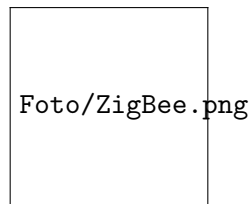
## 3.5 ZigBee 802.15.4

È l'unico standard open source. Protocollo sviluppato per la domotica e i dispositivi IoT, caratteristica fondamentale per questi dispositivi è quella di avere un basso consumo energetico e la quantità di informazioni che inviano è abbastanza limitata. Utilizza frequenze a  $2.4\text{ GHz}$ . La tecnologia ZigBee è supportata da un'alleanza di aziende che rendono i loro dispositivi compatibili tra loro per ottenere l'interoperabilità. Connessioni a bassa affidabilità.

Esistono tre suddivisioni di frequenza  $868\text{ MHz}$  (canale 0),  $915\text{ MHz}$  (canali da 1 a 10) e  $2.4\text{ GHz}$  (canali da 11 a 26) per un totale di **27 canali**.

### 3.5.1 Topologia

ZigBee utilizza una topologia di tipo **mesh** con più path ridondati, è scalabile ed è in grado di coprire una vasta area con dispositivi a basso consumo energetico.



### 3.5.2 Architettura

L'architettura prevede due tipo di indirizzo:

- **64 bit** = identificano lo specifico dispositivo (come il MAC);
- **16 bit** = identificano il servizio (come le porte TCP/IP).

#### Stack di ZigBee

<b>Applications</b>	2*802.15.4	5* ZigBee specification
<b>Application Framework</b>		
<b>Network &amp; Security</b>		
<b>MAC Layer</b>		
<b>PHY Layer</b>		

**Tipi di nodi** Esistono tre tipi di nodi:

1. **PAN coordinator** = coordina la rete, scopre nuovi dispositivi e gli configura;
  - Assegna l'ID da 16 bit e il canale di comunicazione ai nuovi dispositivi.
2. **FFDs (Full Function Devices)** = può lavorare come router ma non gestisce la rete, inoltra i pacchetti;
3. **RFDs (Reduced Function Devices)** = nodo terminale da cui arrivano le informazioni.

### 3.5.3 Medium access control

**Livello MAC** Gestisce dei superframe e il controllo di accesso al canale. Valida i frame e invia gli ack per rendere la comunicazione affidabile.

**Slotted CSMA/CA** Gestito dal PAN coordinator tramite i superframe.

**CSMA/CA** I nodi comunicano direttamente (come in WiFi ad Hoc).

**Superframe** É un frame inteso come intervallo di tempo tra due beacon frame (inviati periodicamente dal master).

É un protocollo adattativo, perché ha una parte in cui si può fare slotted CSMA/CA (sezione CAP) con slot assegnati dal master, e una parte libera (CFP) che viene utilizzata come CSMA/CA classico gestita autonomamente dai nodi. Il master decide e comunica a partire dal beacon i parametri (quanto è lunga la fase slotted, la fase classica e il periodo di inattività).



**Frame** Esistono 4 tipi di frame:

1. **Beacon frame** = utilizzato dal coordinator;
2. **Data frame** = usato per inviare i dati;
3. **Ack frame** = perché l'affidabilità del canale è bassa;
4. **MAC command frame** = utilizzato per la configurazione delle interfacce.

### 3.6 Z-Wave

Protocollo proprietario basato su una topologia mesh. I dispositivi comunicano punto a punto fino a 35 metri. Anche le reti Z-Wave possono essere collegate fra loro per riuscire a coprire una maggiore porzione di spazio. Ogni singola rete può supportare fino a 232 dispositivi e lavora sulle frequenze *868,42 MHz*.

### 3.7 Reti cellulari 4G e 5G

Il termine cellulare si riferisce al fatto che la regione coperta da una rete cellulare è divisa in una serie di aree geografiche, dette **celle**. Ogni cella contiene una base station che trasmette e riceve i segnali dai dispositivi mobili presenti nella cella.

#### Similitudini con le reti cablate

- vi è distinzione tra i dispositivi all'edge e al core;
- entrambi sono composte da reti di reti;
- si interfacciano direttamente con la rete internet pubblica;
- utilizzano i protocolli standard (HTTP, DNS, TCP, UDP, IP, NAT ecc.);
- interconnessi con la rete internet cablata;

#### Differenze con le reti cablate

- physical e data link layer molto diversi tra le due tecnologie;
- focalizzata sulla mobilità dei nodi;
- l'identificati degli utenti avviene tramite SIM card;
- modello di business (gli utenti stipulano un contratto con il provider).

#### 3.7.1 Elementi di una rete 4G

- **Dispositivi mobili (UE)** = Si connettono alla rete cellulare. Hanno un un identificatore unico detto IMSI (International Mobile Subscriber Identity) da 64 bit, memorizzato nella SIM;
- **Base station (eNode-B)**= Gestisce le risorse radio e i dispositivi mobili nella sua cella, coordinando l'autenticazione dei dispositivi e l'allocazione delle risorse, ha un ruolo attivo per la mobilità;
- **HSS (Home Subscriber Server)** = É un database che memorizza informazioni sui dispositivi nell'home network;
- **S-GW (Serving Gateway), P-GW (Packet Data Network Gateway)** = sono due router nel percorso tra il dispositivo mobile e internet. P-GW svolge funzioni simili a un router gateway e fornisce servizi NAT.
- **MME (Mobility Management Entity)** = É un elemento del piano di controllo. Insieme a HSS, fornisce autenticazione dei dispositivi e gestisce il passaggio di cella e gestisce il setup del tunneling tra il dispositivo mobile e internet.

#### 3.7.2 Protocollo LTE

#### 3.7.3 Separazione piano controllo e dati

**Piano di controllo** Gli attori del control plane sono: la base station, l'MME, l'HSS e P-GW. É un nuovo protocollo per la gestione della mobilità, sicurezza e autenticazione.



**Piano dati** Gli attori del piano dati sono: base station, S-GW e P-GW. É un nuovo protocollo a livello collegamento e fisico.



### 3.7.4 Stack data plane

Dispositivo mobile	Base station
Application	IP
Transport	Packet Data Convergence
IP	Radio Link
Packet Data Convergence	Medium Access
Radio Link	Physical
Medium Access	
Physical	

- **Packet Data Convergence** = si occupa della compressione dei pacchetti e la cifratura;
- **RLC (Radio Link Control)** = si occupa della frammentazione e riassemblaggio dei datagrammi IP, e garantisce trasporto affidabile grazie al protocollo ARQ;



- **Medium Access** = gestisce l'uso degli slot di trasmissione assegnati in modo dinamico ai vari dispositivi.

**LTE radio access network** I canali sono divisi in:

- **Downstream** = dalla base station al dispositivo. Si utilizzano FDM, TDM tra le frequenze del canale (OFDM (Orthogonal Frequency Division Multiplexing))
- **Upstream** = dal dispositivo alla base station. Si utilizzano FDM, TDM simile a OFDM.

Ad ogni dispositivo possono essere allocati, in modo dinamico, due o più slot di tempo da  $0,5\ ms$  su  $12\ frequenze$  diverse. Ci sono poi degli algoritmi di scheduling, non standardizzati, che assegnano in modo dinamico queste risorse radio (slot e frequenze) ai diversi dispositivi.

**Packet core**

Base station	S-GW
IP	GTP-U
Packet Data Convergence	UDP
Radio Link	IP
Medium Access	Link
Physical	Physical

La base station oltre ai livelli utilizzati per comunicare con il dispositivo mobile gestisce altri livelli. GTP-U permette di definire dei tunnel (su UDP) tra la base station e il S-GW. A sua volta il S-GW ritunnellizza il datagramma al P-GW. Cambio solo gli endpoint quando il dispositivo mobile si sposta da una parte all'altra della rete.

**Associazione con una base station**

1. La BS fa un broadcast su tutti i canali per dare informazioni sui servizi che offre e si identifica per permettere ai dispositivi di sincronizzarsi con la base station;
2. il dispositivo mobile fa uno scanning (partizionato) della rete finché non trova un primo segnale, da questo primo segnale trova la frequenza che bisogna usare. Quando trova le informazioni viene a conoscenza dell'ampiezza del canale e le configurazioni. Se è coperto da diverse celle può ricevere diverse informazioni da diverse base station;
3. il dispositivo mobile seleziona con quale BS associarsi;
4. ulteriori passi per l'autenticazione, stabilire la connessione e fare il setup del piano dati.

**Sleep mode** Anche i dispositivi mobili, per risparmiare batteria, hanno una modalità di sleep, esistono due modalità:

- **light sleep** = ogni  $100\ msec$  di inattività si sveglia e controlla se ci sono delle trasmissioni downstream, o se deve trasmettere;
- **deep sleep** = ogni  $5-10\ sec$  di inattività si risveglia ma deve ristabilire l'associazione con la BS.

**Rete cellulare globale** L'home network di un utente è collegato alle reti degli altri carrier mobili e ad internet tramite router gateway. Le reti mobili sono connesse tra loro dall'internet pubblico o da un IPX (Internet Protocol Packet eXchange) Network.

### 3.7.5 Reti 5G

L'obiettivo del 5G è quello di incrementare il bitrate di picco di 10 volte, diminuire la latenza di 10 volte e incrementare la capacità di traffico di 100 volte rispetto al 4G.

Questi obiettivi si cercano di raggiungere utilizzando due bande di frequenze molto alte: FR1 (450 MHz - 6 GHz) e FR2 (24 GHz - 52 GHz), queste ultime sono chiamate millimeter wave frequencies perché la lunghezza d'onda è nell'ordine dei millimetri e le distanze raggiunte sono dai 10 ai 100 metri. Il 5G non è retrocompatibile con il 4G. Per avere la stessa copertura dovrà utilizzare più BS (MIMO (Multiple Directional Antennae)).

#### Vantaggi

- con frequenze più elevate avremo più banda di trasmissione e maggiore velocità.

#### Svantaggi

- frequenze più elevate, simile alla luce, portata minore;
- più facilmente assorbita dagli ostacoli materiali.

### 3.8 Gestione della mobilità

Se un dispositivo si muove dal suo access network, continuando a inviare datagrammi IP, la rete dovrà svolgere l'**handover**.

#### Spettro di mobilità

No mobilità					
[HTML]CBCEFB	[HTML]9698ED	[HTML]6665CD	[HTML]6434FC	[HTML]6200C9	[HTML]303498
Il dispositivo si muove ma stacca la connessione durante il passaggio		Il dispositivo si muove nella stessa AP in una provider network		Il dispositivo si muove tra diverse AP in una provider network	

**Approccio** Non posso gestirlo a livello del core della rete. Si cerca di gestire nell'edge della rete utilizzando due modalità:

- **Routing indiretto** = la comunicazione passa per la home network che fa da tramite verso la nuova rete in cui si è spostato il nodo;
- **Routing diretto** = il corrispondente ottiene le informazioni che servono per raggiungere il nodo mobile che si è spostato.

**Home network** Rete di cui si ha un contratto con il provider. HSS conserva le informazioni dell'identificativo e dei servizi.

**Visited network** Ogni altra rete che non sia la home network. I servizi sono stabiliti con le altre reti per fornire l'accesso ai dispositivi visitatori.

Normalmente in ISP/WiFi non c'è l'idea di gestire a livello globale un certo utente. Le credenziali dall'ISP sono conservate nel dispositivo o ricordate dall'utente. Differenti reti hanno differenti credenziali, ci sono alcune eccezioni come eduroam.

**Registrazione** La rete home deve sapere dove si trova il dispositivo. Il dispositivo mobile si associa con il visited mobility manager che registra la posizione del dispositivo con l'HSS home.

Alla fine il visited mobility manager sa dell'esistenza del dispositivo mobile e l'HSS home sa dove si trova il dispositivo.

### 3.8.1 Routing indiretto

Il corrispondente ha l'indirizzo permanente.

1. I datagrammi sono inviati all'home network, indirizzati all'indirizzo permanente del dispositivo mobile da raggiungere;
2. Il gateway dell'home network intercetta i datagrammi, consulta l'HSS per determinare la visited network dove il dispositivo mobile risiede, incapsula tutto in un altro datagramma (lasciando quello originale intatto) e lo inoltra al gateway router del visited network utilizzando il tunneling;
3. Il visited network gateway riceve il datagramma, lo decapsula e lo inoltra al dispositivo mobile;
4. Due opzioni:
  - (a) Il datagramma generato dal dispositivo mobile viene spedito all'home gateway router e al corrispondente;
  - (b) Il datagramma viene inviato direttamente al corrispondente dal visited network (local breakout).

### Vantaggi

- Totalmente trasparente al corrispondente;
  - mantenere delle connessioni che sono state instaurate.

### Svantaggi

- routing triangolare = inefficiente quando corrispondente e dispositivo mobile sono nella stessa rete, perché devo prima passare dalla home network per poi ritornare nella visited network.

**Cambio di rete** Quando il dispositivo si sposta su un'altra rete l'agente della seconda visited network informa la prima che c'è stato lo spostamento e si cambia l'estremo del tunnel.

### 3.8.2 Routing diretto

I dati non passano dalla home network, il corrispondente comunica direttamente con il dispositivo mobile.

1. Il corrispondente contatta la home network e richiede l'indirizzo del dispositivo mobile;
2. riceve l'indirizzo del dispositivo mobile;
3. il corrispondente invia il datagramma all'indirizzo della visited network;
4. il visited gateway router inoltra il datagramma al dispositivo.

### **Vantaggi**

- Risolto il problema del routing triangolare.

### **Svantaggi**

- Non è più trasparente al corrispondente;
- Se il dispositivo cambia rete, può essere gestito ma con complessità aggiuntiva.

**Cambio di rete** Se il dispositivo si sposta su un'altra rete, faccio un tunnel tra la prima visited network e la seconda visited network.