417

Distance Vector Routing

- Distributed Algorithm
- Routing Protocol
- Forwarding

---

what are we trying to compute?

Model

- nodes have unique addresses
- can send on incident links
- can detect neighbors
- links are not reliable

Notation

at node x

$D^x(y, z)$

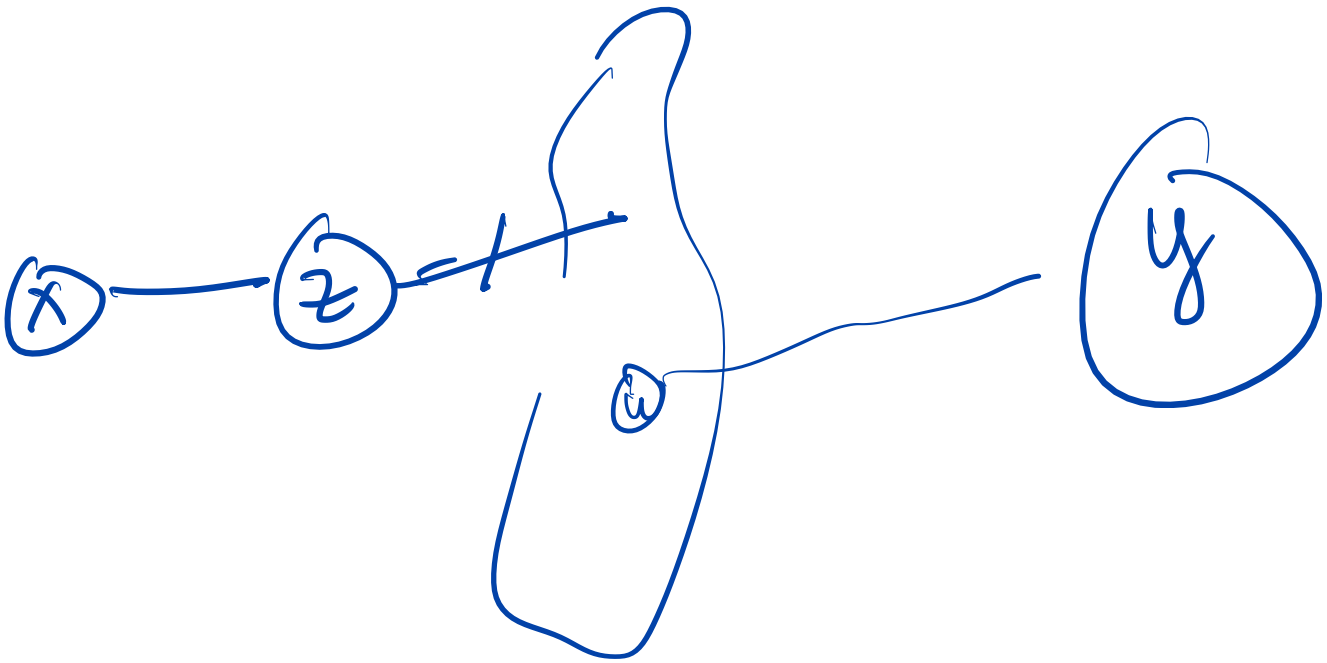
Distance known at x to dest. y via neighbor z.

$cost(x, y)$: link cost between x, y (non-negative)

Convergence

$$D^x(y,z) \leftarrow \text{cost}(x,z) + \min_{w} D^z(y,w)$$

w: neighbors of z



DV

$G = (N, E)$

initially

$D^x(*, v) \leftarrow \infty$

$D^x(v, v) \leftarrow$ cost (x, v)

// direct links

V : n'bor

$\forall$ dest y

send $\min_{w \in N} D^x(y, w)$

to

each neighbor

forever

get rt. update from v
- can be new link cost
or update for some dest

if cost(x, v) changed by δ

∀ dest y

0. $D^x(y, v) \leftarrow D^x(y, v) + \delta$

else
if cost to y via v changed

1. $D^x(y, v) \leftarrow cost(x, v)$

+ new cost

forever

get rt. update from v
- can be new link cost
or update for some dest

if cost(x, v) changed by δ

∀ dest y

update

$D^x(y,v) \leftarrow D^x(y,v) + \delta$

else

if cost to y via v changed

$D^x(y,v) \leftarrow$ cost(x,v)

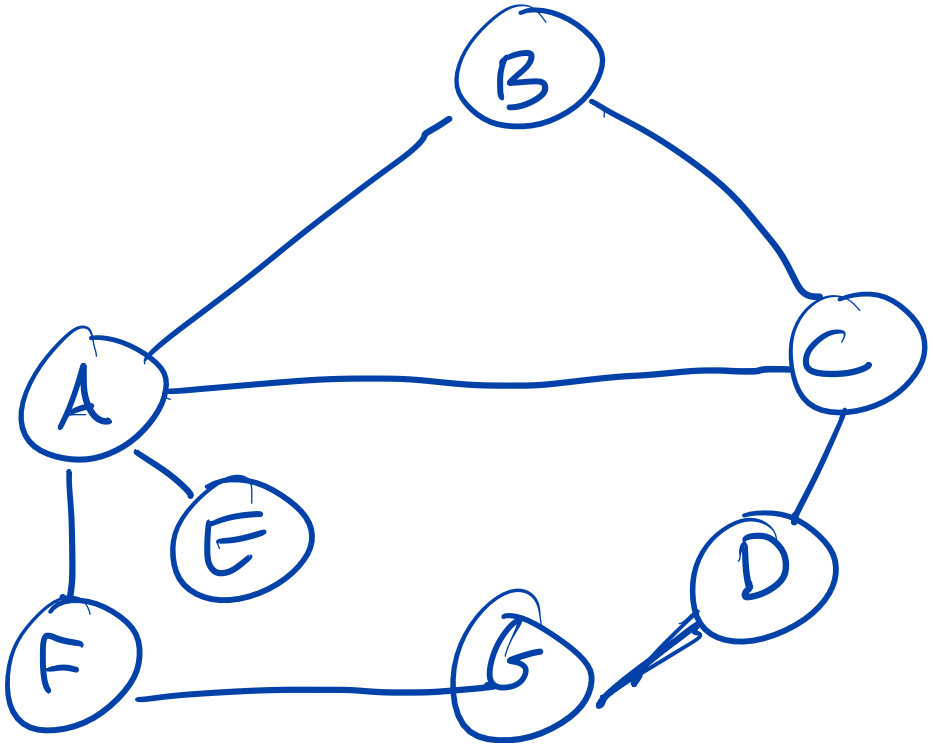
+ new cost

Propagate

if $\min_w D^x(y,w)$ changed

Send $D^x(y,w)$ to

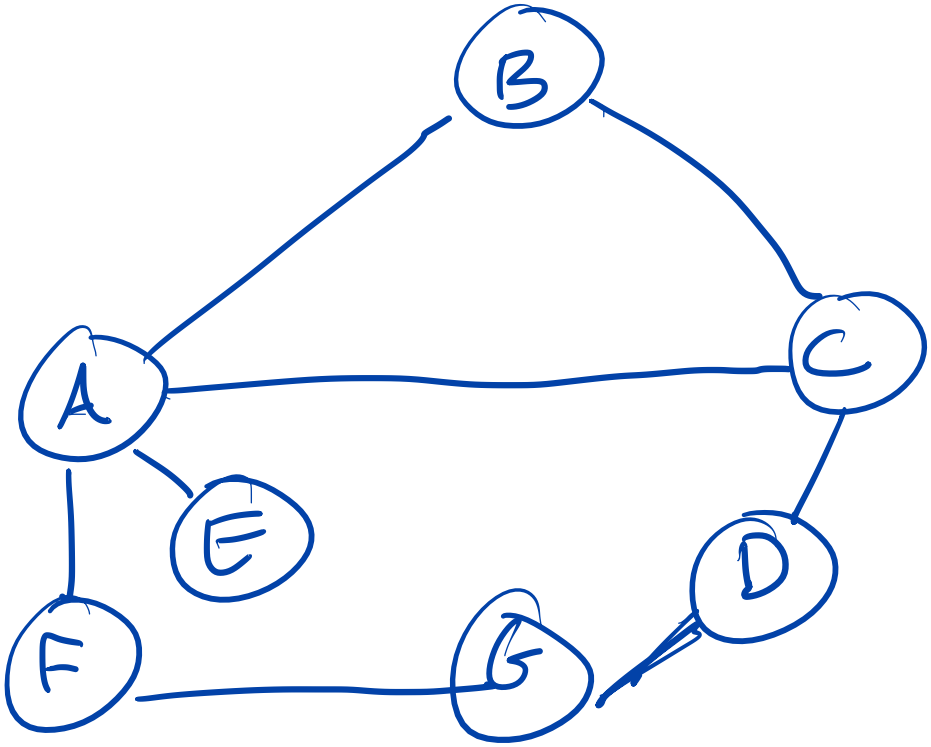
all neighbors.



all links unit cost

at A

Dest	NH	Cost
B	B	1
C	C	1
D	—	∞
E	E	1
F	F	1
G	—	∞



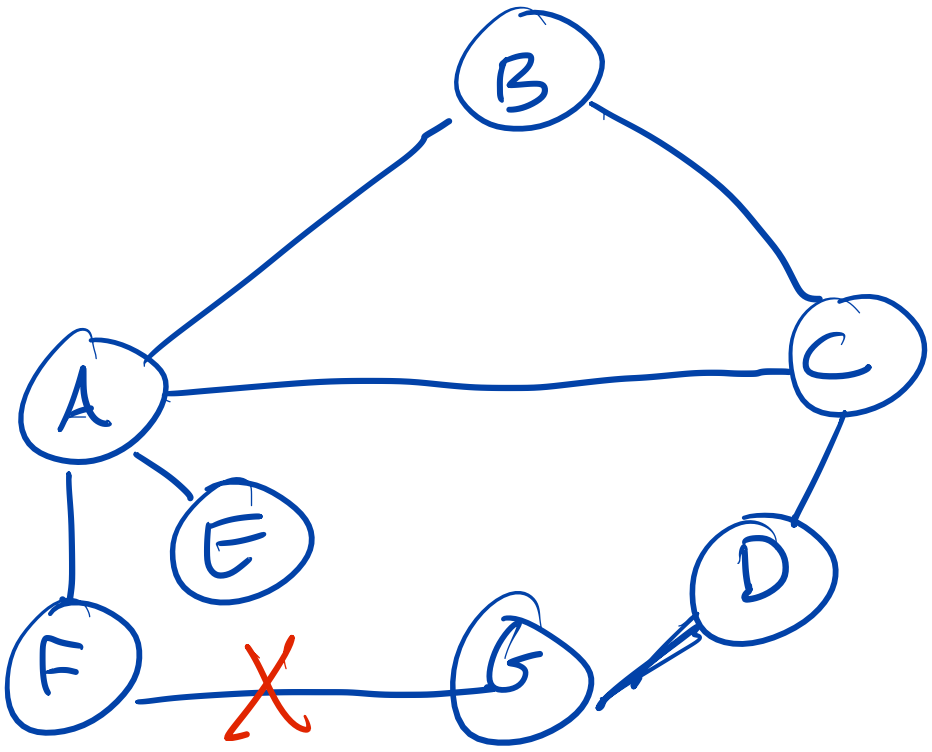
all links unit cost

at A

Dest	NH	Cost
B	B	1
C	C	1
D	~~C~~ C	~~2~~
E	E	1
F	F	1
G	~~E~~ E	~~3~~ 2

From F
(F → G), 1
at A *

From C
C → D, 1



F → G link fails

at F, (F → G) → ∞

update

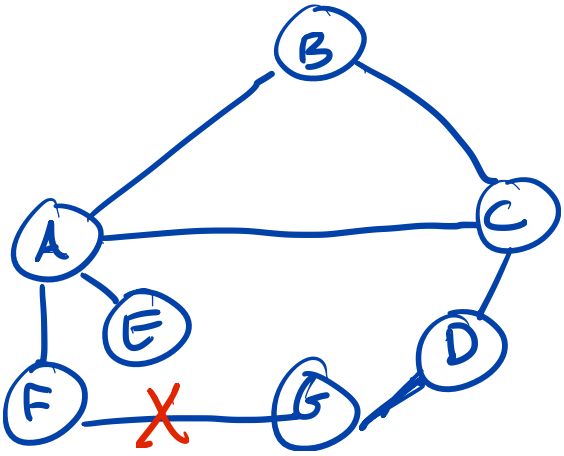
F → A : F → G, ∞

at A : A → G, ~~∞~~ ?

periodic updates

vs. Full table

Periodic update



later

C → A : C → G, 2

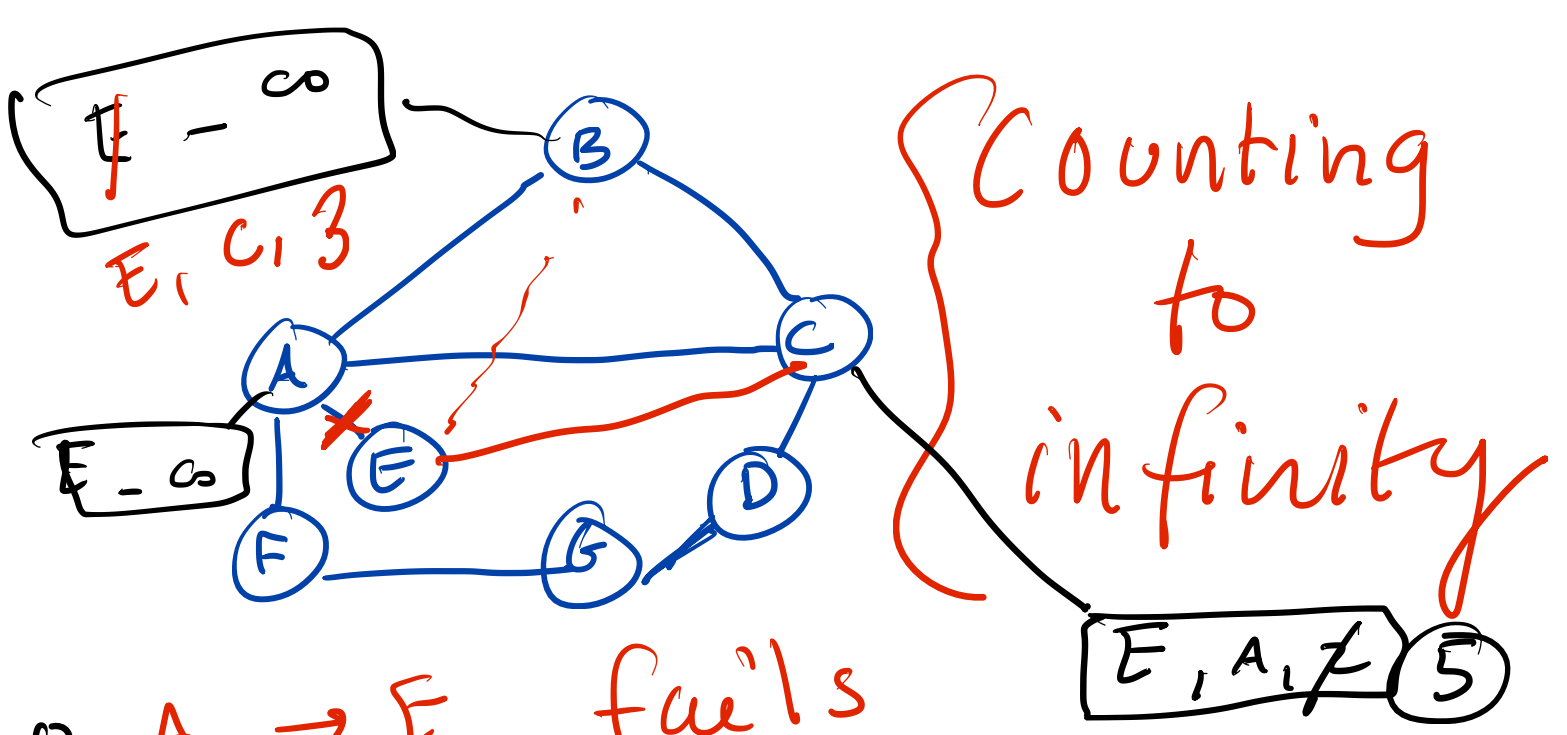
at A : A → G, C, 3

update to F

A → F : A → G, 3

at F

F → G : A, 4



Counting to infinity

0 A → E fails

1 at A : (A → E) → ∞ dropped

2 sends updates to B, ~~C~~ ...

3 B hears for C :

C → E, 2

4 at B, B → E : C, 3 sent to A

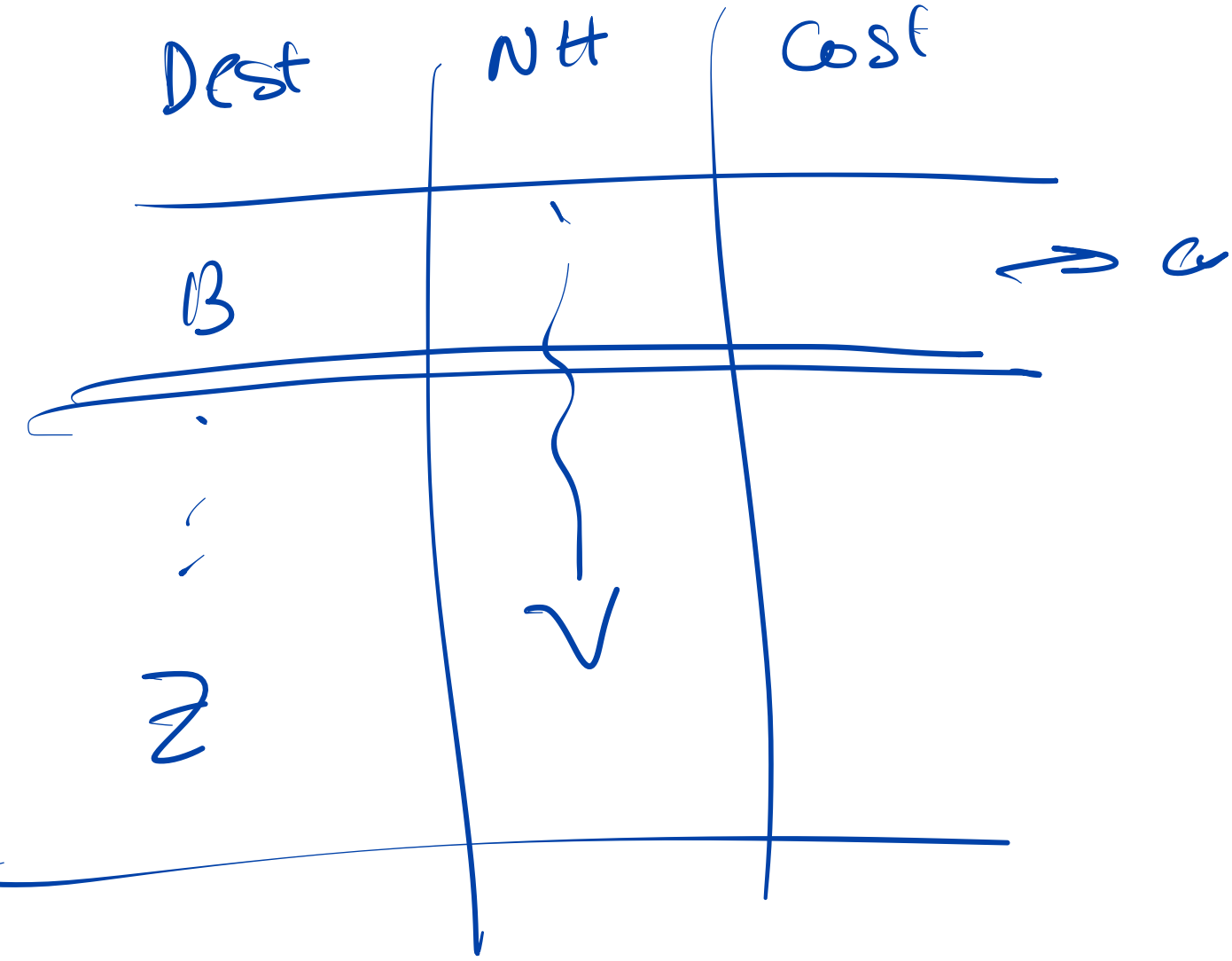
5 at A, A → E : B, 4 sent to C

6 at C, C → E : A, 5, sent to B

7 at B, B → E ; C, 6, sent to A

⋮

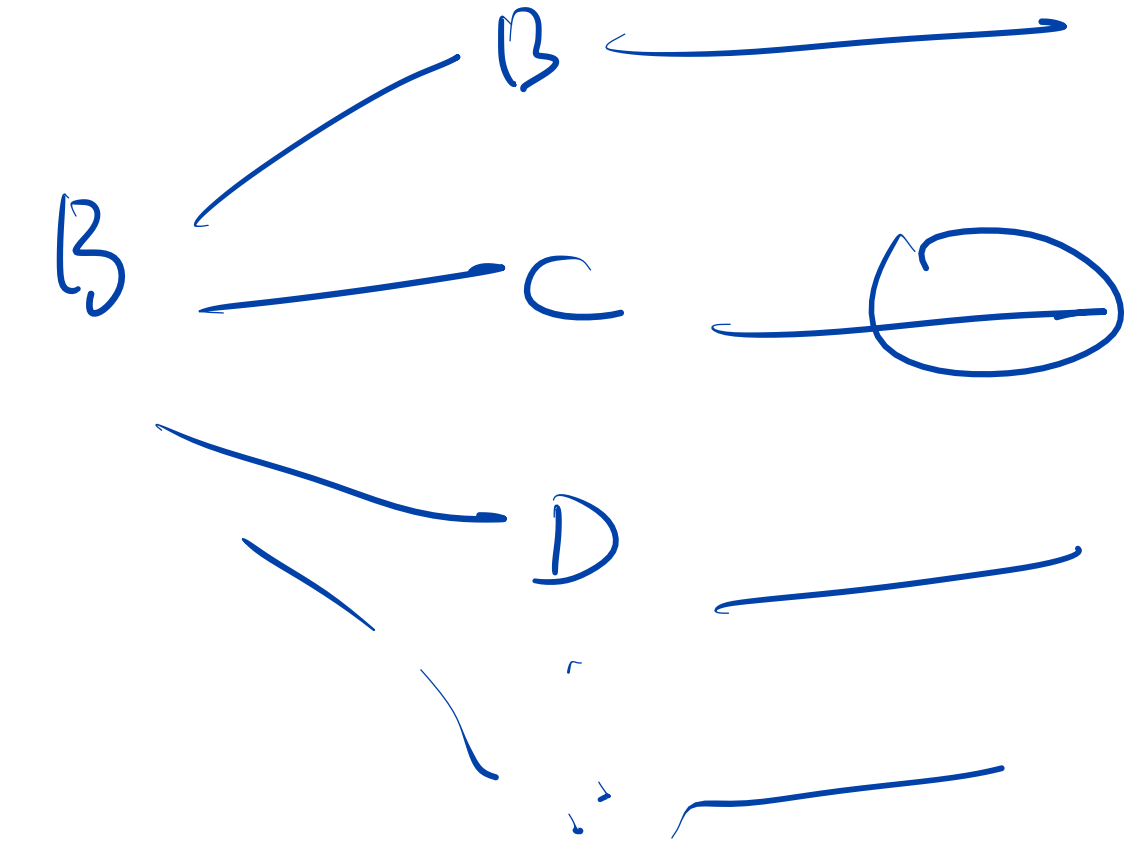
at A



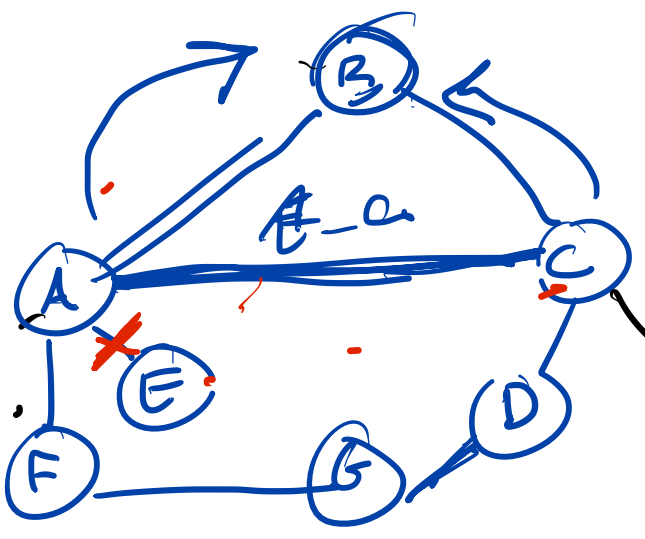
Full Table

Full Table

at A



~~E~~ E, C, 3



Counting to infinity

0. A → E fails
1. at A : (A → E) → ∞
2. sends updates to B, ~~C~~ ...
3. B hears for C :
   C → E, 2
4. at B, B → E : C, 3 sent to
5. at A, A → E : B, 4 sent to
6. at C, C → E : A, 5, sent to
7. at B, B → E ; C, 6, sent to

⋮