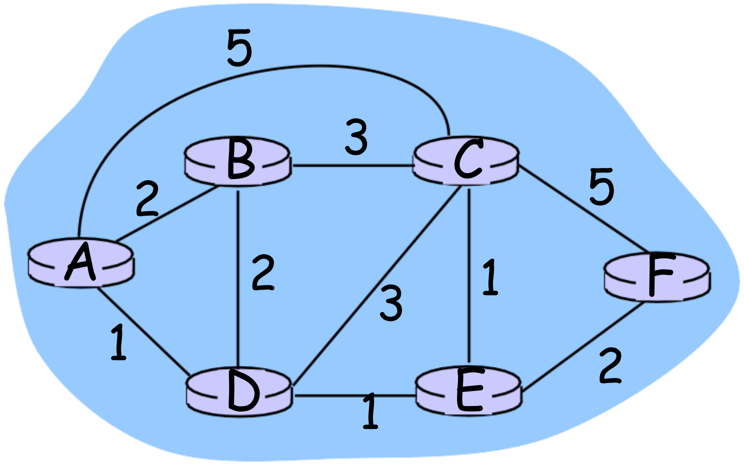
Chapter 4 roadmap

1. Introduction and Network Service Models
2. Routing Principles
3. Hierarchical Routing
4. The Internet (IP) Protocol
5. Routing in the Internet
6. What's Inside a Router
7. IPv6
8. Multicast Routing
9. Mobility

Routing

Routing protocol

Goal: determine "good" path (sequence of routers) thru network from source to dest.



Graph abstraction for routing algorithms:

- graph nodes are routers
- graph edges are physical links
  - link cost: delay, $ cost, or congestion level

"good" path:

- typically means minimum cost path
- other def's possible

Routing Algorithm classification

Global or decentralized information?

Global:

- all routers have complete topology, link cost info
- "link state" algorithms

Decentralized:

- router knows physically-connected neighbors, link costs to neighbors
- iterative process of computation, exchange of info with neighbors
- "distance vector" algorithms

Static or dynamic?

Static:

- routes change slowly over time

Dynamic:

- routes change more quickly
  - periodic update
  - in response to link cost changes

A Link-State Routing Algorithm

Dijkstra’s algorithm

- net topology, link costs known to all nodes
  - accomplished via “link state broadcast”
  - all nodes have same info
- computes least cost paths from one node (‘source”) to all other nodes
  - gives routing table for that node
- iterative: after k iterations, know least cost path to k dest.’s

Notation:

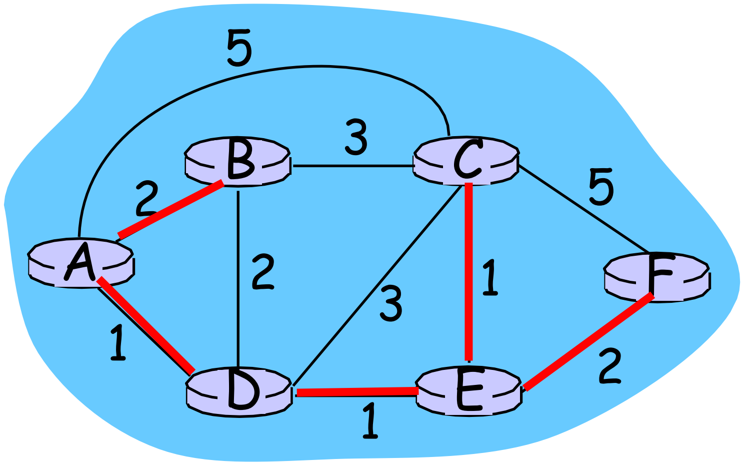
- c(i,j): link cost from node i to j. cost infinite if not direct neighbors
- D(v): current value of cost of path from source to dest. V
- p(v): predecessor node along path from source to v, that is next v
- N: set of nodes whose least cost path definitively known

Dijsktra's Algorithm

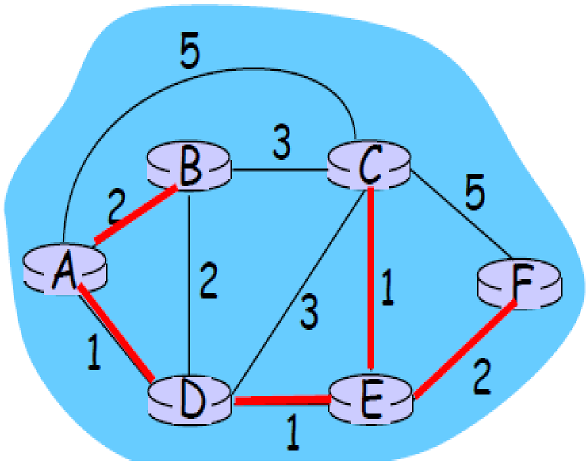
```
Initialization:
  N = {A}
  for all nodes v
    if v adjacent to A
      then $D(v) = c(A,v)$
else $D(v)$ = infinity

Loop
   find w not in N such that $D(w)$ is a minimum
   add w to N
update $D(v)$ for all v adjacent to w and not in N:
   $D(v) = \min( D(v), D(w) + c(w,v) )$
   /* new cost to v is either old cost to v or known
    shortest path cost to w plus cost from w to v */
until all nodes in N
```

Dijkstra's algorithm: example

Step	start N	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	infinity	infinity
1	AD	2,A	4,D		2,D	infinity
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5	ADEBCF					



Least Cost Path and forwarding Table for Node A



Destination	Link
B	(A.B)
C	(A, D)
D	(A, D)
E	(A, D)
F	(A, D)