

PTC 3450 - Aula 2 I

5.1 Introdução ao plano de controle

5.2 Algoritmos de roteamento

(Kurose, p. 271 - 283)

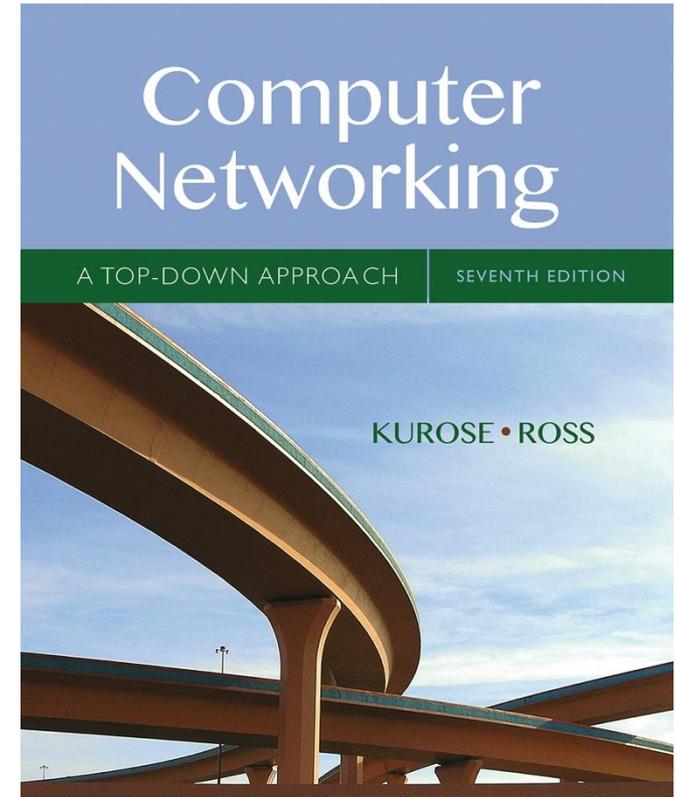
(Peterson, p. 147-163)

13/06/2017

Capítulo 5

Camada de rede:

O Plano de Controle



Computer Networking: A Top Down Approach

7th edition

Jim Kurose, Keith Ross

Pearson/Addison Wesley

April 2016

Capítulo 5: plano de controle da camada de rede

objetivos do capítulo: entender princípios que fundamental o plano de controle da rede

- algoritmos de roteamento tradicionais
- controladores SDN
- *Internet Control Message Protocol (ICMP)*
- administração de rede

e sua implementação na Internet:

- OSPF, BGP, *OpenFlow*, controladores ODL e ONOS, ICMP, SNMP

Capítulo 5: roteiro

5.1 introdução

5.2 protocolos de roteamento

- estado de enlace
- vetor de distâncias

5.3 Roteamento intra-AS na Internet: OSPF

5.4 roteamento entre os ISPs: BGP

5.5 O plano de controle do SDN

5.6 ICMP: o *Internet Control Message Protocol*

5.7 Administração de rede e o SNMP

Funções da camada de rede

Lembrando: 2 funções da camada de rede

- *repasse*: mover pacotes de da entrada do roteador para a saída apropriada

plano de dados

- *roteamento*: determinar a rota tomada pelos pacotes da fonte ao destino

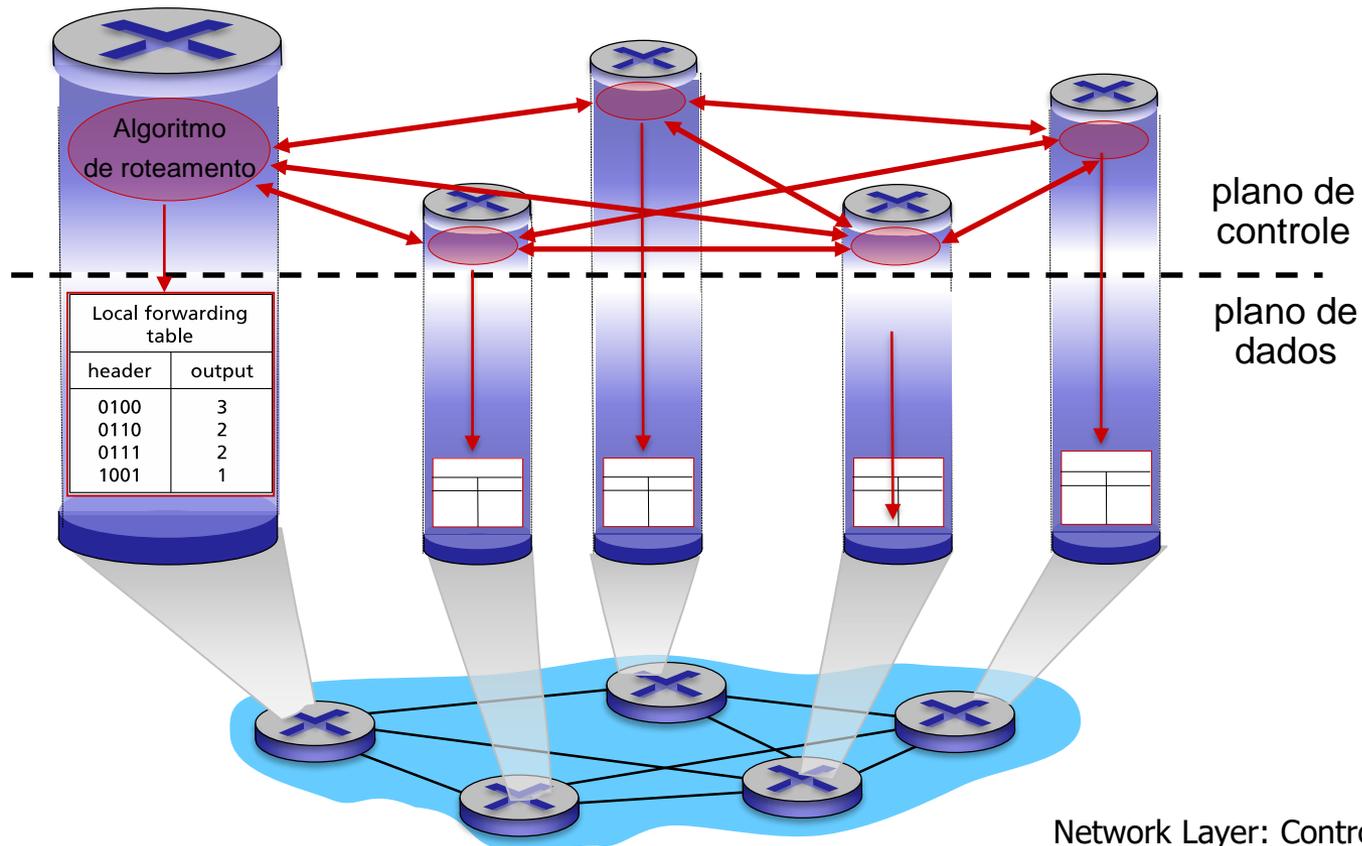
plano de controle

2 abordagens para estruturar plano de controle da rede:

- controle por roteador (tradicional)
- controle centralizado logicamente (*software defined networking*)

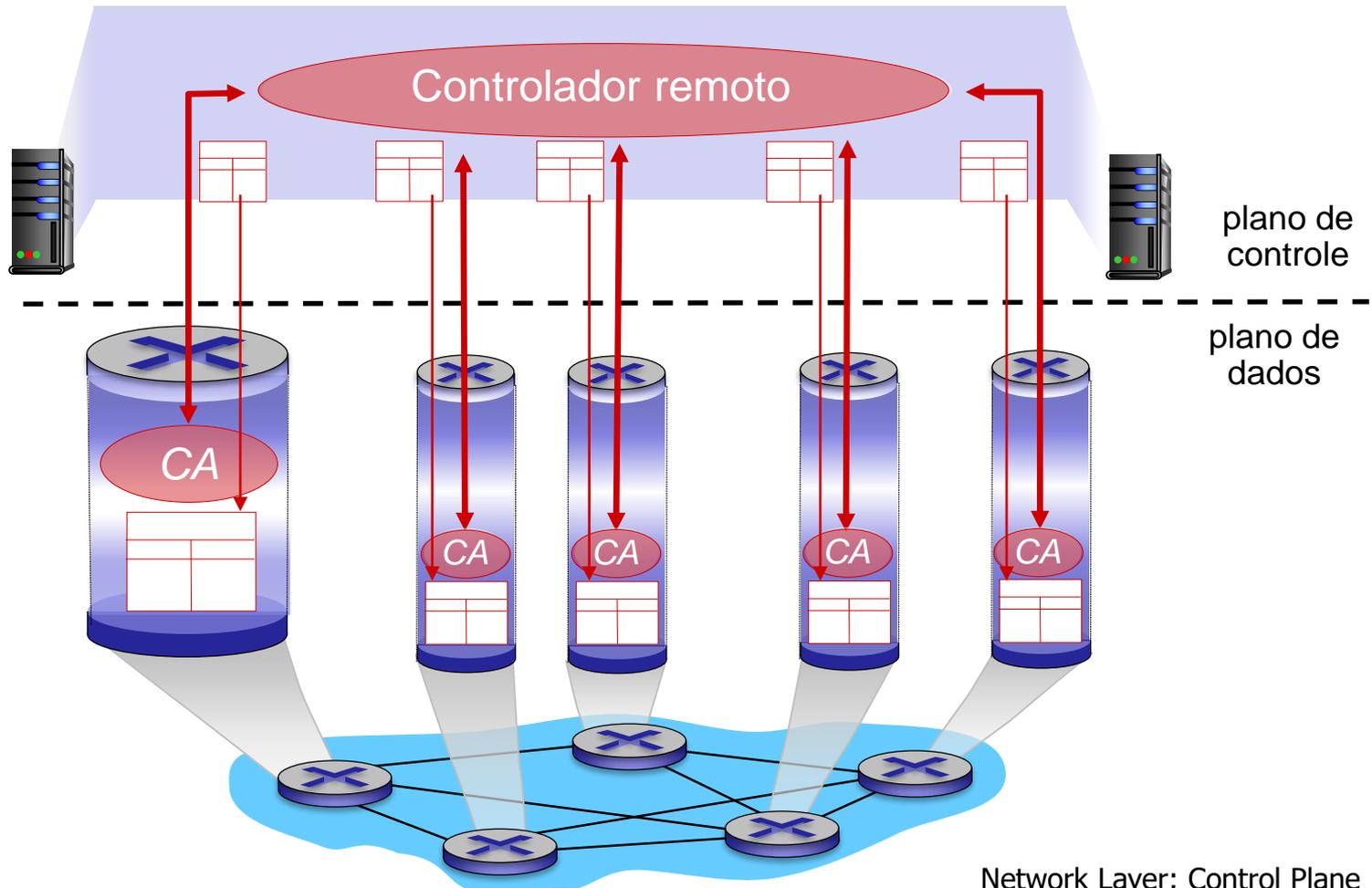
Plano de controle por roteador

Componentes do algoritmo de roteamento individuais *em cada e todo roteador* interagem um com os outros no plano de controle para computar as tabelas de repasse



Plano de controle centrado logicamente

Controlador distinto (tipicamente remoto) interage com agentes de controle local (CAs – *Control Agents*) nos roteadores para computar as tabelas de repasse



Exemplos de redes baseadas em SDN

- [B4 – Google](#): ligando globalmente seus datacenters [palestra](#)
- [SWAN – Microsoft Research](#): experimental [palestra](#)
- [China Telecom + Huawei](#)
- [AT&T](#)

Capítulo 5: roteiro

5.1 introdução

5.2 protocolos de roteamento

- estado de enlace
- vetor de distâncias

5.3 Roteamento intra-AS na Internet: OSPF

5.4 roteamento entre os ISPs: BGP

5.5 O plano de controle do SDN

5.6 ICMP: o *Internet Control Message Protocol*

5.7 Administração de rede e o SNMP

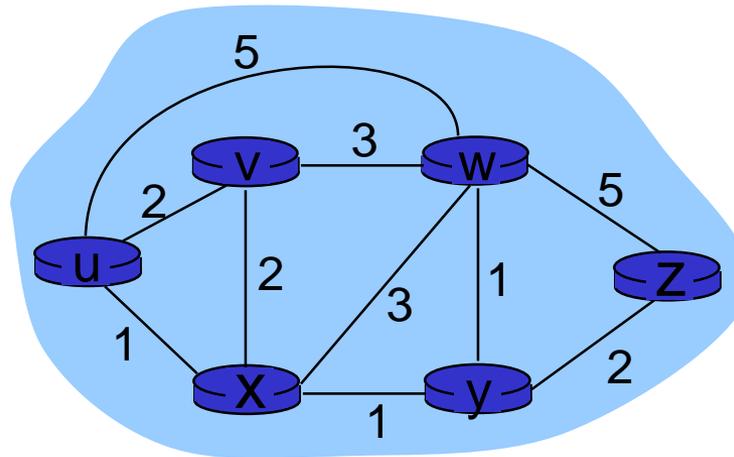
Protocolos de Roteamento

Objetivo de um protocolo de roteamento:

determinar “bons” caminhos (equivalentemente, rotas), do *host* fonte ao *host* destino, através da rede de roteadores

- **caminho**: sequência de roteadores que os pacotes atravessarão indo de um dado *host* fonte inicial até um dado *host* destino
- “**bom**”: menor “custo”, “mais rápido”, “menos congestionado”
- roteamento: um desafio “top-10” da área de redes!

Abstração - Grafo



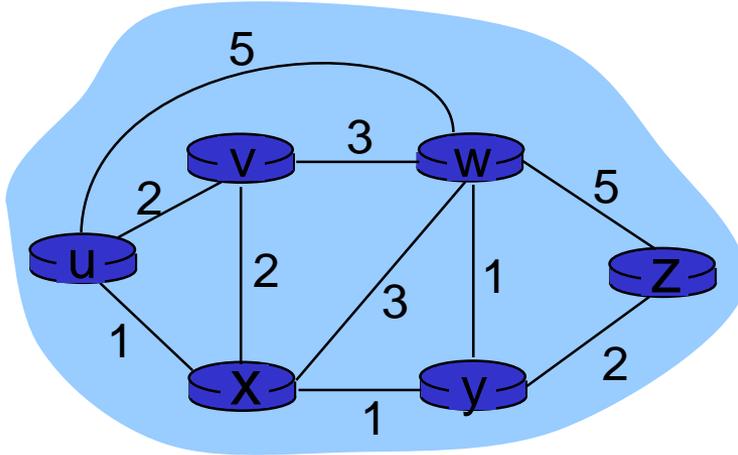
grafo: $G = (N, E)$

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

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

nota: grafos também são úteis em outros contextos de rede, e.g., P2P, em que N é conjunto de peers e E é conjunto de conexões TCP

Grafo: custos



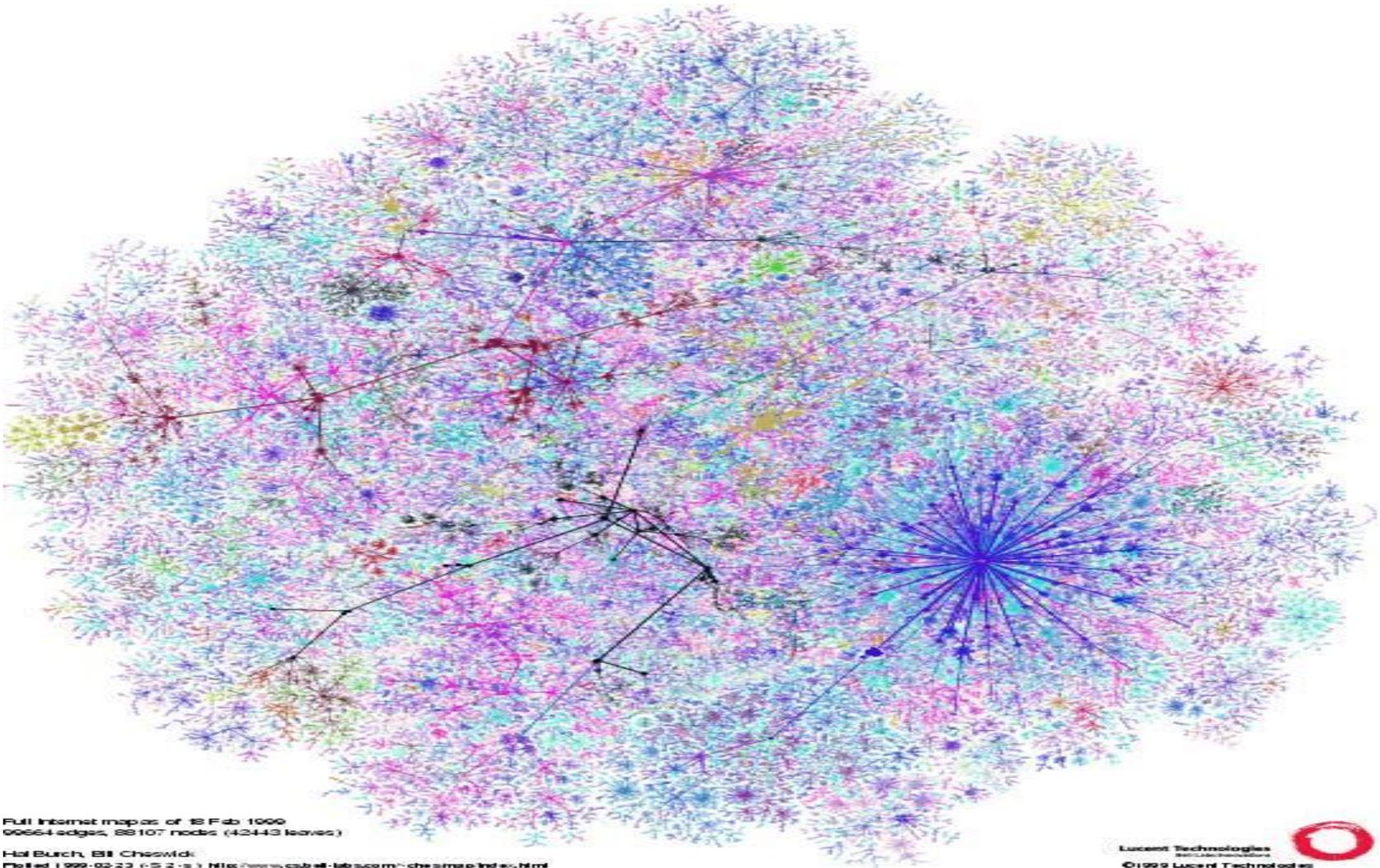
$c(x,x')$ = custo do enlace (x,x')
e.g., $c(w,z) = 5$

custo pode ser sempre 1, ou
inversamente relacionado à capacidade,
ou inversamente relacionado ao
congestionamento, etc.

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

questão chave: qual é o caminho de menor custo entre u e z?
algoritmo de roteamento: algoritmo que encontra o caminho de menor custo

Exemplo de Grafo da Internet



<https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/atlas.html>

Classificação dos algoritmos de roteamento

Q: informação global ou descentralizada?

global:

- ❖ todos roteadores têm informações completas sobre topologia e custos dos enlaces
- ❖ algoritmos “estado de enlace”

descentralizado:

- ❖ roteadores conhecem vizinhos fisicamente conectados e custos dos enlaces para vizinhos
- ❖ processo iterativo de computação, troca de informações com vizinhos
- ❖ algoritmos “vetor de distâncias”

Q: estático ou dinâmico?

estático:

- ❖ rotas mudam lentamente no tempo

dinâmico:

- ❖ rotas mudam mais rapidamente
 - atualização periódica
 - em resposta a mudanças nos custos dos enlaces

Capítulo 5: roteiro

5.1 introdução

5.2 protocolos de roteamento

❖ estado de enlace

❖ vetor de distâncias

5.3 Roteamento intra-AS na Internet: OSPF

5.4 roteamento entre os ISPs: BGP

5.5 O plano de controle do SDN

5.6 ICMP: o *Internet Control Message Protocol*

5.7 Administração de rede e o SNMP

Um algoritmo de roteamento estado de enlace

Algoritmo de Dijkstra

- ❖ topologia da rede e custos dos enlaces conhecidos por todos os nós
 - realizado via “broadcast do estado de enlace”
 - todos os nós têm mesma informação
- ❖ computa caminhos de menor custo de um nó (“fonte”) para todos os outros nós
 - fornece *tabela de repasse* para aquele nó
- ❖ iterativo: depois de k iterações, sabe caminho de menor custo para k destinos

notação:

- ❖ $C(x,y)$: custo do enlace do nó x para y ; $= \infty$ se não são vizinhos diretos
- ❖ $D(v)$: valor atual do custo do caminho da fonte até dest. v
- ❖ $p(v)$: nó predecessor no caminho da fonte até v
- ❖ N^i : conjunto de nós cujo caminho de menor custo é definitivamente conhecido num certo passo do algoritmo

Algoritmo de Dijkstra

1 **Inicialização – calculo da tabela de repasse do nó u :**

2 $N' = \{u\}$

3 para todos nós v

4 se v adjacente a u

5 então $D(v) = c(u,v)$

6 senão $D(v) = \infty$

7

8 **Laço**

9 encontrar w não em N' tal que $D(w)$ é mínima

10 adicionar w a N'

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

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

13 /* novo custo até v é ou o custo anterior até v ou o custo do

14 menor caminho conhecido até w mais o custo de w até v */

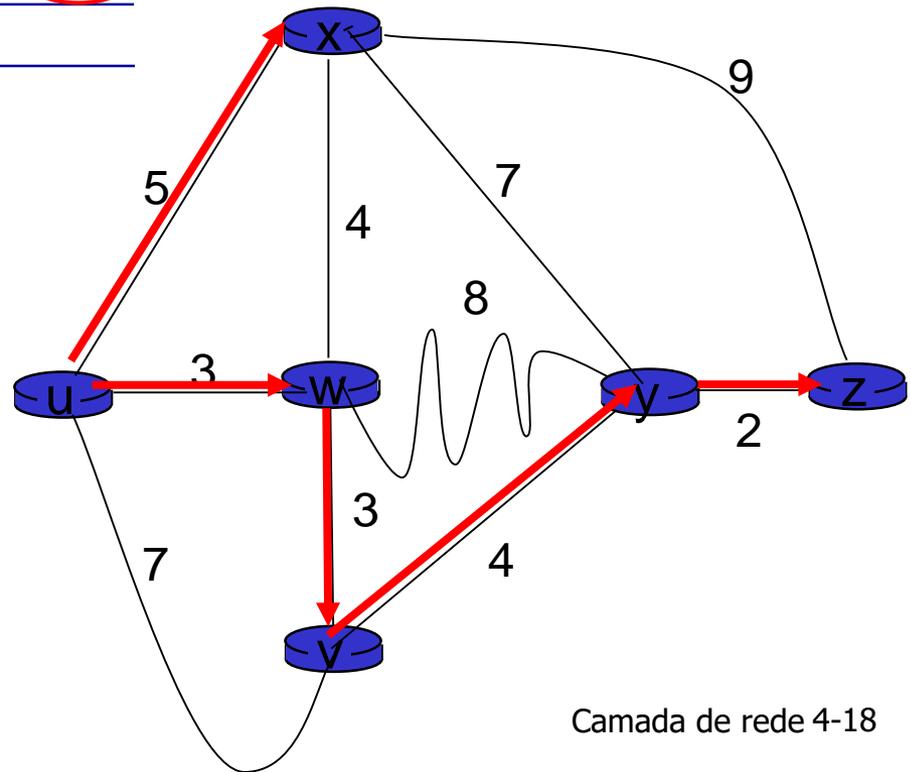
15 **até que todos os nós estejam em N' ($N'=N$)**

Algoritmo de Dijkstra: exemplo

Passo	N'	$D(v)$ $p(v)$	$D(w)$ $p(w)$	$D(x)$ $p(x)$	$D(y)$ $p(y)$	$D(z)$ $p(z)$
0	u	$7, u$	$3, u$	$5, u$	∞	∞
1	uw	$6, w$		$5, u$	$11, w$	∞
2	uwx		$6, w$		$11, w$	$14, x$
3	$uwxv$				$10, v$	$14, x$
4	$uwxvy$					$12, y$
5	$uwxvyz$					

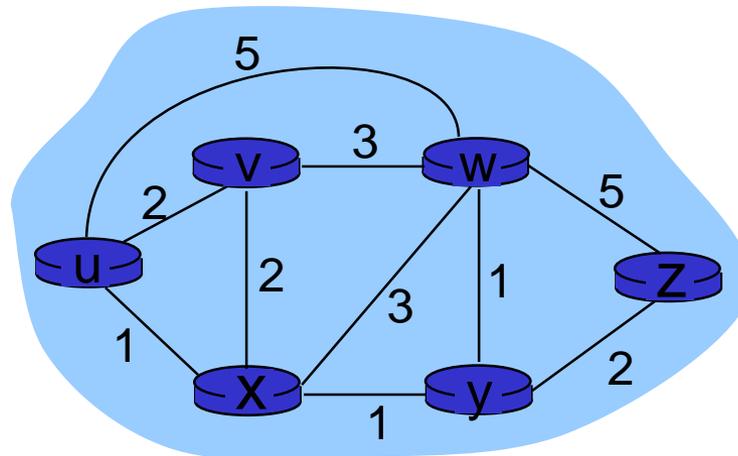
notas:

- ❖ construir árvore de caminhos mais curtos seguindo nós precedentes
- ❖ empates podem existir (podem ser resolvidos arbitrariamente)



Algoritmo de Dijkstra: um outro exemplo

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,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



Algoritmo de Dijkstra: um outro exemplo (cont)

árvore de menores caminhos resultante a partir de u :

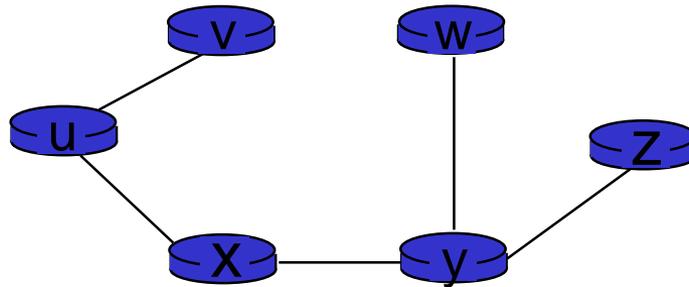


tabela de repasse resultante em u :

destino	enlace
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Exercício: Algoritmo de Dijkstra

1. (Kurose e Ross, 2014, p. 312) Considere a seguinte rede. Com os custos de enlace indicados, use o algoritmo do caminho mais curto de Dijkstra para calcular o caminho mais curto de x até todos os nós da rede. Obtenha a tabela de repasse e a árvore de menores caminhos a partir de x .

