Network layer: "control plane" roadmap

- introduction
- **routing protocols**
  - link state
  - **distance vector**
- intra-ISP routing: OSPF
- routing among ISPs: BGP
- Internet Control Message Protocol

Distance vector algorithm

Based on *Bellman-Ford* (BF) equation (dynamic programming):

Bellman-Ford equation

Let $D_x(y)$: cost of least-cost path from *x* to *y*.
Then:

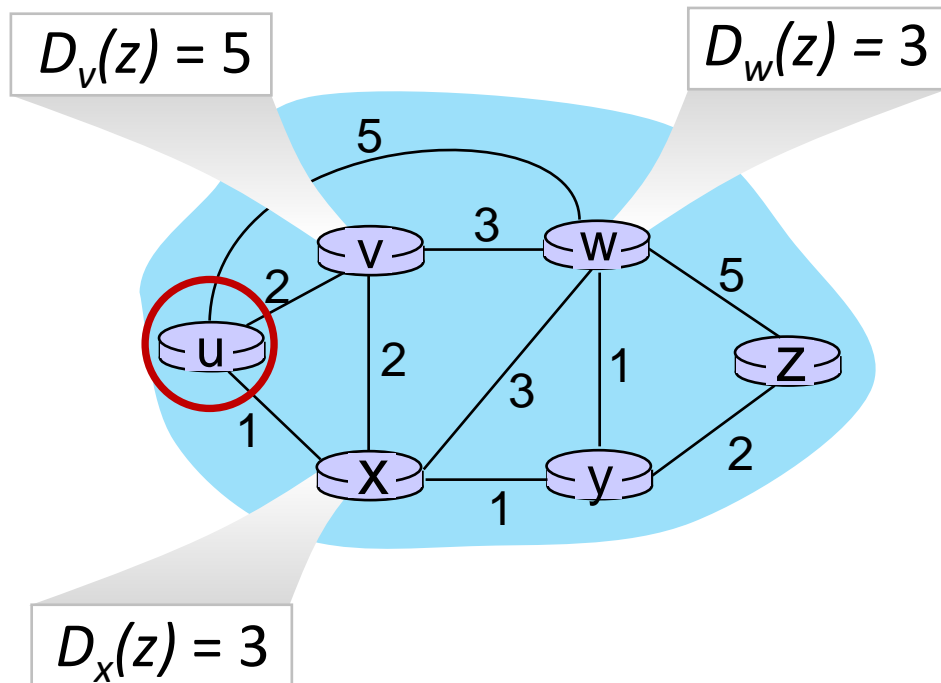$$D_x(y) = \min_v \{ c_{x,v} + D_v(y) \}$$

v's estimated least-cost-path cost to *y*

min taken over all neighbors *v* of *x*

direct cost of link from *x* to *v*

Bellman-Ford Example

Suppose that *u*'s neighboring node*s, x,v,w,* know that for destination *z*:

$D_v(z) = 5$

$D_w(z) = 3$

$D_x(z) = 3$

Bellman-Ford equation says:

$D_u(z)$ = min { $c_{u,v} + D_v(z)$,
$c_{u,x} + D_x(z)$,
$c_{u,w} + D_w(z)$ }

= min {2 + 5,
1 + 3,
5 + 3} = 4

node achieving minimum (x) is next hop on estimated least-cost path to destination (z)

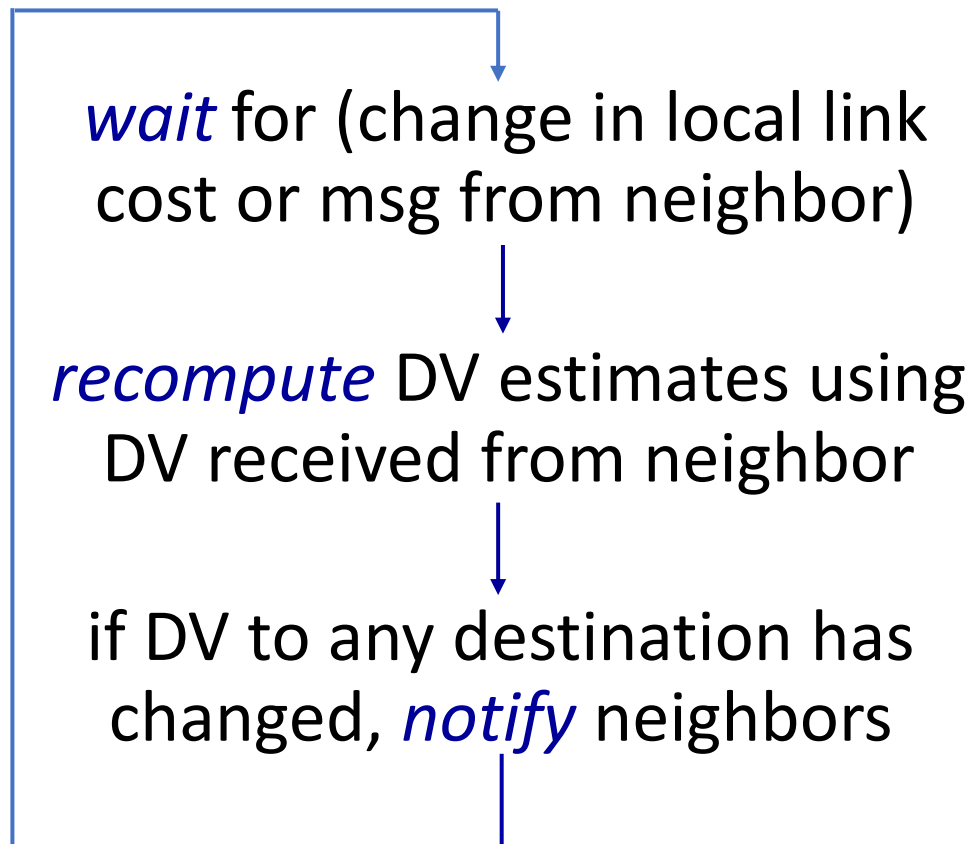# Distance vector algorithm

key idea:

- from time-to-time, each node sends its own distance vector estimate to neighbors

- when *x* receives new DV estimate from any neighbor, it updates its own DV using B-F equation:

$$D_x(y) \leftarrow min_v\{c_{x,v} + D_v(y)\} \text{ for each node } y \in N$$

- under minor, natural conditions, the estimate $D_x(y)$ *converge to the actual least cost* $d_x(y)$

Distance vector algorithm:

each node:

wait for (change in local link cost or msg from neighbor)

recompute DV estimates using DV received from neighbor

if DV to any destination has changed, notify neighbors

iterative, asynchronous: each local iteration caused by:

- local link cost change
- DV update message from neighbor

distributed, self-stopping: each node notifies neighbors only when its DV changes

- neighbors then notify their neighbors – only if necessary
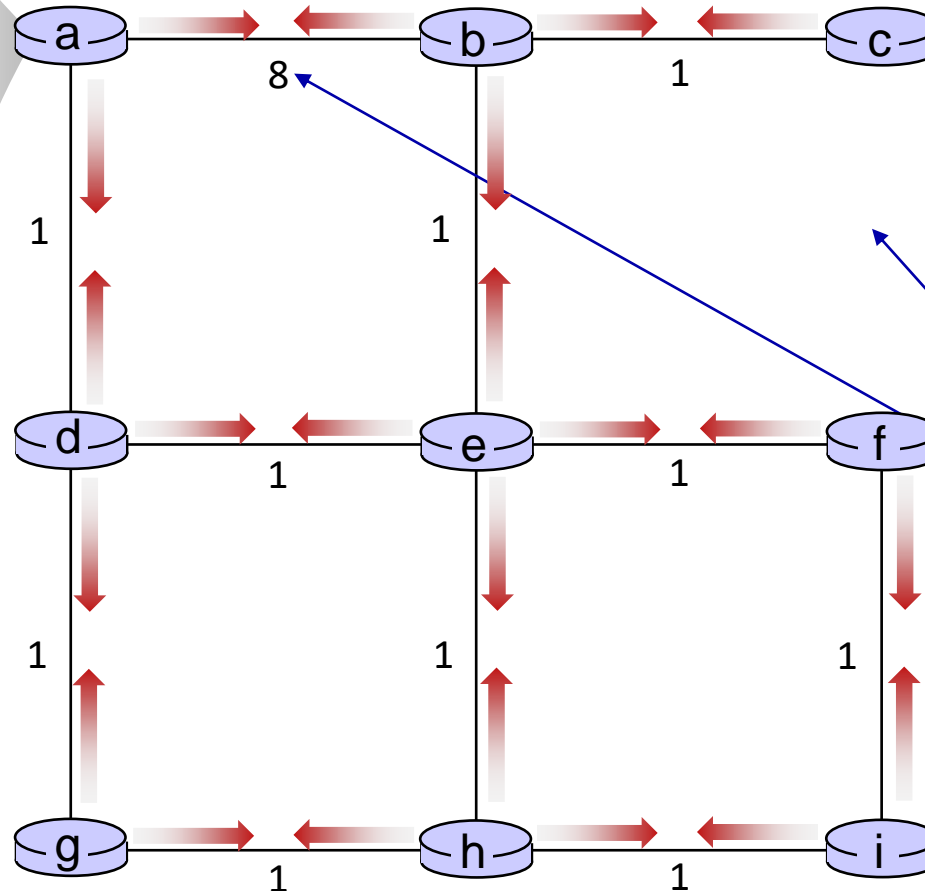- no notification received, no actions taken!

Distance vector: example

DV in a:

$D_a(a)=0$
$D_a(b) = 8$
$D_a(c) = \infty$
$D_a(d) = 1$
$D_a(e) = \infty$
$D_a(f) = \infty$
$D_a(g) = \infty$
$D_a(h) = \infty$
$D_a(i) = \infty$

t=0

- All nodes have distance estimates to nearest neighbors (only)

- All nodes send their local distance vector to their neighbors
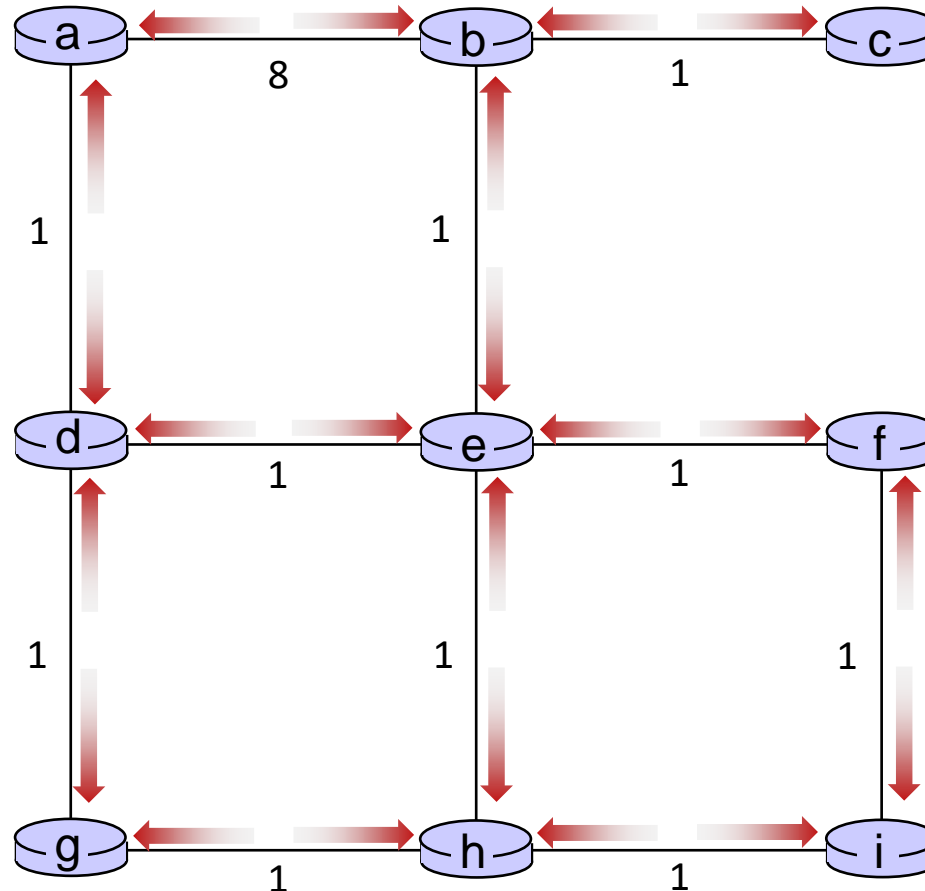
A few asymmetries:
- missing link
- larger cost

Distance vector example: iteration

t=1

All nodes:
- receive distance vectors from neighbors
- compute their new local distance vector
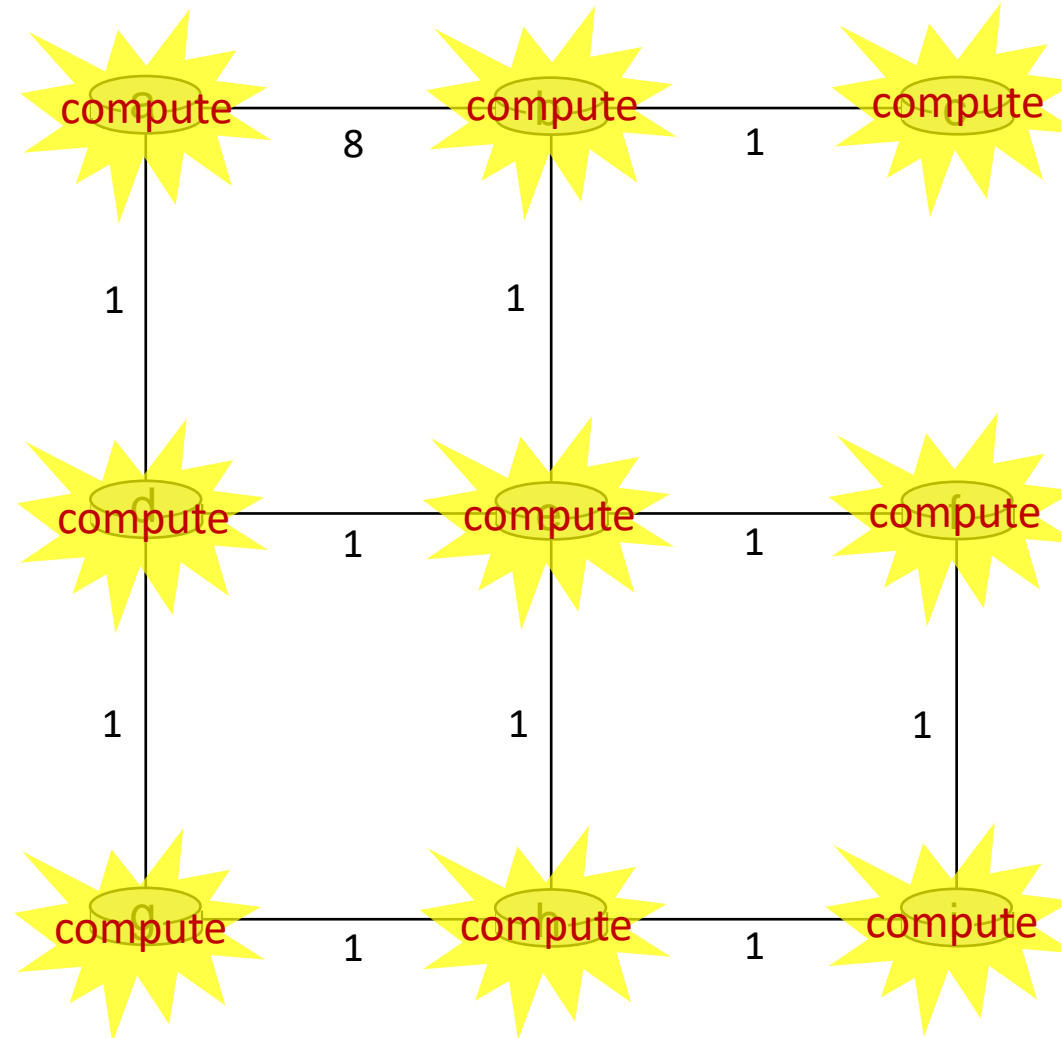- send their new local distance vector to neighbors

Distance vector example: iteration

t=1

All nodes:
- receive distance vectors from neighbors
- **compute their new local distance vector**
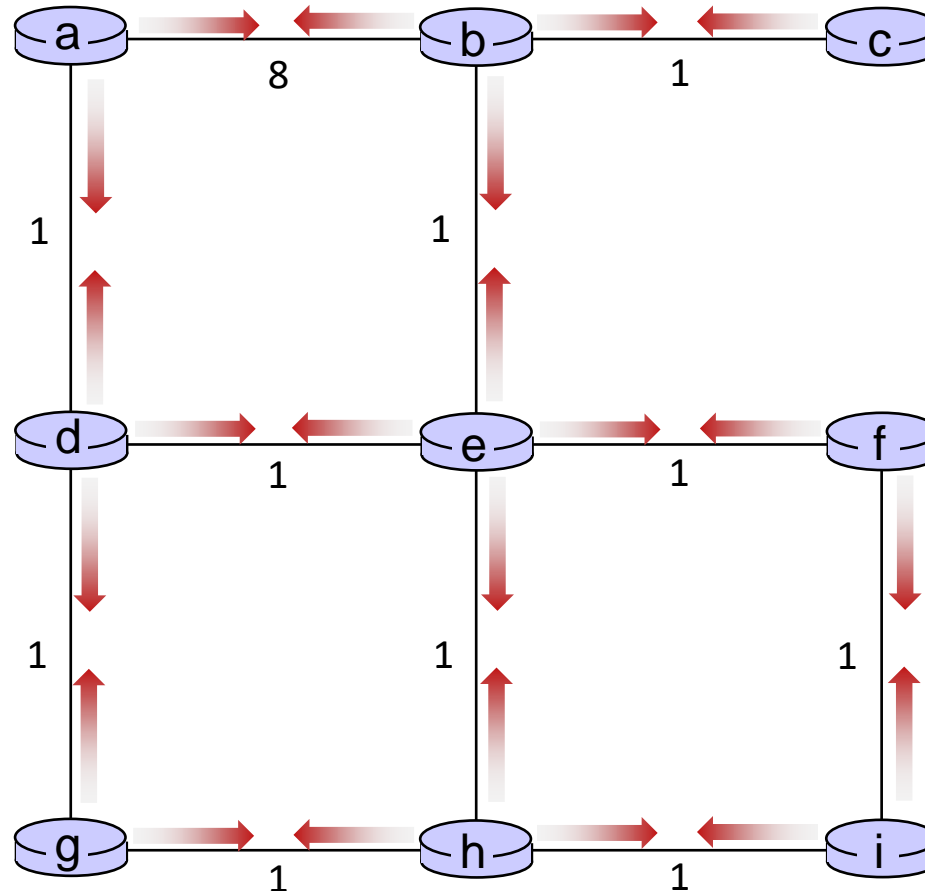- send their new local distance vector to neighbors

Distance vector example: iteration

t=1

All nodes:
- receive distance vectors from neighbors
- compute their new local distance vector
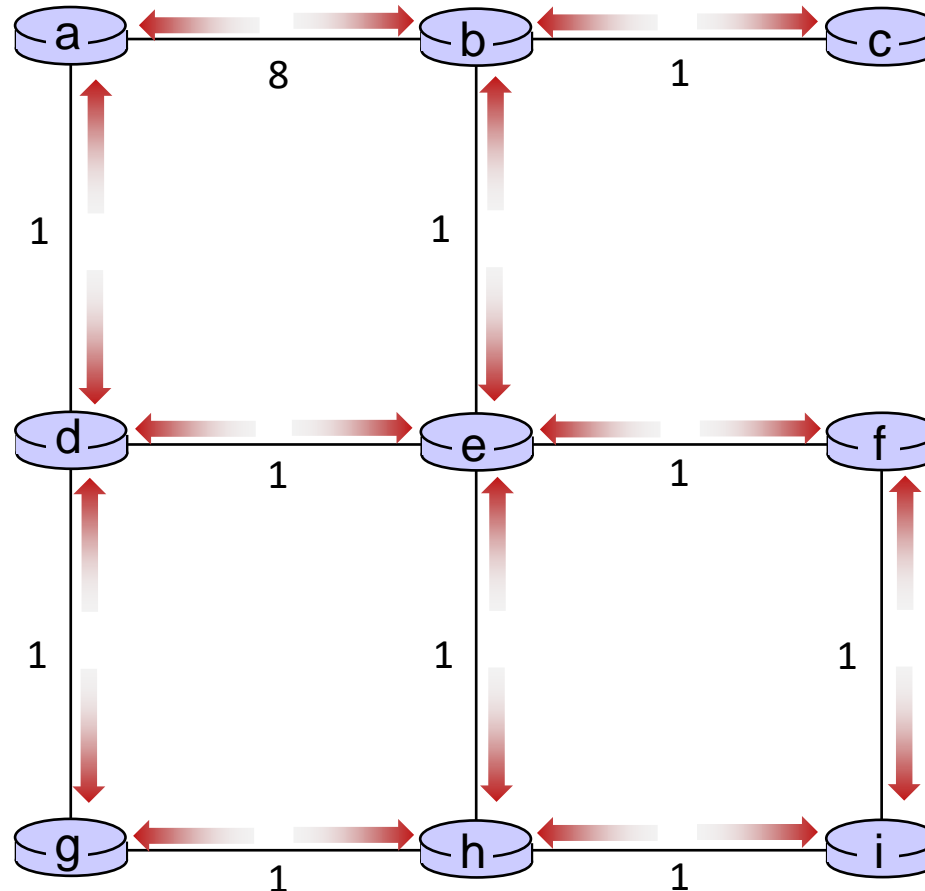- **send their new local distance vector to neighbors**

Distance vector example: iteration

t=2

All nodes:
- **receive distance vectors from neighbors**
- compute their new local distance vector
- send their new local distance vector to neighbors

Distance vector example: iteration

t=2

All nodes:
- receive distance vectors from neighbors
- **compute their new local distance vector**
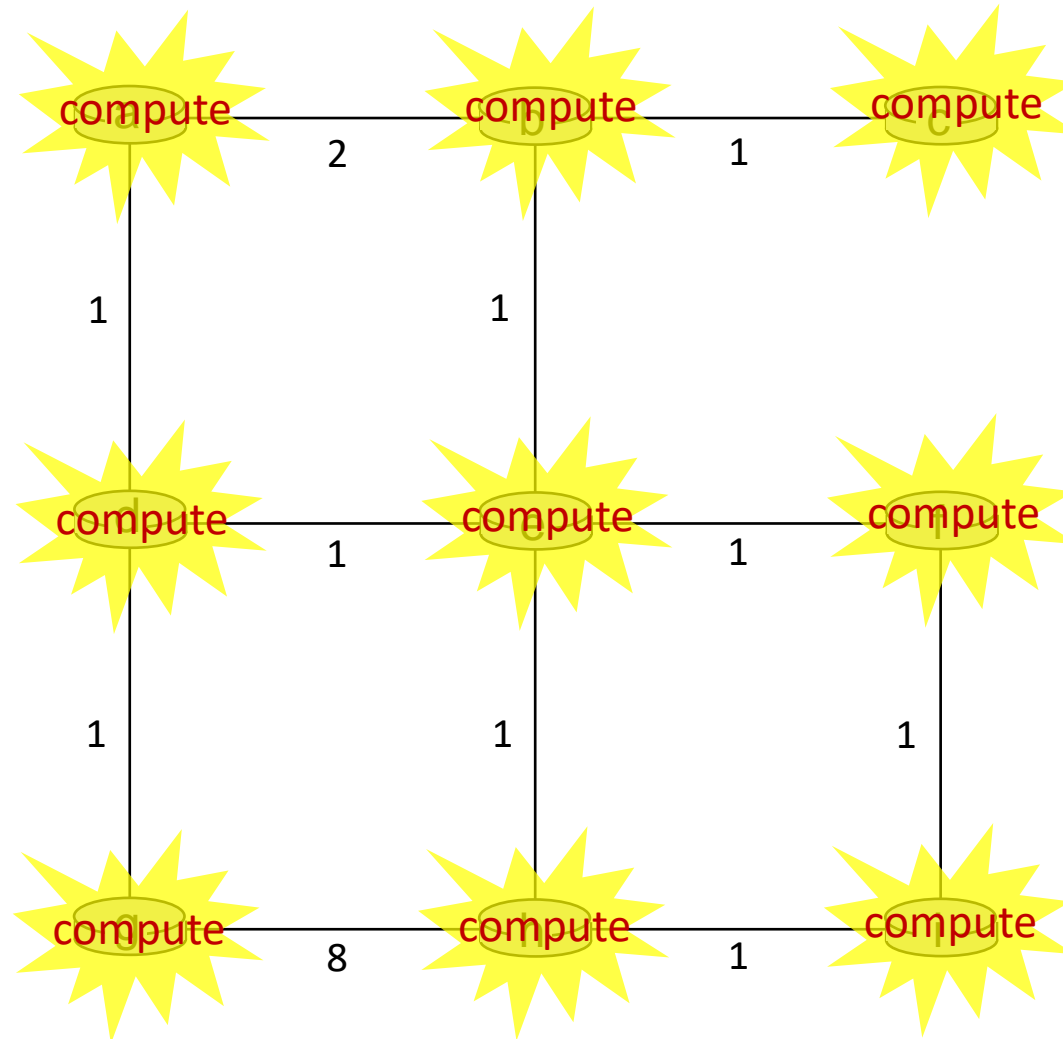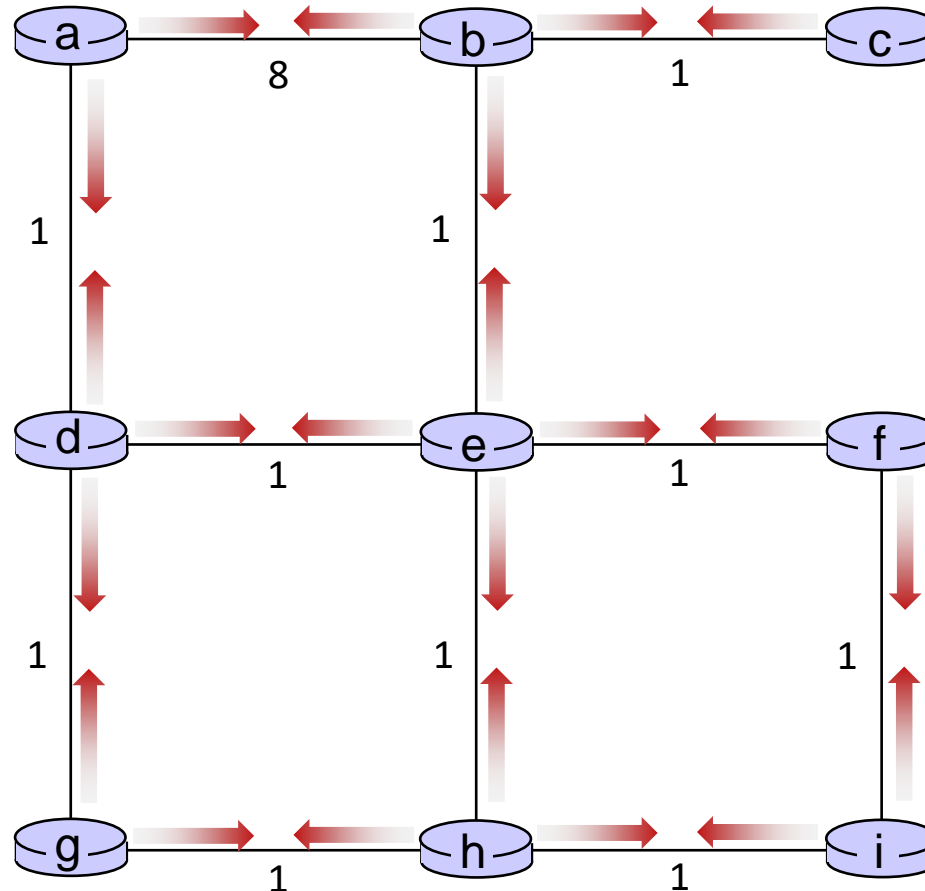- send their new local distance vector to neighbors

Distance vector example: iteration

t=2

All nodes:
- receive distance vectors from neighbors
- compute their new local distance vector
- **send their new local distance vector to neighbors**

Distance vector example: iteration

.... and so on

Let's next take a look at the iterative *computations* at nodes

Distance vector example: c

DV in a:

$D_a(a)=0$
$D_a(b) = 8$
$D_a(c) = \infty$
$D_a(d) = 1$
$D_a(e) = \infty$
$D_a(f) = \infty$
$D_a(g) = \infty$
$D_a(h) = \infty$
$D_a(i) = \infty$

DV in b:

$D_b(a) = 8$ $D_b(f) = \infty$
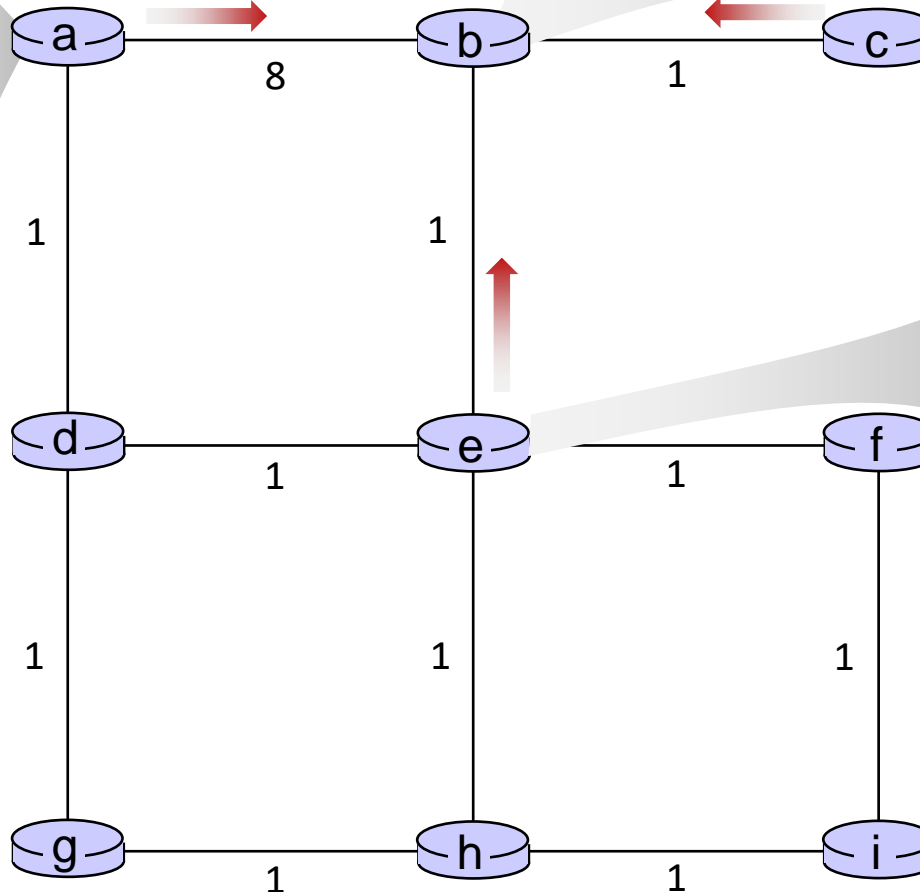$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = \infty$ $D_b(h) = \infty$
$D_b(e) = 1$ $D_b(i) = \infty$

DV in c:

$D_c(a) = \infty$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

DV in e:

$D_e(a) = \infty$
$D_e(b) = 1$
$D_e(c) = \infty$
$D_e(d) = 1$
$D_e(e) = 0$
$D_e(f) = 1$
$D_e(g) = \infty$
$D_e(h) = 1$
$D_e(i) = \infty$

t=1

- b receives DVs from a, c, e

Distance vector example: c

t=1

- b receives DVs from a, c, e, computes:

DV in a:

$D_a(a)=0$
$D_a(b) = 8$
$D_a(c) = \infty$
$D_a(d) = 1$
$D_a(e) = \infty$
$D_a(f) = \infty$
$D_a(g) = \infty$
$D_a(h) = \infty$
$D_a(i) = \infty$

DV in b:

$D_b(a) = 8$ $D_b(f) = \infty$
$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = \infty$ $D_b(h) = \infty$
$D_b(e) = 1$ $D_b(i) = \infty$

DV in c:

$D_c(a) = \infty$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

DV in e:

$D_e(a) = \infty$
$D_e(b) = 1$
$D_e(c) = \infty$
$D_e(d) = 1$
$D_e(e) = 0$
$D_e(f) = 1$
$D_e(g) = \infty$
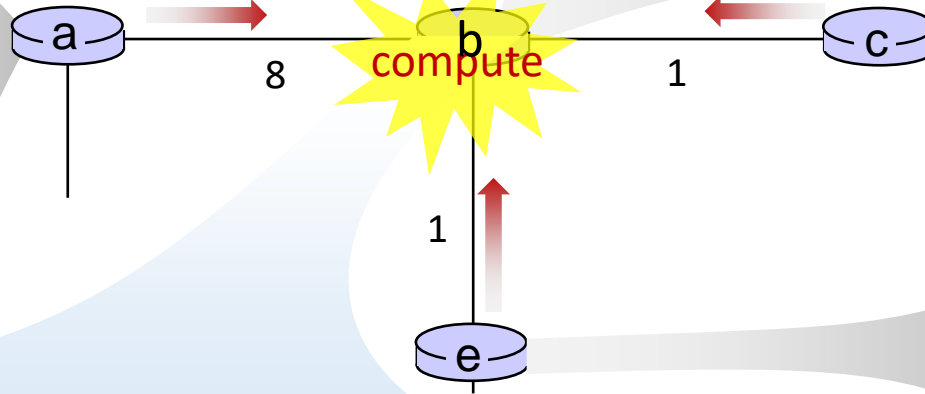$D_e(h) = 1$
$D_e(i) = \infty$

a — 8 — b (compute) — 1 — c

b — 1 — e

$D_b(a) = \min\{c_{b,a}+D_a(a),\ c_{b,c}+D_c(a),\ c_{b,e}+D_e(a)\} = \min\{8,\infty,\infty\} = 8$

$D_b(c) = \min\{c_{b,a}+D_a(c),\ c_{b,c}+D_c(c),\ c_{b,e}+D_e(c)\} = \min\{\infty,1,\infty\} = 1$

$D_b(d) = \min\{c_{b,a}+D_a(d),\ c_{b,c}+D_c(d),\ c_{b,e}+D_e(d)\} = \min\{9,\infty,2\} = 2$

$D_b(e) = \min\{c_{b,a}+D_a(e),\ c_{b,c}+D_c(e),\ c_{b,e}+D_e(e)\} = \min\{\infty,\infty,1\} = 1$

$D_b(f) = \min\{c_{b,a}+D_a(f),\ c_{b,c}+D_c(f),\ c_{b,e}+D_e(f)\} = \min\{\infty,\infty,2\} = 2$

$D_b(g) = \min\{c_{b,a}+D_a(g),\ c_{b,c}+D_c(g),\ c_{b,e}+D_e(g)\} = \min\{\infty,\infty,\infty\} = \infty$

$D_b(h) = \min\{c_{b,a}+D_a(h),\ c_{b,c}+D_c(h),\ c_{b,e}+D_e(h)\} = \min\{\infty,\infty,2\} = 2$

$D_b(i) = \min\{c_{b,a}+D_a(i),\ c_{b,c}+D_c(i),\ c_{b,e}+D_e(i)\} = \min\{\infty,\infty,\infty\} = \infty$

DV in b:

$D_b(a) = 8$ $D_b(f) = 2$
$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = 2$ $D_b(h) = 2$
$D_b(e) = 1$ $D_b(i) = \infty$
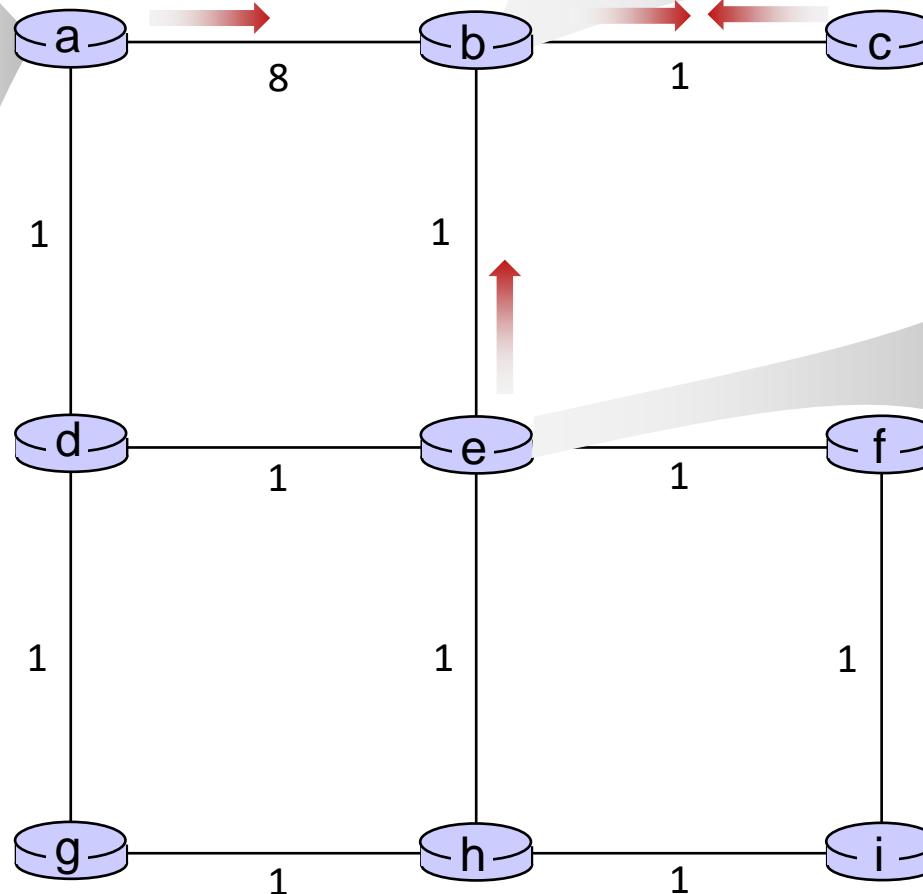
Distance vector example: c

DV in b:

$D_b(a) = 8$ $D_b(f) = \infty$
$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = \infty$ $D_b(h) = \infty$
$D_b(e) = 1$ $D_b(i) = \infty$

DV in c:

$D_c(a) = \infty$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

DV in a:

$D_a(a)=0$
$D_a(b) = 8$
$D_a(c) = \infty$
$D_a(d) = 1$
$D_a(e) = \infty$
$D_a(f) = \infty$
$D_a(g) = \infty$
$D_a(h) = \infty$
$D_a(i) = \infty$

DV in e:

$D_e(a) = \infty$
$D_e(b) = 1$
$D_e(c) = \infty$
$D_e(d) = 1$
$D_e(e) = 0$
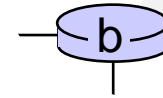$D_e(f) = 1$
$D_e(g) = \infty$
$D_e(h) = 1$
$D_e(i) = \infty$

t=1

- c receives DVs from b

Distance vector example: c

DV in b:

$D_b(a) = 8$ $D_b(f) = \infty$
$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = \infty$ $D_b(h) = \infty$
$D_b(e) = 1$ $D_b(i) = \infty$

DV in c:

$D_c(a) = \infty$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
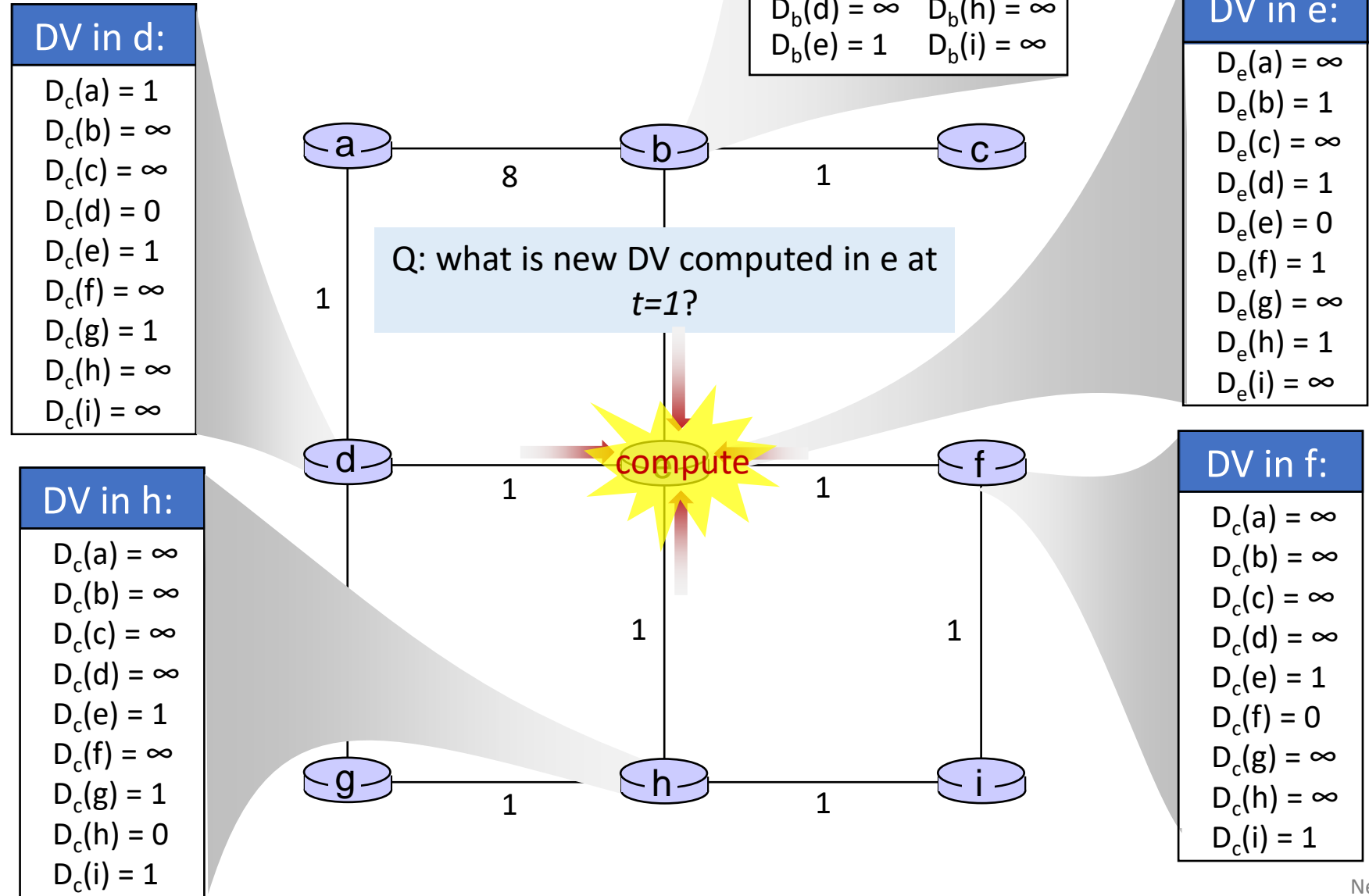$D_c(h) = \infty$
$D_c(i) = \infty$

b —→ 1 compute

t=1

- c receives DVs
  from b computes:

$D_c(a) = \min\{c_{c,b} + D_b(a)\} = 1 + 8 = 9$

$D_c(b) = \min\{c_{c,b} + D_b(b)\} = 1 + 0 = 1$

$D_c(d) = \min\{c_{c,b} + D_b(d)\} = 1 + \infty = \infty$

$D_c(e) = \min\{c_{c,b} + D_b(e)\} = 1 + 1 = 2$

$D_c(f) = \min\{c_{c,b} + D_b(f)\} = 1 + \infty = \infty$

$D_c(g) = \min\{c_{c,b} + D_b(g)\} = 1 + \infty = \infty$

$D_c(h) = \min\{c_{bc,b} + D_b(h)\} = 1 + \infty = \infty$

$D_c(i) = \min\{c_{c,b} + D_b(i)\} = 1 + \infty = \infty$

DV in c:

$D_c(a) = 9$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = 2$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

Distance vector example: c

DV in b:

$D_b(a) = 8$ $D_b(f) = \infty$
$D_b(c) = 1$ $D_b(g) = \infty$
$D_b(d) = \infty$ $D_b(h) = \infty$
$D_b(e) = 1$ $D_b(i) = \infty$

DV in d:

$D_c(a) = 1$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = 0$
$D_c(e) = 1$
$D_c(f) = \infty$
$D_c(g) = 1$
$D_c(h) = \infty$
$D_c(i) = \infty$

DV in e:

$D_e(a) = \infty$
$D_e(b) = 1$
$D_e(c) = \infty$
$D_e(d) = 1$
$D_e(e) = 0$
$D_e(f) = 1$
$D_e(g) = \infty$
$D_e(h) = 1$
$D_e(i) = \infty$

DV in h:

$D_c(a) = \infty$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = \infty$
$D_c(e) = 1$
$D_c(f) = \infty$
$D_c(g) = 1$
$D_c(h) = 0$
$D_c(i) = 1$

DV in f:

$D_c(a) = \infty$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = \infty$
$D_c(e) = 1$
$D_c(f) = 0$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = 1$

t=1

- e receives DVs from b, d, f, h

Q: what is new DV computed in e at *t=1*?

compute

a — 8 — b — 1 — c

1

d — 1 — compute — 1 — f

1

1

g — 1 — h — 1 — i

Distance vector: state information diffusion

Iterative communication, computation steps diffuses information through network:

t=0   c's state at t=0 is at c only

t=1   c's state at t=0 has propagated to b, and may influence distance vector computations up to **1** hop away, i.e., at b

t=2   c's state at t=0 may now influence distance vector computations up to **2** hops away, i.e., at b and now at a, e as well

t=3   c's state at t=0 may influence distance vector computations up to **3** hops away, i.e., at d, f, h

t=4   c's state at t=0 may influence distance vector computations up to **4** hops away, i.e., at g, i
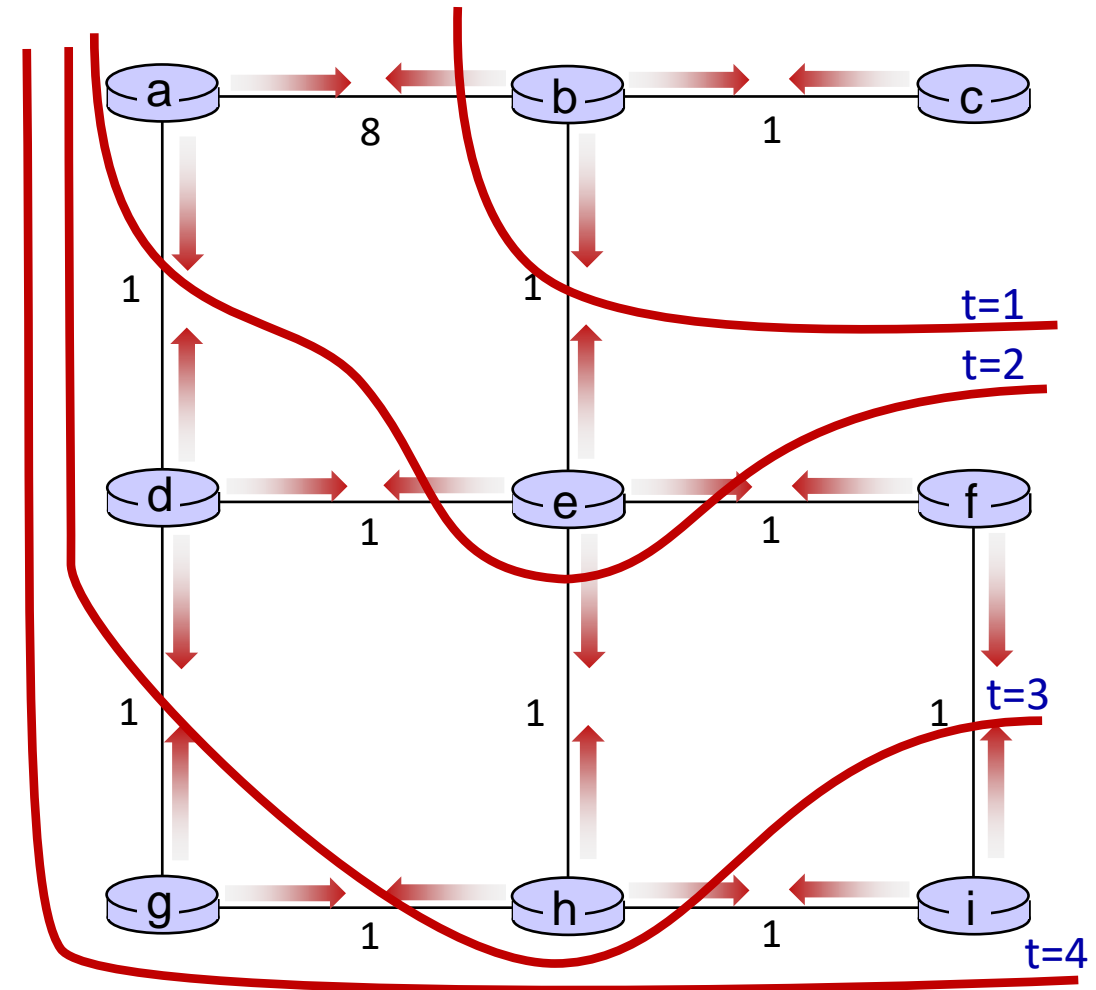
Distance vector: link cost changes

link cost changes:

- node detects local link cost change
- updates routing info, recalculates local DV
- if DV changes, notify neighbors
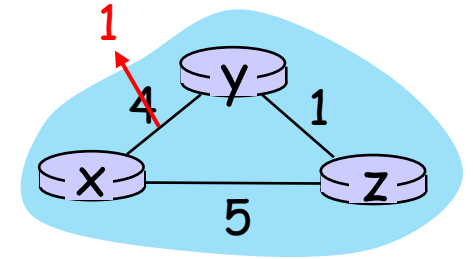
t_0 : y detects link-cost change, updates its DV, informs its neighbors.

"good news travels fast"

t_1 : z receives update from y, updates its DV, computes new least cost to x , sends its neighbors its DV.

t_2 : y receives z's update, updates its DV. y's least costs do *not* change, so y does *not* send a message to z.

Distance vector: link cost changes

link cost changes:

■ node detects local link cost change

■ "bad news travels slow" – count-to-infinity problem:

- *y* sees direct link to *x* has new cost 60, but z has said it has a path at cost of 5. So *y* computes "my new cost to x will be 6, via z); notifies z of new cost of 6 to *x.*

- *z* learns that path to *x* via *y* has new cost 6, so *z* computes "my new cost to x will be 7 via y), notifies *y* of new cost of 7 to *x.*

- *y* learns that path to *x* via *z* has new cost 7, so *y* computes "my new cost to x will be 8 via y), notifies *z* of new cost of 8 to *x.*

- *z* learns that path to *x* via *y* has new cost 8, so *z* computes "my new cost to x will be 9 via y), notifies *y* of new cost of 9 to *x.*
  ...

■ see text for solutions. *Distributed algorithms are tricky!*
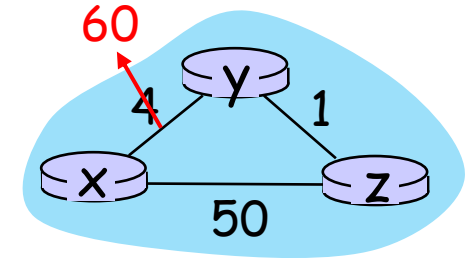
Comparison of LS and DV algorithms

message complexity

LS: n routers, O(n^2) messages sent

DV: exchange between neighbors; convergence time varies

speed of convergence

LS: O(n^2) algorithm, O(n^2) messages
- may have oscillations

DV: convergence time varies
- may have routing loops
- count-to-infinity problem

robustness: what happens if router malfunctions, or is compromised?

LS:
- router can advertise incorrect *link* cost
- each router computes only its *own* table

DV:
- DV router can advertise incorrect *path* cost ("I have a *really* low-cost path to everywhere"): *black-holing*
- each router's DV is used by others: error propagate thru network

Network layer: "control plane" roadmap

- introduction
- routing protocols
- **intra-ISP routing: OSPF**
- routing among ISPs: BGP
- Internet Control Message Protocol

Copyright Information
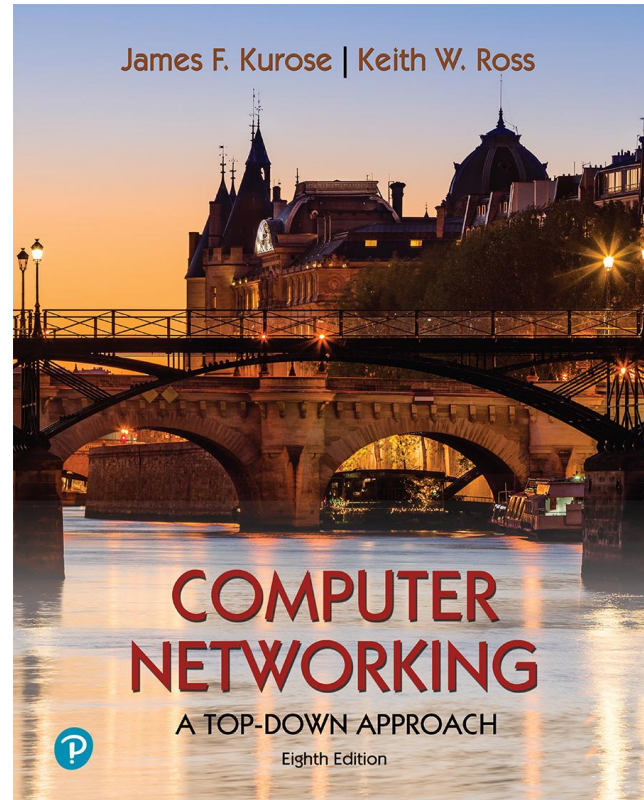
Computer Networking: A Top-Down Approach
8th edition
Jim Kurose, Keith Ross
Pearson, 2020

The Slides are adapted from,