

Livello di Rete:
protocollo OSPF, protocollo BGP

Prof.ssa Gaia Maselli
maselli@di.uniroma1.it

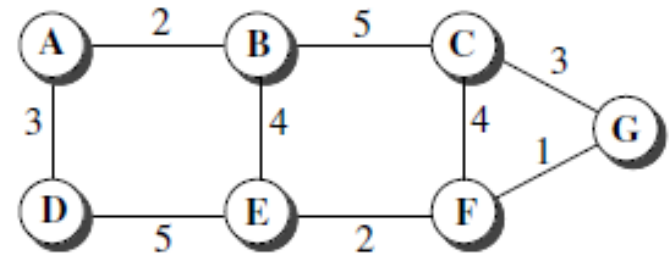
Parte di queste slide sono state prese dal materiale associato ai libri:

- 1) B.A. Forouzan, F. Mosharraf – Reti di calcolatori. Un approccio top-down. Copyright © 2013 McGraw-Hill Education Italy srl. Edizione italiana delle slide a cura di Gabriele D'Angelo e Gaia Maselli
- 2) Computer Networking: A Top Down Approach , 6th edition. All material copyright 1996-2009 J.F Kurose and K.W. Ross, All Rights Reserved

Algoritmo d'instradamento a stato del collegamento (LS)

Algoritmo di Dijkstra:

- La topologia di rete e tutti i costi dei collegamenti sono noti a tutti i nodi
 - attraverso il "link-state broadcast".
 - tutti i nodi dispongono delle stesse informazioni
- Calcola il cammino a costo minimo da un nodo (origine) a tutti gli altri nodi della rete.
 - Crea una **tabella d'inoltro** per quel nodo
- È iterativo: dopo la k -esima iterazione i cammini a costo minimo sono noti a k nodi di destinazione.



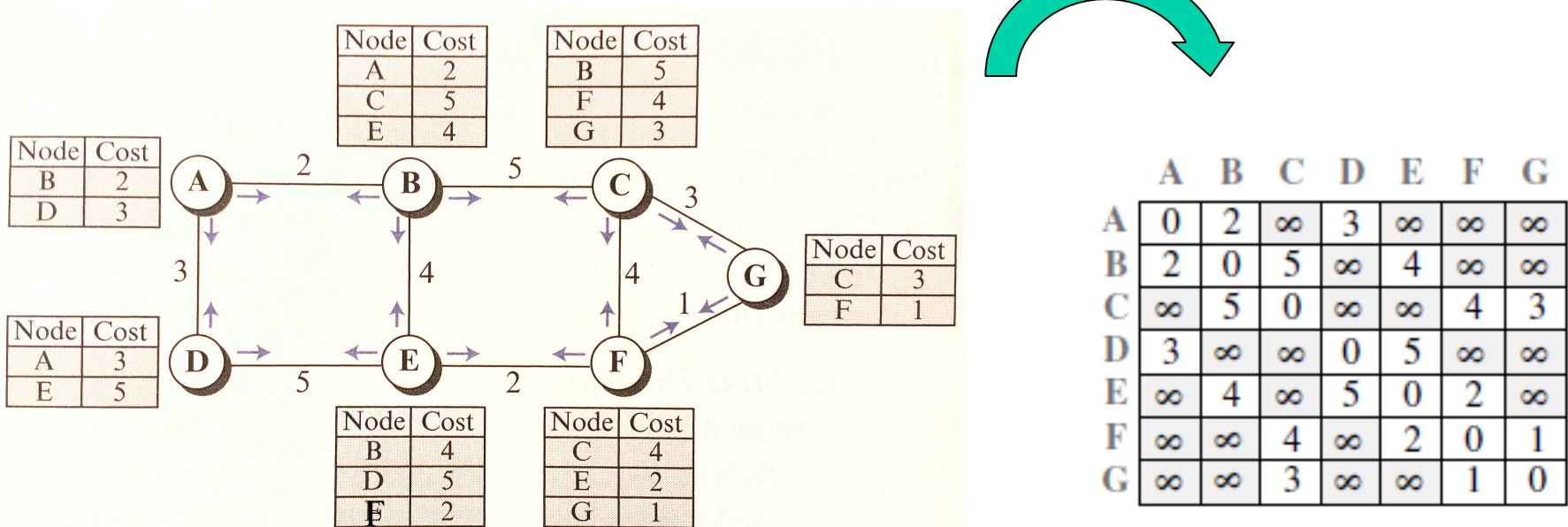
a. Il grafo pesato

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Il link-state database

Link-state database (LSDB)

- ❑ LSDB è unico per tutta la rete
- ❑ Si costruisce mediante flooding
- ❑ Ogni nodo invia messaggi di benvenuto ai propri vicini per scoprire identità del vicino e costo del nodo (creando il LS packet - LSP)
- ❑ Flooding del LSP (invio su ogni interfaccia ad esclusione di quella da cui si è ricevuto il pacchetto)



Costruire la tabella di routing

- ❑ Per costruire il suo albero a costo minimo utilizzando il LSDB ogni nodo deve eseguire l'algoritmo di Dijkstra
- ❑ Ogni nodo sceglie se stesso come radice dell'albero
- ❑ Ogni nodo applica l'algoritmo indipendentemente

Algoritmo di routing a stato del collegamento (LS)

Definiamo la seguente notazione:

- N : insieme dei nodi della rete
- $c(x,y)$: costo del collegamento dal nodo x al nodo y
- $D(v)$: costo del cammino minimo dal nodo origine alla destinazione v per quanto riguarda l'iterazione corrente.
- $p(v)$: immediato predecessore di v lungo il cammino.
- N' : sottoinsieme di nodi per cui il cammino a costo minimo dall'origine è definitivamente noto.

Algoritmo di Dijkstra

- Calcola i percorsi di costo minimo da un nodo u a tutti gli altri nodi della rete
- È iterativo e viene eseguito un numero di volte pari al numero di nodi nella rete

Algoritmo di Dijkstra

1 **Inizializzazione:**

2 $N' = \{r\}$ /* r è il nodo che esegue l'algoritmo (radice dell'albero)*/

3 per tutti i nodi n

4 se n è adiacente a r

5 allora $D(n) = c(r,n)$

6 altrimenti $D(n) = \infty$

7

8 **Ciclo**

9 determina un n non in N' tale che $D(n)$ sia minimo

10 aggiungi n a N'

11 per ogni nodo a adiacente a n e non in N' aggiorna $D(a)$:

12 $D(a) = \min(D(a), D(n) + c(n,a))$

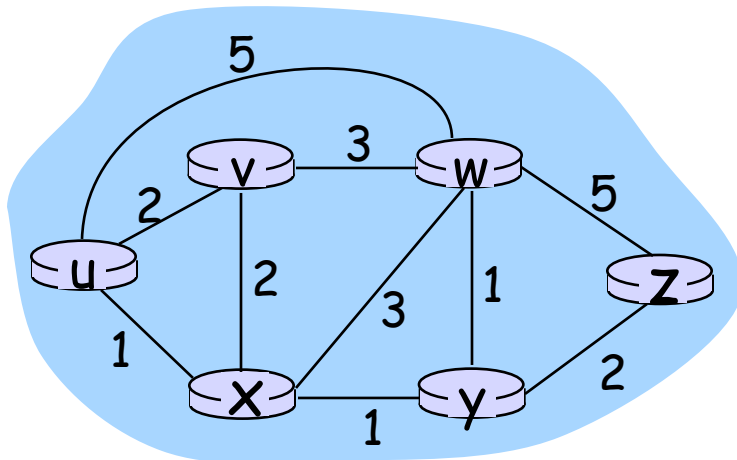
13 /* il nuovo costo verso a è il vecchio costo verso a oppure

14 il costo del cammino minimo noto verso a più il costo da n a a */

15 **Finché** $N' = N$

Algoritmo di Dijkstra: esempio

passo	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux					
2	uxy					
3	uxyv					
4	uxyvw					
5	uxyvwz					



1 **Inizializzazione:**

- 2 N' = {r} /* r è il nodo che esegue l'algoritmo (radice dell'albero)*/
- 3 per tutti i nodi n
- 4 se n è adiacente a r
- 5 allora $D(n) = c(r,n)$
- 6 altrimenti $D(n) = \infty$

8 **Ciclo**

- 9 determina un n non in N' tale che D(n) sia minimo
- 10 aggiungi n a N'
- 11 per ogni nodo a adiacente a n e non in N' aggiorna D(a):
- 12 $D(a) = \min(D(a), D(n) + c(n,a))$
- 13 /* il nuovo costo verso a è il vecchio costo verso a oppure
- 14 il costo del cammino minimo noto verso a più il costo da n a a */
- 15 **Finché** N' = N

Algoritmo di Dijkstra: un altro esempio

Cammino a costo minimo da u:

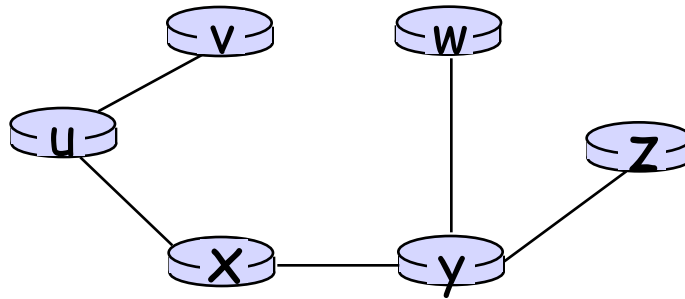
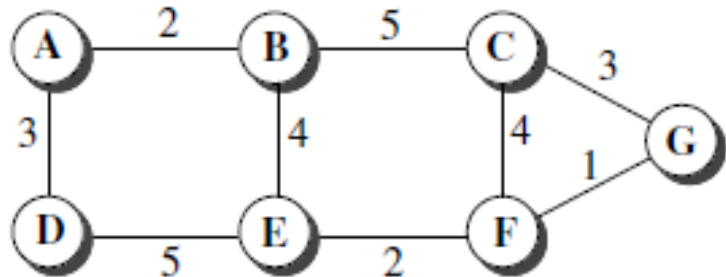


Tabella d'inoltro in u:

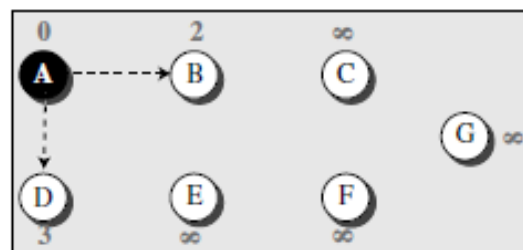
destinazione	collegamento
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Altro esempio

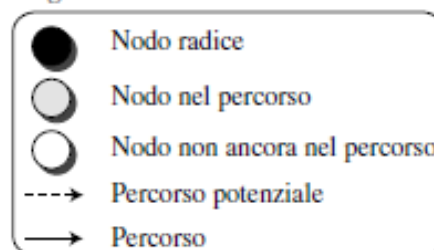


a. Il grafo pesato

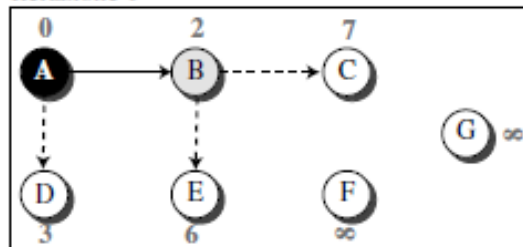
Inizializzazione



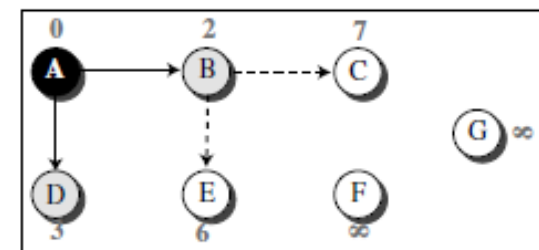
Legenda



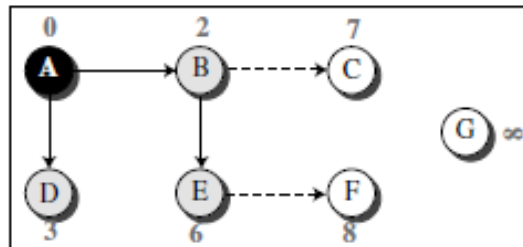
Iterazione 1



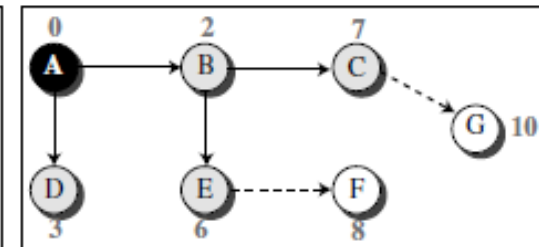
Iterazione 2



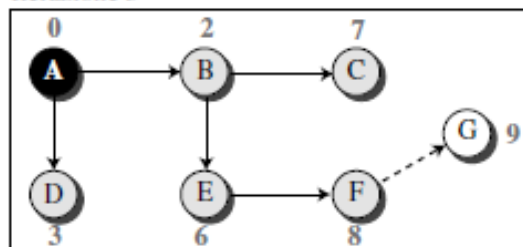
Iterazione 3



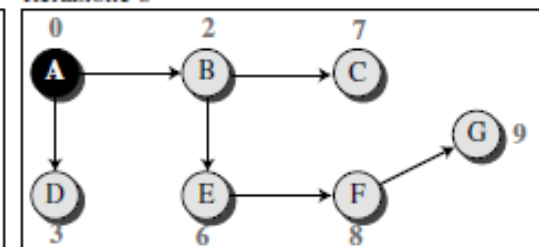
Iterazione 4



Iterazione 5



Iterazione 6



Protocollo di routing

- ❑ Un protocollo è qualcosa di più di un algoritmo!
- ❑ Un protocollo deve definire il suo ambito di funzionamento, i messaggi che vengono scambiati, la comunicazione tra router e l'interazione con gli altri protocolli
- ❑ **OSPF: Open Shortest Path First** basato su algoritmo LS

OSPF (Open Shortest Path First)

- ❑ “open”: le specifiche del protocollo sono pubblicamente disponibili.
- ❑ È un protocollo a stato del collegamento:
 1. Utilizza il **flooding** di informazioni di stato del collegamento
 2. Utilizza l'algoritmo di Dijkstra per la determinazione del percorso a costo minimo.
- ❑ Flooding
 - ❑ Con OSPF, ogni volta che si verifica un cambiamento nello stato di un collegamento, il router manda informazioni d'instradamento a *tutti* gli altri router.
 - ❑ Invia periodicamente (ogni 30 minuti) messaggi OSPF all'intero sistema autonomo, utilizzando il flooding.
- ❑ I messaggi OSPF vengono trasportati direttamente in datagrammi IP usando il numero di protocollo 89 nel campo IP protocol

Messaggi OSPF

- ❑ **Hello:** usato dai router per annunciare la propria esistenza e i vicini che conosce
- ❑ **Database description:** risposta ad hello (consente di ottenere il LSDB a chi si è appena connesso)
- ❑ **Link-state request:** usato per richiedere specifiche informazioni su un collegamento
- ❑ **Link-state update:** messaggio principale usato da OSPF per la costruzione del LSDB
- ❑ **Link-state ack:** riscontro ai link-state update (per fornire affidabilità)

Confronto tra gli algoritmi LS e DV

Complessità dei messaggi:

- LS: con n nodi, E collegamenti, implica l'invio di $O(nE)$ messaggi (ogni nodo deve conoscere il costo degli E link).
- DV: richiede scambi tra nodi adiacenti.
 - Il tempo di convergenza può variare.

Velocità di convergenza:

- LS: l'algoritmo ha complessità $O(n^2)$ (procede prima n nodi, poi $n-1$, $n-2$, etc. quindi $n(n+1)/2$)
- DV: può convergere lentamente.
 - può presentare cicli d'instradamento.
 - può presentare il problema del conteggio all'infinito.

Robustezza: OSPF è più robusto di RIP.

cosa avviene se un router funziona male?

LS:

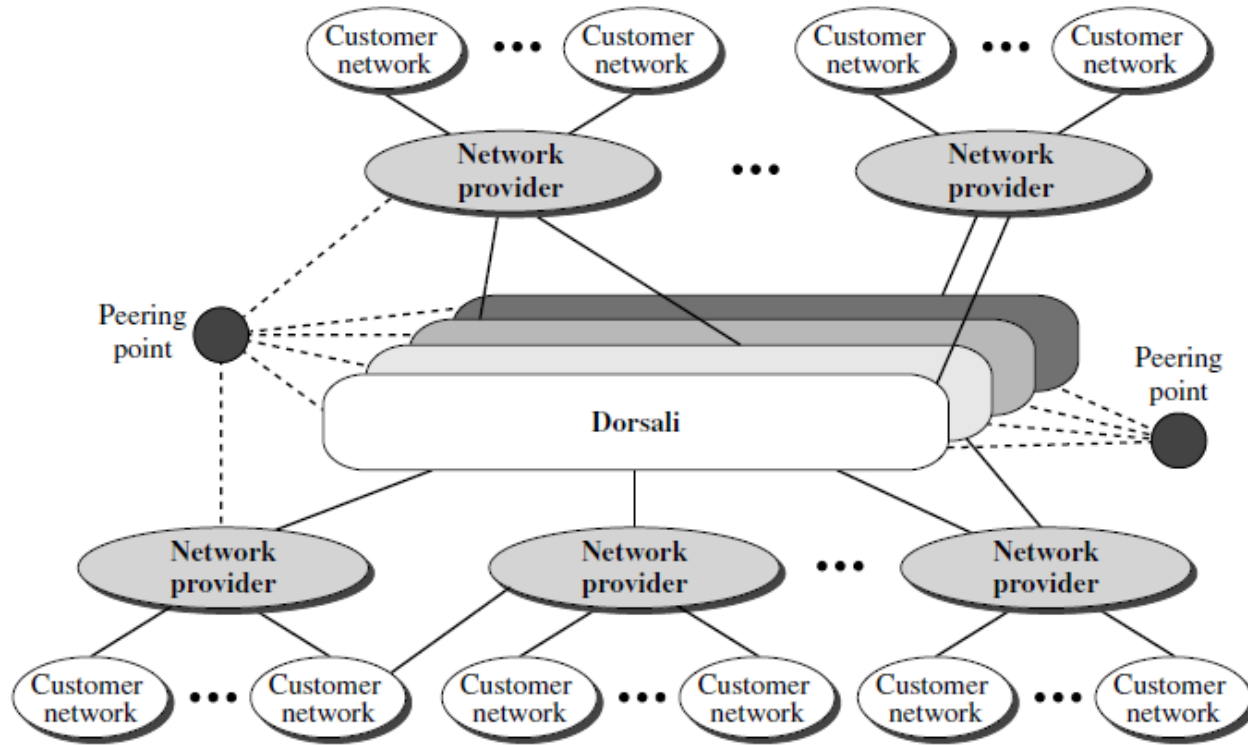
- un router può comunicare via broadcast un costo sbagliato per uno dei suoi collegamenti connessi (ma non per altri).
- i nodi si occupano di calcolare soltanto le proprie tabelle.

DV:

- un nodo può comunicare **cammini** a costo minimo errati a tutte le destinazioni.
- la tabella di ciascun nodo può essere usata dagli altri.
 - Un calcolo errato si può diffondere per l'intera rete.

Internet routing

Struttura di Internet



Come si esegue il routing su Internet?
Può essere usato un solo protocollo?

ISP (forniscono servizi a livelli differenti)

- Dorsali, gestite da società private di telecomunicazioni, che forniscono la connettività globale (connesse tramite peering point)
- Network provider, (fornitori di rete) che utilizzano le dorsali per avere connettività globale e forniscono connettività ai clienti Internet
- Customer network (reti dei clienti) che usano i servizi dei network provider

Impossibilità di usare un singolo protocollo di routing

Abbiamo fin qui visto la rete come una collezione di router interconnessi

- ❑ Ciascun router era indistinguibile dagli altri
- ❑ Visione omogenea della rete

... nella pratica le cose non sono così semplici

Scala: con 200 milioni di destinazioni:

- ❑ Archiviare le informazioni d'instradamento su ciascun host richiederebbe un'enorme quantità di memoria.
- ❑ Il traffico generato dagli aggiornamenti LS non lascerebbero banda per i pacchetti di dati!
- ❑ DV non convergerebbe mai

Autonomia amministrativa:

- ❑ Internet = la rete delle reti
- ❑ Da un punto di vista ideale, ciascuno dovrebbe essere in grado di amministrare la propria rete nel modo desiderato, pur mantenendo la possibilità di connetterla alle reti esterne.
- ❑ **Ogni ISP è un'autorità amministrativa autonoma:**
 - ❑ Usare sottoreti che vuole
 - ❑ Imporre politiche specifiche sul traffico

Instradamento gerarchico

- ❑ Ogni ISP è un *sistema autonomo (AS, autonomous system)*
- ❑ Ogni AS può eseguire un protocollo di routing che soddisfa le sue esigenze
- ❑ I router di un AS eseguono lo stesso algoritmo di routing.
 - Protocollo di routing interno al sistema autonomo (intra-AS), o intradominio, o interior gateway protocol (IGP)
 - I router appartenenti a differenti AS possono eseguire protocolli d'instradamento intra-AS diversi
- ❑ Dobbiamo avere un solo protocollo inter-dominio che gestisce il routing tra i vari AS
- ❑ Protocollo di routing inter-AS o inter-dominio, o Exterior Gateway Protocol (EGP)

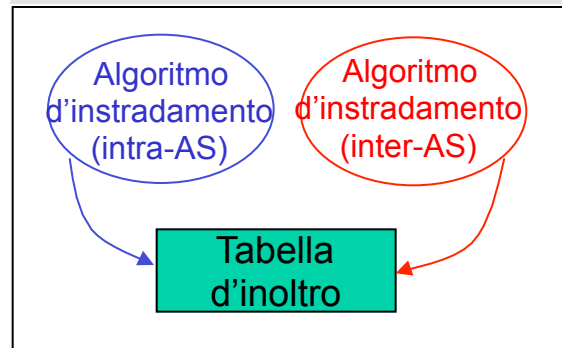
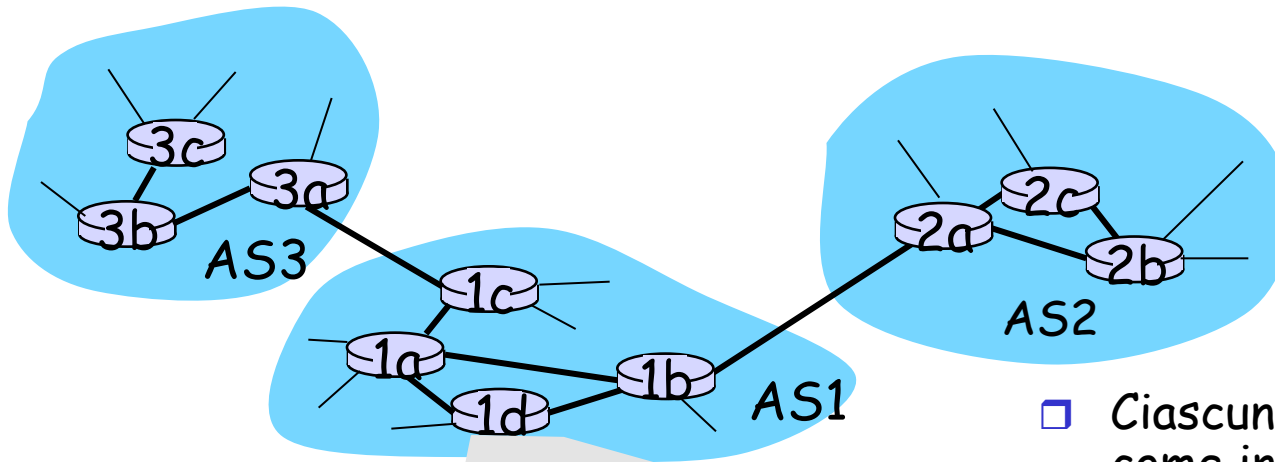
Router gateway

- ❑ Connettono gli AS tra loro
- ❑ Hanno il compito aggiuntivo d'inoltrare pacchetti a destinazioni esterne.

Sistemi autonomi

- ❑ Ogni ISP è un sistema autonomo
- ❑ Ad ogni AS viene assegnato un numero identificativo univoco di 16 bit (autonomous number - ASN) dall'ICANN
- ❑ Gli AS possono avere diverse dimensioni
- ❑ Gli AS sono classificati in base al modo in cui sono connessi ad altri AS
 - AS stub: ha un solo collegamento verso un altro AS. Il traffico è generato o destinato allo stub ma non transita attraverso di esso (es. grande azienda)
 - AS multihomed: ha più di una connessione con altri AS ma non consente transito di traffico (azienda che usa servizi di più di un network provider ma non fornisce connettività agli altri AS)
 - AS di transito: è collegato a più AS e consente il traffico (network provider e dorsali)

Sistemi autonomi interconnessi



- Ciascun sistema autonomo sa come inoltrare pacchetti lungo il percorso ottimo verso qualsiasi destinazione interna al gruppo
 - Il sistema AS1 ha quattro router
 - I sistemi AS2 e AS3 hanno tre router ciascuno
 - I protocolli d'instradamento dei tre sistemi autonomi non sono necessariamente gli stessi
 - I router 1b, 1c, 2a e 3a sono gateway

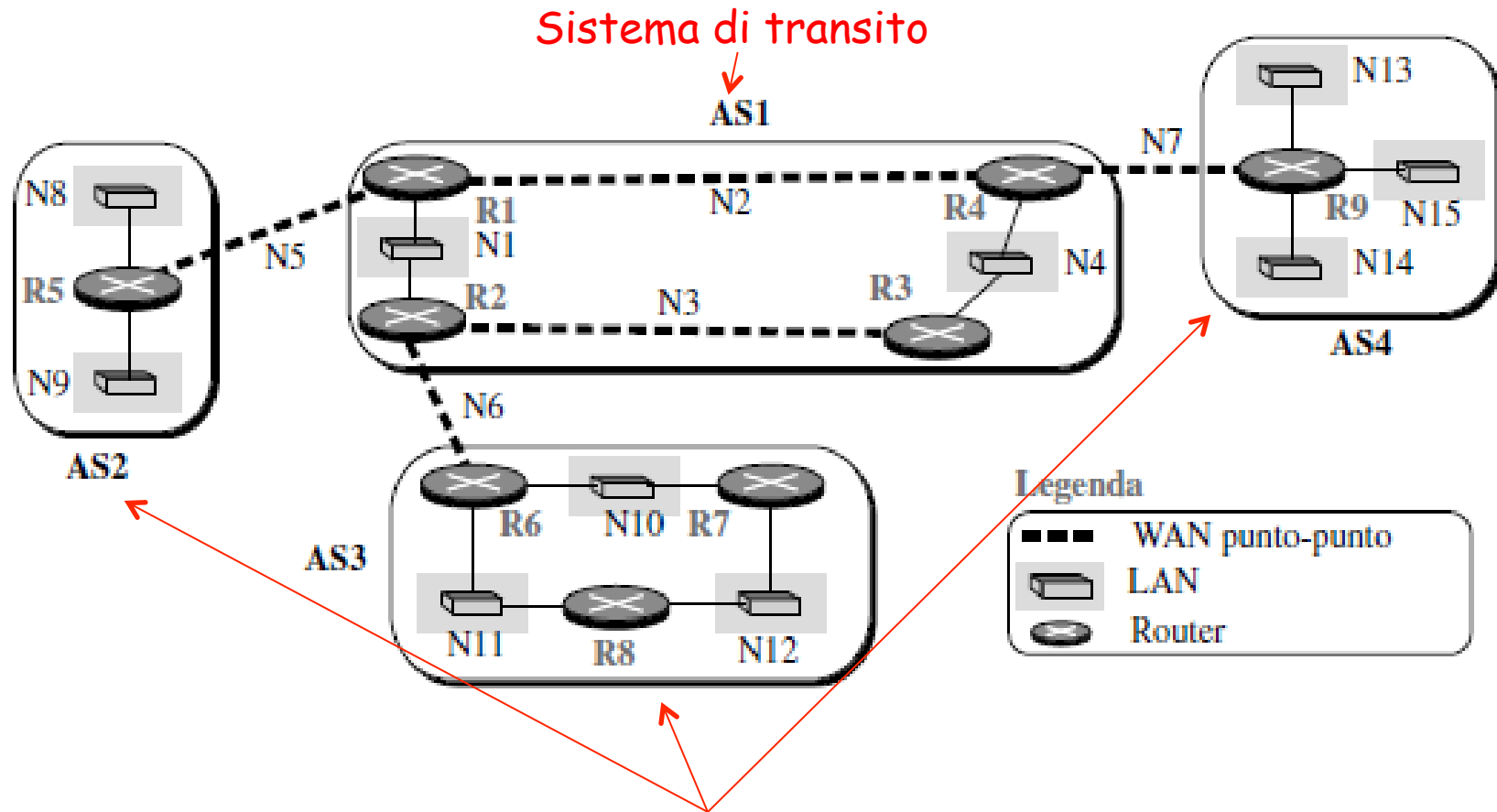
Routing intra-dominio

- ❑ RIP: Routing Information Protocol
- ❑ OSPF: Open Shortest Path First

Routing inter-dominio

- ❑ BGP: Border Gateway Protocol

Instradamento inter-AS



Stub AS - lo scambio di dati tra questi AS passa attraverso AS1

Ogni router all'interno degli AS sa come raggiungere tutte le reti che si trovano nel suo AS ma non sa come raggiungere una rete che si trova in un altro AS 4-21

Border Gateway Protocol

- ❑ RIP e OSPF vengono utilizzati per determinare i percorsi ottimali per le coppie origine-destinazione interne a un sistema autonomo
- ❑ **BGP (Border Gateway Protocol)** - proprietà di CISCO
 - ❑ Usato per determinare percorsi per le coppie origine-destinazione che interessano più sistemi autonomi
 - ❑ rappresenta l'attuale standard *de facto*.
- ❑ BGP è un protocollo **path vector** (distance vector con percorsi)
- ❑ BGP mette a disposizione di ciascun AS un modo per:
 1. ottenere informazioni sulla raggiungibilità delle sottoreti da parte di AS confinanti
 2. propagare le informazioni di raggiungibilità a tutti i router interni di un AS
 3. determinare percorsi "buoni" verso le sottoreti sulla base delle informazioni di raggiungibilità e delle politiche dell'AS
- ❑ BGP consente a ciascuna sottorete di comunicare la propria esistenza al resto di Internet.
- ❑ In BGP le destinazioni non sono host ma prefissi CIDR che rappresentano una sottorete o una collezione di sottoreti

Path-vector routing

- ❑ Sia LS che DS si basano sul costo minimo
- ❑ Tuttavia ci sono casi in cui il costo minimo non è l'obiettivo prioritario

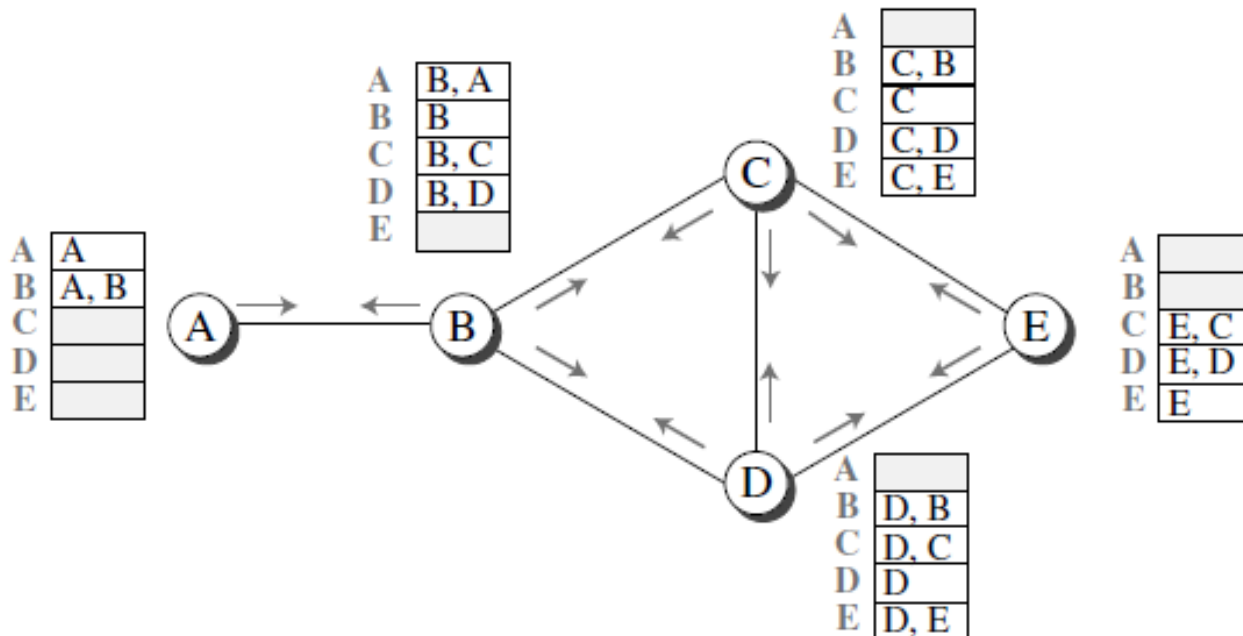
Esempio: un mittente non vuole che i suoi pacchetti passino attraverso determinati router

- ❑ Il routing a costo minimo non consente di applicare questo tipo di politiche nella scelta del percorso
- ❑ Path-vector routing (routing a vettore di percorso): la sorgente può controllare il percorso
 - Politiche: minimizzare il numero di hop
 - Evitare alcuni nodi

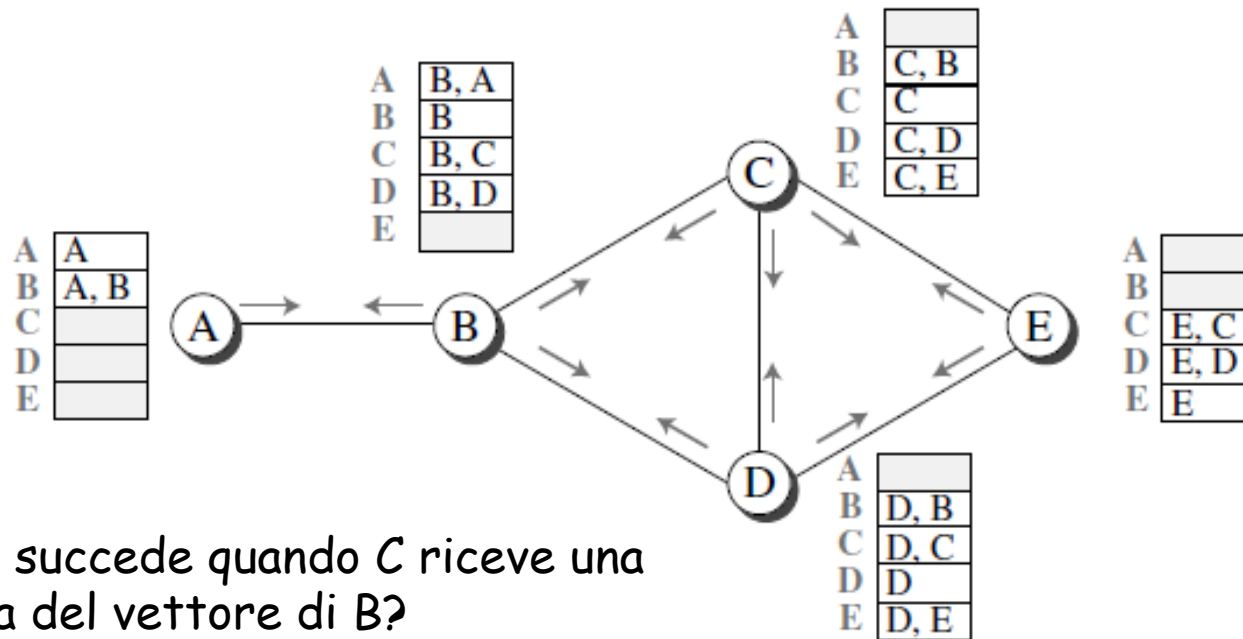
Path vector routing

- Simile a distance vector ma vengono inviati percorsi invece che solo destinazioni
- Ogni nodo quando riceve un path vector da un vicino, aggiorna il suo path vector applicando la sua politica invece del costo minimo

Inizializzazione



Aggiornamento dei path vector



Cosa succede quando C riceve una Copia del vettore di B?

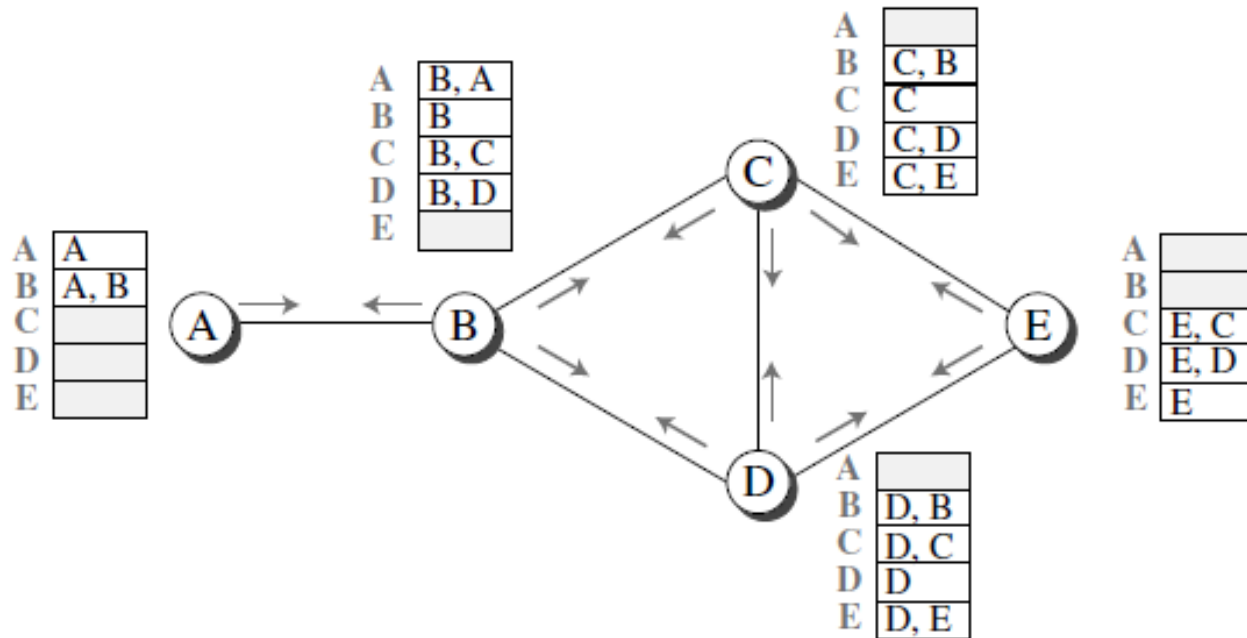
	Nuovo C	Vecchio C	B
A	C, B, A		B, A
B	C, B	C, B	B
C	C	C	B, C
D	C, D	C, D	B, D
E	C, E	C, E	

$C[] = \text{migliore}(C[], C + B[])$

Nota:
 X []: vettore X
 Y: nodo Y

Evento 1: C riceve una copia del vettore di B

Aggiornamento dei path vector



Nessun cambiamento

	Nuovo C	Vecchio C	D
A	C, B, A	C, B, A	
B	C, B	C, B	D, B
C	C	C	D, C
D	C, D	C, D	D
E	C, E	C, E	D, E

$C[] = \text{migliore}(C[], C + D[])$

Algoritmo path-vector

```
1 Path_Vector_Routing ()
2 {
3   // Inizializzazione
4   for (y = 1 to N)
5     {
6       if (y è me_stesso)
7         Path[y] = me_stesso
8       else if (y è un vicino)
9         Path[y] = me_stesso + il_nodo_vicino
10      else
11        Path[y] = vuoto
12    }
13  Spedisci il vettore {Path[1], Path[2], ..., Path[y]} a tutti i vicini
14  // Aggiornamento
15  repeat (sempre)
16    {
17      wait (un vettore Pathw da un vicino w)
18      for (v = 1 to N)
19        {
20          if (Pathw comprende me_stesso)
21            scarta il percorso           // Evita ogni ciclo
22          else
23            Path[y] = il_migliore_tra {Path[y], (me_stesso + Pathw[y])}
24        }
25      If (c'è un cambiamento nel vettore)
26        Spedisci il vettore {Path[1], Path[2], ..., Path[y]} a tutti i vicini
27    }
28 } // Fine del path-vector
```

eBGP e iBGP

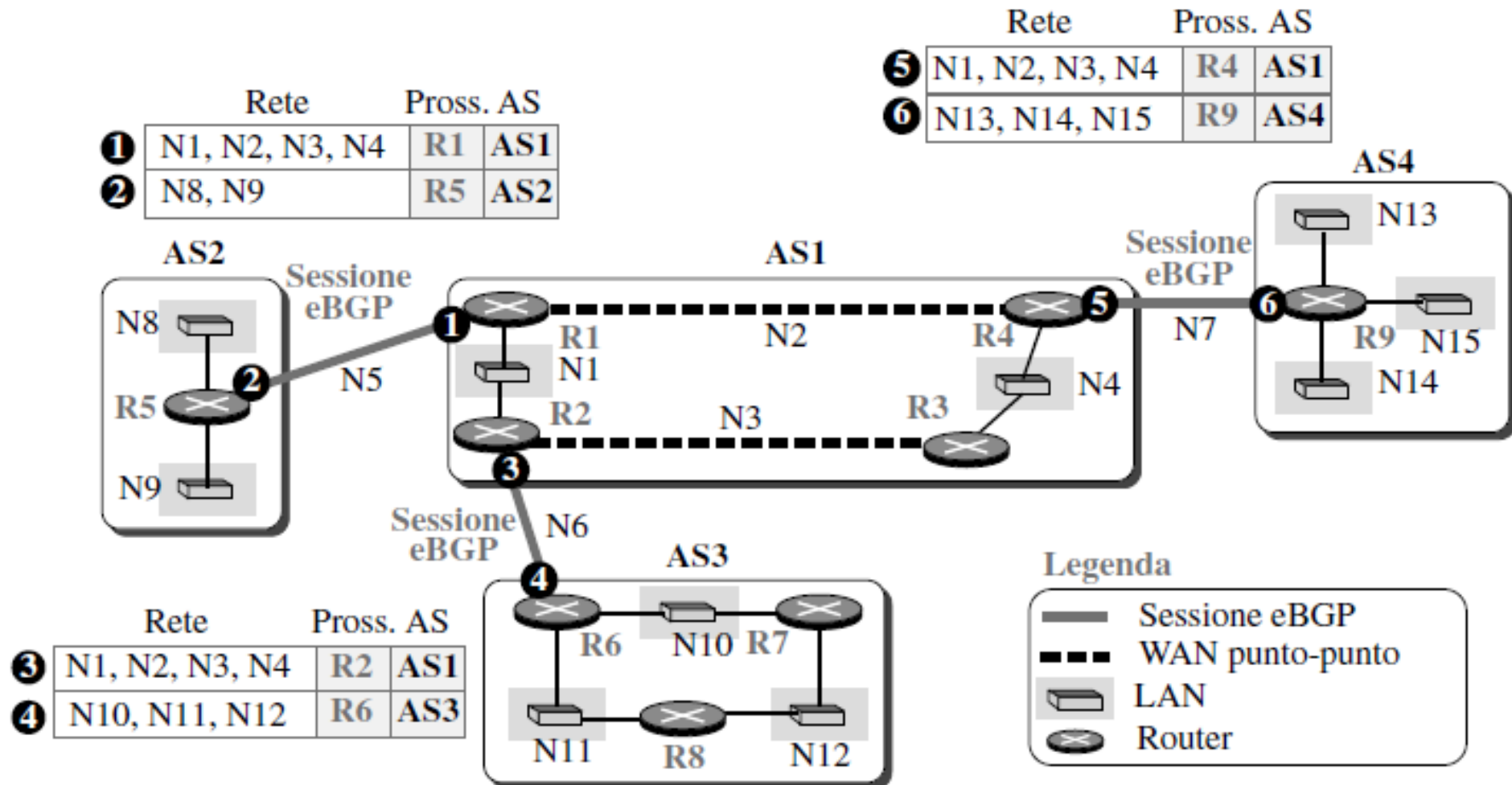
- ❑ Per permettere ad ogni router di instradare correttamente i pacchetti, qualsiasi sia la destinazione, è necessario installare su tutti i router di confine (border router) dell'AS una variante del BGP chiamata *BGP esterno (external BGP o eBGP)*
- ❑ Tutti i router (non solo quelli di confine) dovranno invece usare la seconda variante del BGP, chiamata *BGP interno (internal BGP o iBGP)*
- ❑ I router di confine devono eseguire tre protocolli di routing (intra-dominio, eBGP, iBGP) e tutti gli altri router ne eseguono due (intra-dominio e iBGP)

Fondamenti di BGP

- ❑ Coppie di router si scambiano informazioni di instradamento su connessioni TCP usando la porta 179
- ❑ I router ai capi di una connessione TCP sono chiamati **peer BGP**, e la connessione TCP con tutti i messaggi BGP che vi vengono inviati è detta **sessione BGP**.
- ❑ Notiamo che le linee di sessione BGP non sempre corrispondono ai collegamenti fisici.

eBGP

- Due router di confine che si trovano in due diversi AS formano una coppia di peer BGP e si scambiano messaggi



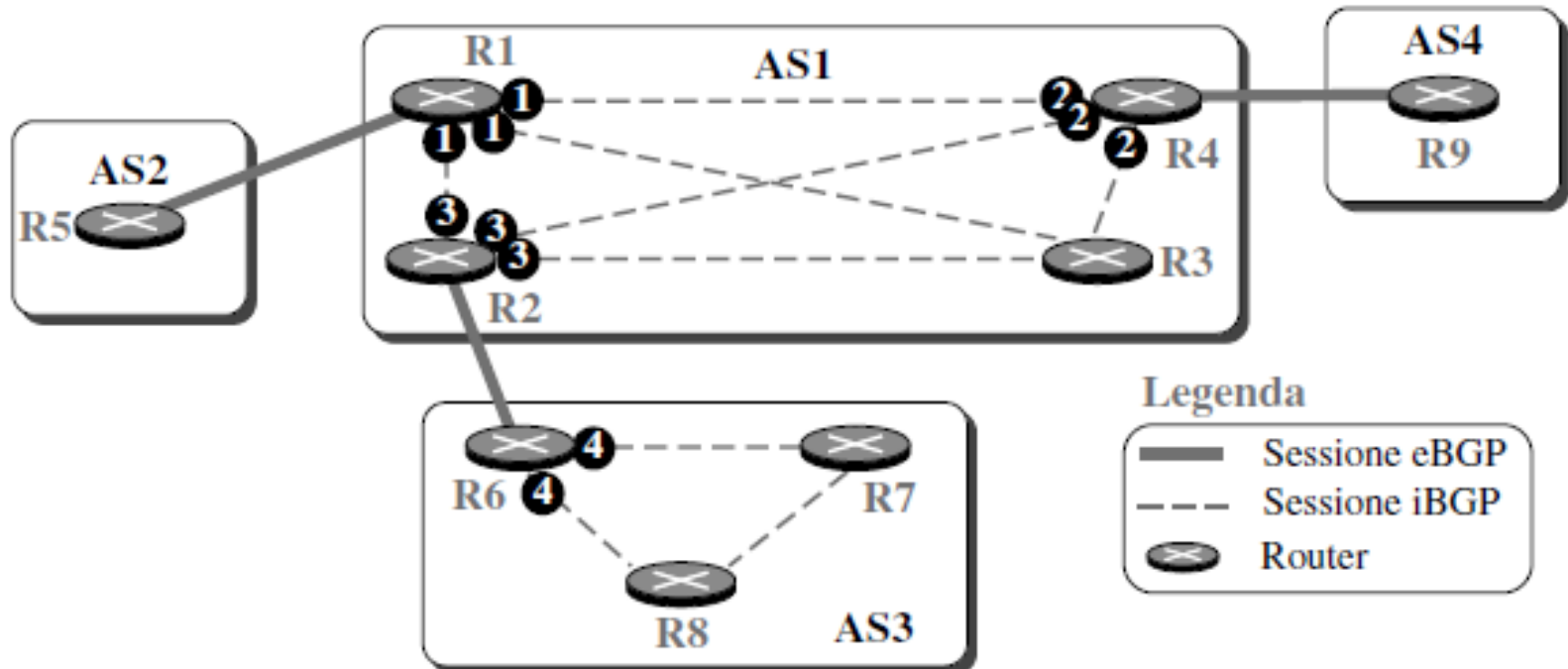
eBGP

- ❑ I messaggi scambiati durante le sessioni eBGP servono per indicare ad alcuni router come instradare i pacchetti destinati ad alcune reti, ma le informazioni di raggiungibilità non sono complete
- ❑ Problemi da risolvere
 1. I router di confine sanno instradare pacchetti solo ad AS vicini
 2. Nessuno dei router non di confine (interno agli AS) sa come instradare un pacchetto destinato alle reti che si trovano in altri AS
- ❑ Soluzione: iBGP

iBGP

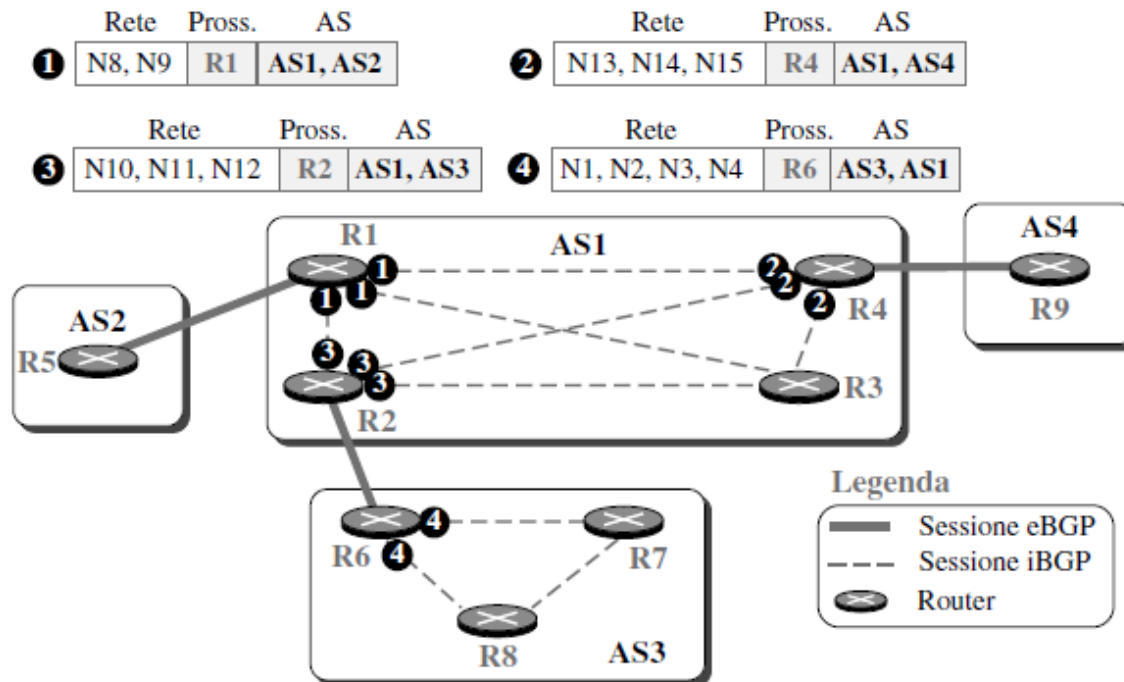
- ❑ Crea una sessione tra ogni possibile coppia di router all'interno di un AS
- ❑ Non tutti i nodi hanno messaggi da inviare (es. R3, R7, R8 non sono collegati a router esterni), ma tutti ricevono

	Rete	Pross.	AS		Rete	Pross.	AS
❶	N8, N9	R1	AS1, AS2	❷	N13, N14, N15	R4	AS1, AS4
❸	N10, N11, N12	R2	AS1, AS3	❹	N1, N2, N3, N4	R6	AS3, AS1



Scambio di messaggi

- Il processo di aggiornamento non termina dopo il primo scambio di messaggi
- Esempio: R1 dopo che ha ricevuto il messaggio di aggiornamento di R2, combina le informazioni circa la raggiungibilità di AS3 con quelle che già conosceva relativamente a AS1 e invia un nuovo messaggio d'aggiornamento a R5 (che quindi sa come raggiungere AS1 e AS3)



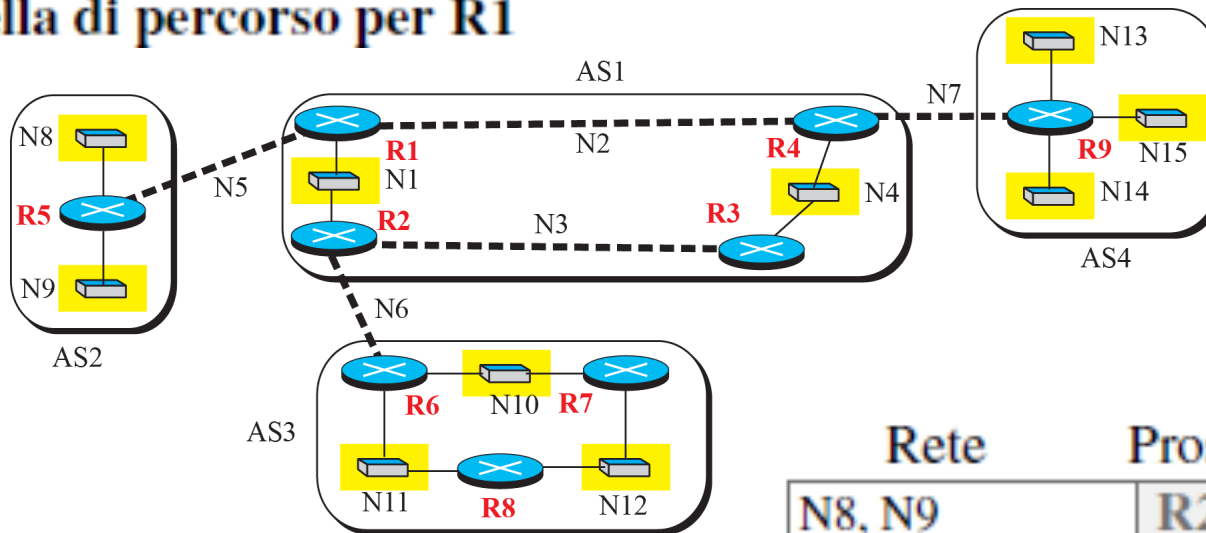
Scambio di messaggi

- ❑ Il processo di aggiornamento continua finché non ci sono più aggiornamenti
- ❑ Le informazioni ottenute da eBGP e iBGP vengono combinate per creare le tabelle dei percorsi

Tabelle di percorso

Rete	Pross.	Percorso
N8, N9	R5	AS1, AS2
N10, N11, N12	R2	AS1, AS3
N13, N14, N15	R4	AS1, AS4

Tabella di percorso per R1



Rete	Pross.	Percorso
N8, N9	R1	AS1, AS2
N10, N11, N12	R6	AS1, AS3
N13, N14, N15	R1	AS1, AS4

Tabella di percorso per R2

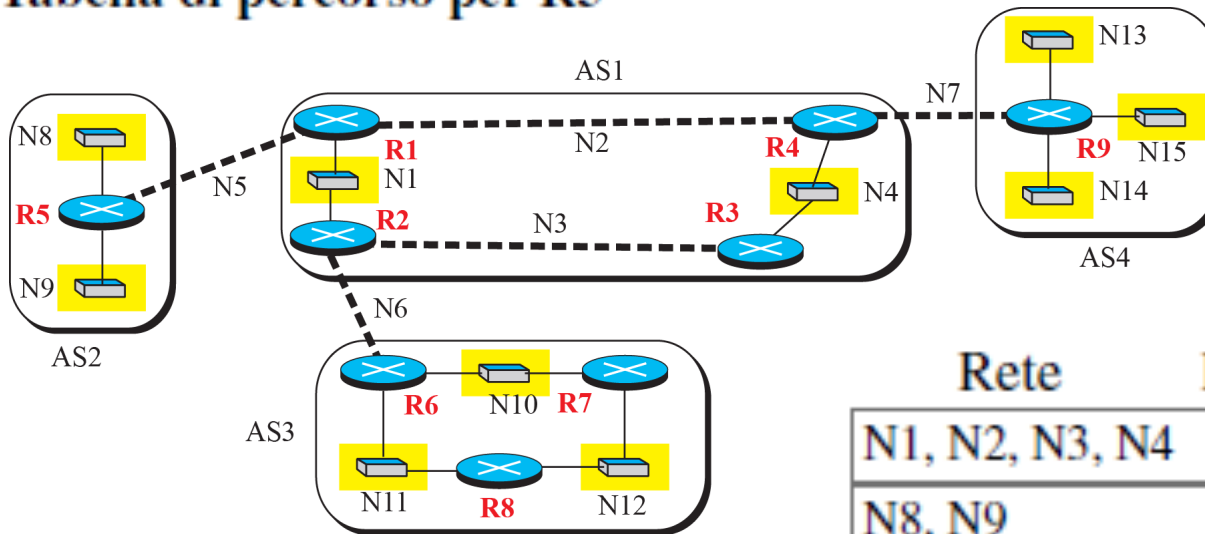
Rete	Pross.	Percorso
N8, N9	R2	AS1, AS2
N10, N11, N12	R2	AS1, AS3
N13, N14, N15	R4	AS1, AS4

Tabella di percorso per R3

Tabelle di percorso

Rete	Pross.	Percorso
N1, N2, N3, N4	R1	AS2, AS1
N10, N11, N12	R1	AS2, AS1, AS3
N13, N14, N15	R1	AS2, AS1, AS4

Tabella di percorso per R5



Rete	Pross.	Percorso
N8, N9	R1	AS1, AS2
N10, N11, N12	R1	AS1, AS3
N13, N14, N15	R9	AS1, AS4

Tabella di percorso per R4

Rete	Pross.	Percorso
N1, N2, N3, N4	R2	AS3, AS1
N8, N9	R2	AS3, AS1, AS2
N13, N14, N15	R2	AS3, AS1, AS4

Tabella di percorso per R6

Tabelle di routing

- ❑ Le tabelle di percorso ottenute da BGP non vengono usate di per sé per l'instradamento dei pacchetti bensì inserite nelle tabelle di routing intra-dominio (generate da RIP o OSPF)
- ❑ Nel caso di stub, l'unico router di confine dell'area aggiunge una regola di default alla fine della sua tabella di routing e definisce come prossimo router quello che si trova dall'altro lato della connessione eBGP
- ❑ Nel caso di AS di transito, il contenuto della tabella di percorso deve essere inserito nella tabella di routing ma bisogna impostare il costo (RIP e OSPF usano metriche differenti)
 - Si imposta il costo pari a quello per raggiungere il primo AS nel percorso

Tabelle d'inoltro dopo l'aggiunta delle informazioni BGP

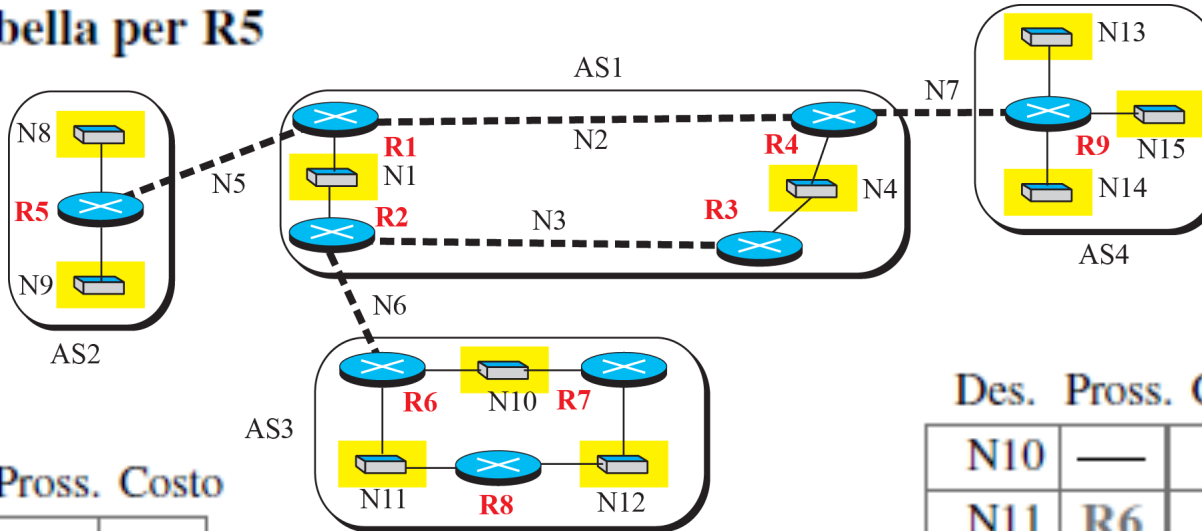
Des.	Pross.	Costo
N8	—	1
N9	—	1
0	R1	1

Tabella per R5

Nel caso di stub, l'unico router di confine dell'area aggiunge una regola di default alla fine della sua tabella di routing e definisce come prossimo router quello che si trova dall'altro lato della connessione eBGP

Des.	Pross.	Costo
N13	—	1
N14	—	1
N15	—	1
0	R4	1

Tabella per R9



Des.	Pross.	Costo
N10	—	1
N11	—	1
N12	R7	2
0	R2	1

Tabella per R6

Des.	Pross.	Costo
N10	R6	2
N11	—	1
N12	—	1
0	R6	2

Tabella per R8

Des.	Pross.	Costo
N10	—	1
N11	R6	2
N12	—	1
0	R6	2

Tabella per R7

Tabelle d'inoltro dopo l'aggiunta delle informazioni BGP

Des.	Pross.	Costo
N1	—	1
N4	R4	2
N8	R5	1
N9	R5	1
N10	R2	2
N11	R2	2
N12	R2	2
N13	R4	2
N14	R4	2
N15	R4	2

Tabella per R1

Des.	Pross.	Costo
N1	—	1
N4	R3	2
N8	R1	2
N9	R1	2
N10	R6	1
N11	R6	1
N12	R6	1
N13	R3	3
N14	R3	3
N15	R3	3

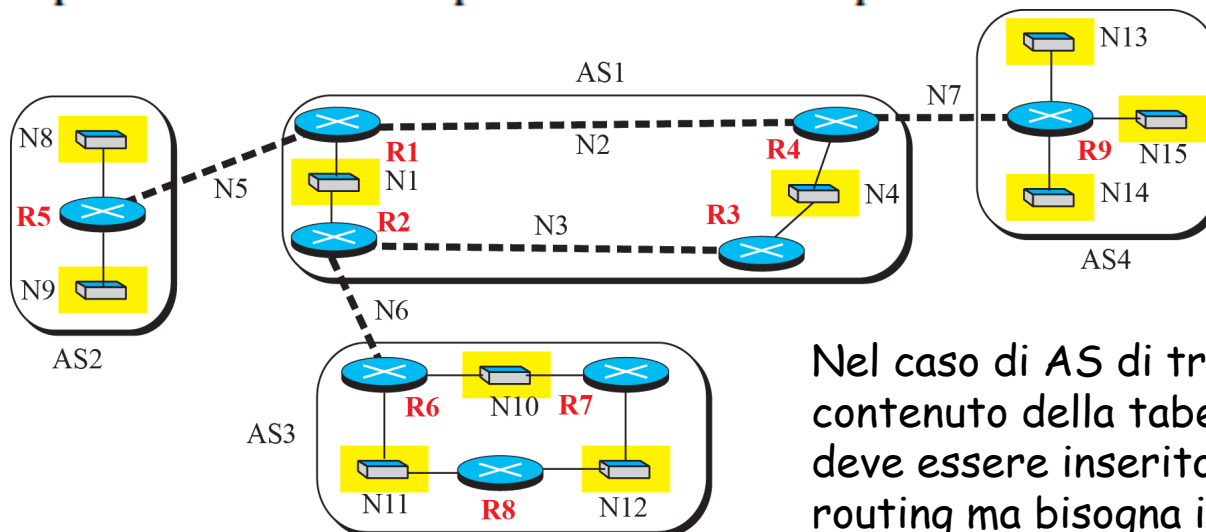
Tabella per R2

Des.	Pross.	Costo
N1	R2	2
N4	—	1
N8	R2	3
N9	R2	3
N10	R2	2
N11	R2	2
N12	R2	2
N13	R4	2
N14	R4	2
N15	R4	2

Tabella per R3

Des.	Pross.	Costo
N1	R1	2
N4	—	1
N8	R1	2
N9	R1	2
N10	R3	3
N11	R3	3
N12	R3	3
N13	R9	1
N14	R9	1
N15	R9	1

Tabella per R4



Nel caso di AS di transito, il contenuto della tabella di percorso deve essere inserito nella tabella di routing ma bisogna impostare il costo

Attributi del percorso e rotte BGP

- ❑ Quando un router annuncia una rotta per un prefisso (di rete) per una sessione BGP, include anche un certo numero di **attributi BGP**.
 - prefisso + attributi = "rotta"
- ❑ Due dei più importanti attributi sono:
 - **AS-PATH**: serve per selezionare i percorsi. Elenca i sistemi autonomi attraverso i quali è passato l'annuncio del prefisso (e quindi gli hop intermedi della rotta). Ogni sistema autonomo non stub ha un identificativo univoco (in questo modo si evitano cicli)
 - **NEXT-HOP**: indirizzo IP dell'interfaccia su cui viene inviato il pacchetto. (Un router ha più indirizzi IP, uno per ogni interfaccia)
- ❑ Quando un router gateway riceve un annuncio di rotta, utilizza le proprie **politiche d'importazione** per decidere se accettare o filtrare la rotta.
 - ❑ Il sistema autonomo può non voler inviare traffico su uno degli AS presenti nel AS-PATH
 - ❑ Router conosce rotta migliore

Selezione dei percorsi BGP

- ❑ Un router può ricavare più di una rotta verso una destinazione (percorsi multipli), e deve quindi sceglierne una.
- ❑ Regole di eliminazione:
 1. Alle rotte viene assegnato come attributo un valore di **preferenza locale**. Si selezionano quindi le rotte con i più alti valori di preferenza locale (riflette la politica imposta dall'amministratore)
 2. Si seleziona la rotta con valore **AS-PATH** più breve.
 3. Si seleziona quella il cui router di **NEXT-HOP** ha costo **minore**: hot-potato routing.
 4. Se rimane ancora più di una rotta, il router si basa sugli **identificatori BGP**.

Messaggi BGP

- ❑ I messaggi BGP vengono scambiati attraverso TCP.
- ❑ Messaggi BGP:
 - **OPEN**: apre la connessione TCP e autentica il mittente
 - **UPDATE**: annuncia il nuovo percorso (o cancella quello vecchio)
 - **KEEPALIVE** mantiene la connessione attiva in mancanza di UPDATE
 - **NOTIFICATION**: riporta gli errori del precedente messaggio; usato anche per chiudere il collegamento.

Perché i protocolli d'instradamento inter-AS sono diversi da quelli intra-AS?

Politiche:

- ❑ Inter-AS: il controllo amministrativo desidera avere il controllo su come il traffico viene instradato e su chi instrada attraverso le sue reti.
- ❑ Intra-AS: unico controllo amministrativo, e di conseguenza le questioni di politica hanno un ruolo molto meno importante nello scegliere le rotte interne al sistema

Prestazioni:

- ❑ Intra-AS: orientato alle prestazioni
- ❑ Inter-AS: le politiche possono prevalere sulle prestazioni