



UNIVERSITÀ  
DI PARMA

DIPARTIMENTO DI SCIENZE MATEMATICHE, FISICHE ED INFORMATICHE  
Corso di Laurea in Informatica

# Il Livello Network

## Parte III : Routing

RETI DI CALCOLATORI - a.a. 2022/2023

Roberto Alfieri

# Il livello Network: sommario

## PARTE I

- ▶ Scopi del livello Network
- ▶ Commutazione di circuito e di pacchetto
- ▶ La rete ATM
- ▶ La famiglia dei protocolli TCP/IP
- ▶ Il protocollo IP: trama indirizzi, instradamento
- ▶ Protocolli di servizio: ARP, ICMP, DHCP

## PARTE II

- ▶ IPv6

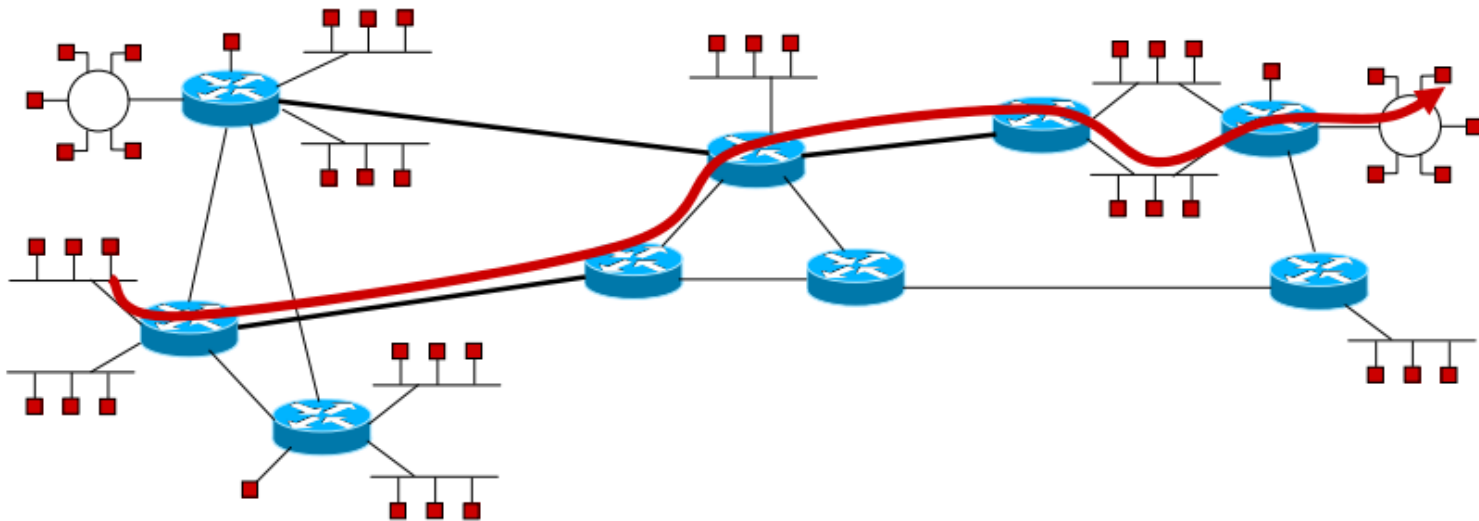
## PARTE III

- ▶ Routing: Algoritmi e protocolli. Distance Vector e Link State.

# Routing

E' la scelta del percorso su cui inviare i dati quando mittente e destinatario appartengono a 2 reti diverse e quindi la consegna non può avvenire direttamente a livello Link.

In questo caso il mittente affida la consegna ad una struttura interconnessa di **Router** i quali passano i datagrammi dall'uno all'altro finché raggiungono quello che può consegnarli direttamente al destinatario.

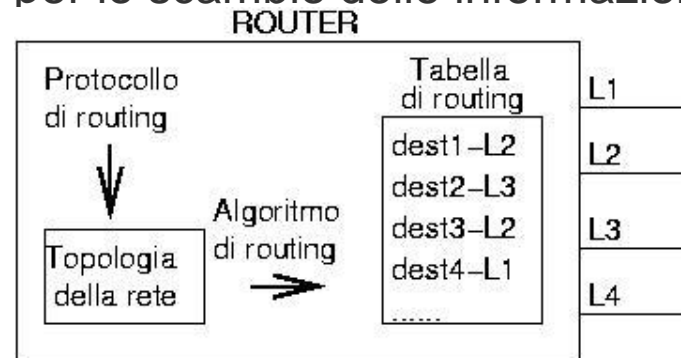


# Router

I Router sono dotati di due componenti funzionali: **instradamento e inoltro**.

**Instradamento:** Creazione di **una tabella di routing (RT)**, che contiene le informazioni riguardo la porta di uscita per le destinazioni dei Frame. Componenti:

- ▶ **Algoritmi di routing (RA)** si preoccupano di scegliere lungo quale linea di uscita vanno instradati i pacchetti in arrivo. Sono usati per il calcolo della tabella di routing in base alla topologia della rete.
- ▶ **Protocolli di routing (RP)** utilizzati per lo scambio delle informazioni necessarie per determinare la topologie della rete.



**Inoltro:** Applicazione dell'instradamento sui singoli datagrammi ricevuti.

- ▶ **Lettura dell'intestazione IP** ed estrazione dell'indirizzo di destinazione.
- ▶ **Look-up della tabella di routing** ed identificazione dell'interfaccia di uscita
- ▶ **Switching:** trasferimento fisico dei datagrammi da ingresso a uscita.

# Algoritmo di Flooding

Flooding è un semplice algoritmo di routing in cui ogni pacchetto entrante è spedito verso tutte le linee di uscita eccetto quella di cui è arrivato.

Per gestire la propagazione di pacchetti duplicati le possibili tecniche sono:

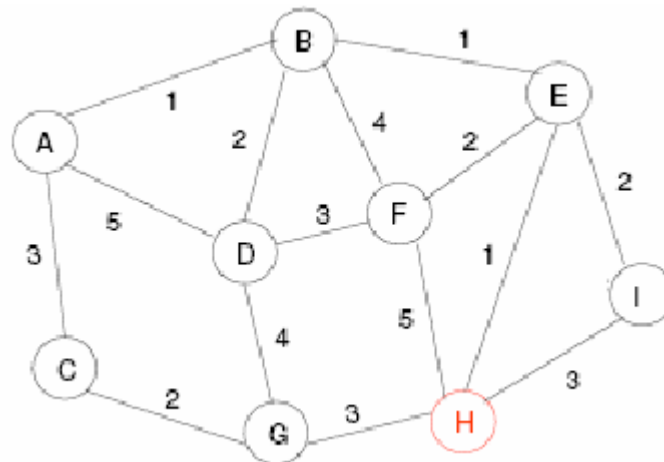
- Conteggio del numero di salti, decrementando il valore ad ogni salto.
- Numero di sequenza assegnato dal mittente ad ogni messaggio. Ogni nodo deve gestire una lista con i messaggi già trasmessi, scartando eventuali repliche.

Flooding è utilizzato

- nei bridge per l'invio di broadcast e nella fase di autoapprendimento
- nei protocolli link state.

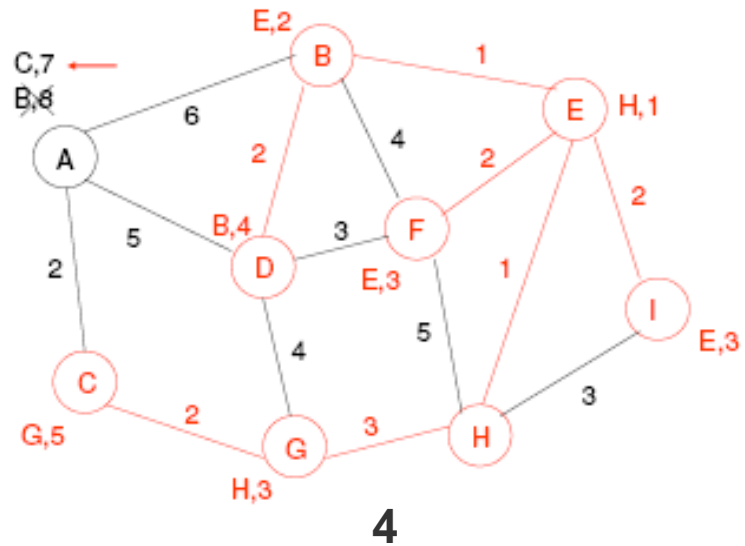
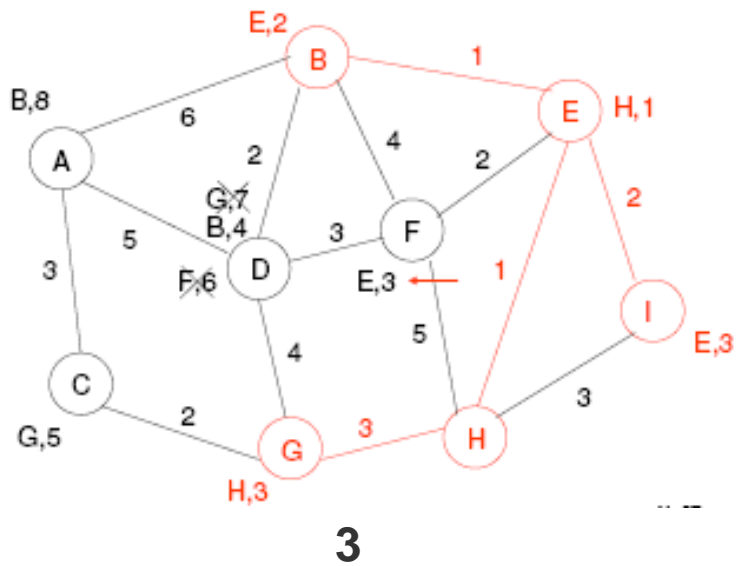
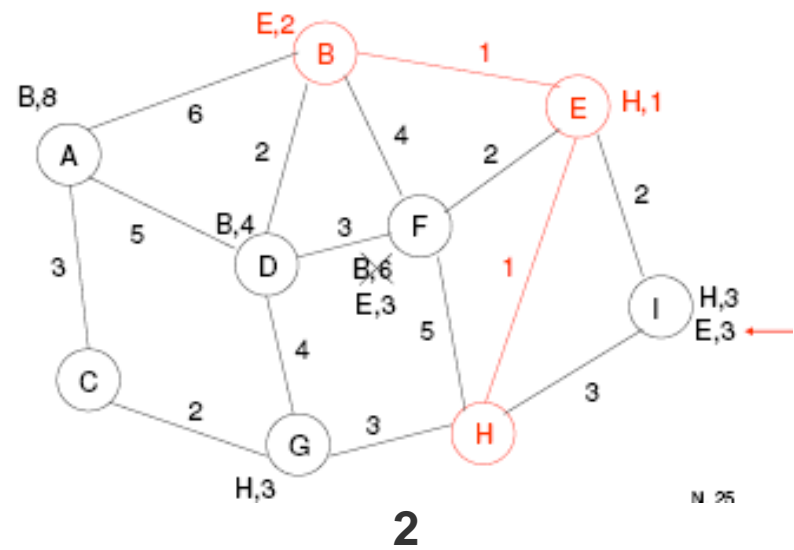
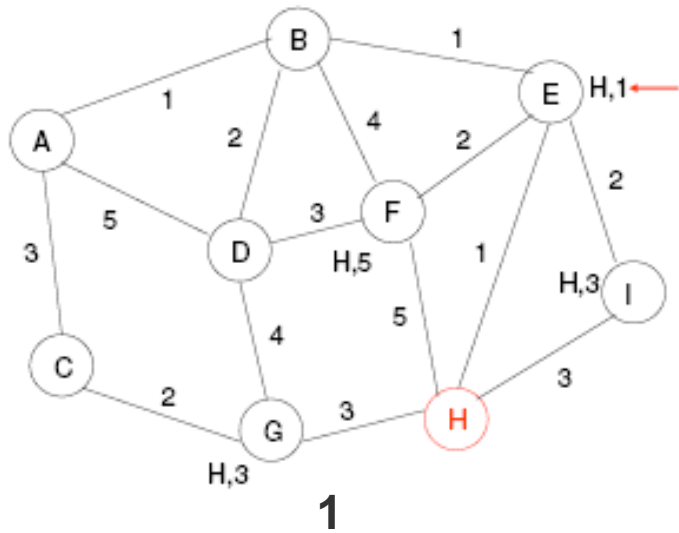
# Algoritmo Shortest Path First

- ▶ La topologia della rete può essere rappresentata da un grafo pesato non orientato. Il peso attribuito ad ogni arco viene determinato mediante l'attribuzione di una metrica che tiene conto vari parametri di rete (velocità, latenza, ..)
- ▶ Per il principio di ottimalità il cammino minimo che un nodo deve percorrere per raggiungere qualsiasi altro nodo del grafo è un albero detto “**Sink Tree**” (a partire dalla destinazione) o **Source Tree** (a partire dall'origine).
- ▶ Si conoscono diversi algoritmi per elaborare il percorso più breve tra due nodi. Il più utilizzato è stato ideato da Dijkstra nel 1959 ed è noto con il nome di **Shortest Path First (SPF)**.



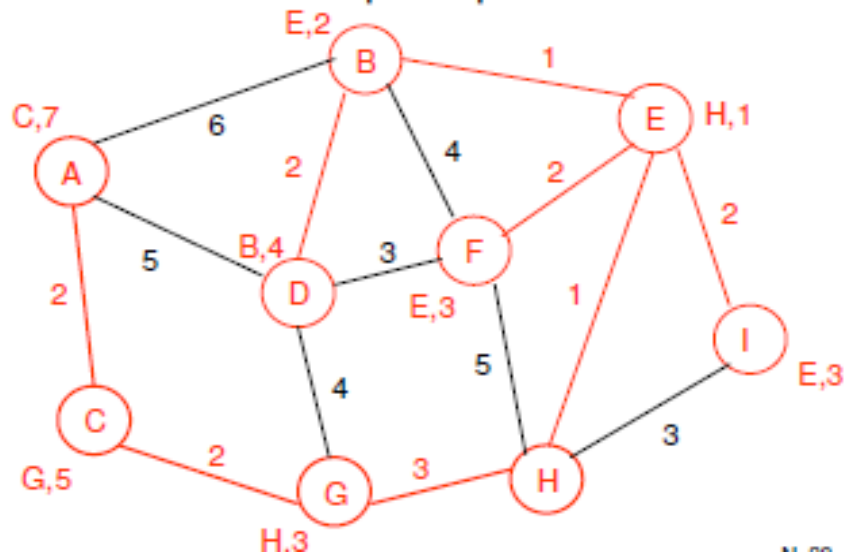
Ricerca del cammino minimo da A verso H. A partire da H (Sink Tree) mettiamo etichette provvisorie sui nodi adiacenti. Scegliamo la più piccola delle distanze.

# Sink Tree del nodo H con SPF



# Risultato finale

- ▶ A va ad H attraverso C con distanza 7.





# Protocolli di Routing

I RP stabiliscono le modalità di comunicazione tra i router per la costruzione della topologia di rete. Possono essere statici o dinamici (adattivi).

## ► Routing Statico:

- La topologia e tabella di routing vengono definite in fase di setup della rete.
- In caso di variazioni (inserimento o eliminazione di nodi o collegamenti) è necessario l'intervento dell'operatore.

## ► Routing Dinamico:

- La topologia della rete è costruita dinamicamente in modo automatico, in base ai cambiamenti della topologia di rete o al traffico.

### ■ **Routing Dinamico Centralizzato:**

- - un nodo centrale raccoglie le informazioni sullo stato della rete
- - calcola (RA) la tabella per ogni nodo e la spedisce.
- - tabelle consistenti, ma abbiamo un punto di criticità

### ■ **Routing Dinamico Distribuito:**

- - i nodi si scambiano informazioni sullo stato della rete
- - ogni nodo calcola la propria tabella sulla base delle informazioni ricevute.
- - Tre categorie di protocolli: **Distance Vector, Link State e Gerarchici**

# Protocolli Distance Vector

- ▶ Nel grafo ogni coppia di nodi ha una **distanza** che dipende dalla “metrica” utilizzata.
- ▶ Una metrica ragionevole tra 2 nodi adiacenti potrebbe dipendere dalla velocità, la latenza e il Throughput del canale.
- ▶ La distanza di un percorso potrebbe dipendere dalla somma delle singole distanze e/o dal numero di salti.
- ▶ **Nel Protocollo Distance Vector (DV) ogni nodo invia ai primi vicini l'elenco delle distanze (a lui note) con tutti gli altri nodi (ovvero il DV), periodicamente e ogni volta che c'è un cambiamento.**
- ▶ Le distanze con i primi vicini vengono misurate (ad esempio con un ECHO), mentre le altre distanze sono derivate dalle informazioni ricevute
- ▶ Tutte le volte che un Router calcola una nuova tabella di instradamento, la invia agli IS adiacenti (cioè quelli collegati da un cammino fisico diretto) sotto forma di DV
- ▶ **La tabella** contiene una entry per ogni nodo presente in rete
- ▶ Ogni **entry** è composta da quattro parametri:
  - Indirizzo (del nodo remoto)
  - Hops (numero di salti per raggiungerlo)
  - Costo (determinato in base alla metrica)
  - Linea
- ▶ Il DV inviato contiene Indirizzo-Hops-Costo di ogni entry (non la linea).

# Protocolli Distance Vector

Il router che riceve il DV prima di tutto verifica se vi sono delle modifiche dal precedente e, in caso affermativo, aggiorna i campi hops e costo, sommando 1 a tutti gli hops e sommando il costo della linea da cui è arrivato il messaggio al campo costo.

Il passo successivo è l'aggiornamento della propria tabella tramite un processo di **fusione** (merge) di tutti i Distance Vector a lui pervenuti da ogni linea attiva.

| Indirizzo | Hops | Costo | Linea |
|-----------|------|-------|-------|
| 1         | 3    | 25    | 3     |
| 2         | 5    | 35    | 2     |
| 3         | 9    | 50    | 6     |
| 4         | 1    | 5     | 7     |
| 5         | 0    | 0     | 0     |

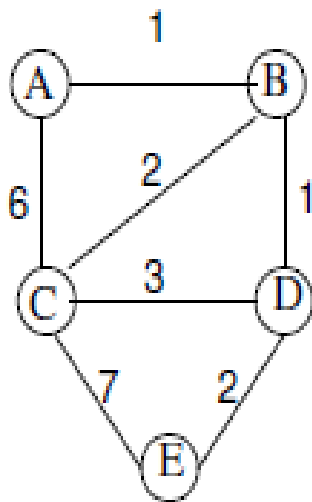
DV

Nella fusione vengono esaminate le entry con lo stesso indirizzo di destinazione, scartando quelle con i costi maggiori.

**A parità di costo** si seleziona quella che ha il **minor numero di hops**.

Il protocollo è semplice ma a lenta convergenza: l'informazione di una modifica della topologia (linea interrotta, router spento, ..) si propaga lentamente.

# Distance Vector: Esempio di calcolo



Distance Vector A =  $\{(A,0), (B,1), (C,6), (D,\infty), (E, \infty)\}$

DV B =  $\{(A,1), (B,0), (C,2), (D,1), (E,\infty)\}$

DV C =  $\{(A,6), (B,2), (C,0), (D,3), (E,7)\}$

DV D =  $\{(A,\infty), (B,1), (C,3), (D,0), (E,2)\}$

DV E =  $\{(A,\infty), (B,\infty), (C,7), (D,2), (E,0)\}$

Tabella di routing alla fine del periodo di convergenza (al termine dello scambio dei DV)

1. A riceve DV di B

| dest | Costo, next hop |
|------|-----------------|
| A    | 0               |
| B    | 1, A            |
| C    | 3, B            |
| D    | 2, B            |
| E    | $\infty$        |

2. A riceve DV di C

| dest | Costo, next hop |
|------|-----------------|
| A    | 0               |
| B    | 1, A            |
| C    | 3, B            |
| D    | 2, B            |
| E    | 13, C           |

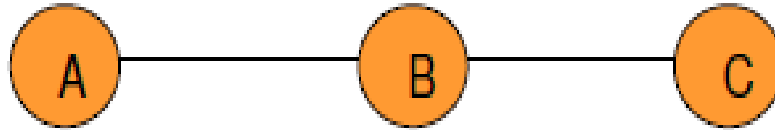
3. B riceve DV di D

| dest | Costo, next hop |
|------|-----------------|
| A    | 1, B            |
| B    | 0               |
| C    | 2, B            |
| D    | 1, B            |
| E    | 3, D            |

4. A riceve DV di B

| dest | Costo, next hop |
|------|-----------------|
| A    | 0               |
| B    | 1, A            |
| C    | 3, B            |
| D    | 2, B            |
| E    | 4, B            |

# Distance Vector: il problema “count to infinity”



- Situazione iniziale:  $D_{AC} = 2$  e  $D_{BC} = 1$ 
  - Link BC va fuori servizio
  - B riceve il DV di A che contiene l'informazione  $D_{AC} = 2$ , per cui esso computa una nuova  $D_{BC} = D_{BA} + D_{AC} = 3$  e la comunica ad A
  - A calcola la nuova distanza  $D_{AC} = D_{AB} + D'_{BC} = 4$
  - Il processo può continuare all'infinito
- Vari rimedi sono stati proposti, nessuno risolutivo

# RIP protocol

**RIP (Routing Information Protocol)** è la prima implementazione di un protocollo DV. Ne esistono 3 versioni:

- ▶ RIPv1 (RFC 1058) usa il routing "classful" (reti senza NetMask)
- ▶ RIPv2 (RFC 2453) usa il routing "classless" (CIDR)
- ▶ RIPv1 (RFC 1058) estensione del protocollo RIPv1 per supportare IPv6.

## Caratteristiche:

- ▶ Metrica: basata solo sulla minimizzazione degli hops (max 15)
- ▶ Nodi RIP Attivi (tipicamente Router): annunciano il loro percorsi
  - **ogni 30 secondi e quando si verificano cambiamenti di topologia**
- ▶ Nodi RIP Passivi (tipicamente Host): aggiornano senza annunciare

# Protocolli Link State

E' un RP con cui ogni nodo determina e mantiene aggiornata la topologia della rete da cui calcola la Tabella di Routing applicando un RA.

Il protocollo si sviluppa nelle seguenti fasi:

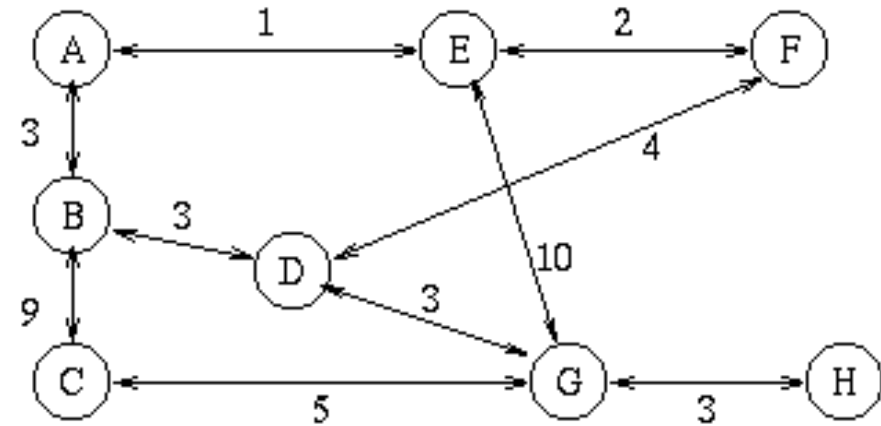
1. **Scoperta dei vicini** (neighbor greetings) : invio di un pacchetto HELLO su tutte le linee.
2. **Misurazione costo linea**: invio di un ECHO ai router che hanno risposto all'HELLO.
3. **Costruzione di un pacchetto** (Link State Packet - LSP) con tutte le informazioni ricavate nella fase 2: l'identità del trasmittente, numero di sequenza, dall'età e lista di vicini con il relativo ritardo misurato.
4. **Distribuzione periodica del LSP a TUTTI i nodi della rete**, con un numero di sequenza, utilizzando il Flooding. Se arriva un pacchetto con un numero di sequenza inferiore al numero più alto visto fino a quel momento, il pacchetto viene scartato (ritenuto obsoleto).
5. **Ogni nodo**, dopo aver ricevuto gli LSP da tutti gli altri, **costruisce la topologia della rete** e applica un RA (Shortest Path First di Dijkstra) per il **calcolo della tabella**.

# Protocolli Link State: esempio 1/2

- ▶ Raccolta Info (HELLO - ECHO)
- ▶ Propagazione info (LSP) in flooding multicast
- ▶ Per il nodo D il pacchetto LSP sarebbe:

| Adiacente | Costo |
|-----------|-------|
| B         | 3     |
| F         | 4     |
| G         | 3     |

- ▶ Ogni nodo ricostruisce la mappa della rete fondendo i LSP ricevuti in una tabella come quella a fianco.



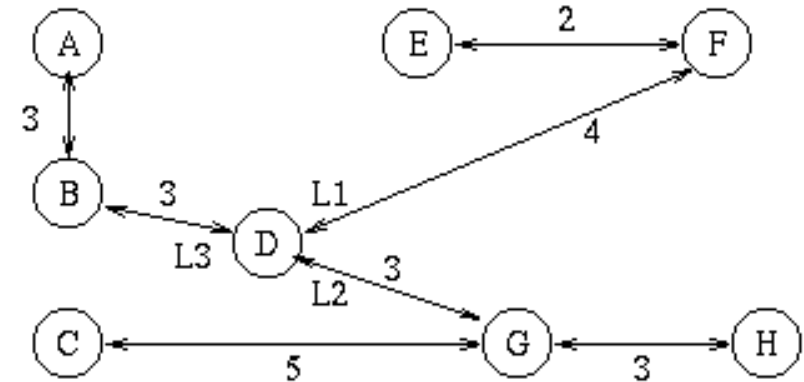
|   |     |     |      |
|---|-----|-----|------|
| A | B/3 | E/1 |      |
| B | A/3 | C/9 | D/3  |
| C | B/9 | G/5 |      |
| D | B/3 | F/4 | G/3  |
| E | A/1 | F/2 | G/10 |
| F | D/4 | E/2 |      |
| G | C/5 | D/3 | E/10 |
| H | G/3 |     |      |

H/3



# Protocolli Link State: esempio 2/2

- ▶ Il calcolo della tabella di instradamento si riduce ora al calcolo dello spanning tree di tipo SPF (Shortest Path First) e lo si effettua tramite il noto algoritmo di Dijkstra
- ▶ Lo spanning tree ad esempio del nodo D risulterà come nella figura a lato e, a seguire la relativa tabella di instradamento
- ▶ L'algoritmo può gestire reti di grandi dimensioni grazie alla sua rapida convergenza ed il suo comportamento è prevedibile, poiché ogni nodo ha in memoria la mappa intera della rete.
- ▶ Difficilmente si generano loop e, comunque, risulta facile identificarli ed eliminarli.



|   |    |
|---|----|
| A | L3 |
| B | L3 |
| C | L2 |
| E | L1 |
| F | L1 |
| G | L2 |
| H | L2 |

# OSPF

**OSPF** (Open Shortest Path First, RFC2328)

È un protocollo IGP di tipo **Link State Packet** ed è raccomandato da IETF per Internet.

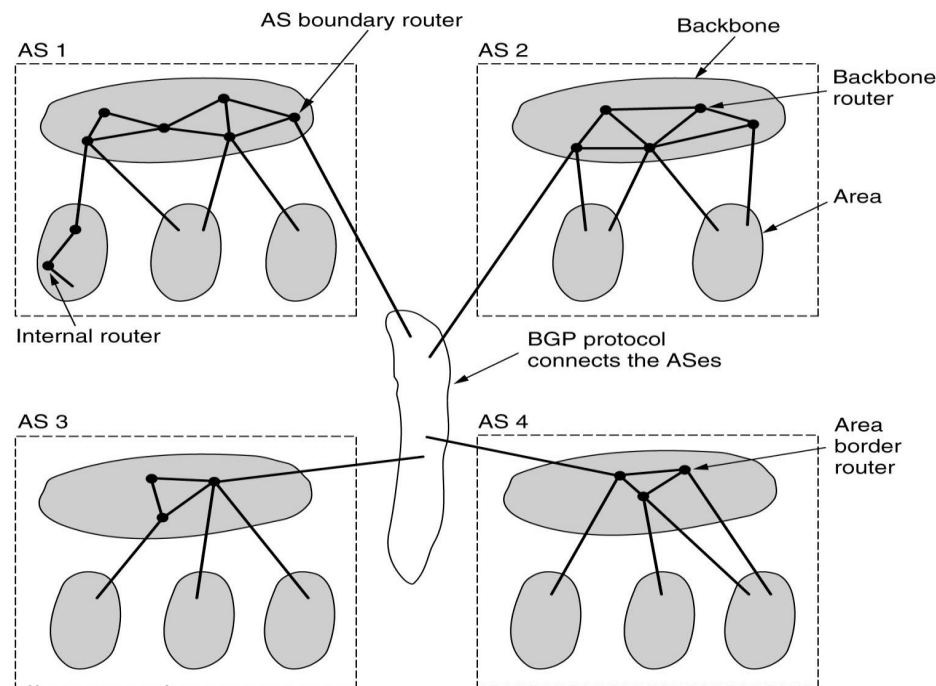
- ▶ Ciascun router emette periodicamente (default 10 s) dei **pacchetti Hello** multicast (244.0.0.5) , per valutare possibili modifiche topologiche
- ▶ Ogni Router costruisce un pacchetto con l'elenco delle linee attive e dei loro costi ( 1/larghezza di banda della linea)
- ▶ Invia in **flooding** pacchetti **Link State Update (LS-Update)** multicast 244.0.0.5
- ▶ Questi vengono riscontrati con un **Link State Ack (LS-Ack)**
- ▶ Se, in base ai pacchetti di update ricevuti, si sono verificate modifiche della topologia ricalcola la tabella di routing con l'algoritmo **Shortest Path First (SPF)**
  
- ▶ Scalabilità:
  - Il calcolo del SPF ha complessità  $O(N \log N)$ , dove N è il numero di router e reti.
  - Il problema viene risolto suddividendo un AS in aree:
    - Ogni AS-OSPF contiene almeno un'area: l'area di Backbone (area 0)
    - Le eventuali altre aree sono connesse al Backbone
    - Ogni router mantiene informazioni solo riguardo la topologia della propria area.

# Protocolli Gerarchici

Nel caso di reti di grandi dimensioni non è possibile gestire le tabelle di routing per l'intera rete in tutti i router, in questo caso il routing deve essere gerarchico:

- ▶ la rete viene ripartita in aree, chiamate Autonomous-System
- ▶ i router all'interno di un area sono in grado di effettuare l'instradamento relativamente alla sola area
- ▶ per destinazioni al di fuori dell'area si limitano ad inviare i pacchetti a dei router “di bordo” che sono a conoscenza della topologia esterna dell'area
- ▶ i router “di bordo” si occupano solamente dell'instradamento dei pacchetti fra aree

In linea di principio la ripartizione può essere effettuata tante volte quante si vuole creando più livelli nella gerarchia di routing



# Protocolli di Routing in Internet

In TCP/IP i router sono suddivisi in due classi, **Exterior Router** ed **Interior Router**. I primi interconnettono due insiemi di reti distinti. Ogni insieme di reti, gestito da una singola autorità amministrativa, è un Autonomous System ed i router interni ad essi sono proprio gli Interior.

Un sistema è detto autonomo, poiché è libero di scegliere un'architettura di instradamento interna, ma deve raccogliere informazioni su tutte le sue reti e progettare uno o più gateway, gli Exterior Router, che passino le informazioni di raggiungibilità ad altri sistemi autonomi.

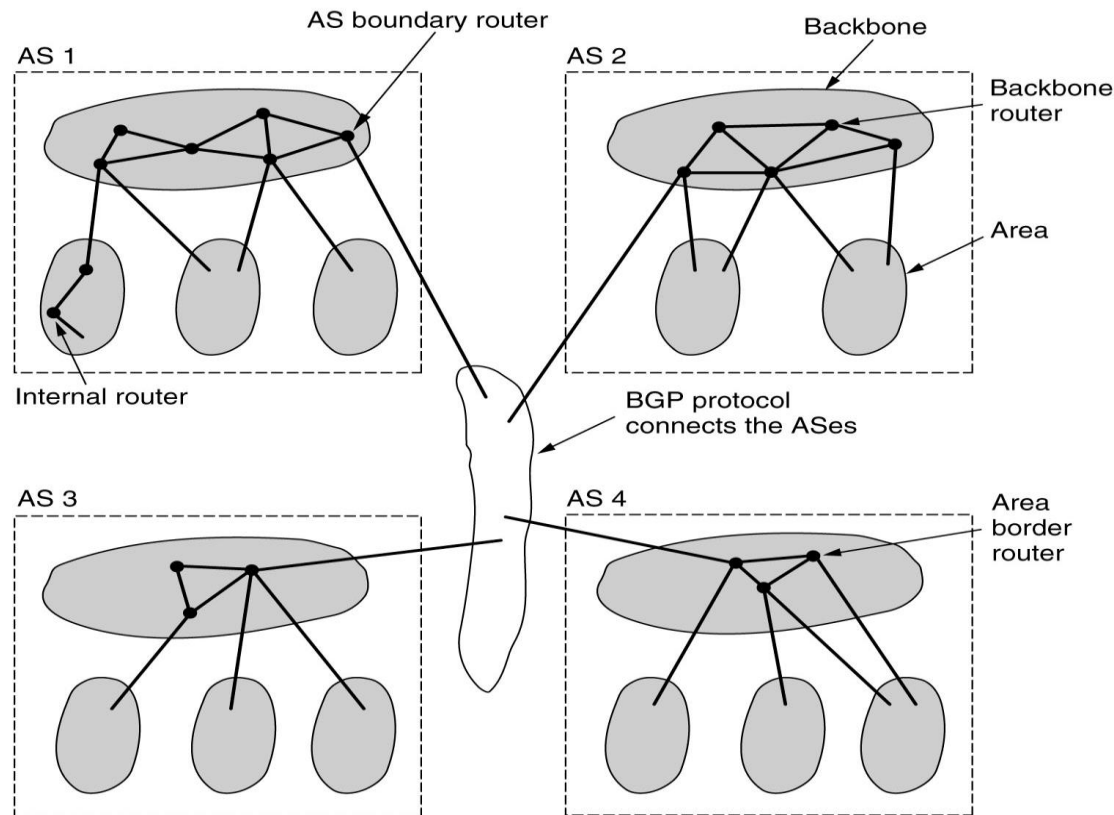
Gli Exterior Router utilizzano protocolli denominati EGP (Exterior Gateway Protocol), mentre gli Interior Router scambiano informazioni di instradamento tramite gli IGP (Interior Gateway Protocol).

I protocolli IGP più utilizzati sono **RIP** (Distance Vector) e **OSPF** (Link State). **BGP** è il protocollo raccomandato in Internet per l'interconnessione di Autonomous System.

# BGP

**BGP** ( Border Gateway Protocol, RFC 1771).

Protocollo Path Vector : invece di propagare i costi propaga la sequenza di AS da attraversare per arrivare a destinazione



# Routing Anycast

IP Anycast è una tecnica che consente a diverse macchine (server) di condividere lo stesso indirizzo IP, in modo che un client raggiunga il server più vicino (a minore costo) per ridurre la latenza e aumentare la ridondanza.

Gli algoritmi di routing basati sui protocolli Distance Vector o Link State gestiscono automaticamente percorsi multipli per raggiungere una destinazione, selezionando il percorso a minor costo e quindi la destinazione più conveniente.

Ovviamente il routing non è stabilito sugli indirizzi IP, ma sulle reti, quindi sarà necessario definire una rete Anycast, (anche piccola) replicata in diversi punti della rete.

