Last class

- Routing: how to get packets to their destination
  - **Forwarding**: local calculation to decide next hop for each packet
  - **Routing**: global calculation to ensure that forwarding decisions ultimately take packets to the right place

- Intra-domain routing protocols
  - Also called Interior Gateway Protocols (IGP)
  - Distance Vector
    - Local exchange of global topology information
    - In steady-state converges to correct solution
    - Problems during failures: count-to-infinity
This class

- Finish Intra-domain routing
  - Link-state protocols
Link State routing

- Same goal as DV, but a different approach
- Two phases
  - Reliable flooding
    - Tell all routers what you know about your local topology
  - Path calculation (Dijkstra’s algorithm)
    - Each router computes best path over complete network

Motivation
- Using DV, routers only have local information, making it difficult to decide what to do when there are changes
- With LS, faster convergence and better stability (hopefully)
- But,… more complex
Reliable flooding

- Goal: tell everyone what you know about local topology

- Periodically send link state packets (LSPs) on all links
  - LSP contains [node, neighbors, costs]

- If node X receives an LSP from node Y over link Q
  - Save it in local link state database
  - Forward LSP on all links except Q

- Use explicit ACKs and retransmits to make flooding reliable

- Each LSP will travel exactly once over each link
Flooding example

- LSP generated by X at T=0
- Nodes become orange as they receive it

\[ T=0 \hspace{1cm} T=1 \hspace{1cm} T=2 \hspace{1cm} T=3 \]
Dijkstra’s Shortest Path Tree (SPT) algorithm

- Graph algorithm for single-source shortest path tree

\[
\begin{align*}
S & \leftarrow \{\} \\
Q & \leftarrow \{\text{all nodes keyed by distance}\} \\
\text{While } Q & \neq \{\} \\
& \quad u \leftarrow \text{extract-min}(Q) \\
& \quad S \leftarrow S \text{ plus } \{u\} \\
& \quad \text{for each node } v \text{ adjacent to } u \\
& \quad \quad \text{“relax” the cost of } v \\
\end{align*}
\]

\[\Rightarrow u \text{ is done}\]
Example – Step 2

\[ Q = \{B, E\} \]
\[ u = E \]

A \rightarrow B: 10
A \rightarrow E: 5
B \rightarrow C: 1
B \rightarrow E: 3
C \rightarrow D: 6
D \rightarrow E: 2
Example – Step 3

Q = \{B, C, D\}

u = D
Example – Step 4

Q={B,C}

u=B

A

B

C

E

D

\[ Q=\{B, C\} \]
\[ u=B \]
Example - Step 5

Q = {C}  

u = C

A

B

C

D

E

0

1

2

3

4

5

6

7

8

9

10
Example - Done

Diagram:
- Nodes: A, B, C, D, E
- Edges:
  - A to B: 10
  - A to D: 2
  - B to C: 1
  - C to D: 6
  - C to A: 4
  - D to E: 7
  - D to C: 4
  - E to 0: 5
  - E to D: 2
  - 0 to A: 3
  - 0 to C: 5
- Q = 0
**Reliable flooding challenges**

- When link/router fails need to remove old data…how?
  - LSPs carry sequence numbers to distinguish new from old
  - Only accept (and forward) the “newest” LSP seen from a node
  - Send a new LSP with cost infinity to signal a link down

- What happens when a router fails and restarts?
  - What sequence # should it use? Don’t want data ignored
  - Aging
    - Put a TTL in the LSP, periodically decremented by each router
    - When TTL = 0, purge the LSP and flood the LSP with TTL 0 to tell everyone else to do the same
    - If router waits for LSP to age out can use any sequence number
  - Alternative: when receiving an “old” LSP from a node, tell the node what the current sequence # is rather than simply dropping the LSP
More challenges

- What happens if the network is partitioned and heals?
  - Different LS databases must be synchronized
  - Use version #s on each LSP (incremented for each update)
  - Compare version #s when a link comes back up and request out of date LSPs
Link State evaluation

- **Strengths**
  - Loop free as long as LSDB’s are consistent
    - Can have transient routing loops
  - Messages are small (esp compared to DV)
  - Converges quickly (esp compared to DV)

- **Weaknesses**
  - Must flood data across entire network (scalability?)
  - Must maintain state for entire topology
Link State in practice

- OSPF (Open Shortest Path First) and IS-IS
  - Most widely used intra-domain routing protocol
  - Run by almost all ISPs and many large organizations

- Basic link state algorithm plus many features:
  - Authentication of routing messages
  - Extra hierarchy: Partition into routing areas
  - Load balancing: Multiple equal cost routes
For next time...

- Inter-domain routing
- Read 4.3-4.3.3
Flooding

- Each router maintains link state database and periodically sends link state packets (LSPs) to neighbor
  - LSPs contain [router, neighbors, costs]
- Each router forwards LSPs not already in its database on all ports except where received
  - Each LSP will travel over the same link at most once in each direction
- Flooding is fast, and can be made reliable with acknowledgments