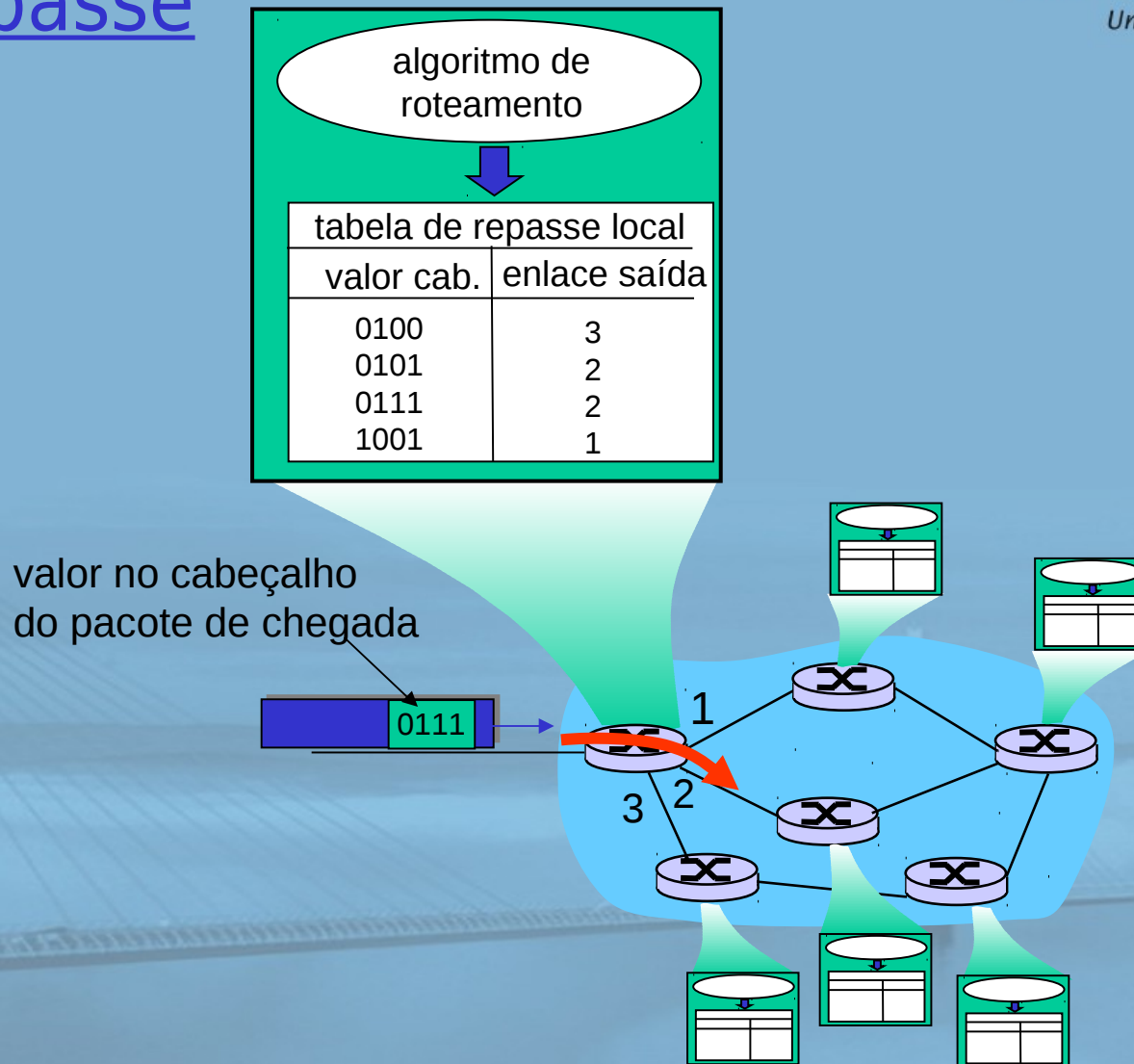


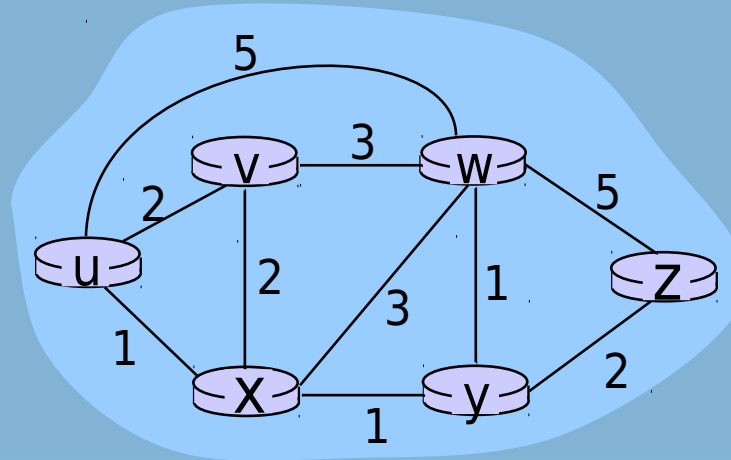
Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Interação entre roteamento e repasse



Abstração de grafo

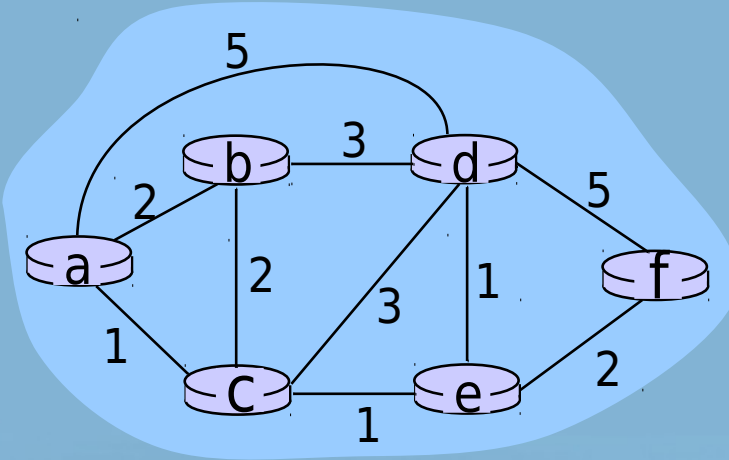


Grafo: $G = (N, E)$

N = conjunto de roteadores = $\{ u, v, w, x, y, z \}$

E = conjunto de enlaces = $\{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

Abstração de grafo: custos



- $c(x, x') =$ custo do enlace (x, x')
 - p. e., $c(d, f) = 5$
- custo poderia ser sempre 1, ou inversamente relacionado à largura ou inversamente relacionado ao congestionamento

Custo do caminho $(x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$

Pergunta: Qual é o caminho de menor custo entre a e f?

algoritmo de roteamento: algoritmo que encontra o caminho de menor custo

Classificação do algoritmo de roteamento

informação global ou descentralizada?

global:

- ❑ todos os roteadores têm topologia completa, informação de custo do enlace
- ❑ algoritmos de “estado do enlace” (Link State - LS)

descentralizada:

- ❑ roteador conhece vizinhos conectados fisicamente, custos de enlace para vizinhos
- ❑ processo de computação iterativo, troca de informações com vizinhos
- ❑ algoritmos de “vetor de distância” (Distance Vector - DV)

Estático ou dinâmico?

estático:

- ❑ rotas mudam lentamente com o tempo

dinâmico:

- ❑ rotas mudam mais rapidamente
 - atualização periódica
 - em resposta a mudanças no custo do enlace

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Algoritmo de roteamento de estado do enlace

algoritmo de Dijkstra

- ❑ topologia e custos de enlace conhecidos de todos os nós
 - realizado por “broadcast de estado do enlace”
 - todos os nós têm a mesma informação
- ❑ calcula caminhos de menor custo de um nó (“origem”) para todos os outros nós da **tabela de repasse**
- ❑ iterativo: após k iterações, sabe caminho de menor custo para k destinos

notação:

- ❑ $c(x,y)$: custo do enlace do nó x até y ; $= \infty$ se não forem vizinhos diretos
- ❑ $D(v)$: valor atual do custo do caminho da origem ao destino v
- ❑ $p(v)$: nó predecessor ao longo do caminho da origem até v
- ❑ N' : conjunto de nós cujo caminho de menor custo é definitivamente conhecido

Algoritmo de Dijkstra

1 **Inicialização:**

2 $N' = \{u\}$

3 para todos os nós v

4 se v adjacente a u

5 então $D(v) = c(u,v)$

6 senão $D(v) = \infty$

7

8 **Loop**

9 acha w não em N' tal que $D(w)$ é mínimo

10 acrescenta w a N'

11 atualiza $D(v)$ para todo v adjacente a w e não em N' :

12 $D(v) = \min(D(v), D(w) + c(w,v))$

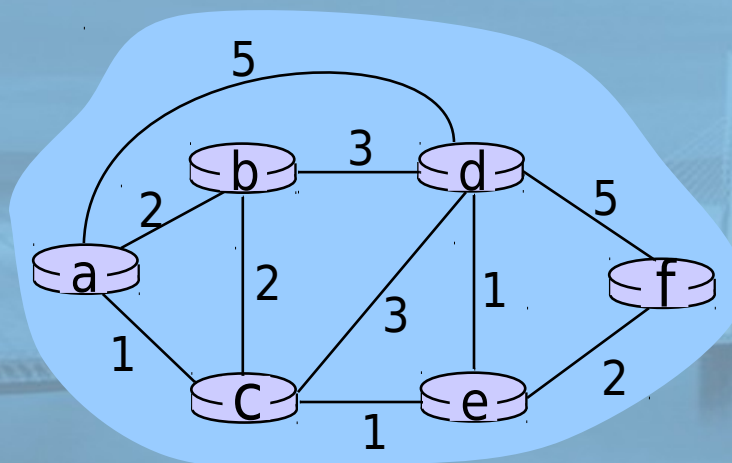
13 /* novo custo para v é custo antigo para v ou custo conhecido

14 do caminho mais curto para w + custo de w para v */

15 **até todos os nós em N'**

Algoritmo de Dijkstra: exemplo

Etapa	N'	D(b),p(b)	D(d),p(d)	D(c),p(c)	D(e),p(e)	D(f),p(f)
0	a	2,a	5,a	1,a	∞	∞
1	ac	2,a	4,c		2,c	∞
2	ace	2,a	3,e			4,e
3	aceb		3,e			4,e
4	acebd					4,e
5	acebdf					



Algoritmo de Dijkstra: exemplo (2)

árvore resultante do caminho mais curto a partir de u:

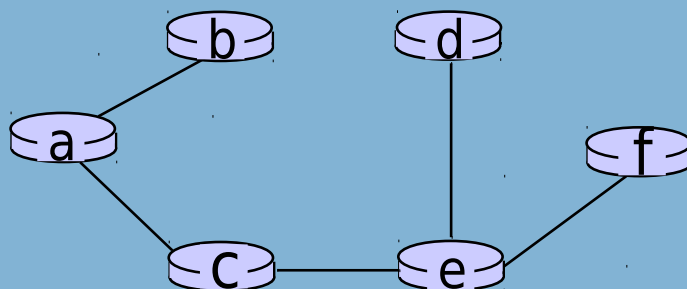


tabela de repasse resultante em u:

destino	enlace
b	(u,b)
c	(u,c)
d	(u,c)
e	(u,c)
f	(u,c)

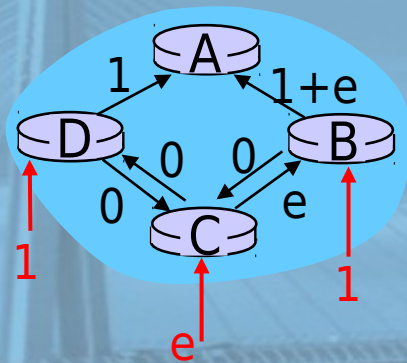
Algoritmo de Dijkstra, discussão

complexidade do algoritmo: n nós

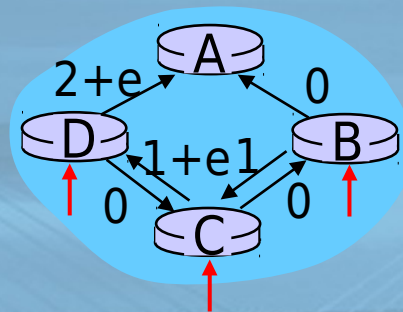
- ❑ cada iteração: precisa verificar todos os nós, w , não em N
- ❑ $n(n+1)/2$ comparações: $O(n^2)$
- ❑ implementações mais eficientes possíveis: $O(n \log n)$

oscilações possíveis:

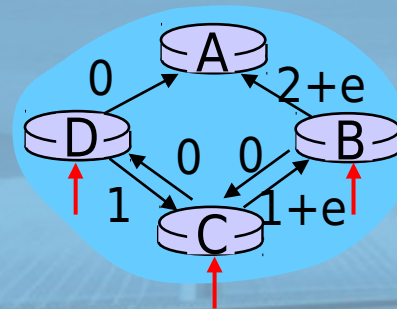
- ❑ p. e., custo do enlace = quantidade de tráfego transportado



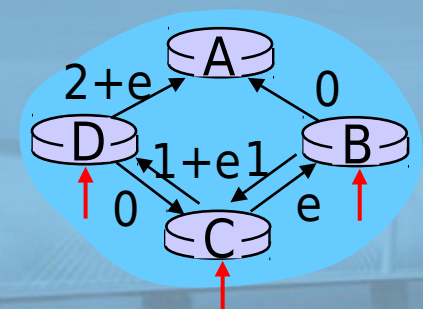
inicialmente



... recalcula
roteamento



... recalcula



... recalcula

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - **vetor de distâncias**
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Algoritmo de vetor de distância

Equação de Bellman-Ford (programação dinâmica)

defina

$d_x(y) :=$ custo do caminho de menor custo de x
para y

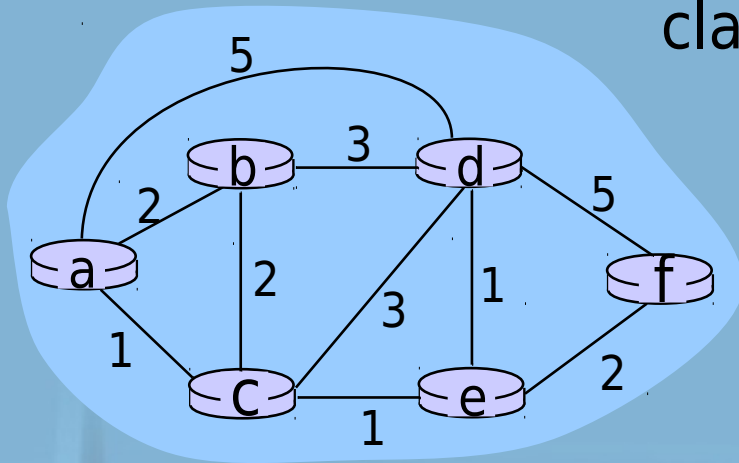
depois

$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

onde min assume todos os vizinhos v de x

Exemplo de Bellman-Ford

claramente, $d_b(f) = 5$, $d_c(f) = 3$, $d_d(f) = 3$



equação B-F diz:

$$\begin{aligned} d_u(z) &= \min \{ c(a,b) + d_b(f), \\ &\quad c(a,c) + d_c(f), \\ &\quad c(a,d) + d_d(f) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

nó que alcança mínimo é o próximo salto
no caminho mais curto → tabela de repasse

Algoritmo de vetor de distância

- ❑ $D_x(y)$ = estimativa do menor custo de x para y
- ❑ nó x sabe custo de cada vizinho v: $c(x,v)$
- ❑ nó x mantém vetor de distância $D_x = [D_x(y): y \in N]$
- ❑ nó x também mantém vetor de distância de seus vizinhos
 - para cada vizinho v, x mantém $D_v = [D_v(y): y \in N]$

Algoritmo de vetor de distância (4)

ideia básica:

- ❑ de tempos em tempos, cada nó envia sua própria estimativa de vetor de distância aos vizinhos
- ❑ assíncrono
- ❑ quando um nó x recebe nova estimativa DV do vizinho, ele atualiza seu próprio DV usando a equação de B-F:

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \quad \text{para cada nó } y \in N$$

- ❑ sob condições modestas, naturais, a estimativa $D_x(y)$ converge para o menor custo real $d_x(y)$

Algoritmo de vetor de distância (5)

iterativo, assíncrono: cada iteração local causada por:

- ❑ mudança de custo do enlace local
- ❑ mensagem de atualização do DV do vizinho

distribuído:

- ❑ cada nó notifica os vizinhos *apenas* quando seu DV muda
 - vizinhos, então, notificam seus vizinhos, se necessário

Cada nó:

```
graph TD; A[espera (mudança no custo do enlace local ou msg do vizinho)] --> B[recalcula estimativas]; B --> C[se DV a qualquer destino tiver mudado, notifica vizinhos]; C --> A;
```

espera (mudança no custo do enlace local ou msg do vizinho)

recalcula estimativas

se DV a qualquer destino tiver mudado, notifica vizinhos

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$$

$$= \min\{2+0, 7+1\} = 2$$

tabela nó x

		custo para		
		x	y	z
de	x	0	2	7
	y	∞	∞	∞
	z	∞	∞	∞

tabela nó y

		custo para		
		x	y	z
de	x	∞	∞	∞
	y	2	0	1
	z	∞	∞	∞

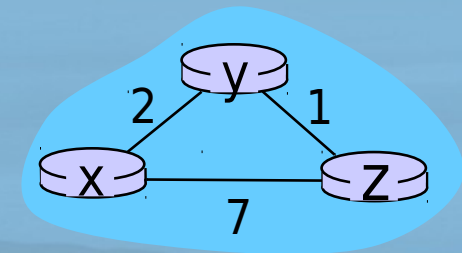
tabela nó z

		custo para		
		x	y	z
de	x	∞	∞	∞
	y	∞	∞	∞
	z	7	1	0

		custo para		
		x	y	z
de	x	0	2	3
	y	2	0	1
	z	7	1	0

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$$

$$= \min\{2+1, 7+0\} = 3$$



► tempo

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$$

$$= \min\{2+0, 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$$

$$= \min\{2+1, 7+0\} = 3$$

tabela nó x

custo para

	x	y	z
x	0	2	7
y	∞	0	∞
z	∞	∞	0

tabela nó y

custo para

	x	y	z
x	∞	0	∞
y	2	0	1
z	∞	∞	0

tabela nó z

custo para

	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

custo para

	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

custo para

	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

custo para

	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

custo para

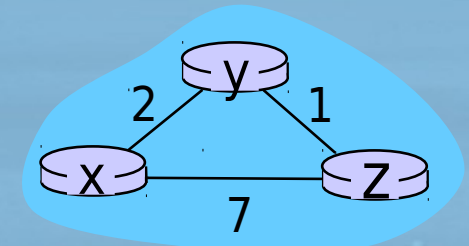
	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

custo para

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

custo para

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

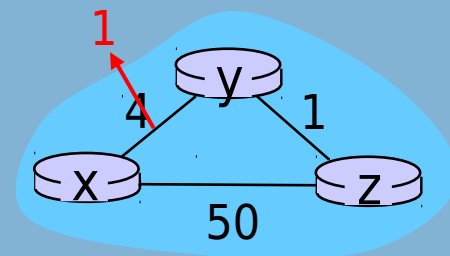


tempo

Vetor de distância: mudanças de custo do enlace

mudanças de custo do enlace:

- ❑ nó detecta mudança de custo no enlace local
- ❑ atualiza informação de roteamento, recalcula vetor de distância
- ❑ se DV mudar, notifica vizinhos



“boas
notícias
correm
rápido”

no tempo t_0 , y detecta a mudança do custo do enlace, atualiza seu DV e informa aos seus vizinhos.

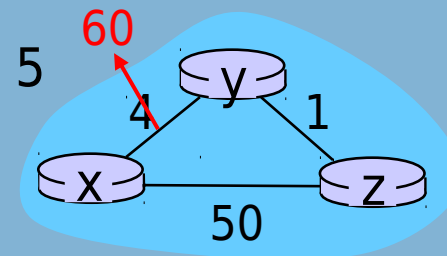
no tempo t_1 , z recebe a atualização de y e atualiza sua tabela. Calcula um novo custo mínimo para x e envia seu DV aos vizinhos.

no tempo t_2 , y recebe a atualização de z e atualiza sua tabela de distância. Menores custos de y não mudam, e daí y *não* envia qualquer mensagem a z.

Vetor de distância: mudanças de custo do enlace

mudanças de custo do enlace:

- ❑ antes da mudança temos: $D_y(x) = 4$ e $D_z(x) = 5$
- ❑ y detecta a mudança e atualiza $D_y(x)=6$ e repassa para z
- ❑ z atualiza $D_z(x)=7$ e repassa para y
- ❑ y atualiza $D_y(x)=8$ e repassa para z
- ❑



solução:

- ❑ ao repassar vetor de distância aos vizinhos, se utiliza o vizinho como rota, mude valor para infinito
 - y detecta a mudança e atualiza $D_y(x)=6$, mas repassa $D_y(x)=\infty$ para z
- ❑ z atualiza $D_z(x)=50$ e repassa para y
- ❑ y atualiza $D_y(x)=51$ e repassa $D_y(x)=\infty$ para z
- ❑ não ocorre mais alteração em z

Comparação dos algoritmos LS e DV

complexidade da mensagem

- ❑ LS: com n nós, E enlaces, $O(nE)$ mensagens enviadas
- ❑ DV: troca apenas entre vizinhos
 - tempo de convergência varia

velocidade de convergência

- ❑ LS: algoritmo $O(n \log n)$ requer $O(nE)$ mensagens
 - pode ter oscilações
- ❑ DV: tempo de convergência varia
 - podem ter loops de roteamento
 - problema da contagem até o infinito

robustez: o que acontece se roteador der defeito?

LS:

- nó pode anunciar custo do *enlace* incorreto
- cada nó calcula apenas sua *própria* tabela

DV:

- nó DV pode anunciar custo do *caminho* incorreto
- tabela de cada nó usada por outros
 - erro se propaga pela rede

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Roteamento hierárquico

nosso estudo de roteamento até aqui – o ideal:

- ❑ todos os roteadores idênticos
- ❑ rede “achatada”

... *não* acontece na prática

escala: com 200 milhões de destinos:

- ❑ não pode armazenar todos os destinos nas tabelas de roteamento!
- ❑ troca de tabela de roteamento atolaria os enlaces!

autonomia administrativa

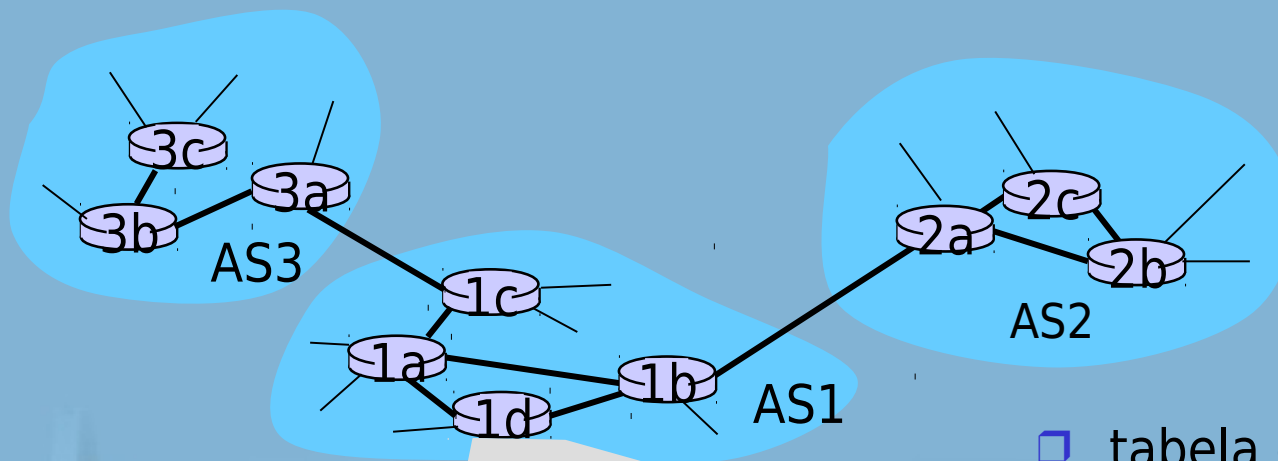
- ❑ Internet = rede de redes
- ❑ cada administrador de rede pode querer controlar o roteamento em sua própria rede

- ❑ roteadores agregados em regiões, “sistemas autônomos” (AS)
- ❑ roteadores no mesmo AS rodam o mesmo protocolo de roteamento
 - protocolo de roteamento “intra-AS”
 - roteadores em ASes diferentes podem executar protocolo de roteamento intra-AS diferente

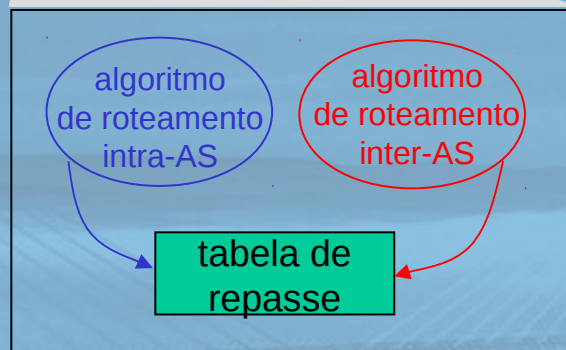
roteador de borda

- ❑ Enlace direto com roteador em outro AS

ASes interconectados

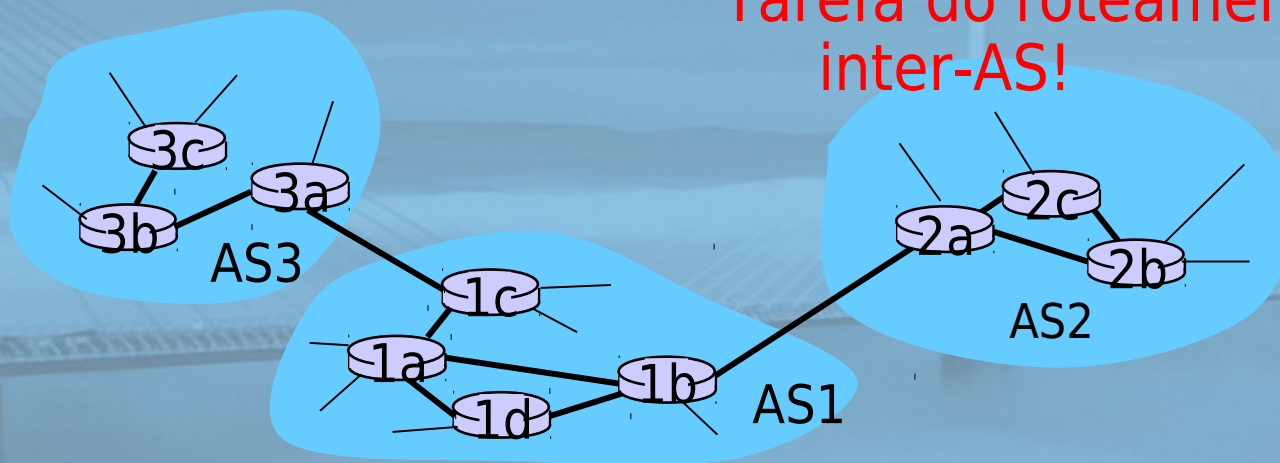


- tabela de repasse configurada por algoritmo de roteamento intra e inter-AS
 - intra-AS define entradas para destinos internos
 - inter-AS & intra-AS definem entradas para destinos externos



Tarefas inter-AS

- suponha que roteador no AS1 recebe datagrama destinado para fora do AS1:
 - roteador deve encaminhar pacote ao roteador de borda, mas qual?



AS1 deve:

1. descobrir quais destinos são alcançáveis por AS2 e quais por AS3
2. propagar essa informação de acessibilidade a todos os roteadores no AS1

Tarefa do roteamento inter-AS!

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Roteamento intra-AS

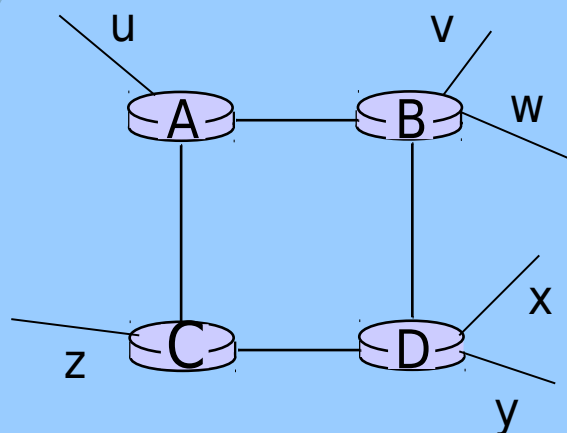
- ❑ também conhecido como **Interior Gateway Protocols (IGP)**
- ❑ protocolos de roteamento intra-AS mais comuns:
 - RIP: Routing Information Protocol
 - OSPF: Open Shortest Path First
 - IGRP: Interior Gateway roteamento Protocol (proprietário da Cisco)

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

RIP (Routing Information Protocol)

- ❑ algoritmo de vetor de distância
- ❑ incluído na distribuição BSD-UNIX em 1982
- ❑ métrica de distância: # de saltos (máx. = 15 saltos)



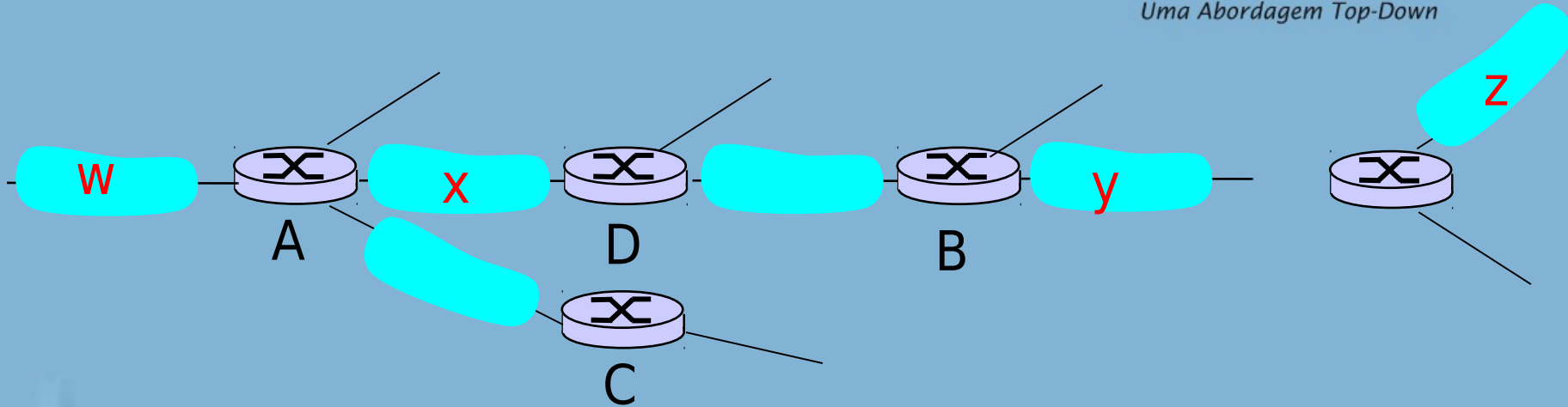
Do roteador A às sub-redes:

<u>destino</u>	<u>saltos</u>
u	1
v	2
w	2
x	3
y	3
z	2

Anúncios RIP

- ❑ vetores de distância: trocados entre vizinhos a cada 30 s por meio de mensagem de resposta (também conhecida como **anúncio**)
- ❑ cada anúncio: lista de até 25 sub-redes de destino dentro do AS

RIP: Exemplo



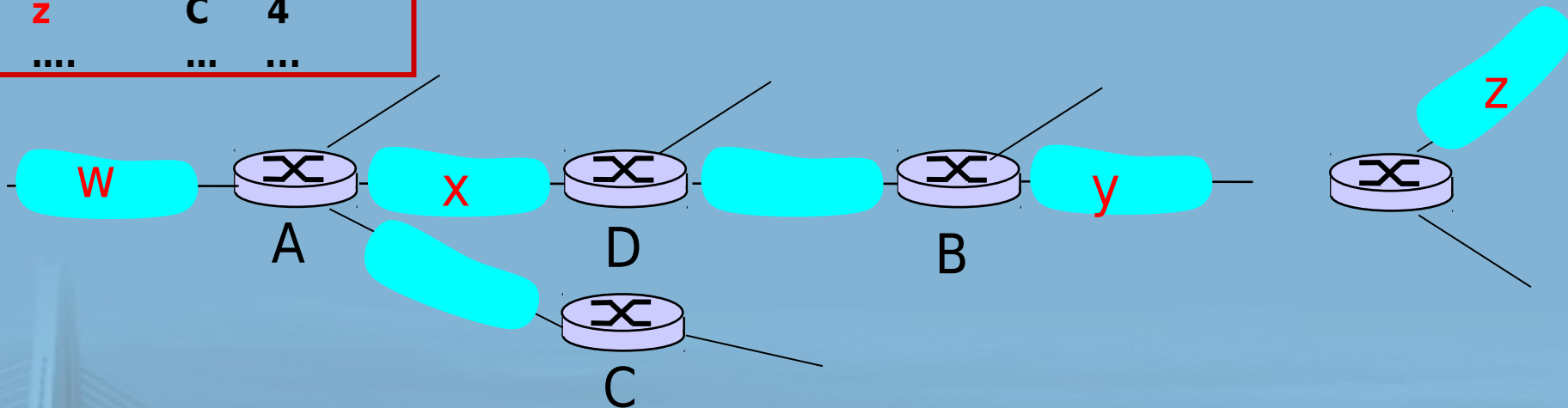
Rede de destino	Roteador seguinte	Núm. saltos até dest.
W	A	2
y	B	2
z	B	7
x	--	1
....

tabela de roteamento/repassse em D

anúncio de A para D

Destino Próx. saltos

W	-	1
X	-	1
Z	C	4
....



Rede de destino até dest.

Roteador seguinte

Núm. saltos

W	A	2
Y	B	2
Z	B A	7 5
X	--	1
....

tabela de roteamento/repasse em D

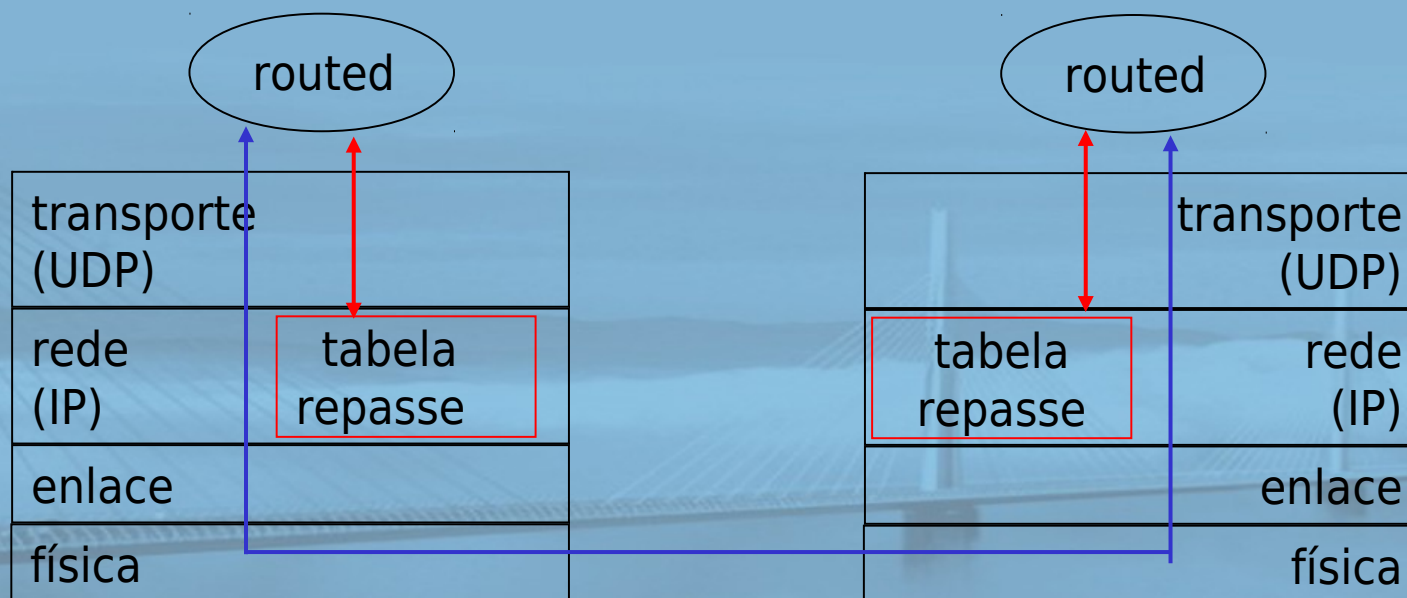
RIP: falha e recuperação do enlace

se nenhum anúncio for ouvido após 180 s -->
vizinho/enlace declarado morto

- rotas via vizinho invalidadas
- novos anúncios enviados aos vizinhos
- vizinhos por sua vez enviam novos anúncios
- informação de falha do enlace se propaga para rede inteira
- *reversão envenenada* usada para impedir loops de pingue-pongue (distância infinita = 16 saltos)

Processamento de tabela RIP

- ❑ tabelas de roteamento RIP controladas por processo em **nível de aplicação** chamado routed (daemon)
- ❑ anúncios enviados em pacotes UDP, repetidos periodicamente



Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

OSPF (Open Shortest Path First)

- ❑ “open”: publicamente disponível
- ❑ usa algoritmo Link State
 - disseminação de pacote LS
 - mapa de topologia em cada nó
 - cálculo de rota usando algoritmo de Dijkstra
- ❑ anúncio OSPF transporta uma entrada por roteador vizinho
- ❑ anúncios disseminados ao AS **inteiro** (com inundação)
 - transportados nas mensagens OSPF diretamente por IP (em vez de TCP ou UDP)

Recursos “avancados” do OSPF (não no RIP)

- ❑ **segurança**: todas as mensagens OSPF autenticadas (para impedir intrusão maliciosa)
- ❑ múltiplos caminhos de mesmo custo permitidos (apenas um caminho no RIP)
- ❑ para cada enlace, múltiplas métricas de custo para diferentes tipos de serviços (p. e., custo de enlace de satélite definido “baixo” para melhor esforço; alto para tempo real)
- ❑ OSPF **hierárquico** em grandes domínios

- ❑ **hierarquia em dois níveis:** área local, backbone.
 - anúncios de estado do enlace somente na área
 - cada nó tem topologia de área detalhada; somente direção conhecida (caminho mais curto) para redes em outras áreas.
- ❑ **roteadores de borda:**
 - “resumem” distâncias às redes na própria área, anunciam para outros roteadores de borda.
 - conectam-se a outros AS's.
- ❑ **roteadores de backbone:** executam roteamento OSPF limitado ao backbone.

Capítulo 4: Camada de rede

- ❑ 4.1 Introdução
- ❑ 4.2 Redes de circuitos virtuais e de datagramas
- ❑ 4.3 O que há dentro de um roteador?
- ❑ 4.4 IP: Internet Protocol
 - formato do datagrama
 - endereçamento IPv4
 - ICMP
 - IPv6
- ❑ 4.5 Algoritmos de roteamento
 - estado de enlace
 - vetor de distâncias
 - roteamento hierárquico
- ❑ 4.6 Roteamento na Internet
 - RIP
 - OSPF
 - BGP
- ❑ 4.7 Roteamento broadcast e multicast

Roteamento inter-AS da Internet: BGP

- ❑ **BGP (Border Gateway Protocol):** o padrão de fato
- ❑ BGP oferece a cada AS um meio de:
 1. obter informação de acessibilidade da sub-rede a partir de ASs vizinhos.
 2. propagar informação de acessibilidade a todos os roteadores internos ao AS.
 3. determinar rotas “boas” para sub-redes com base na informação e política de acessibilidade.
- ❑ permite que a sub-rede anuncie sua existência ao resto da Internet: *“Estou aqui”*

Por que roteamento intra e inter-AS diferente?

política:

- ❑ inter-AS: admin deseja controle sobre como seu tráfego é roteado, quem roteia através de sua rede
- ❑ intra-AS: único admin, de modo que nenhuma decisão política é necessária

escala:

- ❑ roteamento hierárquico salva tamanho de tabela, tráfego de atualização reduzido

desempenho:

- ❑ intra-AS: pode focalizar no desempenho
- ❑ inter-AS: política pode dominar sobre desempenho