Panel 1

Last Time:

Network Layer

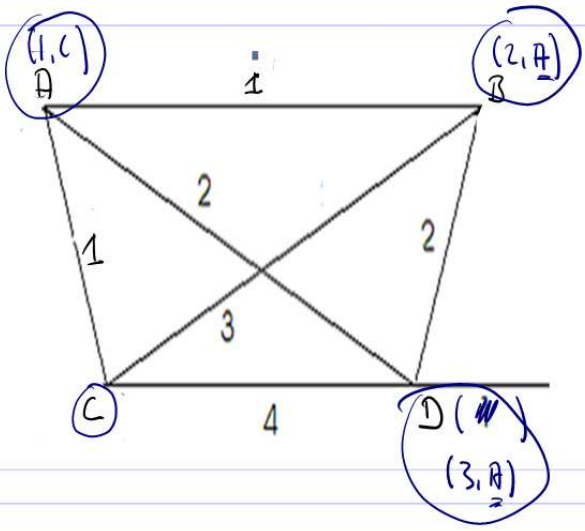
Routing — Shortest Path
→ Flooding
Distance Vector
Link State Rt.

Also: Parity Bit
Stuffing
Framing

1

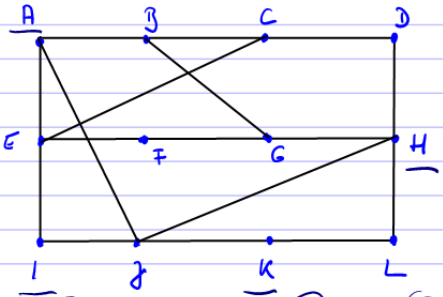
Panel 2

Construct the sink tree for C



2

Panel 3



Each router knows the delay to each neighbor (via ping packet).

	A	B	C	D
A	0	20	24	21
B	12	31	36	28
C	25	19	18	36
D	40	8	27	24
⇒ E	14	30	7	22
F	23	19	20	40
G	18	6	31	31
H	17	0	20	19
I	21	14	0	22
J	9	7	11	10
K	24	22	22	0
L	29	9	33	9

J

8	A	8
20	A	
28	I	
20	H	
17	I	
30	I	
18	H	
12	H	12
10	I	10
0	-	
6	K	6
15	K	

Find the missing entries in the routing table for J

3

Panel 4

Distance vector routing problem:

Count-to-infinity problem - explain

"Bad news travel too slow"

"Alg. converges too slowly if router / connection goes down

4

Panel 5

Link State Routing

Each router must:

1. Discover its neighbors & unique network address
2. Measure delay to each neighbor
3. Construct a packet with this information
4. Distribute this info to all routers
5. Compute shortest path to every router

5

Panel 6

1. Discover neighbors

Protocol | Send "HELLO" packet with your address to your neighbors.

6

Panel 7

2. Measure Delays

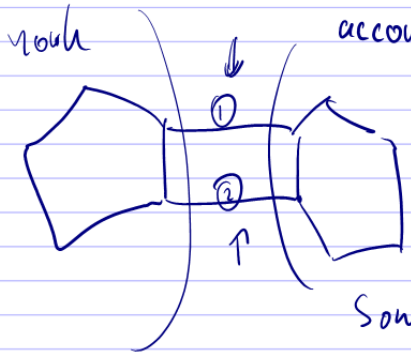
Send ECHO packet, start timer.

Wait for return, divide time in half

≈ delay

Should you take load into account or not

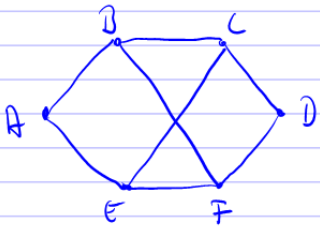
might get oscillation

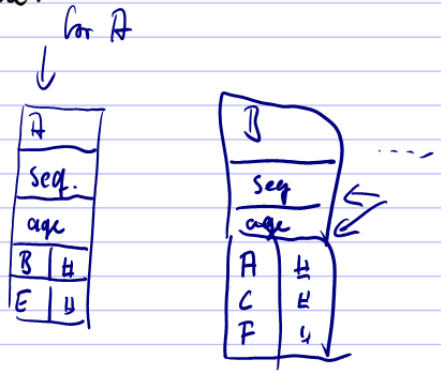


7

Panel 8

3. Building Link State Packet





Distr. these packets

- when state changes
- every 10 minutes

8

Panel 9

4. Distribution of Link State Packets

Today: Routers getting first packets will change their routes ⇒ different routers will have different info of network

Use flooding,
use sequence/age field to control the flood

9

Panel 10

5. Computing the new Routes

Use Dijkstra's Algorithm

Works well, is use today (with modifications)

Trouble: large Routing tables

⇒ Use hierarchy

Thm: Use $\ln(N)$ - levels in hierarchy for N routers!

10

Panel 11