



Corso di
“Reti di Calcolatori e Comunicazione Digitale”

Modulo 4 : TCP/IP : Il routing

*Prof. Sebastiano Pizzutilo
Dipartimento di Informatica*

Il Routing

Il routing consiste di due attività fondamentali:

*1) Determinazione del percorso ottimale dei messaggi (**CALCOLO DEL PERCORSO**).*

Centrale nella determinazione del percorso ottimale è il modo in cui si creano e si aggiornano le tabelle di routing. I protocolli di ROUTING svolgono questo tipo di attività.

*2) Trasporto delle informazioni tra 2 reti diverse (**INOLTRO**).*

*L'**INOLTRO** dei pacchetti verso l'host destinazione avviene secondo modalità standard:*

- Acquisito l'indirizzo IP di un pacchetto da inoltrare, il router controlla, attraverso la propria netmask, se è relativo ad un host della propria rete;
- Se l'indirizzo appartiene alla stessa rete del router, l'IP del router utilizzerà i servizi dello strato inferiore (data-link) per spedire il pacchetto direttamente all'host destinatario.
- Se l'indirizzo appartiene ad un'altra rete, il router consulterà la propria Routing Table, che associa ad ogni rete l'indirizzo del router di frontiera delle reti connesse.
- Se il router (che è collegato a più reti) ha una connessione alla rete dove è collegato l'host destinatario, gli inoltra direttamente il pacchetto, altrimenti lo passa al router più vicino, fino a raggiungere un router in grado di consegnare il pacchetto.

Metriche di routing

Per selezionare il miglior percorso gli algoritmi scelgono diversi criteri di scelta.

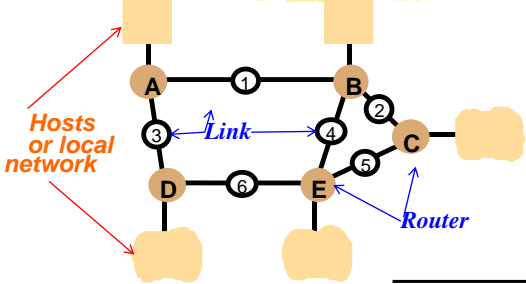
Due parametri (metriche) universalmente accettati sono:

- ✓ **Hops:** numero di salti effettuati, cioè il numero di IS attraversati lungo il cammino,
- ✓ **Costo:** somma dei costi di tutte le linee attraversate; il costo di una linea è inversamente proporzionale alla sua velocità (banda trasmissiva, tipo e affidabilità del mezzo trasmissivo, lunghezza del percorso, traffico di rete,...)

Distance Vector Routing

Gli algoritmi del tipo *distance vector* effettuano l'invio da parte di ciascun router della propria tabella ai soli router vicini.

La tabella inviata ha l'aspetto di un vettore (ad es. il router A invia sul link 1 a B il vettore "A=0,0, B=1,1, C=1,2, D=3,1, E=1,2")



Routing from A			Routing from B			Routings from C			Routings from D			Routings from E		
To	Link	Cost	To	Link	Cost	To	Link	Cost	To	Link	Cost	To	Link	Cost
A	local	0	A	1	1	A	2	2	A	3	1	A	4	2
B	1	1	B	local	0	B	2	1	B	3	2	B	4	1
C	1	2	C	2	1	C	local	0	C	6	2	C	5	1
D	3	1	D	1	2	D	5	2	D	local	0	D	6	1
E	1	2	E	4	1	E	5	1	E	6	1	E	local	0

ALGORITMO DI BELLMAN-FORD

I protocolli **distance vector** sono basati sull'algoritmo di **Bellman-Ford**, che e' una **versione distribuita** di un algoritmo molto semplice per la ricerca del percorso più breve.

- ☞ N = numero dei nodi, M = numero dei links.
- ☞ L = tabella dei link di dimensione M , dove : $L[l].m$ = **metrica del link**,
 $L[l].s$ = **sorgente del link**, $L[l].d$ = **destinazione del link**.
- ☞ D = matrice delle distanze di dimensione $[N,N]$, dove $D[i,j]$ e' la **distanza da i a j**
- ☞ H = matrice dei link di dimensione $[N,N]$, dove $H[i,j]$ e' il **link** sul quale i instrada i pacchetti destinati a j .

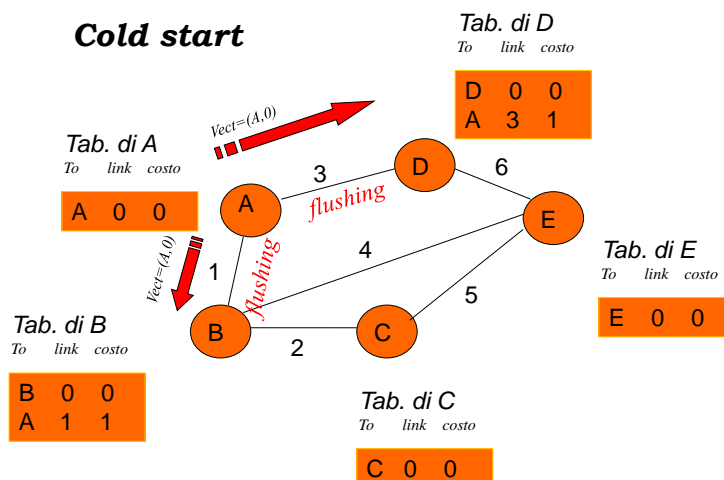
Algoritmo :

- 1) Se $i = j$ allora tutti $D[i,j] = 0$, altrimenti $D[i,j] = \infty$.
- 2) Inizializza tutti gli $H[i,j] = -1$.
- 3) Per tutti i link l e per tutte le destinazioni k : $i = L[l].s$; $j = L[l].d$;
- 4) Calcola per tutti i link l $Dist = L[l].m + D[j,k]$.
- 5) Se $Dist < D[i,k]$, aggiorna $D[i,k] = Dist$; $H[i,k] = l$
- 6) Se almeno un $D[i,k]$ e' stato aggiornato, ripeti lo step 3, altrimenti stop.

Università di Bari Aldo Moro - Cdl. ICD - corso di Reti - A.A.2011-2012

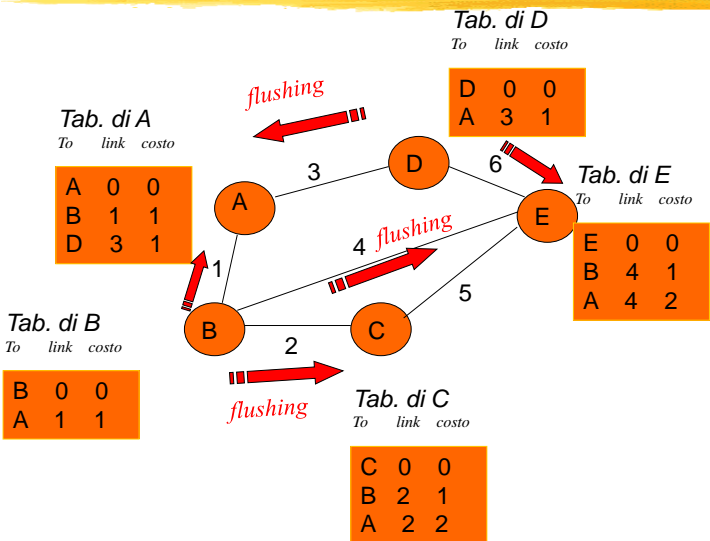
Distance vector start up

Cold start



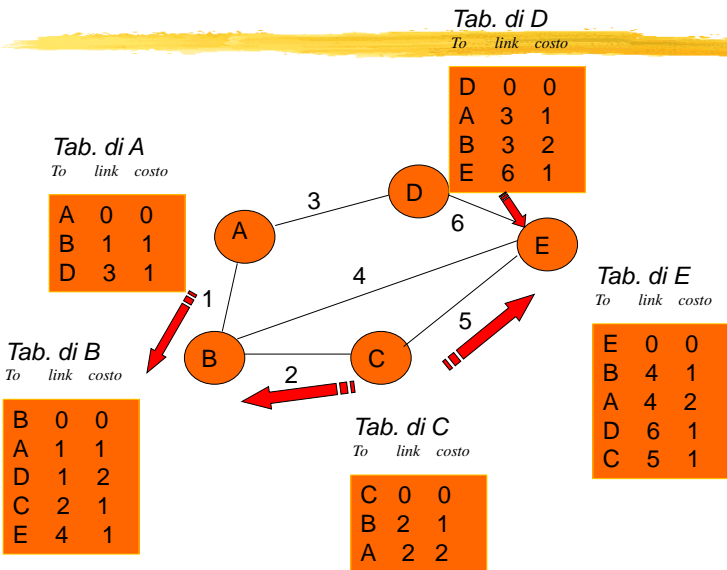
Università di Bari Aldo Moro - Cdl. ICD - corso di Reti - A.A.2011-2012

D.V. Start up : Flushing 1



Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

D.V. Start up: Flushing 2



Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

D.V. Start up: Flushing 3

Tab. di A

To	link	costo
A	0	0
B	1	1
D	3	1
C	1	2
E	1	2

Tab. di B

To	link	costo
B	0	0
A	1	1
D	1	2
C	2	1
E	4	1

Tab. di D

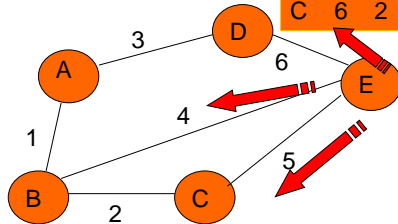
To	link	costo
D	0	0
A	3	1
B	3	2
E	6	1
C	6	2

Tab. di E

To	link	costo
E	0	0
B	4	1
A	4	2
D	6	1
C	5	1

Tab. di C

To	link	costo
C	0	0
B	2	1
A	2	2
E	5	1
D	5	2



... a questo punto
l'algoritmo converge !!!

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

Problema della instabilità : ad es. crash di un ramo (tra A e B)

Tab. di A

To	link	costo
A	0	0
B	1	∞
D	3	1
C	1	∞
E	1	∞

Tab. di B

To	link	costo
B	0	0
A	1	∞
D	1	∞
C	2	1
E	4	1

Tab. di D

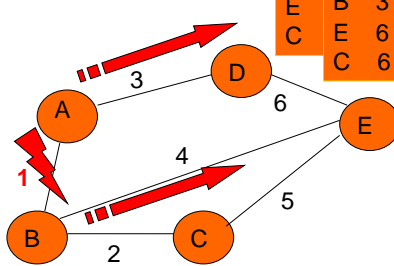
To	link	costo
D	0	0
A	3	1
B	3	∞
E	6	1
C	6	2

Tab. di E

To	link	costo
E	0	0
B	4	1
A	4	∞
D	6	1
C	5	1

Tab. di C

To	link	costo
C	0	0
B	2	1
A	2	∞
E	5	1
D	5	2



Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

Flushing del crash, 1

Tab. di A

To	link	costo
A	0	0
B	A	0
D	B	1
C	D	3
E	C	3
E	E	3

Tab. di B

To	link	costo
B	0	0
A	B	0
D	A	1
C	D	4
E	C	2
E	E	4

Tab. di D

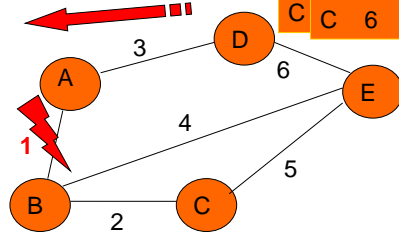
To	link	costo
D	0	0
A	D	3
B	A	6
E	B	6
C	E	6

Tab. di E

To	link	costo
E	0	0
B	E	0
A	B	4
D	A	6
C	D	6
C	C	5

Tab. di C

To	link	costo
C	0	0
B	C	2
A	B	2
E	A	5
D	E	5



Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

Flushing del crash, 2

Tab. di A

To	link	costo
A	0	0
B	A	0
D	B	3
C	D	3
E	C	3
E	E	3

Tab. di B

To	link	costo
B	0	0
A	B	0
D	A	4
C	D	4
E	C	2
E	E	4

Tab. di D

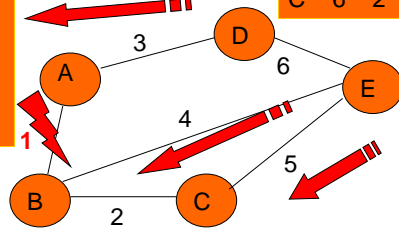
To	link	costo
D	0	0
A	D	3
B	A	6
E	B	6
C	E	6

Tab. di E

To	link	costo
E	0	0
B	E	0
A	B	4
D	A	6
C	D	6
C	C	5

Tab. di C

To	link	costo
C	0	0
B	C	0
A	B	2
E	A	5
D	E	5
D	D	5



**Connettività
ripristinata !!....**

Sicuro ?

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

count-to-infinity problem

L'algoritmo di Bellman-Ford NON risolve il problema della possibilità di **routing loop** ("count-to-infinity problem").

Supponiamo di avere la rete A-B-C-D-E-F del lucido precedente, con la metrica degli hop:

Se A è *out of order*, B non riceve il vettore delle distanze da A.
Ma B riceve da E, che non sa ancora che A è *down*, l'informazione che A è raggiungibile con due hop ($E - B - A$), ma questo è falso.

Questo tipo di propagazione del vettore rallenta il processo fino al raggiungimento del valore massimo consentito degli hop (infinito).

Il tempo di convergenza aumenta così fino alla possibilità di formare **bouncing effect** o un **loop**.

Il problema che sta alla base del **distance vector** è il fatto che nessuno conosce la topologia della rete e questo limita la precisione.

Split horizon e poison reverse

Per cercare di risolvere i problemi di loop sono stati introdotti nell'algoritmo base di distance vector alcune soluzioni:

- **split horizon**, principio secondo cui un nodo non può raggiungere una destinazione passando per il nodo precedente. In questo modo però non si risolvono i loop complessi (cioè quelli che coinvolgono maglie di router), ma solo i loop che coinvolgono due router.

- **split horizon with poisonous reverse**, principio secondo il quale le destinazioni (quelle raggiungibili passando per il nodo a cui si sta inviando il distance vector) vengono messe a costo infinito. In questo modo si risolve l'ambiguità legata alla non ricezione di un distance vector.

- **route poisoning**, In presenza di un count to infinity, si è notato che il costo verso una certa destinazione cresce progressivamente. L'idea è quella di bloccare l'utilizzo di tutte le route che aumentano in modo progressivo di costo. La route verrà rimessa in servizio solo quando due annunci consecutivi confermeranno la presenza della route.

RIP (Routing Information Protocol) (1)

E' un protocollo relativamente semplice appartenente alla famiglia di protocolli di tipo **"distance vector"**.

E' il protocollo di routing IP più vecchio ancora in uso : esistono due versioni, la seconda versione aggiunge nuove funzionalità a questo protocollo. **RIPv1** è di tipo **classfull** mentre **RIPv2** è **classless**.

- I processi RIP utilizzano la porta **520** sia per la trasmissione che per la ricezione.
- Gli indirizzi presenti nelle tabelle RIP sono **indirizzi Internet a 32 bit**.
- Una voce (entry) nella tabella di routing può rappresentare un host, una rete o una sottorete. Non sono presenti specifiche sul tipo di indirizzo nei pacchetti RIP; e' compito dei router analizzare l'indirizzo per capire di cosa si tratta.
- Di default, RIP utilizza una **metrica** molto semplice: la distanza (**hop count**) e' il numero di links che vengono attraversati per raggiungere la destinazione. Questa distanza e' espressa come **un numero intero variabile tra 1 e 15. Il valore 16 rappresenta una distanza infinita**.

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

RIP (Routing Information Protocol) (2)

- Normalmente i pacchetti vengono inviati in **modalità broadcast** ogni **30 secondi**, o meno nel caso di aggiornamenti alle tabelle. Se una route non viene aggiornata entro 3 minuti, la distanza viene fissata ad infinito e l'entry verrà successivamente rimossa dalle tabelle.
- Il processo RIP, in seguito alla ricezione di un messaggio di risposta, aggiorna la propria tabella. Ogni voce della tabella sarà al limite composta da :
 - a) Indirizzo di destinazione b) Metrica associata con la destinazione
 - c) Indirizzo del "next router" d) Un "recently updated" flag
- e) **Numerosi timers :**
 - ✓ **Routing update timer** (default 30 s): intervallo di tempo per l'invio degli annunci
 - ✓ **Route invalid timer** (default 90 s): intervallo di tempo dopo il quale una route è dichiarata irraggiungibile (distanza posta ad infinito)
 - ✓ **Route flush timer** (default 270 s): intervallo di tempo dopo il quale la route è cancellata dalla routing table
 - ✓ **Triggered updates**: sono inviate con un ritardo casuale compreso tra 1 e 5 secondi.

Routing Information Protocol (3)

- Processando le risposte in arrivo, il router **esaminerà le voci una ad una** ed eseguirà una serie di **check**, ad esempio verificherà che l'indirizzo sia valido ed appartenente ad una delle classi A, B o C, e che la metrica non sia maggiore di infinito.
- **Se la metrica in arrivo risulta minore di infinito (16)**, viene incrementata di **1** per il successivo hop, quindi la tabella di routing viene scandita per una voce corrispondente alla destinazione e viene eseguito il generico **processo "distance vector"**:
 - a) *Se la voce non è presente e la sua metrica nel messaggio ricevuto non è infinito, la aggiunge alla tabella, inizializzando la metrica al valore ricevuto ed il next router al mittente del messaggio, prima di avviare un timer per quella voce.*
 - b) *Se la voce è presente con una metrica più grande, aggiorna i campi della metrica e del next router e riavvia il timer per quella voce.*
 - c) *Se la voce è presente ed il next router è il mittente del messaggio di risposta, aggiorna la metrica se questa differisce dal valore memorizzato e riavvia il timer.*
 - d) *In tutti gli altri casi, il messaggio ricevuto è ignorato.*
- **Se la metrica o il next router cambiano**, l'entry viene marcata come "aggiornata". Un messaggio di risposta viene inviato ad intervalli regolari di 30 secondi o può essere attivato in seguito ad un aggiornamento alle tabelle di routing.

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

IGRP (Interior Gateway Router Protocol) (1)

E' un protocollo di routing sviluppato dalla CISCO a metà '80.
Agli inizi degli anni '90, e' stata sviluppata la versione "Enhanced " (EIGRP).

Anche questo protocollo si basa sull'algoritmo **distance vector**, ma con alcuni aspetti che lo rendono più efficiente rispetto al RIP.

In particolare, IGRP è caratterizzato da:

- **Metriche più sofisticate**
- **Supporto del multipath routing**
- **Migliore stabilità**
- ✓ IGRP utilizza una **frequenza di update (90 s)** delle **route table inferiore** al RIP.
- ✓ Il protocollo IGRP permette il routing all'interno dell'Autonomous System (l'AS è identificato da un intero su 16 bit).
- ✓ A differenza di RIP, che permette una sola entry per ogni destinazione, il protocollo IGRP permette la gestione di più entry nella routing table per la stessa destinazione. Il carico può essere ripartito tra le diverse route in funzione della metrica associata.

Metriche di IGRP (2)

A differenza del protocollo RIP, dove l'unica metrica possibile è l'"hop count" e si ha un limite massimo di 15 hop count, in IGRP le metriche si basano su 4 parametri:

B - Bandwidth (1 - 224) (1 = 1.2 kbit/s) Il valore effettivo corrisponde al "numero di slot temporali da 10ms necessari per trasmettere 10000 bit"

D - Delay (1 - 224) (1 = 10 ms)

R - Reliability (1 - 255) (255 = 100%)

L - Load (1 - 255) (255 = 100%)

La variazione dei coefficienti permette di privilegiare determinati parametri a scapito di altri (es. banda piuttosto che ritardo)

Bandwidth e **Delay** sono per default associati al tipo di portante fisico.

Ad esempio:

Ethernet -> $B=10000$, $D=100$ CDN 64 kbit/s -> $B=64$, $D=2000$

Per ciascun link B e D possono comunque essere impostati dal gestore.

Viceversa, i parametri **R** e **L** possono essere difficili da quantificare e possono variare con una frequenza piuttosto elevata.

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

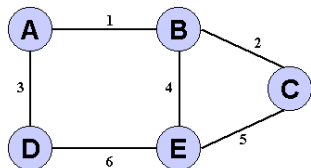
Link State Routing (1)

Gli algoritmi basati sullo **stato dei collegamenti** inviano **tutta la mappa** della rete **a tutti i router**.

Centrale quindi è il problema della **costruzione della mappa della rete** (mediante **l'invio in flooding di pacchetti** di dati sullo stato dei link sulla rete: **LSP**) e del **calcolo del percorso migliore** attraverso un algoritmo di scelta del percorso più breve (**algoritmo di Dijkstra**).

*Il principio alla base dell'instradamento di tipo **Link State** è molto semplice: invece di calcolare i percorsi migliori in modo distribuito, tutti i nodi mantengono una copia intera della mappa della rete ed eseguono un calcolo completo (algoritmo di Dijkstra) dei migliori percorsi da questa mappa locale.*

La mappa è contenuta in un database, dove ciascun record rappresenta un link nella rete.



Da	A	Link	Distanza
A	B	1	1
A	D	3	1
B	A	1	1
B	C	2	1
B	E	4	1
C	B	2	1
C	E	5	1
D	A	3	1
D	E	6	1
E	B	4	1
E	C	5	1
E	D	6	1

Link State Packet (2)

*I pacchetti che sono inviati da un router e che permettono la costruzione della mappa della rete sui vari router, sono detti **Link State Packet (LSP)** che contengono:*

1. Stato di ogni link connesso al router,
2. Identità di ogni vicino connesso all'altro estremo del link (sulle LAN possono esserci migliaia di vicini),
3. Costo del link,
4. Numero di sequenza per il LSP (a seguito di frequenti variazioni di topologia un router può ricevere un LSP vecchio dopo uno nuovo, quindi deve essere in grado di valutare il più recente),
5. Checksum,
6. Lifetime (la validità di ogni LSP è limitata nel tempo, diversamente un errore sul numero di sequenza potrebbe rendere un LSP valido per anni).

*Un LSP viene **generato periodicamente**, oppure **quando viene rilevata una variazione nella topologia locale (adiacenze)**, ossia :*

- a) Viene riconosciuto un nuovo vicino,
- b) Il costo verso un vicino e' cambiato,
- c) Si e' persa la connettività verso un vicino precedentemente raggiungibile.

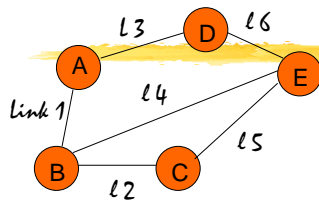
Il LSP è trasmesso in flooding su tutti i link del router e tutti i router del dominio di routing ricevono il LSP.

Link State Packet (3)

All'atto del ricevimento di un **LSP**, il router compie le seguenti azioni:

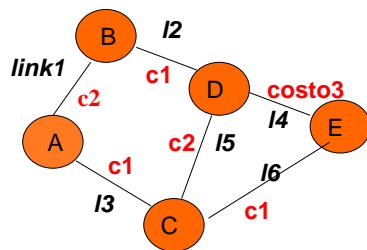
- a) *Se non ha mai ricevuto LSP da quel router o se il LSP è più recente di quello precedentemente memorizzato (campo Sequence Number), memorizza il pacchetto e lo ritrasmette in flooding su tutte le linee eccetto quella da cui l'ha ricevuto;*
- b) *Se il LSP ha lo stesso numero di sequenza di quello posseduto non fa nulla;*
- c) *Se il LSP è più vecchio di quello posseduto trasmette al mittente il pacchetto più recente.*

Tabella del link state



from to Link costo

A	B	1	1
A	D	3	1
B	A	1	1
B	C	2	1
B	E	4	1
C	B	2	1
C	E	5	1
D	A	3	1
D	E	6	1
E	B	4	1
E	C	5	1
E	D	6	1



from to Link costo

A	B	1	2
A	C	3	1
B	A	1	2
B	D	2	1
C	A	3	1
C	E	6	1
C	D	5	2
D	B	2	1
D	E	4	3
D	C	5	2
E	C	6	1
E	D	4	3

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012

L'algoritmo di DIJKSTRA

L'algoritmo di calcolo del percorso piu' breve ed efficiente (rispetto a quello di Bellman-Ford) è stato definito da **E.W. Dijkstra** con il suo "**Shortest Path First**" (SPF).

L'algoritmo **SPF** calcola il percorso piu' breve tra un nodo sorgente ed un altro nodo della rete, definendo:

- 1 nodo **radice (root)**, ossia il nodo che sta calcolando l'algoritmo,
- 1 insieme **PATH** di nodi (ID, cost, link) **per i quali si è già trovato il percorso migliore**,
- 1 insieme **TEMP** di nodi (ID, cost, link) **per i quali si sta cercando un percorso**.

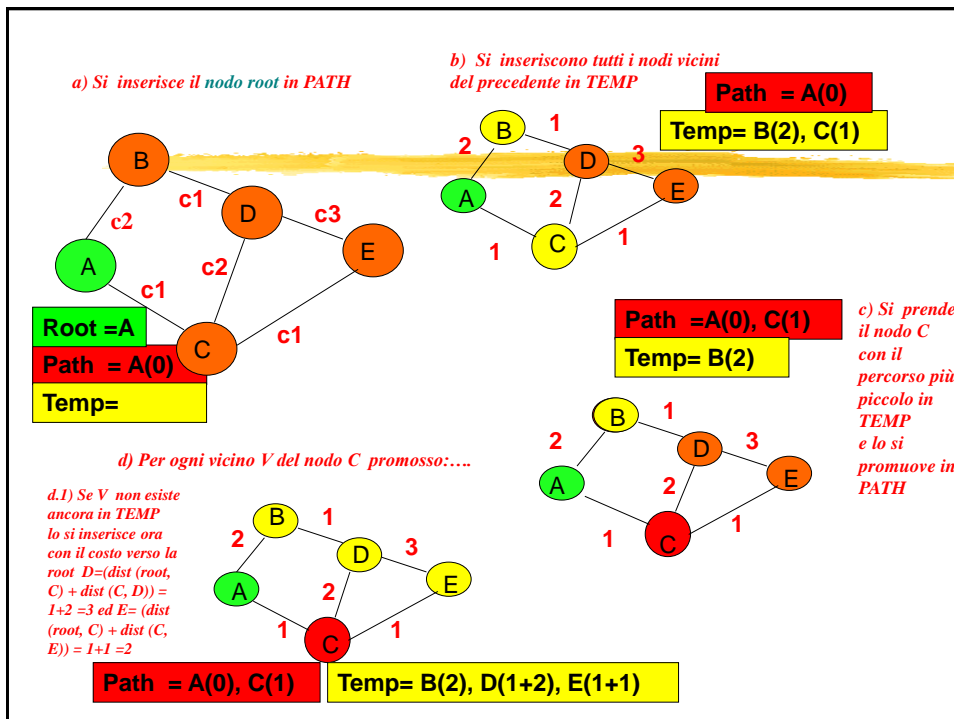
L'Algoritmo di Dijkstra

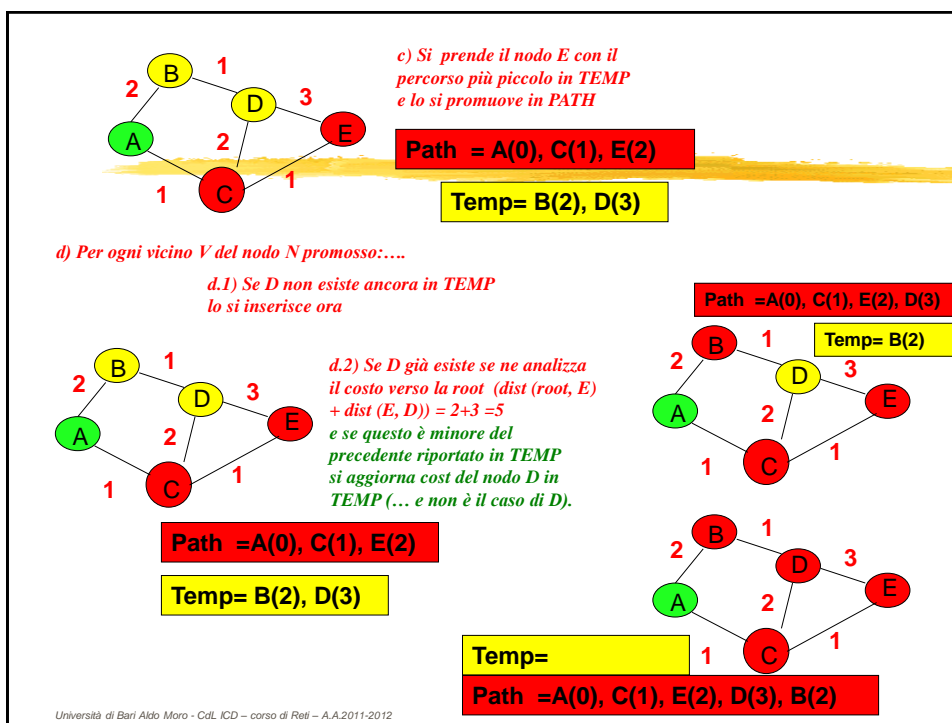
1. Si inserisce il nodo root in **PATH** ;
2. Si inseriscono tutti i nodi vicini del precedente in **TEMP** ;
3. Si prende il **nodo N** con il percorso più piccolo in **TEMP** e lo si promuove in **PATH** ;
4. Per ogni vicino V del nodo N promosso :
 Se V **non** esiste ancora in **TEMP** lo si inserisce ora con il costo **Dist** verso la root ($\text{Dist}(\text{root}, N) + \text{Dist}(N, V)$)
 Se V **già** esiste se ne analizza il costo **Dist** e se questo è minore del precedente riportato in **TEMP** si aggiorna cost e link di quel nodo in **TEMP**.

Principi di funzionamento dell'algoritmo

- Ad ogni nodo è associata una etichetta che rappresenta il costo del cammino migliore trovato per raggiungerlo.
- L'algoritmo etichetta progressivamente i nodi partendo da ciascun nodo ;
- Il prossimo nodo etichettato è quello raggiungibile a costo più basso a partire da un nodo già etichettato.
- L'algoritmo termina quando a tutti i nodi è stata associata un'etichetta.

Università di Bari Aldo Moro - CdL ICD - corso di Reti - A.A.2011-2012





OSPF (Open Shortest Path First) (1)

Questo protocollo di routing si basa sulla tecnologia Link State :

- ✓ Prevede un **database distribuito**, una procedura di **flooding**, una definizione di **adjacency** e record speciali per **route esterne**.
- Il protocollo è basato sullo **stato dei LINK** tra **Autonomous Systems**, che viene comunicato a tutti i router mediante un'operazione di **flooding**. In questa maniera **ciascun router si costruisce una mappa topologica completa dell'intero AS**.
- A questo punto ciascun router mette in funzione localmente l'algoritmo di Dijkstra "Shortest first" per determinare il percorso più breve verso le altre reti, considerando se stesso come il nodo radice.
- ✓ OSPF, utilizzando il concetto di **gerarchia**, permette di gestire reti di dimensioni notevoli.
- ✓ Un AS viene suddiviso in **aree**, le quali contengono un gruppo di reti contigue. Questo garantisce un basso utilizzo di CPU e memoria.
- ✓ OSPF è un protocollo sviluppato dalla IETF, è utilizzato su Internet ed è il protocollo raccomandato dalla IAB in sostituzione di RIP.

OSPF (2)

OSPF e' stato progettato per :

- *separare host e router*

- ✓ Molto spesso, nelle reti moderne, gli host IP sono connessi a reti locali, ad esempio una rete Ethernet. Se si applica strettamente il modello **Link State**, dovremmo descrivere, attraverso un record di **link state** la relazione tra ciascun host ed il router.

- ✓ OSPF permette una semplificazione, basata sul modello "subnet" di IP.

Poiche' tutti gli host della rete appartengono ad una singola sottorete IP, e' sufficiente inviare un messaggio sul link tra il router e la sottorete. Nella terminologia OSPF, questa e' una speciale variante di "router link" chiamata "**link to a stub network**". Il link verso un vicino e' identificato dall'indirizzo IP del vicino stesso, mentre quello verso una "stub network" e' identificato dal suo numero di rete o sottorete.

- *Suddividere grandi reti in aree*

- ✓ OSPF prevede di utilizzare il "routing gerarchico" ossia la suddivisione della rete in un set di parti indipendenti connesse attraverso un "backbone". In OSPF le parti indipendenti sono chiamate aree e la parte superiore e' chiamata "Backbone area".