Link-State (LS) & Distance Vector (DV) Algorithms.

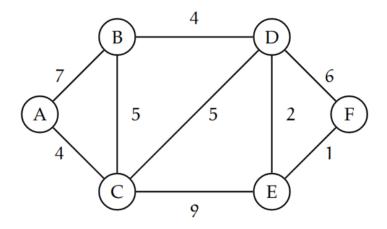
Worked through examples.

Link-State (LS) Algorithm for Source Node *u*

```
Initialization:
  N' = \{u\}
   for all nodes v
     if v is a neighbor of u
     then D(v) = c(u, v)
     else D(v) = \infty
  Loop
  find w not in N' such that D(w) is a minimum
10 add w to N'
11 update D(v) for each neighbor v of w and not in N':
12
          D(v) = \min(D(v), D(w) + C(w,v))
13 /* new cost to v is either old cost to v or known
least path cost to w plus cost from w to v */
15 until N' = N
```

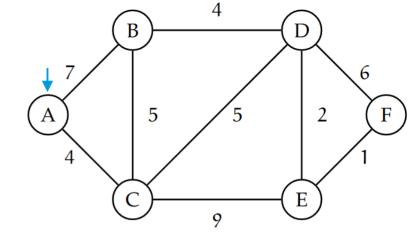
Re-exam 2021-22, Question 3.3.1

Question 3.3.1: Consider the network topology outlined in the graph below.

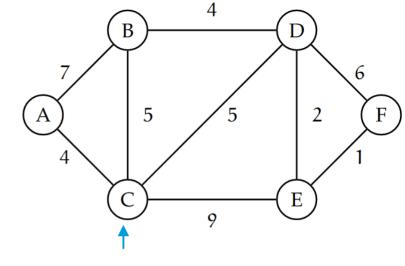


Apply the link state routing algorithm and compute the forwarding table on node **A** by filling out the following tables.

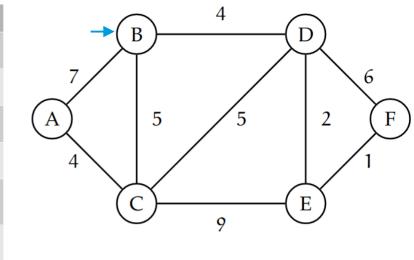
Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	А	7, A	4, A	∞	∞	∞



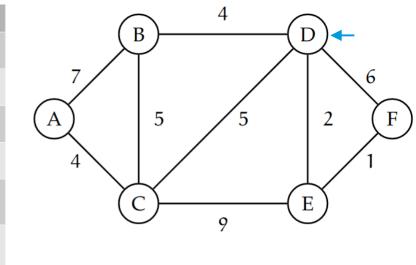
Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞



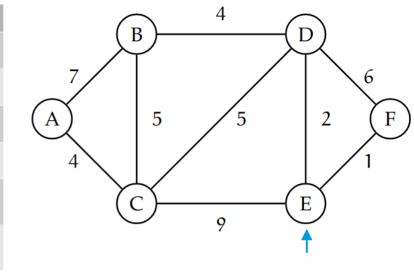
Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞
2	A, C, B	7, A		9, C	13, C	∞



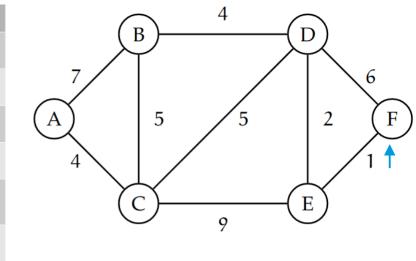
Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞
2	A, C, B	7, A		9, C	13, C	∞
3	A, C, B, D			9, C	11, D	15, D



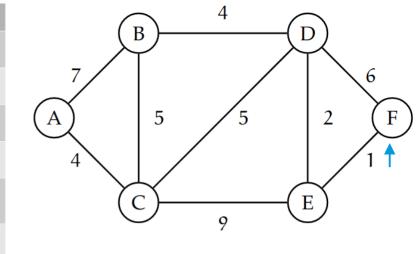
Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞
2	A, C, B	7, A		9, C	13, C	∞
3	A, C, B, D			9, C	11, D	15, D
4	A, C, B, D, E				11, D	12, E



Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞
2	A, C, B	7, A		9, C	13, C	∞
3	A, C, B, D			9, C	11, D	15, D
4	A, C, B, D, E				11, D	12, E
5	A, C, B, D, E, F					12, E



Step	N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E), p(E)	D(F), p(F)
0	Α	7, A	4, A	∞	∞	∞
1	A, C	7, A	4, A	9, C	13, C	∞
2	A, C, B	7, A		9, C	13, C	∞
3	A, C, B, D			9, C	11, D	15, D
4	A, C, B, D, E				11, D	12, E
5	A, C, B, D, E, F					12, E



Forwarding table for A:

	Destination	Edge
	В	(A,B)
des	С	(A,C)
	D	(A,C)
	Е	(A,C)
	F	(A,C)

p(X) = predecessor, the node before X, this can and will change as we relax more nodes
D(X) = distance (cost) to X, from our initial node, in this case A

01/11/2023				10
------------	--	--	--	----

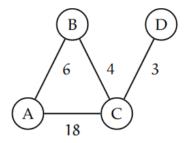
At each node, *x*:

```
Initialization:
       for all destinations y in N:
          D_{\nu}(y) = c(x,y)/* if y is not a neighbor then c(x,y) = \infty */
      for each neighbor w
          D_{..}(y) = ? for all destinations y in N
      for each neighbor w
          send distance vector \mathbf{D}_{\mathbf{y}} = [D_{\mathbf{y}}(\mathbf{y}): \mathbf{y} \text{ in } \mathbf{N}] to w
9 loop
10
              (until I see a link cost change to some neighbor w or
                  until I receive a distance vector from some neighbor w)
11
12
13
       for each y in N:
             D_{x}(y) = \min_{v} \{c(x,v) + D_{v}(y)\}
14
15
16 if D (y) changed for any destination y
            send distance vector \mathbf{D}_{\mathrm{x}} = [D<sub>x</sub>(y): y in N] to all neighbors
17
18
19 forever
```

P. 419 K&R.

Exam 2021-22, Question 3.3.1

Consider the network topology outlined in the graph below



Question 3.3.1: Describe how to apply the Distance Vector Algorithm for Node A in the above diagram, so that it gets a complete routing table to all other Nodes. You do not need to describe each individual step on each Node in detail, but should be able to provide a generalised description that could apply to any Node. For this answer you can assume no connection costs change.

Start of the algorithm, nodes only know price to neighbor

Α

	Α	В	С	D
Α	0	6	18	00
В	00	œ	00	∞
С	00	œ	00	00
D	00	∞	00	00

В

	Α	В	С	D
Α	∞	∞	∞	00
В	6	0	4	00
С	∞	œ	œ	∞
D	œ	00	œ	∞

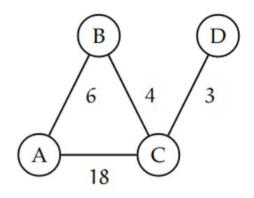
С

	Α	В	С	D	
Α	∞	œ	oo	∞	
В	oo	œ	œ	∞	
С	18	4	0	3	
D	∞	œ	œ	∞	

D

	А	В	С	D
Α	œ	œ	∞	∞
В	00	œ	∞	∞
С	00	œ	œ	∞
D	00	∞	3	0

 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N



Start of the
algorithm, nodes
only know price
to neighbor

Α

В

С

D

Distance vectors are exchanged to neighboring nodes, and local distance vectors are updated

	Α	В	С	D
Α	0	6	18	∞
В	œ	œ	œ	∞
С	œ	œ	œ	∞
D	00	00	00	∞

	Α	В	С	D
Α	0	6	10	21
В	6	0	4	00
С	18	4	0	3
D	00	œ	00	∞

	Α	В	С	D
Α	œ	œ	oo	o
В	6	0	4	∞
С	œ	œ	œ	o
D	co	_∞	∞	œ

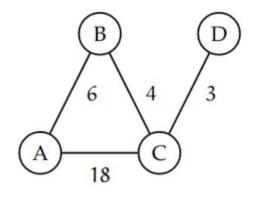
V		Α	В	С	D
X	Α	0	6	18	œ
	В	6	0	4	7
\	С	18	4	0	3
	D	oo	œ	œ	00
- 1	VL.				

	Α	В	С	D
Α	œ	œ	œ	∞
В	œ	∞	∞	∞
С	18	4	0	3
D	œ	∞	∞	∞

	1	Α	В	С	D
	Α	0	6	18	∞
	В	6	0	4	œ
	С	10	4	0	3
V	D	œ	œ	3	œ

	Α	В	С	D	\perp		А	В	С	D
Α	œ	œ	œ	00		Α	œ	œ	œ	∞
В	œ	œ	œ	00		В	œ	œ	œ	∞
С	œ	œ	œ	_∞		С	18	4	0	3
D	00	00	3	0		D	21	7	3	0





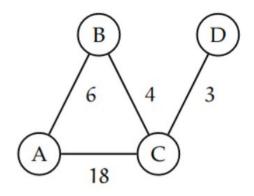
	Start o algorit only k to neig	hm, n now p			are and dis	stance excha d local tance upda	anged vecto	l,			D	_A(D) a _D(A) q al valu	get the	<u>.</u>	
	Α	В	С	D		A	В	С	D			Α	В	С	D
Α	0	6	18	œ	Α	0	6	10	21		Α	0	6	10	13
В	œ	œ	œ	œ	В	6	0	4	∞	\	В	6	0	4	7
С	œ	œ	00	00	С	18	4	0	3	1 4	С	10	4	0	3
D	∞	∞	∞	∞	D	∞	∞	∞	∞		D	∞	∞	∞	∞
	А	В	С	D		А	В	С	D	X		Α	В	С	D
Α	œ	00	00	œ	Α	0	6	18	∞	/V	Α	0	6	10	21
В	6	0	4	œ	В	6	0	4	7		В	6	0	4	7
С	œ	00	00	œ	С	18	4	0	3	V	С	10	4	0	3
D	œ	œ	∞	∞	D	œ	∞	∞	∞	\ \	D	œ	œ	∞	∞
	Α	В	С	D		Α	В	С	D	1 X4		Α	В	С	D
	, ,					, ,				$ I\rangle$, ,			
Α	œ	_∞	∞	oo	Α	0	6	18	∞		A	0	6	10	21
В	œ	00	∞	00	В	6	0	4	∞	Į,	В	6	0	4	7
С	18	4	0	3	С	10	4	0	3	\ 1	С	10	4	0	3
D	œ	œ	00	oo	D	∞	∞	3	0		D	21	7	3	0
	А	В	С	D		Α	В	С	D	X		А	В	С	D
Α	œ	∞	∞	œ	Α	∞	∞	∞	∞		Α	00	00	∞	∞
В	œ	∞	∞	oo	В	∞	∞	∞	∞		В	∞	∞	00	∞
С	œ	∞	∞	∞	С	18	4	0	3		С	10	4	0	3

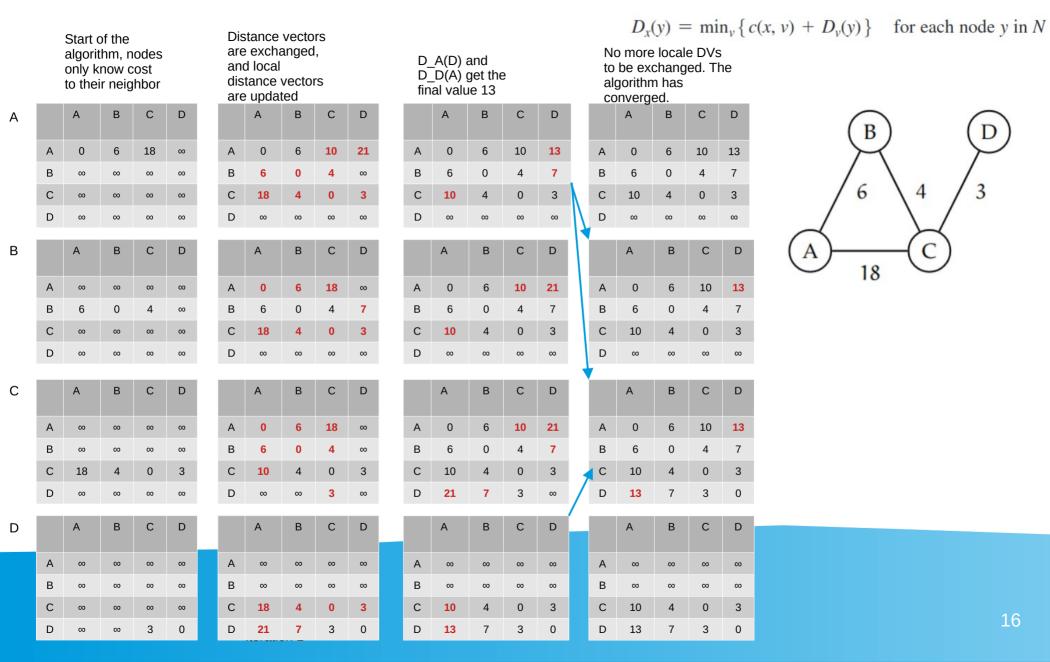
В

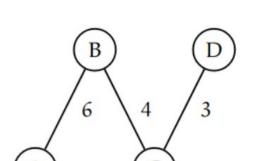
С

D

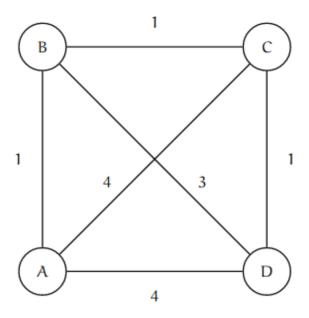
 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N







Exam 2020-21, Question 3.3.3



Question 3.3.3: Now the cost of the edge between A and D changes to 1, would a re-run of the algorithm converge? If yes, how many iterations of the algorithm would be run before convergence and why? If no, explain your reasoning. (Hint: it is recommended that you run the distance vector algorithm on paper to explore the answer.)

Converged start state

В

С

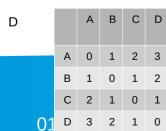
A B C D

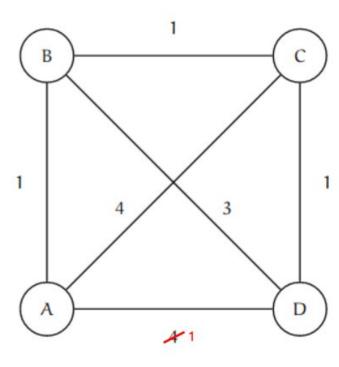
A 0 1 2 3

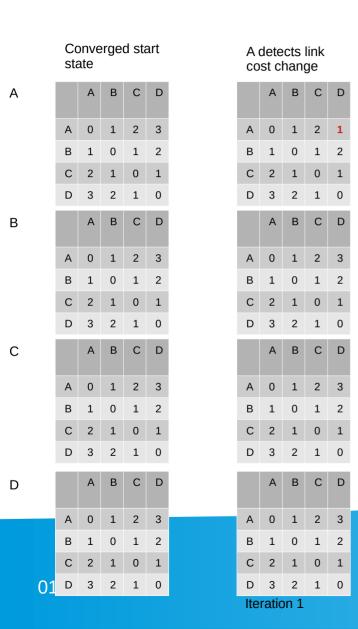
B 1 0 1 2

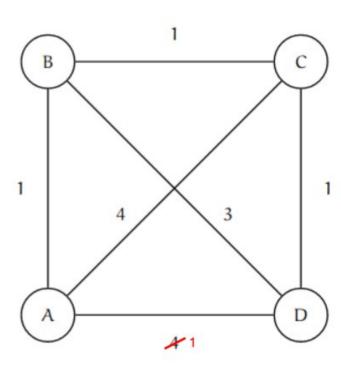
C 2 1 0 1

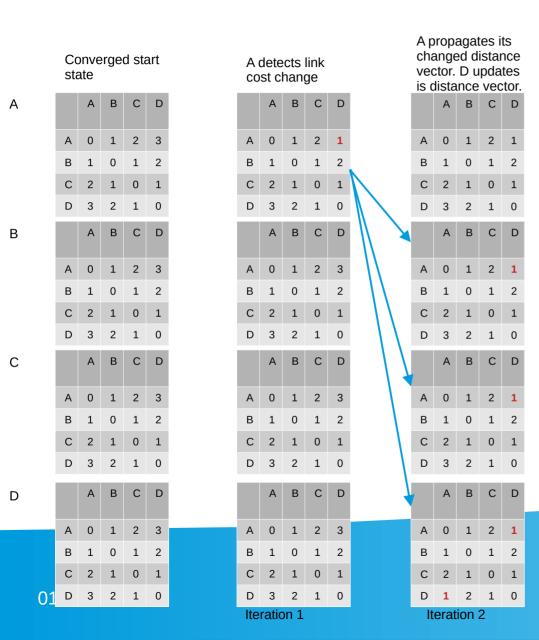
D 3 2 1 0

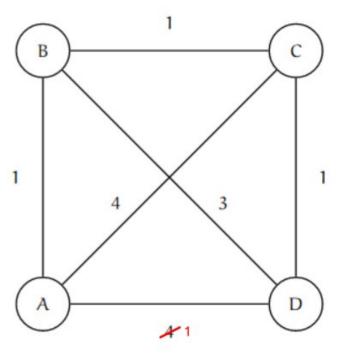


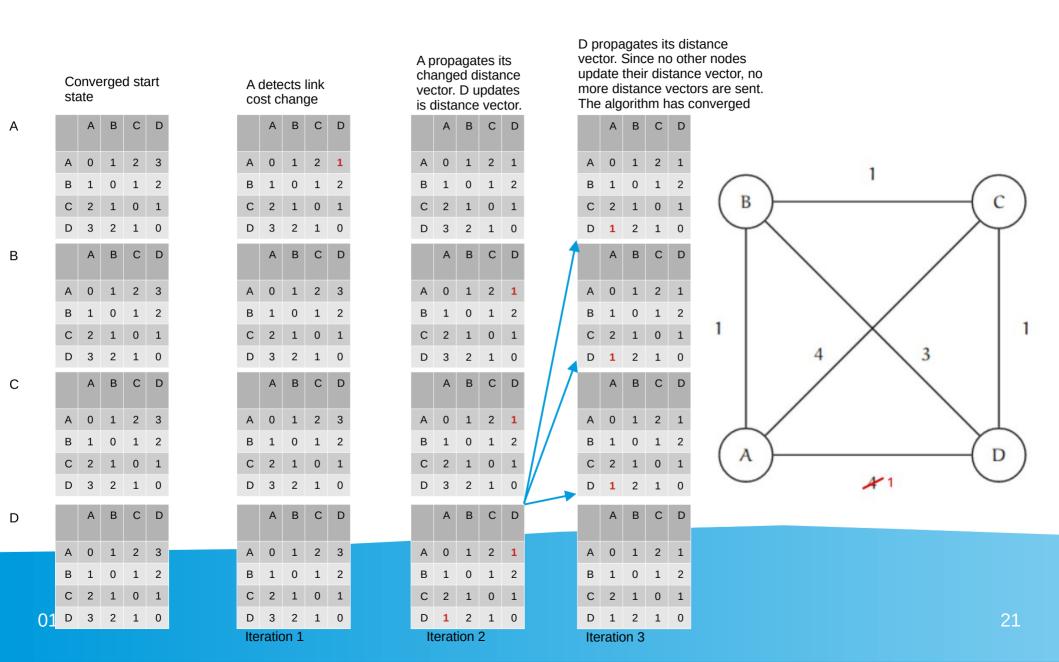




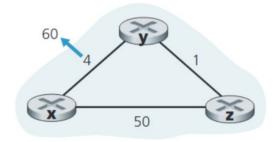








DVA - Example with cost increase from K&R, p. 423,



b.

Converged start state

	X and Y detect lin
	cost change. Y
art	"thinks" there is a
	cheaper way to x
	based on $D^{\prime}z(x)$

X		х	у	Z
	х	0	4	5
	У	4	0	1
	Z	5	1	0

	Х	у	Z
Х	0	51	50
у	4	0	1
Z	5	1	0

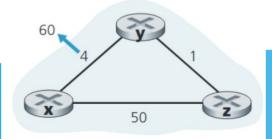
Y		х	у	Z
	х	0	4	5
	у	4	0	1
	Z	5	1	0

	х	У	Z
Х	0	4	5
у	6	0	1
Z	5	1	0

Z		X	У	Z
	Х	0	4	5
	у	4	0	1
	Z	5	1	0

	х	у	Z
х	0	4	5
У	4	0	1
Z	5	1	0
	-		

Iteration 1



b.

	Coi sta		ed sta	art		co: "th ch	X and Y detect link cost change. Y "thinks" there is a cheaper way to x based on D_z(x)				Y propogates its DV. Z calculates its D_z(x) based on the incorrect one received from Y.			
Χ		х	у	Z			х	у	Z			х	у	Z
	х	0	4	5		х	0	51	50		Х	0	51	50
	У	4	0	1		У	4	0	1	M	у	6	0	1
	Z	5	1	0		Z	5	1	0	W	Z	5	1	0
					l					M				
Y		X	У	Z			X	У	Z			X	У	Z
	х	0	4	5		х	0	4	5	M	Х	0	51	50
	У	4	0	1		У	6	0	1	N	у	6	0	1
	Z	5	1	0		Z	5	1	0	1	Z	5	1	0
Z														
۷		X	У	Z			X	У	Z			Х	у	Z
	Х	0	4	5		Х	0	4	5		х	0	51	50
	у	4	0	1		у	4	0	1		у	6	0	1

Iteration 1

z 5 1 0

z 7 1 Iteration 2



	Coi		ed sta	ırt	co: "th	st cha inks" eaper	detec ange. ` there way t n D_z	Y is a to x
Χ		X	у	Z		x	у	Z
	х	0	4	5	х	0	51	50
	У	4	0	1	У	4	0	1
	Z	5	1	0	z	5	1	0
Υ		X	У	Z		X	У	Z
	Х	0	4	5	Х	0	4	5

0

Iteration 1

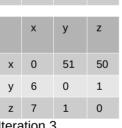
z 5

01/11/2023

ch	eaper	there way t n D_z	to x	or		ncorre eived	ect
	Х	у	Z		Х	У	Z
х	0	51	50	х	0	51	50
у	4	0	1	У	6	0	1
z	5	1	0	Z	5	1	0
	X	У	Z		X	У	Z
х	0	4	5	х	0	51	5
у	6	0	1	У	6	0	1
z	5	1	0	Z	5	1	0
	Х	У	Z		X	У	Z
х	0	4	5	х	0	51	5
у	4	0	1	у	6	0	1
7	5	1	0	7	7	1	0

Y propogates its DV. Z calculates its D_z(x) based on the incorrect one received from Y.					D_\	incor	s its ased rect D	
	Х	У	Z			Х	У	Z
х	0	51	50		х	0	51	50
у	6	0	1		У	6	0	1
Z	5	1	0		z	7	1	0
	X	у	Z			Х	у	Z
х	0	51	50		Х	0	51	50
у	6	0	1	\parallel	у	8	0	1
Z	5	1	0		z	7	1	0
	X	у	Z			Х	У	Z
х	0	51	50		х	0	51	50
у	6	0	1		у	6	0	1
Z	7	1	0		Z	7	1	0
lte	eration	າ 2			Itera	tion 3	3	

And so on, back and forth.



50

	cost change. Y
Converged start	"thinks" there is a
state	cheaper way to x
	based on D. 7(v)

<		х	У	Z
	х	0	4	5
	У	4	0	1
	z	5	1	0

cheaper way to x based on D_z(x)					
	х	У	Z		
Х	0	51	50		
У	4	0	1		
7	5	1	0		

X and Y detect link

	Х	У	Z
х	0	51	50
у	4	0	1
Z	5	1	0
	Х	٧	Z
		,	_

1

5 1

0

Iteration 1

	Х	у	Z
Х	0	51	50
у	6	0	1
Z	5	1	0
	Х	У	Z

Y propogates its

DV. Z calculates

its D z(x) based

on the incorrect

one received

from Y.

Z	5	1	0
	х	у	Z
х	0	51	50
у	6	0	1
z	5	1	0

	Х	у	Z
Х	0	51	50
у	6	0	1
Z	7	1	0
1+0	ration		

,			
Z	7	1	C
Ite	eration	າ 2	

b.

Y updates its $D_{\dot{Y}}(x)$ based on the incorrect DV from Z

	х	у	Z
х	0	51	50
У	6	0	1
Z	7	1	0

	Х	У	Z
х	0	51	50
у	8	0	1
Z	7	1	0

	X	У	Z
х	0	51	50
у	6	0	1
Z	7	1	0

Iteration 3

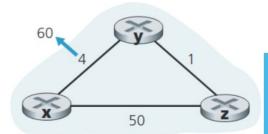
Until D_Y(x) is equal to And so on, back and the cost forth. C(Z,X)

	Х	у	Z
х	0	51	50
У	50	0	1
Z	49	1	0

	Х	У	Z
х	0	51	50
у	50	0	1
Z	49	1	0

	х	у	Z
х	0	51	50
У	50	0	1
Z	50	1	0

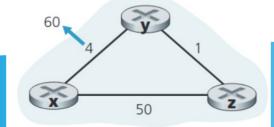
Iteration 46



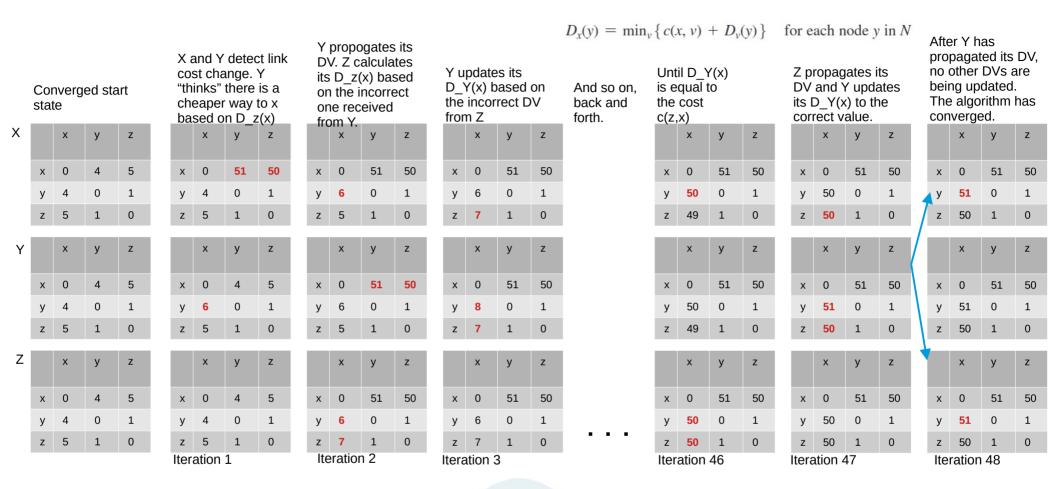
z 5

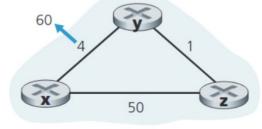
0

	Coi		ed sta	art	cos "thi	st cha inks" eaper	detections detection detec	Y is a to x	D' its or or	V. Z c s D_z n the i ne rec	ogates alcula (x) bas incorre ceived	tes sed ect	D_ the		es its based rrect D	on	And so on, back and forth.	Uni	il D_\ qual i cost		())	Z E it	' pr OV a	opag and \	ates i / upda :) to th	ts ates
Χ		Х	у	Z	bu	х	у	Z	Iro	om Y. ×	у	Z		x	у	Z	TOTUI.	0(2	X	у	Z			х	у	Z
	x	0	4	5	Х	0	51	50	х	0	51	50	х	0	51	50		x	0	51	50		x	0	51	50
	у	4	0	1	у	4	0	1	У	6	0	1	у	6	0	1		У	50	0	1		у	50	0	1
	Z	5	1	0	Z	5	1	0	Z	5	1	0	Z	7	1	0		Z	49	1	0		Z	50	1	0
Υ		Х	у	Z		Х	у	Z		х	у	Z		Х	у	Z			Х	у	Z	ı		Х	у	Z
	х	0	4	5	Х	0	4	5	х	0	51	50	x	0	51	50		x	0	51	50		х	0	51	50
	у	4	0	1	у	6	0	1	у	6	0	1	У	8	0	1		У	50	0	1	1	у	51	0	1
	Z	5	1	0	Z	5	1	0	Z	5	1	0	Z	7	1	0		Z	49	1	0		Z	50	1	0
Z		Х	у	Z		Х	у	Z		Х	у	Z		Х	у	Z			Х	у	Z			Х	У	Z
	х	0	4	5	х	0	4	5	х	0	51	50	х	0	51	50		x	0	51	50		x	0	51	50
	у	4	0	1	у	4	0	1	у	6	0	1	у	6	0	1		У	50	0	1		у	50	0	1
	Z	5	1	0	Z	5	1	0	Z	7	1	0	z		1	0		Z	50	1	0		Z	50	1	0
					Itera	ation	1		lte	eratio	n 2		Iter	ation	3			Iter	ation	46		lt	era	tion 4	17	



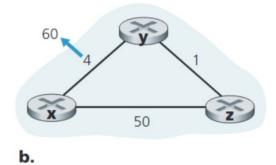
b.





b.

Adding a poisoned reverse, to solve the looping problem.



Converged start	
state	

X and Y detect link cost change. Y "thinks" there is a cheaper way to x based on D_z(x)

X		х	у	Z
	х	0	4	5
	у	4	0	1
	Z	5	1	0

		_	` '
	Х	у	Z
х	0	51	50
у	4	0	1
z	5	1	0

Y		Х	У	Z
	Х	0	4	5
	у	4	0	1
	Z	5	1	0

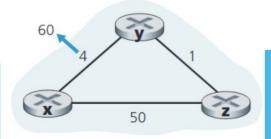
	Х	У	Z
Х	0	4	5
у	6	0	1
Z	5	1	0

_				
Z		X	у	Z
	х	0	4	5
	У	4	0	1
	z	5	1	0

	х	у	Z
Х	0	4	5
у	4	0	1
Z	5	1	0

Iteration 1

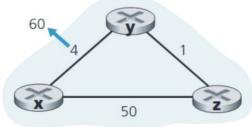
 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N



b.

•	V propagatos its														
	Cor stat		ed sta	ırt		co: "th	st cha inks" eaper	detectange. `there 'way f	D\ its or or	Y propogates its DV. Z calculates its D_z(x) based on the incorrect one received from Y.					
Χ		х	у	Z			х	у	Z			х	у	Z	
	х	0	4	5		х	0	51	50		Х	0	51	50	
	у	4	0	1		У	4	0	1		у	6	0	1	
	z	5	1	0		z	5	1	0		Z	5	1	0	
		_								./.				_	
Υ		X	У	Z			х	У	Z			X	У	Z	
	х	0	4	5		х	0	4	5		х	0	51	50	
	у	4	0	1		У	6	0	1	\setminus	у	6	0	1	
	z	5	1	0		z	5	1	0	$ \setminus $	Z	5	1	0	
Z		_								1				_	
_		X	У	Z			X	У	Z			X	У	Z	
	Х	0	4	5		Х	0	4	5		х	0	51	50	
	у	4	0	1		у	4	0	1		у	6	0	1	
	z	5	1	0		z	5	1	0		Z	7	1	0	
						Iter	ation	1			lte	eration	1 2		

 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N

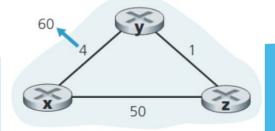


01/11/2023 **b.** 31

	Cor stat		ed sta	art	cos "th ch	st cha inks" eaper	detections detections detection dete	Y is a to x	D\ its or or	V. Z c D_z the	alcula (x) ba incorre	ites sed ect		rou		z is nrough _z(x) =	
Χ		х	у	Z		х	у	Z		х	у	Z			х	у	Z
	Х	0	4	5	х	0	51	50	Х	0	51	50		Х	0	51	50
	у	4	0	1	У	4	0	1	у	6	0	1		у	6	0	1
	Z	5	1	0	z	5	1	0	Z	5	1	0		Z	7	1	0
Υ		Х	у	Z		Х	у	Z		Х	у	Z			Х	у	Z
	х	0	4	5	х	0	4	5	Х	0	51	50		Х	0	51	50
	у	4	0	1	У	6	0	1	у	6	0	1		у	60	0	1
	Z	5	1	0	Z	5	1	0	Z	5	1	0	L	Z	∞	1	0
Z		Х	у	Z		Х	у	Z		Х	у	Z	1		Х	у	Z
	х	0	4	5	х	0	4	5	Х	0	51	50		Х	0	51	50
	у	4	0	1	у	4	0	1	у	6	0	1		у	6	0	1
	Z	5	1	0	Z	5	1	0	Z	7	1	0		Z	7	1	0
					Iter	ation	1		lte	ratio	n 2			tera	ation (3	

Y propogates its

 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N



b.

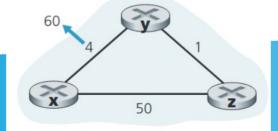
01/11/2023

32

$D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N

	Cor stat	_	ed sta	ırt	co: "th	st cha inks" eapei	detectange. there way n D_z	Y is a to x	D its or or	V. Z c s D_z n the i	ealcula (x) basincorre ceived	ites sed ect	rou		z is nrougl _z(x) =			cori	rect D upda to the	es the OV fror ates its e right	s
X		х	У	Z		X	у	Z		х	у	Z		х	у	Z			х	у	Z
	х	0	4	5	х	0	51	50	×	0	51	50	х	0	51	50		х	0	51	50
	У	4	0	1	У	4	0	1	У	6	0	1	У	6	0	1	4	У	60	0	1
	Z	5	1	0	Z	5	1	0	Z	5	1	0	Z	7	1	0		Z	7	1	0
Υ		х	у	Z		Х	у	Z		X	у	Z		Х	у	Z			Х	у	Z
	х	0	4	5	х	0	4	5	x	0	51	50	х	0	51	50		х	0	51	50
	У	4	0	1	у	6	0	1	У	6	0	1	у	60	0	1	\	У	8	0	1
	Z	5	1	0	Z	5	1	0	z	5	1	0	Z	∞	1	0	$ \setminus $	Z	∞	1	0
Ζ		-															. }				
_		Х	У	Z		X	У	Z		X	У	Z		X	У	Z			X	У	Z
	х	0	4	5	х	0	4	5	x	0	51	50	х	0	51	50		х	0	51	50
	у	4	0	1	У	4	0	1	У	6	0	1	У	6	0	1		У	60	0	1
	Z	5	1	0	Z	5	1	0	Z	7	1	0	Z	7	1	0		Z	50	1	0
					Iter	ation	1		Ite	eratio	n 2		Itera	ation (3			Iter	ation	4	

Y propogates its



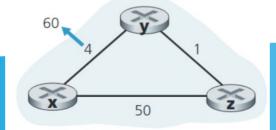
b.

$D_{x}(y) = \min_{y}$	$\{c(x, v) + D_v(y)\}$	for each node y in N
-----------------------	------------------------	----------------------

Converged start state			X and Y detect link cost change. Y "thinks" there is a cheaper way to x based on D_z(x)			y propogates its DV. Z calculates its D_z(x) based on the incorrect one received from Y.			Because z is routing through y, it sets $D_z(x) = \infty$			i i	Z receives the correct DV from y and updates its DV to the right value.			s	Z, no longer routing through y, propagates DV, and Y updates to the correct value.										
Х		х	у	Z			х	у	Z		X	у	Z			х	у	Z			х	у	Z		х	у	Z
	Х	0	4	5		х	0	51	50	Х	0	51	50		х	0	51	50		х	0	51	50	×	0	51	50
	у	4	0	1		У	4	0	1	у	6	0	1		У	6	0	1		у	60	0	1	У	60	0	1
	Z	5	1	0		Z	5	1	0	Z	5	1	0		Z	7	1	0		Z	7	1	0	z	50	1	0
Υ		X	у	Z			X	у	Z		Х	у	Z			Х	у	Z			Х	у	Z		X	у	Z
	Х	0	4	5		х	0	4	5	Х	0	51	50		х	0	51	50		х	0	51	50	х	0	51	50
	у	4	0	1		У	6	0	1	у	6	0	1		У	60	0	1		у	60	0	1	у	51	0	1
	Z	5	1	0		Z	5	1	0	Z	5	1	0		Z	∞	1	0		Z	∞	1	0	z	50	1	0
Z		X	у	Z			X	у	Z		Х	у	Z			Х	у	Z			Х	у	Z		Х	У	Z
	Х	0	4	5		х	0	4	5	х	0	51	50		Х	0	51	50		х	0	51	50	х	0	51	50
	у	4	0	1		У	4	0	1	у	6	0	1		у	6	0	1		у	60	0	1	У	60	0	1
	Z	5	1	0		Z	5	1	0	Z	7	1	0		Z	7	1	0		Z	50	1	0	z	50	1	0
						Iter	ation	1		Ite	ration	า 2		I	Itera	tion 3	3			ter	ation	4		Ite	ration	5	

b.

Y propogates its



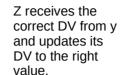
OVA with	poisoned	reverse.
----------	----------	----------

Converged start state	X and Y detect link cost change. Y "thinks" there is a cheaper way to x based on D_z(x)
-----------------------	---

Х	ana y	aetec	t iini
CO	st cha	ange. `	Y
"th	inks"	there	is a
ch	eape	r way t	to x
ba	sed c	on D_z	(x)
	х	V	Z

y propogates its
DV. Z calculates
its D_z(x) based
on the incorrect
one received
from Y.

Because z is
routing through y,
it sets $D_z(x) = \infty$



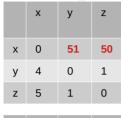
Z, no longer	
routing through y	
propagates DV,	
and Y updates to	
the correct value	

 $D_x(y) = \min_{v} \{ c(x, v) + D_v(y) \}$ for each node y in N

Z

Y propagetes, and no more DVs are being updated. The algorithm has converged after only 6 iterations.

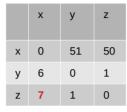
Χ		X	У	Z
	х	0	4	5
	у	4	0	1
	z	5	1	0



ш	11 0111 1.						
	Х	У	Z				
K	0	51	50				
y	6	0	1				
Z	5	1	0				

51

Z

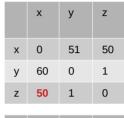


Z

	х	у	Z
Х	0	51	50
у	60	0	1
Z	7	1	0

У

Х



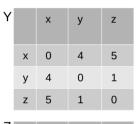
51

	X	У	Z
х	0	51	50
У	51	0	1
Z	50	1	0

51

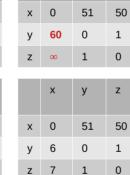
Z

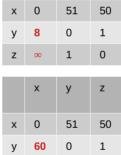
50



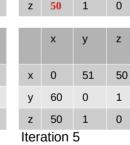








0



51

- \		50	_	Ü			
7							
		Х	У	Z			
	х	0	51	50			
	У	51	0	1			
	Z	50	1	0			
	Iteration 6						

50 1

51



0

5

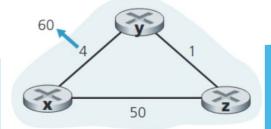
Iteration 2

b.

Iteration 3

z **50** Iteration 4

Iteration 6



z 5