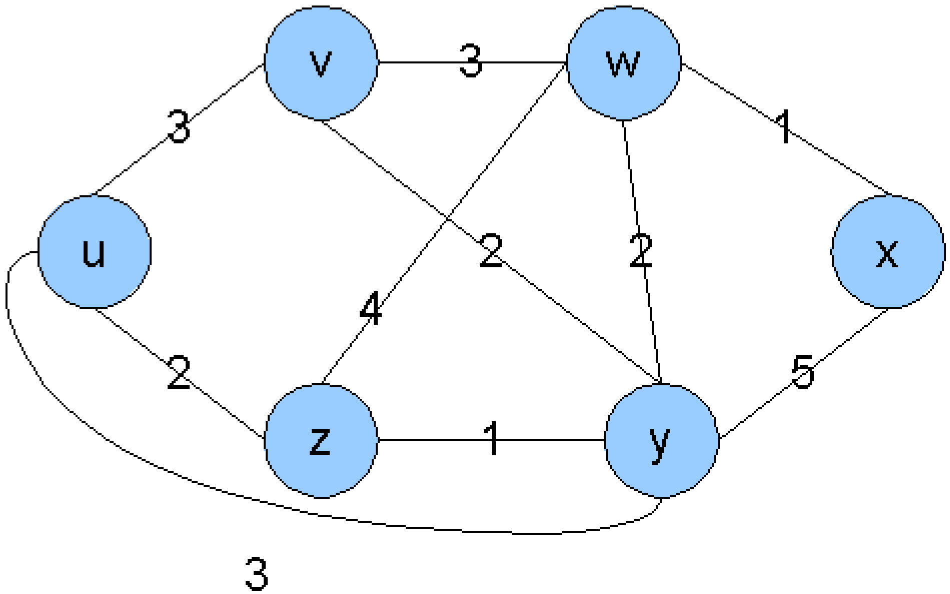
Telematics
Homework #5

Niklas Neumann

26 November 2009

Dijkstra's algorithm

- Given the following network, use Dijkstra's algorithm to find the least cost paths from node u. Please provide a table showing the steps of the algorithm, a graph showing the resulting shortest-path tree from u and the final forwarding table of u.



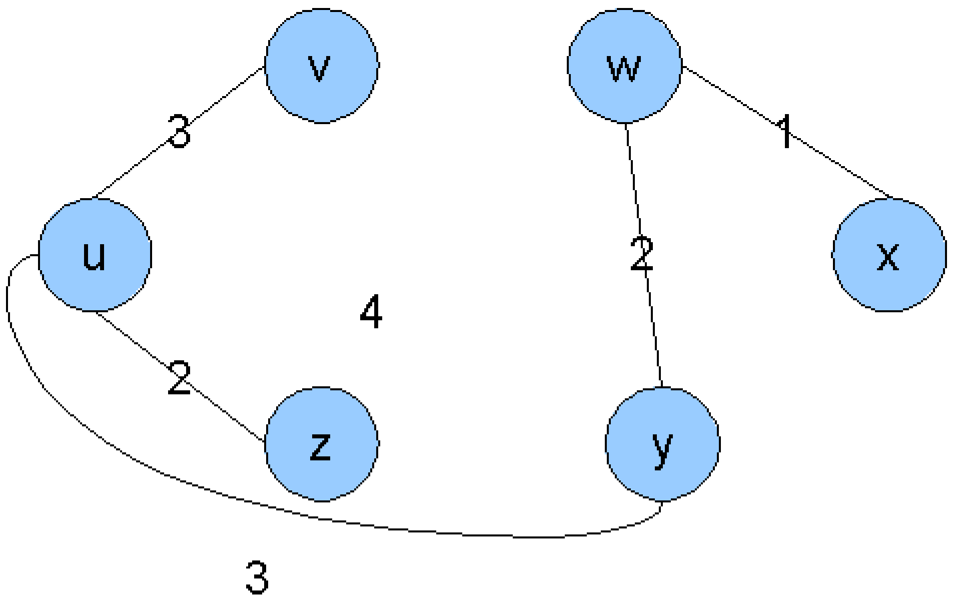
Dijkstra's algorithm (cont’d)

Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)

Dijkstra's algorithm (cont’d)

Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	∞	∞	3,u	2,u
1	uz	3,u	6,z	∞	3,u	
2	uzy	3,u	5,y	8,y		
3	uzyv		5,y	6,w		
4	uzyvw			6,w		
5	uzyvwx					

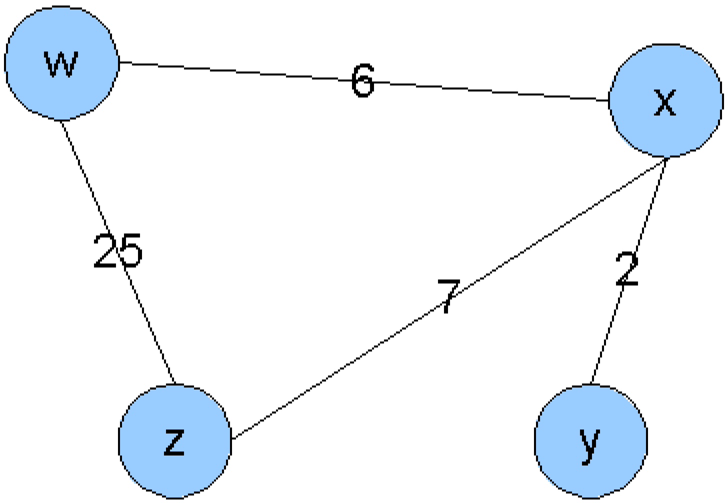
Dijkstra's algorithm (cont’d)



Dest.	Link.
z	z
y	y
v	v
w	y
x	y

Distance Vector algorithm

- Given the following network, use the Distance Vector algorithm to find the least cost paths for all nodes. Fill the provided tables and indicate with arrows between the tables when a node sends a distance vector to another node.



Distance Vector algorithm

Node w		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node w		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node w		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node w		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node x		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node x		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node x		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node x		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node y		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node y		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node y		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node y		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node z		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node z		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node z		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Node z		cost to			
		w	x	y	z
from	w				
	x				
	y				
	z				

Distance Vector algorithm

Node w (from \ cost to)	w	x	y	z
w	**0**	**6**	∞	**25**
x	∞	∞	∞	∞
y	∞	∞	∞	∞
z	∞	∞	∞	∞

Node w (from \ cost to)	w	x	y	z
w	**0**	**6**	**8**	**13**
x	6	0	2	7
y	∞	∞	∞	∞
z	25	7	∞	0

Node w (from \ cost to)	w	x	y	z
w	0	6	8	13
x	6	0	2	7
y	∞	∞	∞	∞
z	13	7	9	0

Node w (from \ cost to)	w	x	y	z
w				3
x				
y				
z				

Node x (from \ cost to)	w	x	y	z
w	∞	∞	∞	∞
x	**6**	**0**	**2**	**7**
y	∞	∞	∞	∞
z	∞	∞	∞	∞

Node x (from \ cost to)	w	x	y	z
w	0	6	∞	25
x	6	0	2	7
y	∞	2	0	∞
z	25	7	∞	0

Node x (from \ cost to)	w	x	y	z
w	0	6	8	13
x	6	0	2	7
y	8	2	0	9
z	13	7	9	0

Node x (from \ cost to)	w	x	y	z
w				
x				
y				
z				

Node y (from \ cost to)	w	x	y	z
w	∞	∞	∞	∞
x	∞	∞	∞	∞
y	∞	**2**	**0**	∞
z	∞	∞	∞	∞

Node y (from \ cost to)	w	x	y	z
w	∞	∞	∞	∞
x	6	0	2	7
y	**8**	**2**	**0**	**9**
z	∞	∞	∞	∞

Node y (from \ cost to)	w	x	y	z
w	∞	∞	∞	∞
x	6	0	2	7
y	8	2	0	9
z	∞	∞	∞	∞

Node y (from \ cost to)	w	x	y	z
w				
x				
y				
z				

Node z (from \ cost to)	w	x	y	z
w	∞	∞	∞	∞
x	∞	∞	∞	∞
y	∞	∞	∞	∞
z	**25**	**7**	∞	**0**

Node z (from \ cost to)	w	x	y	z
w	0	6	∞	25
x	6	0	2	7
y	∞	∞	∞	∞
z	**13**	**7**	**9**	**0**

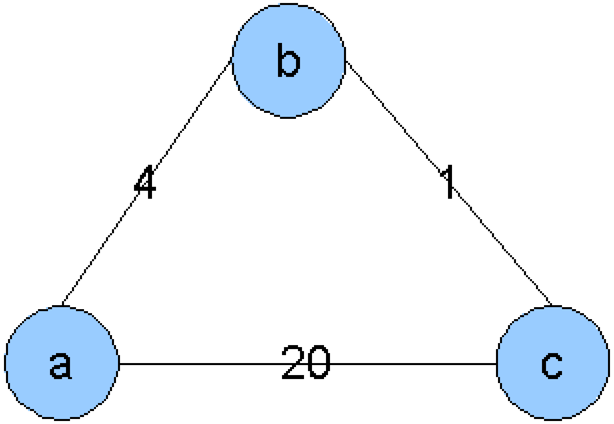
Node z (from \ cost to)	w	x	y	z
w	0	6	8	13
x	6	0	2	7
y	∞	∞	∞	∞
z	13	7	9	0

Node z (from \ cost to)	w	x	y	z
w				
x				
y				
z				

Count-to-infinity problem

- Q: Explain the count-to-infinity problem using a simple example. How can this problem be avoided?

- Consider the following example:



Count-to-infinity problem (con't)

- b sends DV (a,4) to c
- c computes that it can reach a in 5 hops via b and sends DV (a,5) to b
- Now the cost of the a-b changes to 30
- b recomputes its DV to a
  - Using the (old) DV (a,5) from c it computes the DV (a,6) and sends it to c
- c recomputes its DV to a
  - Using the DV from b it computes the DV (a,7) and sends it to b

Count-to-infinity problem (con't)

- The last two steps repeat with an increasing DV to reach a until b sends the DV (a,20)
- c recomputes its DV to a
  - Using the DV from b it determies that DV (a,20) using the link c-a is less costly and sends it back to b
- b recomputes ist DV to a
  - Using the DV (a,20) from c it computes the DV (a,21) and sends it to c
- Now the system is finally stable again

Count-to-infinity problem (con't)

- The count-to-infinitiy problem can be avoided using the poisoned reverse technique.
  - Using the poisoned reverse technique, a router will advertise a distance as infite to another router if that router is on the advertised path
  - In the example, router c will advertise it has an infinite cost to reach router a in its DV to router b as long as router c will route its own packets to a via c

Hierarchical routing

- Q: Explain the concept of hierarchical routing. Why is it needed?
- Hierarchical routing aggregates networks into Autonomous Systems (AS)
  - AS can run different intra-AS routing protocols
  - AS are connected via gateway routers running an inter-AS routing protocol
- HR is needed because flat routing …
  - … does not scale
  - --- does not consider different administrative domains

RIP

- Q: What is RIP and what metric does it use?
- The Routing Information Protocol (RIP) is a simple routing protocol which …
  - … distributes distance vectors
  - … using RIP advertisement messages
  - … which use a simple hop count as metric

Routing policies

- Q: How are routing policies used in BGP. Give one example.
- Routing policies determine …
  - … which BGP advertisements to regard
  - … which routes to advertise
- Example
  - AS x is connected to AS y and AS z
  - Policy : AS x does not want AS y to route traffic via AS x to AS z
  - Therefore, AS x does not advertise any route to reach AS z to AS y

Intra- vs. inter-AS routing

- Q: Why are different inter-AS and intra-AS protocols used in the Internet?
- Different policies
  - Inter-AS: control over how (foreign) traffic is routed via the own network
  - Intra-AS: control over how traffic is routed within the the own network
- Scale
  - Hierarchical routing saves table size, reduced update traffic

Intra- vs. inter-AS routing

- Performance
  - Intra-AS: can focus on performance
  - Inter-AS: policy may dominate over performance

Thank you

Any questions?