

TFA 2014/15
SISTEMI E RETI DI
CALCOLATORI PER L'INSEGNAMENTO
UNITA' DIDATTICA:
ALGORITMI DI ROUTING

21/05/2015
Laura Ricci

STRUTTURA UNITA' DIDATTICA

- Prerequisiti della unità didattica:
 - conoscere le funzionalità del livello IP, struttura di un pacchetto IP
 - funzioni di un router
 - nozione di grafo
- contenuto della lezione: illustrazione di alcuni algoritmi di routing
- metodologia:
 - presentazione algoritmo sequenziale centralizzato
 - dimostrazione funzionamento dell'algoritmo mediante animazioni
 - introdurre il concetto di algoritmo distribuito: tutti i router eseguono lo stesso codice
 - coordinamento comportamento router diversi
 - presentazione algoritmo distribuito
- laboratorio: implementazione dell'algoritmo sequenziale.

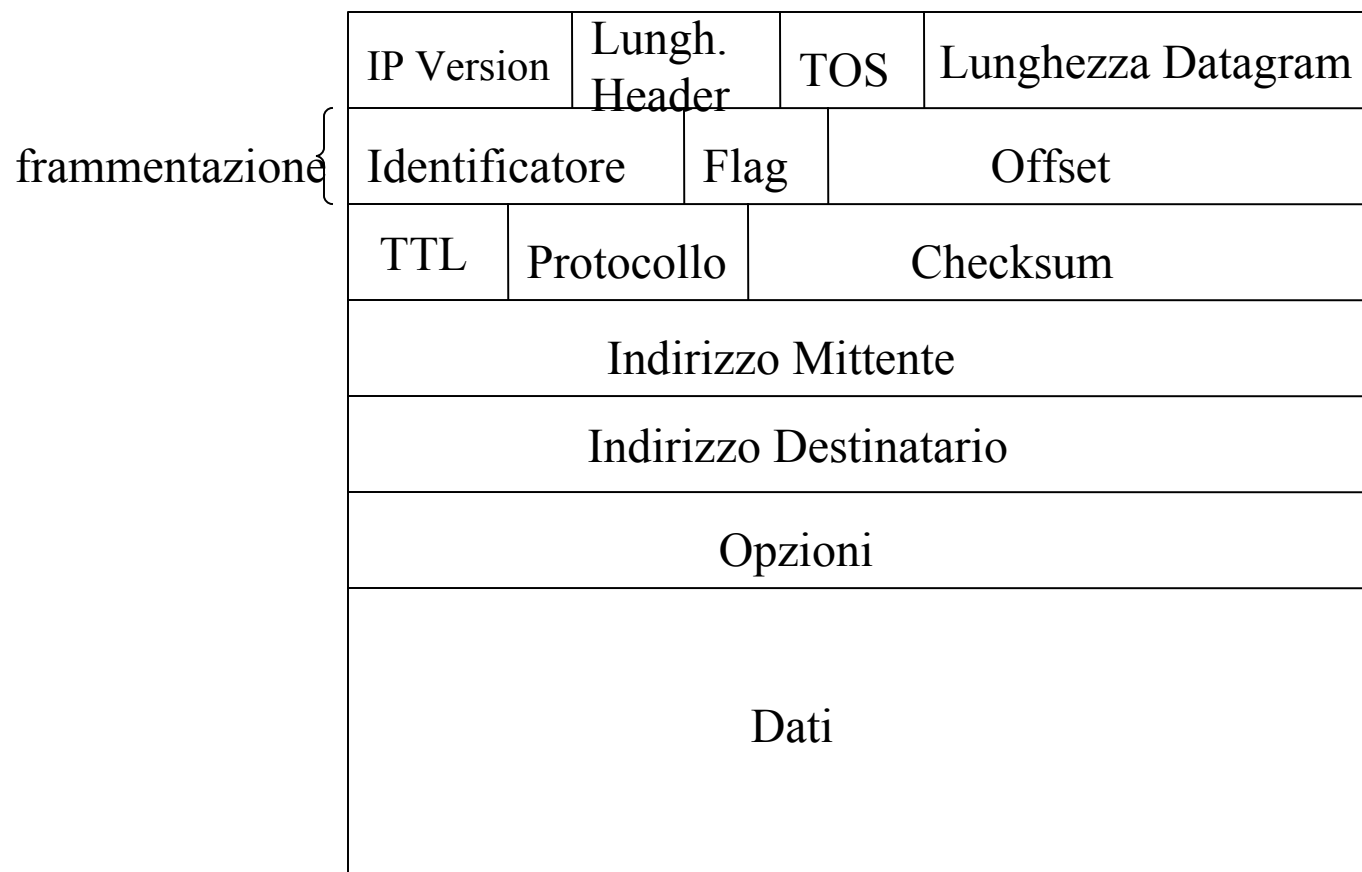
PREREQUISITI: LIVELLO IP

- Livello Network = livello IP
- Modello di servizio di IP: invio di **datagrams** secondo la filosofia **best effort**.
- **datagram** (o pacchetto IP) : contiene informazioni sufficienti affinché la rete (i routers) possa inoltrarlo verso la destinazione
- **best effort**: IP fornisce un servizio **minimale**. Motivazione: possibilità di eseguire il protocollo su reti che utilizzano diverse tecnologie

IP attualmente può essere eseguito su qualsiasi rete, anche su reti con tecnologie non esistenti al momento della sua definizione

metafora: "IP può operare su una rete che trasporta i messaggi mediante piccioni viaggiatori"

PREREQUISITI: FORMATO DI UN PACCHETTO



PREREQUISITI: FORMATO DEL PACCHETTO

IP Version: IPV4 / IPV6

TOS (Type of Service) Consente un trattamento differenziato dei pacchetti. Ad esempio, un particolare valore di **TOS** indica che il pacchetto ha una priorità maggiore rispetto agli altri.

Utile per distinguere per distinguere tipi diversi di traffico (traffico real time, messaggi per la gestione della rete,..)

TTL - Time to Live: consente di limitare la diffusione del pacchetto sulla rete

- valore iniziale impostato dal mittente
- quando il pacchetto attraversa un router, il valore viene decrementato
- quando il valore diventa 0, il pacchetto viene scartato

Introdotta inizialmente per evitare **percorsi circolari** infiniti del pacchetto.

Utilizzato anche per limitare la diffusione del pacchetto nel multicast

PACCHETTI IP: FRAMMENTAZIONE

Trasmissione di pacchetti IP su reti fisiche diverse ed eterogenee:

- quando un pacchetto P viene spedito su una rete fisica R , P deve essere incapsulato in un pacchetto P_{link} .
- la dimensione di P_{link} dipende dal **livello link** della rete fisica R , in particolare dall' **MTU (Maximum Transfer Unit)** di quella rete
- Reti fisiche diverse \Rightarrow MTU diversi
 - Ethernet**: MTU = 1500 bytes
 - FDDI** : MTU = 45500 bytes
- la dimensione di P_{link} in genere è diversa da quella di P

PACCHETTI IP: FRAMMENTAZIONE

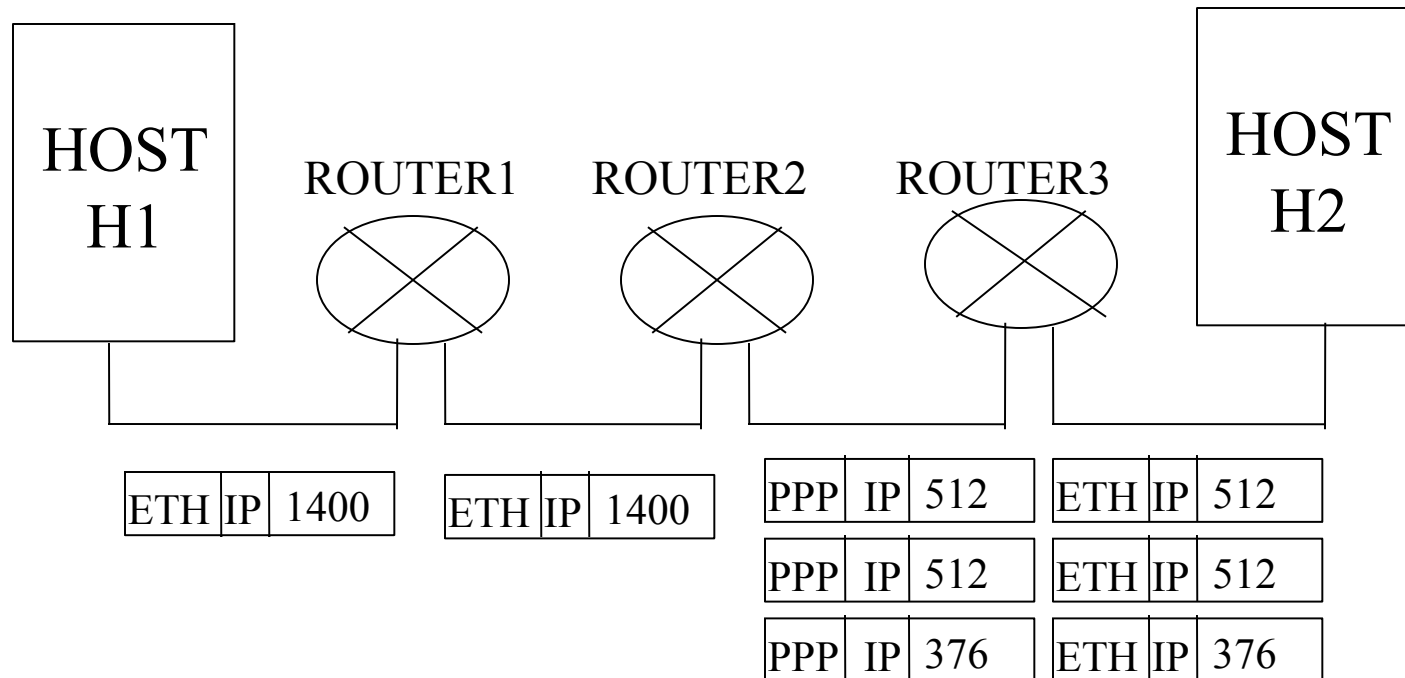
Approcci per la gestione di links eterogenei

- definire la dimensione massima di un pacchetto IP come la minima dimensione dei pacchetti supportata a livello link dalle rete fisiche

svantaggi:

- difficoltà relativa alla definizione a priori di un limite inferiore alla dimensione dei pacchetti a livello link (specie per reti in espansione come Internet)
- spreco di risorse
- **fragmentazione**: dividere un datagram IP in una o più parti (**fragmenti**) nel caso in cui l'MTU di una rete fisica sia inferiore alla dimensione del datagram.

PACCHETTI IP: FRAMMENTAZIONE



PACCHETTI IP: FRAMMENTAZIONE

- la frammentazione del pacchetto viene effettuata dai routers
- la ricostruzione del pacchetto avviene nell'host finale (un pacchetto una volta frammentato viene ricostruito solamente quando arriva a destinazione)
- ricostruzione del pacchetto frammentato:
 - utilizza i campi identificatore, flag, offset contenuti nel pacchetto IP
 - **identificatore**: identifica in modo univoco i pacchetti generati da un host. Quando un router frammenta un pacchetto assegna a tutti i segmenti generati l'identificatore del pacchetto da cui provengono
 - **offset**: identifica la posizione del frammento all'interno del datagram originale
 - **flag M** (more): indica se questo è l'ultimo frammento del pacchetto IP

PACCHETTI IP: FRAMMENTAZIONE

IP	1400
----	------

Pacchetto IP, identificatore 233

PPP IP	512
--------	-----

Fragmento 1, identificatore 233, offset 0, M=1

PPP IP	512
--------	-----

Fragmento 2, identificatore 233, offset 512, M=1

PPP IP	376
--------	-----

Fragmento 3, identificatore 233, offset 1024, M=0

PREREQUISITI: INDIRIZZI IP

Schemi di indirizzamento IP

- Classful Addressing
- Subnetting
- Classless Addressing **CIDR**

Classfull Addressing:

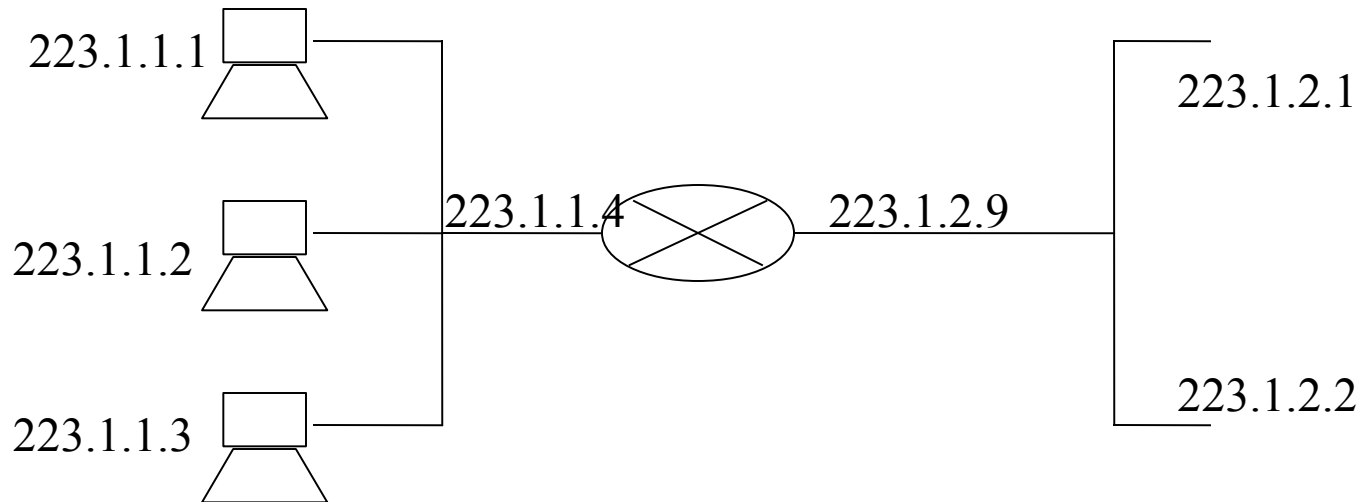
ogni indirizzo IP rappresenta una gerarchia a due livelli

- la prima parte dell'indirizzo identifica **la rete fisica** a cui appartiene l'host individuato da quell'indirizzo
- la seconda parte individua **l'host**

171.69.210.245
rete host

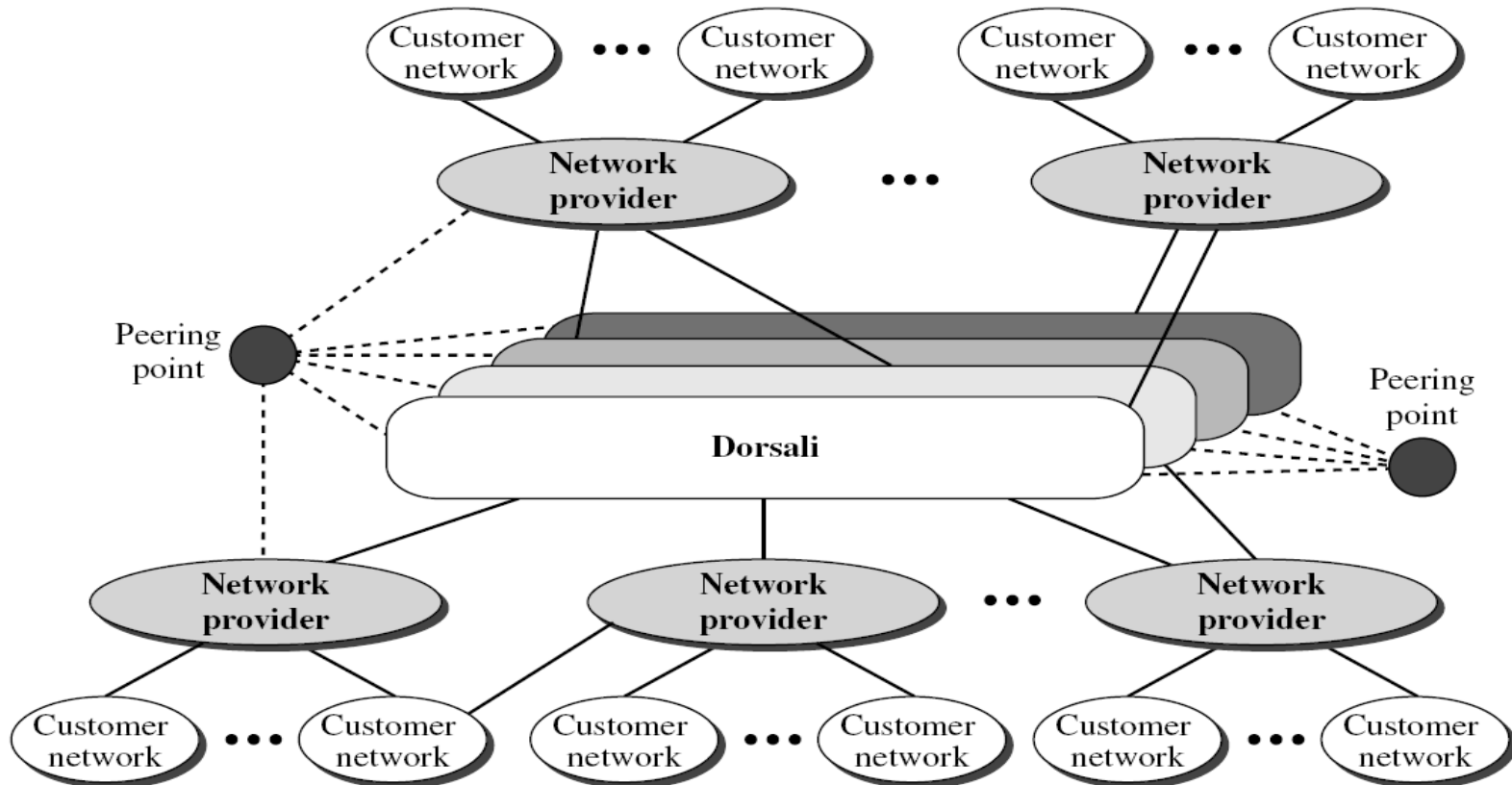
PREREQUISITI: INDIRIZZI IP

- Tutti gli hosts ed i routers che condividono lo stesso indirizzo di rete sono connessi alla stessa rete fisica



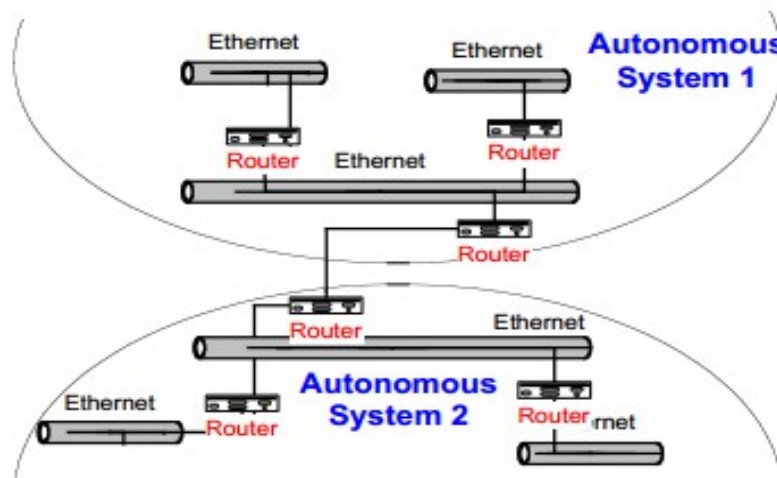
- ogni rete fisica connessa ad Internet ha almeno un router che è connesso ad un'altra rete fisica
- un router possiede un insieme di **interfacce di rete**

PREREQUISITI: LA STRUTTURA DI INTERNET



PREREQUISITI: LA STRUTTURA DI INTERNET

- **Autonomous System (AS)** o **routing domain** è una regione di Internet amministrata da una singola entità
 - i prefissi di tutti gli host sono caratterizzati da un prefisso comune
 - le politiche di routing appartengono ad un dominio amministrativo unico
- Esempi:
 - IBM corporate network
 - in generale gestite da grosse compagnie, università, enti



PREREQUISITI: LA STRUTTURA DI INTERNET

- AS diversi diversi possono utilizzare algoritmi di routing diversi
- Routing all'interno di un AS Intra-AS Routing:
 - focus sulla performance
 - intra-domain or internal gateway (IGP) routing
 - l'amministratore decide quale protocollo di routing adottare (interior gateway protocol)
 - decisioni locali, non visibili all'esterno
- Routing tra AS diversi
 - focus sulla politica di instradamento
 - inter-domain or external gateway (EGP) routing

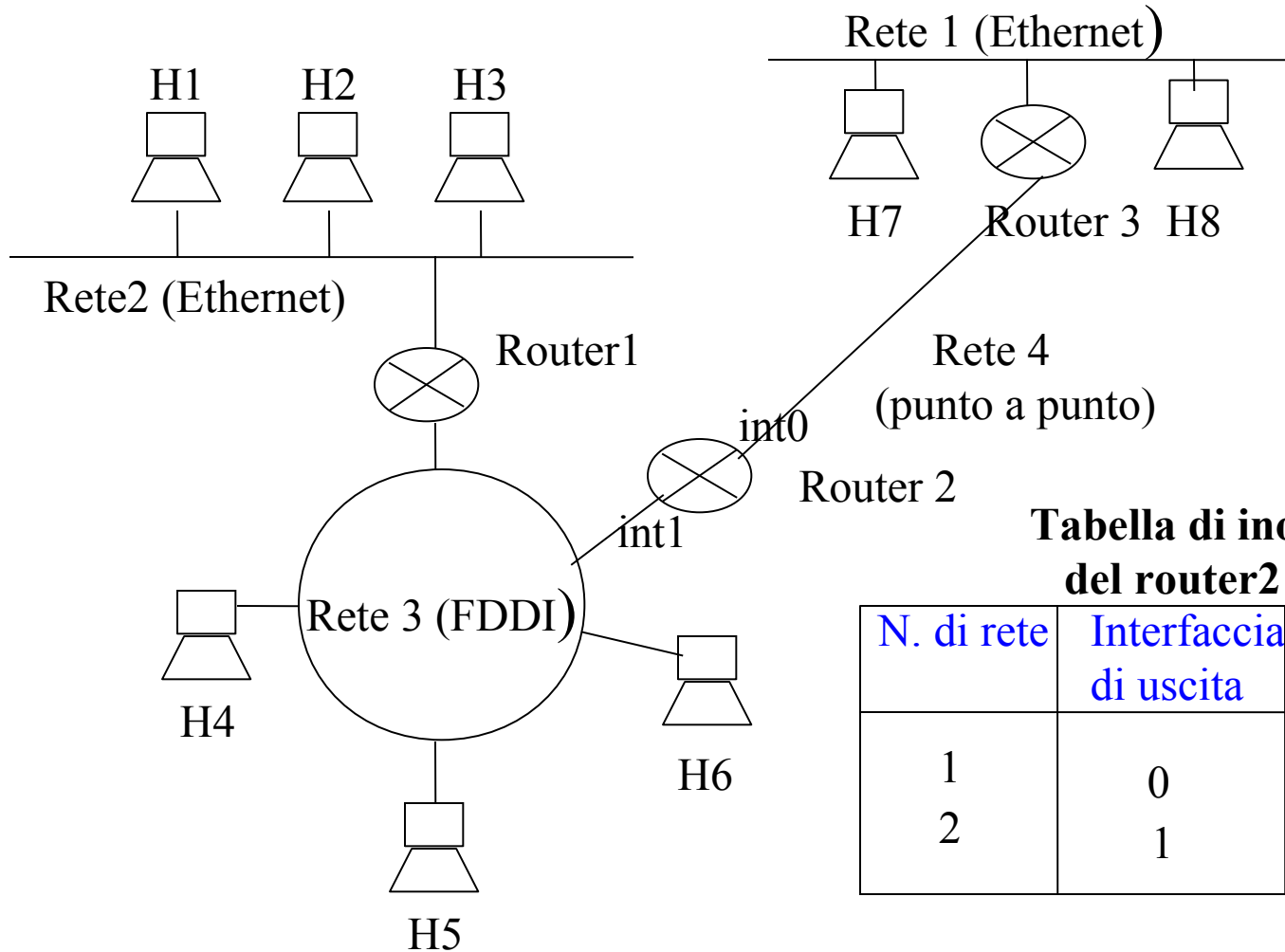
PREREQUISITI: INOLTRO DI PACCHETTI

- routing: meccanismo di inotro utilizzato da routers ed hosts per decidere su quale porta di uscita inviare un pacchetto ricevuto su una porta di ingresso
- il meccanismo di inoltro sfrutta un insieme di **tabelle di inoltro** che contengono un insieme di coppie

(numero di rete, interfaccia di uscita, prossimo router)

- **indirizzamento gerarchico**
 - nelle tabelle dei routers vengono memorizzati solo gli indirizzi delle reti, piuttosto che gli indirizzi dei singoli hosts della rete.
 - il router associato alla rete destinataria invia il messaggio all'host selezionato.
- **indirizzamento gerarchico**= aumento della scalabilità del sistema

TABELLE DI ROUTING



**Tabella di inoltro
del router2**

N. di rete	Interfaccia di uscita	Next router
1	0	Router3
2	1	Router1

INOLTRO DEI PACCHETTI

Algoritmo di inoltro eseguito dal nodo N (host o router)

NetworkAdd = indirizzo di rete contenuto nel pacchetto da inoltrare

- **Networkadd è uguale all' indirizzo di rete di una delle interfacce di N**
in questo caso il destinatario si trova sulla stessa rete del mittente, il pacchetto può essere consegnato direttamente dal livello link
- **Networkadd è contenuto nella tabella di inoltro di N**
in questo caso il pacchetto va consegnato al router **next router**
- **altrimenti**
il pacchetto va consegnato ad un default router
(ad esempio un host può avere un default router a cui consegnare tutti i pacchetti non destinati alla rete locale su cui tale host è connesso)

INOLTRO DEI PACCHETTI

- la trasmissione di un pacchetto IP su una rete fisica avviene mediante il livello link
- ogni nodo connesso ad una rete fisica possiede un indirizzo detto indirizzo fisico o *MAC* (media access control)
- quando un pacchetto IP viene spedito su una rete fisica va utilizzato il *MAC* del nodo destinazione
- traduzione indirizzo IP-MAC address avviene mediante
ARP (Address Resolution Protocol)
- Pacchetto IP incapsulato in un frame a livello link che contiene il *MAC* address del prossimo router o dell'host destinatario

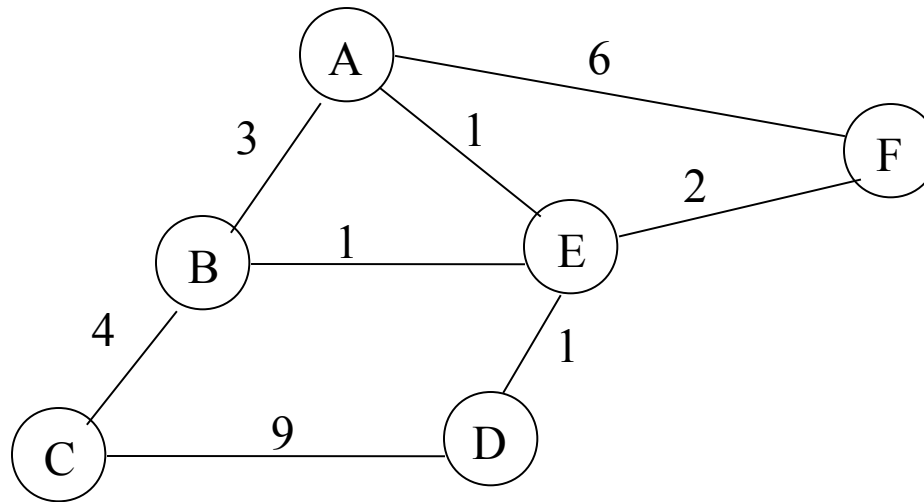
INSTRADAMENTO (ROUTING)

- **Autonomous System Architecture**
 - la rete viene vista come un insieme di gruppi indipendenti
 - ogni gruppo corrisponde ad un sistema autonomo(AS)
 - ogni AS comprende un gruppo di routers gestiti in modo indipendente da una organizzazione
 - il tipo di routing utilizzato all'interno di un sistema autonomo è diverso da quello utilizzato tra sistemi autonomi diversi
- All'interno di un sistema autonomo, si distinguono due tipi di routers
 - **routers interni:** sono connessi unicamente ad altri routers dello stesso sistema autonomo. Eseguono solo **interior routing protocols**
 - **border routers:** sono connessi sia a routers interni che a routers di altri sistemi autonomi. Eseguono **interior ed exterior routing protocols**

INTERIOR ROUTING PROTOCOLS

- **assunzione base:** la rete è rappresentata come un grafo, in cui i nodi rappresentano routers, gli archi linee fisiche di collegamento
- Ad ogni arco è associato un valore che indica il **costo** dell'invio di un pacchetto su quel link. Nel caso più semplice i costi sono unitari ed il percorso di costo minimo è quello che attraversa il minor numero di routers
- **Assegnazione dei costi ai link:** possono essere assegnati in base
 - traffico sulla linea
 - alla **latenza** delle linea
 - alla **banda** delle linea

MODELLARE UNA RETE



Problema: dato un nodo mittente N1 ed un nodo destinatario N2, trovare il cammino di costo minimo tra N1 ed N2 (costo di un cammino= somma dei costi associati agli archi che definiscono il cammino)

INTERIOR ROUTING PROTOCOLS

Algoritmi di routing intradominio: tutti gli algoritmi vengono eseguiti in **modo distribuito** = ogni nodo esegue lo stesso algoritmo, i nodi cooperano scambiandosi informazioni sulla topologia della rete e sui costi associati ai links

- **Algoritmi di routing con conoscenza globale:** ogni nodo deve avere conoscenza del grafo dell'intera rete e dei costi associati ai links.

Algoritmi basati sullo **Stato delle Linee (OSPF)**

- **Algoritmi di routing decentralizzati:** ogni nodo ha solo conoscenza dei costi dei costi relativi ai links che lo collegano ai nodi vicini

Algoritmi basati su **vettori distanza (RIP)**

CLASSIFICAZIONE PROTOCOLLI DI ROUTING

Classificazione dei protocolli di routing:

- interior routing protocols: utilizzati all'interno di un unico sistema autonomo
 - RIP (Routing Information Protocol)
 - OSPF (Open shortestest Path First)
 - IS-IS
 - EIGRP
 - Protocolli proprietari
- Exterior gateway protocols : utilizzati tra sistemi autonomi diversi
 - BGP



CLASSIFICAZIONE ALGORITMI DI ROUTING

- **Distance-Vector:**

- vettori (destinazione-distanza) inviati ai vicini
 - “Tell your neighbors about the rest of the network”
- distanza definita in termini di diverse metriche: hop count, delay, banda
- utilizza **Distributed Bellman-Ford path selection algorithm**
- protocolli: Routing Information Protocol (RIP)

- **Link-State**

- flooding della descrizione dei propri links (link state)
 - “Tell the rest of the network about your neighbors”
- usa l'algoritmo di **selezione dei path di Dijkstra**
- protocol: Open Shortest Path First (OSPF)

OVERVIEW

Bellman-Ford shortest path algorithm

- Distributed Bellman-Ford (DBF) e Routing Information Protocol (RIP)
- Uso di algoritmi shortest-path algorithms in reti reali
 - La debolezza principale del DBF: counting-to-infinity

Dijkstra algorithm

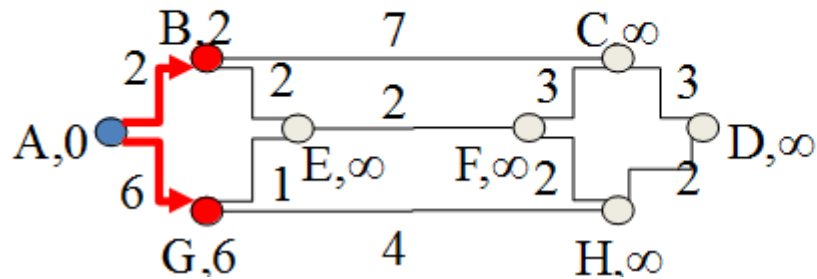
- Dijkstra e OSPF

L'ALGORITMO DI BELLMANN-FORD

- consideriamo un nodo A
- il nodo ha raccolto informazioni su tutti gli altri nodi e collegamenti della rete
- il nodo modella la rete **mediante un grafo**
- Bellman Ford:
 - A esegue l'algoritmo per calcolare la sua distanza da qualsiasi altro nodo della rete
 - quindi costruisce la sua tabella di routing che indica, per ogni altro nodo N della rete, il vicino verso cui inoltrare il messaggio per raggiungere N tramite un cammino minimo

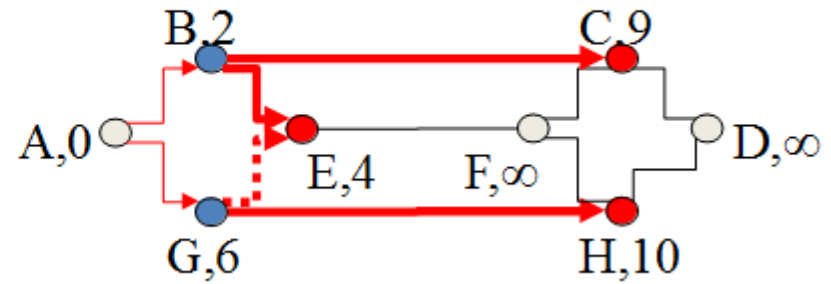
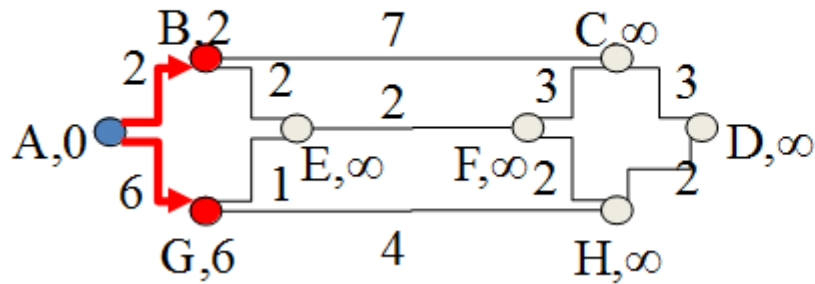
L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, Dashed = rejected



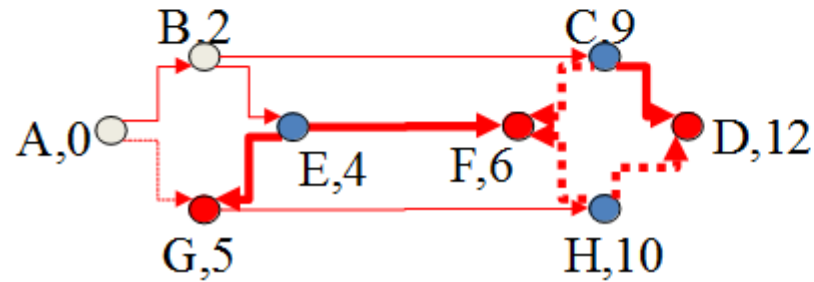
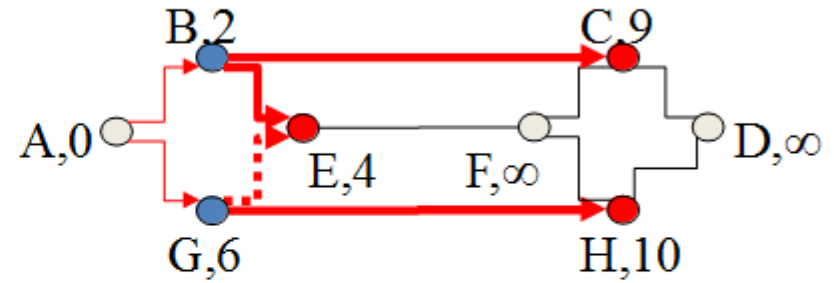
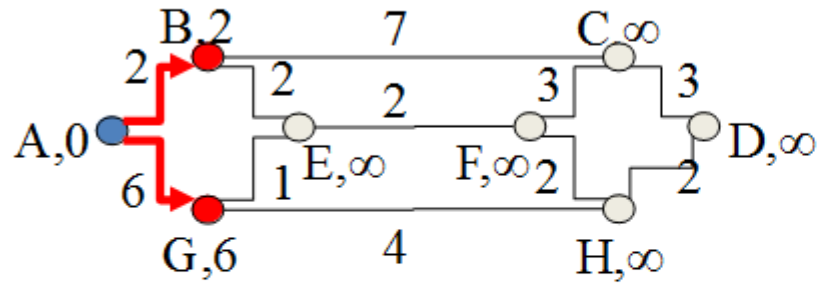
L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, **Dashed** = rejected



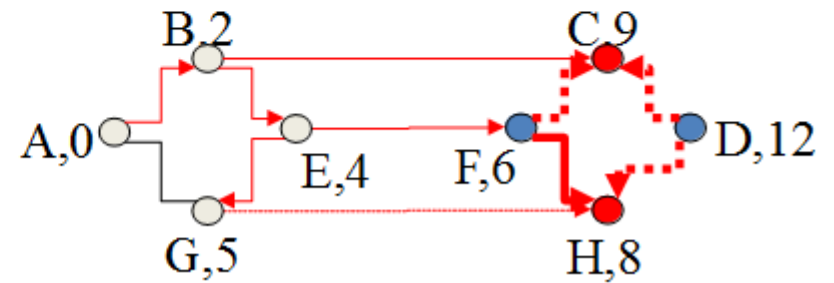
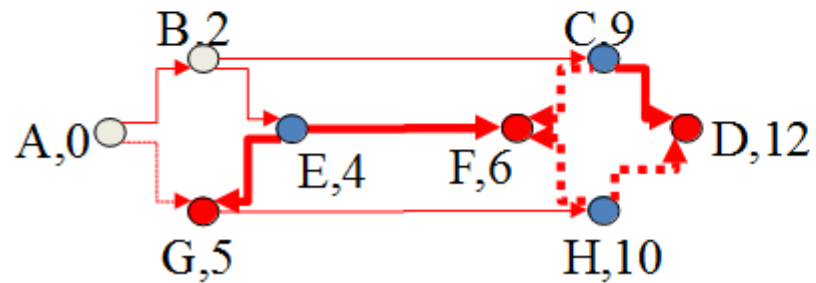
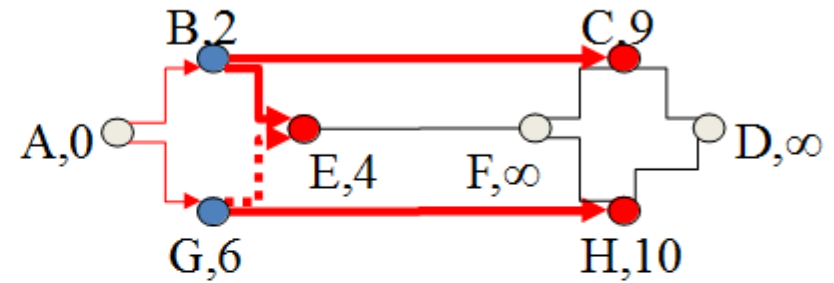
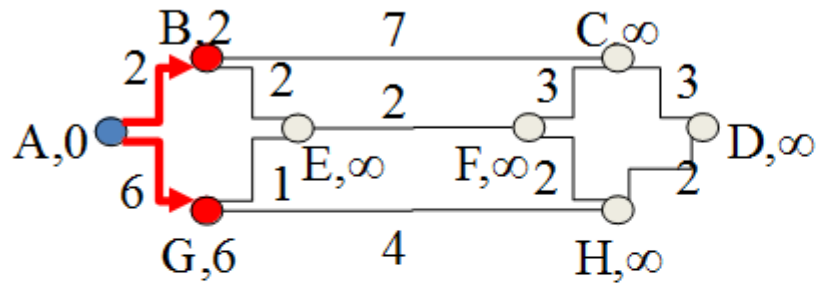
L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, **Dashed** = rejected



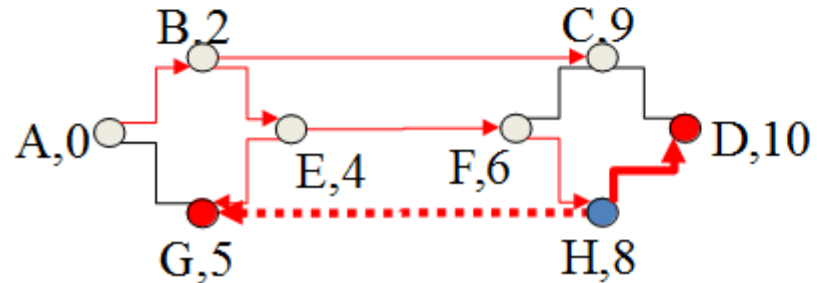
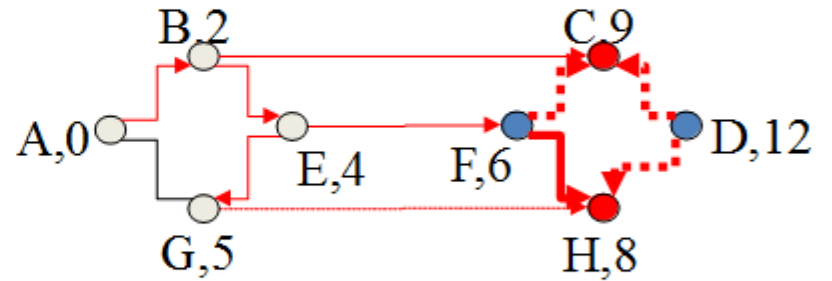
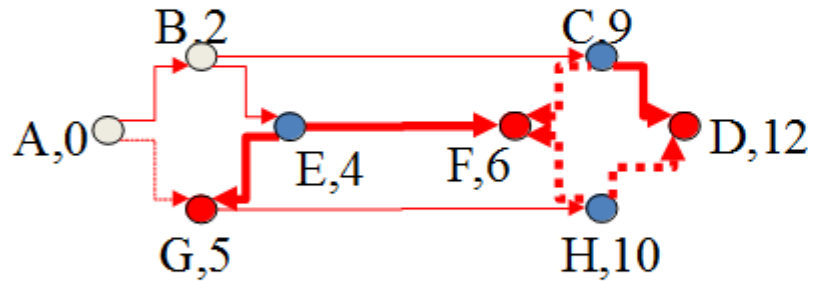
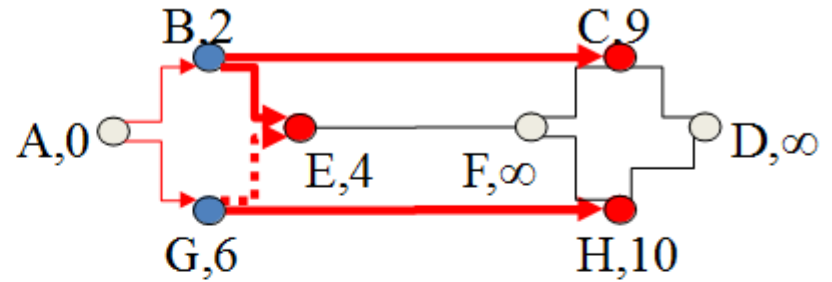
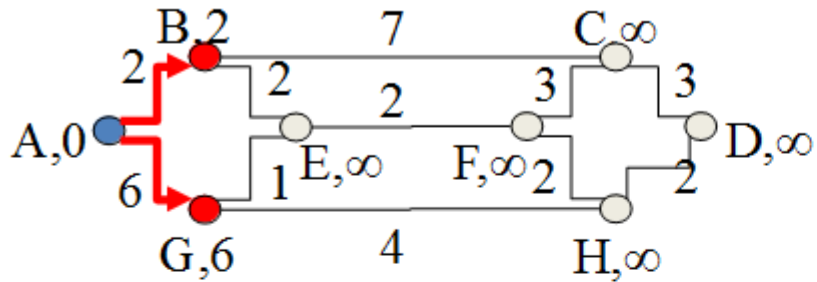
L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, **Dashed** = rejected



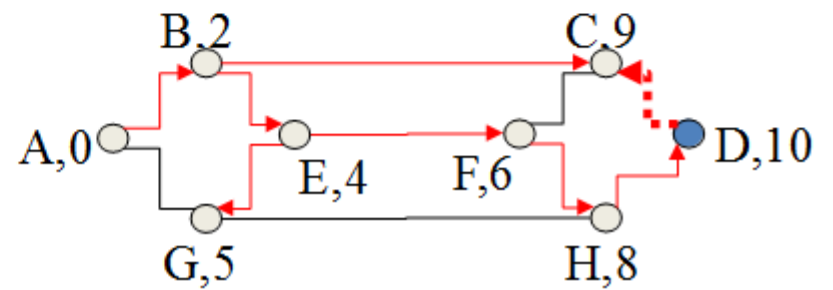
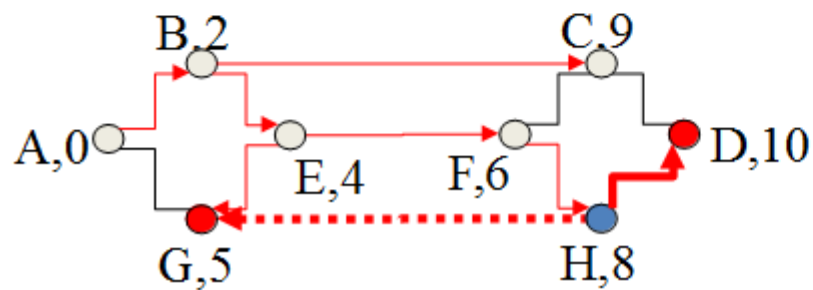
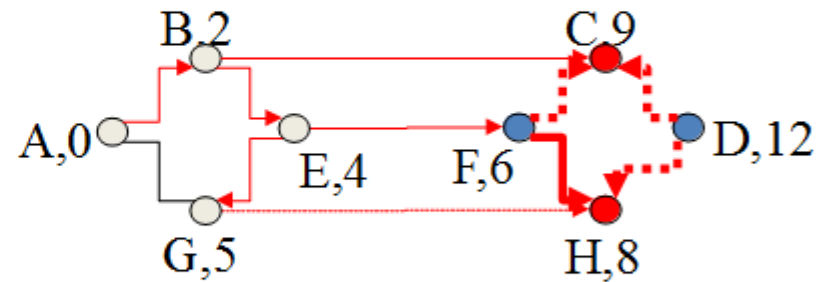
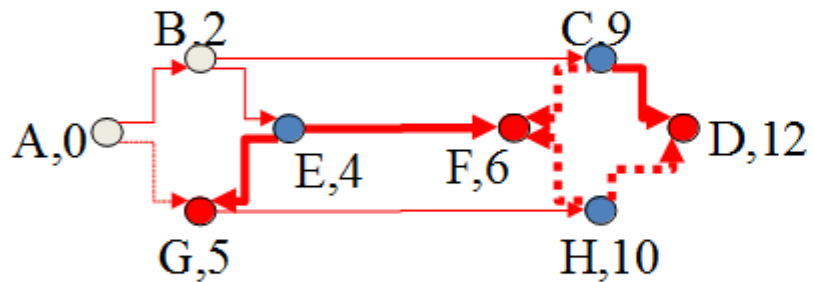
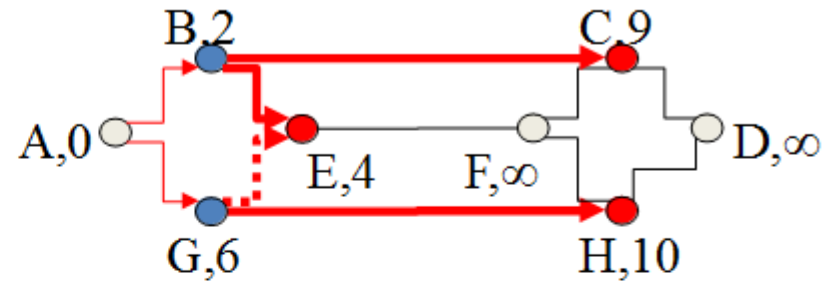
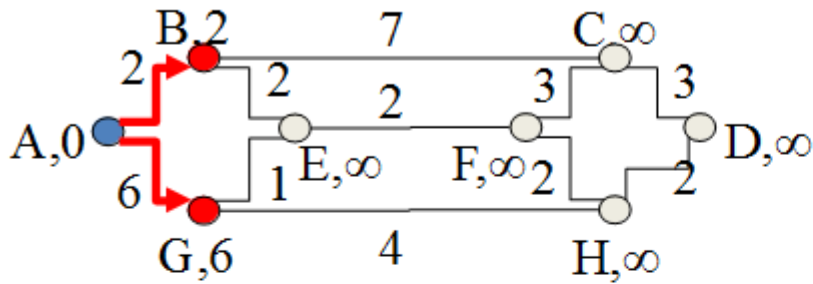
L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, **Dashed** = rejected



L'ALGORITMO DI BELLMAN-FORD

Bold = this iteration, Dashed = rejected



L'ALGORITMO DI BELLMAN-FORD

- Idea generale:
 - ricerca **breadth-first** dei cammini, incremento del numero di hops, ricerca dei cammini migliori dal nodo sorgente a tutte le destinazioni
 - termina quando non è possibile trovare un miglioramento
- L'algorithm in breve:
 - R: insieme dei nodi per cui è stato trovato un cammino
 - passo iterativo:
 - analizza tutti i nodi vicini di R, sia questo insieme R'
 - se si individua un nodo n in R' per cui un cammino tramite uno dei nodi in R è migliore del cammino migliore attuale associato al nodo n, si aggiorna il cammino di n in R
 - ripeti il passo iterativo viene ripetuto fino a che nessun cammino dei nodi in R può essere migliorato
- alla fine da R si ricava la **routing table**

L'ALGORITMO DI BELLMAN-FORD

algorithm CBF run at node i

begin

1 $d_i \leftarrow 0; p_i \leftarrow i;$

2 for each $\{x \in V - \{i\}\};$

3 $d_x \leftarrow \infty; p_x \leftarrow \emptyset;$

4 $R \leftarrow \{i\};$

do

5 $R' \leftarrow \{j \mid x \in R, (x, j) \in E, d_j > d_x + \omega_{xj}\};$

6 for each $\{j \in R'\}$

begin

7 $d_j \leftarrow \text{Min}\{d_x + \omega_{xj} \mid x \in R', (x, j) \in E\};$

8 $p_j \leftarrow \{x \mid (x, j) \in E, d_x + \omega_{xj} = d_j\};$

end

9 $R \leftarrow R \cup R';$

10 until $(R' = \emptyset)$

end

V insieme di vertici

E insieme di archi

w_{ij} peso del link tra i nodi i e j

R insieme dei nodi per cui si è trovato un cammino

R' insieme dei nodi vicini dei nodi in R

d_j distanza dal nodo i al nodo j

p_j predecessore del nodo j nel cammino che parte da i

FIGURE 1. Centralized Bellman-Ford Algorithm.