Vknals Connect two nodes w. cable

M ① **Network layer**

~~F~~Forwarding — once you have a path.

Routing — finding a path (abstract graphalog)

Scale — hierarchy.

↑ difficult.

Slides: ① USA: 18 sites — can eyeball paths

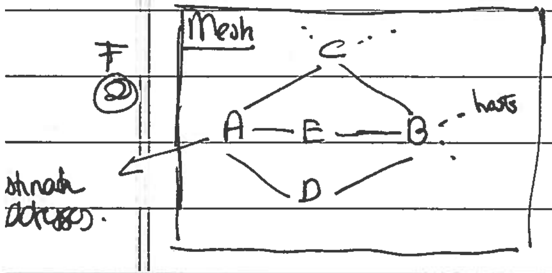
② same: only two paths!

③ 1977: 5x routers, fixed-speed links.

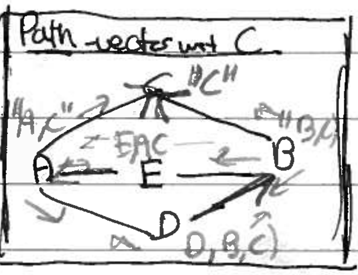
④. 10,000 capacity, spider topology.
city hierarchy
one of dozen big backbones interconnected!

⑤ Still research: ad-hoc + p2p.

F ②



⑤

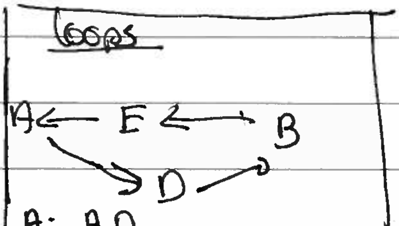


) example

2P

No cycles: tree rooted at C
(also shortest path)

③ ⑥



A: A,D ...

~~E~~: EAD

B: ~~B~~EAD ≪ D

⇒ failure!

③ M **Forwarding table**

dest	B link		C
A	C	A	A
B	-	B	B
C	C	C	-
D	D	D	A
E	E	E	B

(index by dest, fetch link)
fast ⇒ critical path!

Need consistent tables!

What if:

~~E~~	B
	C

loop!!

④ F **Source routing:**

B → A | B,C,A | ....

C → A | C,B,E,A

- No agreement required
- gives end host control
- Rarely done:
  large headers
  ~~routing tables, help from host;~~
  prevents backbones to do opt

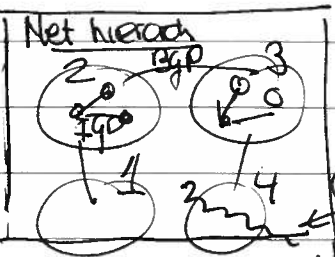
M ⑧ **Path-vector complexity**

N nodes

$N * N * \sqrt{n}$

~~Inet~~: 10^6 routers?

↑ 9



② partitions!

dest: 2.A. (see tomorrow)

⑤ M **Routing table**

① Centralized

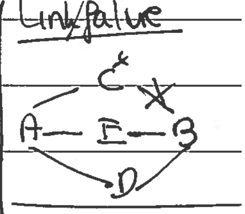
② Distributed
flooding (graph
path-vector.

Slide w code

(explain)

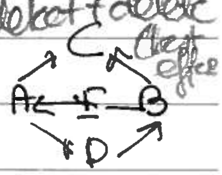
add rule; when
don't hear neighbor
delete routes to
to ~~A~~+B.
node.

F ⑦



but can lead
to temp cycles

⇒ ~~detect~~ delete



A hears: D, B, C

B hears: E, A, C

packets to C
loop
— will be fixed in
next round

⇒ difficult.

⇒ reliable conversation