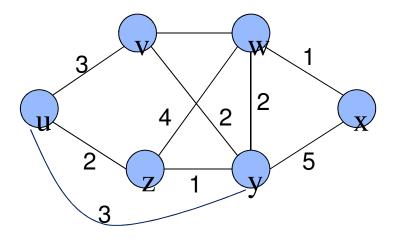
Exercise 5

Narisu Tao narisu.tao@informatik.uni-goettingen.de



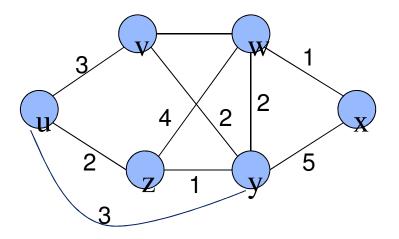
Dijkstra's algorithm

 Q1: 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 shortestpath tree from u and the final forwarding table of u.



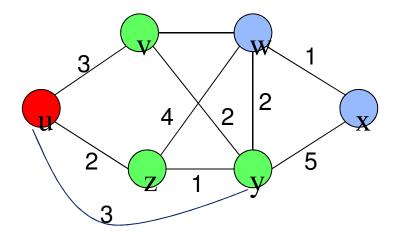


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)



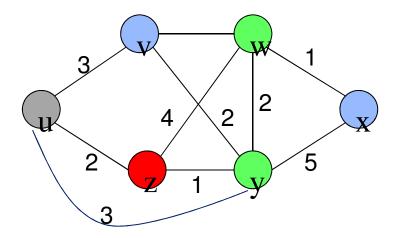


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	×	×	3,u	2,u
1						
2						
3						
4						
5						



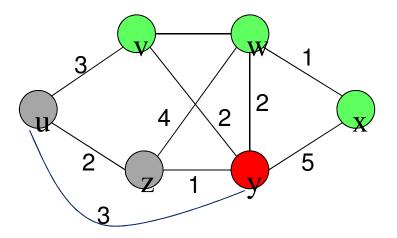


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	×	×	3,u	2,u
1	uz	3,u	6,z	×	3,u	
2						
3						
4						
5						



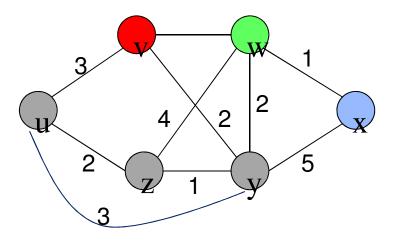


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	∞	×	3,u	2,u
1	uz	3,u	6,z	×	3,u	
2	uzy	3,u	5,y	8,y		
3						
4						
5						



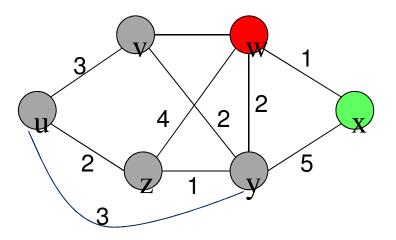


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	×	×	3,u	2,u
1	uz	3,u	6,z	×	3,u	
2	uzy	3,u	5,y	8,y		
3	uzyv		5,y	6,w		
4						
5						



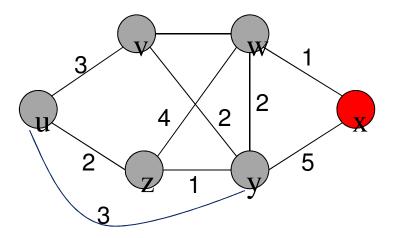


Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	×	×	3,u	2,u
1	uz	3,u	6,z	×	3,u	
2	uzy	3,u	5,y	8,y		
3	uzyv		5,y	6,w		
4	uzyvw			6,w		
5						

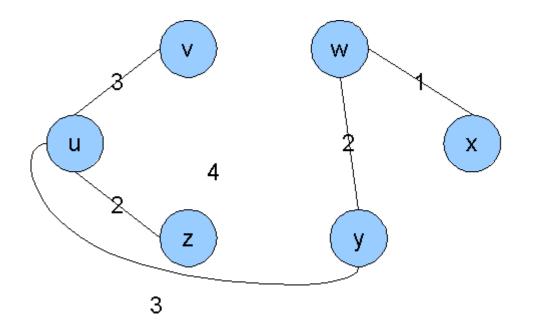




Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	3,u	∞	∞	3,u	2,u
1	uz	3,u	6,z	×	3,u	
2	uzy	3,u	5,у	8,y		
3	uzyv		5,y	6,w		
4	uzyvw			6,w		
5	uzyvwx					





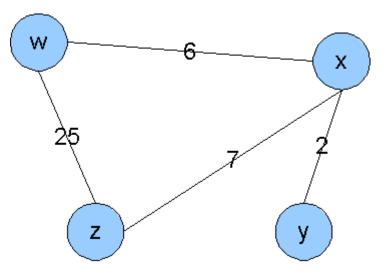


Dest.	Link.
Z	Z
у	у
V	V
W	у
X	у



Distance Vector algorithm

 Q2: 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

x w

c

x w

w

Node		cost to					Node	
W	7	w	x	у	z		w	
	w							w
rom	х						nom	x
ffro	у						fro	y
	z							z

cos	ost to			Node			cos	t to
х	у	z		w		w	x	у
					w			
				rom	x			
				ffc	y			
					z			

	No	Node w		cost to				
	v			x	у	z		
		w						
	from	x						
		y						
	z							

z

Node		cost to					Node	
3	2	w	x	у	z		x	
	w							w
from	x						rom	x
ffrc	y						ffrc	у
	z							z

cos	t to		No	de		cos	t to	
х	у	z	3	2	w	x	у	z
				w				
			rom	x				
			ffc	y				
				z				

	No	de		cost to				
	3		w	x	у	z		
		w						
	rom	x						
	frc	у						
		z						

ſ	Node y		cost to					No	de
			w	х	у	z		3	7
		w							w
	from	x						m	x
	ffro	у						fro	у
l		z							z

cos	t to		No	de		cos	t to	
х	у	z	у		w	x	у	z
				w				
			m	x				
			ffro	y				
				z				

w x

	No	de		cos	t to	
	3	7	w	x	у	z
		w				
	from	x				
		у				
		z				

No	de		N			
z	2	w	x	у	z	
	w					
rom	x					www
firo	у					3
	z					

No	de		cos	t to		No	de
z		w	x	у	z	2	z
	w						w
m	x					E	x
fro	y					ffro	у
	z						z

cost to			No	de	cost to			
x	у	z	z		w	x	у	z
				w				
			rom	x				
			ffro	у				
				z				



Distance Vector algorithm

cost to

z v

13

7 2

00

0 œ

Node w

w

х

w x

0 6 8

6 0

8 œ œ

25

w х v

œ

6

 ∞

w х

0 6

6

80

13

w

х

v

z

No	de		COS	t to		
v	v	w	x	у	z	
	w	0	6	œ	25	
from	x	80	80	8	8	
ffc	у	œ	œ	80	8	ľ
	z	80	80	8	80	l

	z	œ	œ	œ	80	z
						IXT
Nod	1		cos	t to		Nela
1400	le x	w	х	у	z	INCLE
	w	œ	80	80	80	W
E	x	6	0	2	7	
from	у	œ	œ	œ	80	Jaly
	z	œ	œ	œ	80	

	f	У	80	80	80	80	у у
		z	8	8	8	80	z
							W T
	Node y			cos			
			w	x	у	z	ode y
		w	80	8	8	œ	w
	Ξ	x	8	8	8	80	x

	w	80	80	80	80	
E	x	8	8	8	8	/
ffo	у	8	2	0	8	
	z	8	8	8	8	
						n

Ne	Node z		cos	t to			Ţ
1404			x	у	z	ľ.,	1.1
	w	œ	œ	æ	80		
Ξ	x	8	80	8	8		ε
fron	у	œ	œ	80	80	F	fror
	z	25	7	œ	0	ſ	

	ł						
	Y.			cos	t to		
1	ļ	еx	w	x	у	z	
		w	0	6	8	25	
ſ		x	6	0	2	7	
E		у	8	2	0	8	
ļ	ļ	z	25	7	8	0	

cost to

2

0

00

cost to

y

œ

2

œ œ

0

2 8

8

0

œ œ

7 9 z

80

00

z

25

7

80

0

				11
	cos	t to		N
w	x	у	z	1.4
0	6	8	25	
6	0	2	7	
8	2	0	8	
25	7	80	0	ľ

No	Node		cost to			
w		w	x	у	z	
	w	0	6	8	13	
from	x	6	0	2	7	
ffrc	у	8	8	8	8	
	z	13	7	9	0	

	No	Node w		cost to				
	W			x	у	z		
3		w				3		
	from	x						
	fic	у						
		z						

	1	cost to					
T	le x	w	x	у	z		
Т	w	0	6	8	13		
ł	x	6	0	2	7		
3	у	8	2	0	9		
11	z	13	7	9	0		

le y

w œ

Noce z

from х

w

ly

z 13 7

w х

6

8

00

w х v

0

6

œ

cost to

2

œ œ

0

2 0

œ œ

6

0 2

œ

cost to

8

œ

9

		Ned	Node x		cos	t to	
		1400			x	у	z
;			w				
		from	x				
			у				
			z				

	Ned	la 11		cos	t to	
z	Nod	ie y	w	x	у	z
8		w				
7	from	x				
9	fro	у				
8		z				

	Ned			cos	t to	
z	Nod	le z	w	x	у	z
13		w				
7	from	x				
8	fro	у				
0		z				



Comparison LV vs. DV

- Q3:Compare Link State routing algorithms to Distance Vector algorithms in terms of scalability and robustness.
- o Scalability
 - LS uses broadcasts to disseminate complete knowledge about all links to entire network
 - DV only sends (local) information to neighboring nodes.
 Convergence time and DV size still increase with network size
- o Robustness
 - $_{\circ}$ LS: every router does its own calculations
 - DV: wrong DV will be used by neighboring nodes and further propagate the error



Count-to-infinity problem

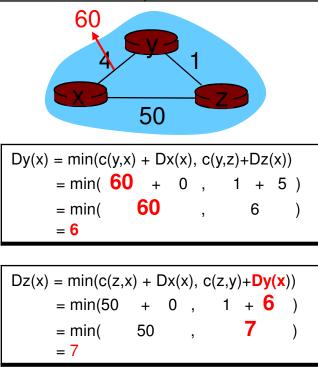
- Q4: Explain the count-to-infinity problem using a simple example. How can this problem be avoided?
- Link cost changes:

44 iterations before algorithm stabilizes: see textbook

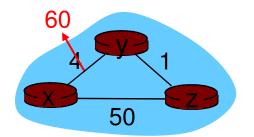
50



Dy(x) = min(c(y,x) + Dx(x), c(y,z)+Dz(x))	Dy(z) = min(c(y,x) + Dx(z), c(y,z)+Dz(z))
= min(4 + 0, 1 + 5)	= min(4 + 5, 1 + 0)
= min(4 , 6)	= min(9, 1)
= 4	= 1
Dz(x) = min(c(z,x) + Dx(x), c(z,y)+Dy(x))	Dz(y) = min(c(z,x) + Dx(y), c(z,y)+Dy(y))
= min(50 + 0 , 1 + 4)	= min(50 + 4 , 1 + 0)
= min(50 , 5)	= min(54 , 1)
= 5	= 1







 $\begin{array}{rl} Dy(x) = \min(c(y,x) + Dx(x), \ c(y,z) + Dz(x)) \\ &= \min(\begin{array}{cc} 60 & + & 0 \\ &= \min(\begin{array}{cc} 60 \\ &, \end{array} \begin{array}{c} 50 \end{array} \end{array}) \\ &= 50 \end{array}$

Dz(x) = min(c(z,x) + Dx(x), c(z,y)+Dy(x))= min(50 + 0 , 1 + 50) = min(50 , 51) = 50



Count-to-infinity problem (con't)

- Count-to-infinitiv problem can be avoided using the poisoned reverse technique.
 - Router A will advertise a distance as infinite to Router B if Router B is on the advertised path
 - In the example: In its advertisements to Router B, Router C will advertise the cost to reach Router A as infinite as long as it routes packets to A via B
 - Poisoned reverse will only prevent routing loops that involve just two gateways. It is still possible to end up with patterns in which three gateways are engaged in mutual deception. E.g. A may believe it has a route through B, B through C, and C through A.



Routing policies

- Q5: How are routing policies used in BGP.
 Give one example.
- Routing policies determine ...
 - ... which BGP advertisements to regard
 - ... which routes to advertise
- \circ Example
 - AS A is connected to AS B and AS C
 - Policy : AS A does not want AS B to route traffic via AS A to AS C

В

С

Α

Therefore, AS A does not advertise any route to reach AS C to AS B



Intra- vs. inter-AS routing

- Q6 :What is the difference between Intra-AS and Inter-AS routing? Why are different inter-AS and intra-AS protocols used in the Internet? ? Name one example for each category.
- 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
- o 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
- \circ Example:
 - Intra-AS: RIP, OSPF
 - Inter-AS: BGP





Any questions?

