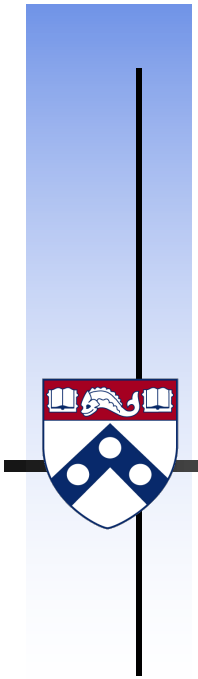


CIS 553: Networked Systems

Interdomain Routing

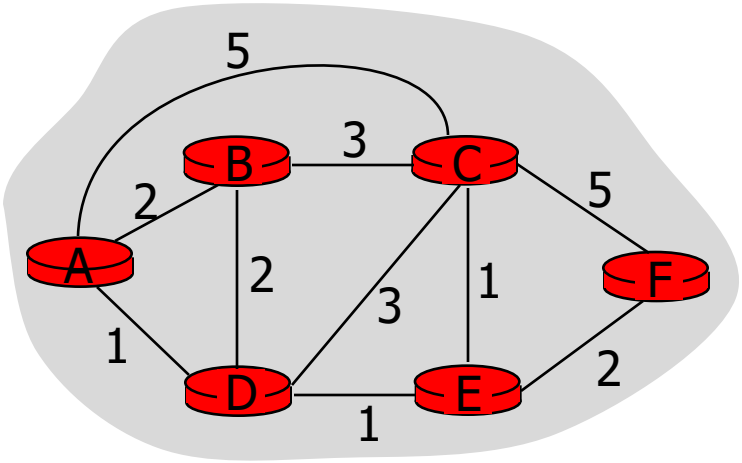
February 19, 2020

Agenda

- Discovery ✓
  - DNS ✓
  - ARP ✓
  - DHCP ✓
- Intradomain Routing ✓
  - Distance Vector ✓
  - Link State NEXT
- Layer-3 Checkpoint
- Interdomain Routing

Example: Dijkstra's Algorithm

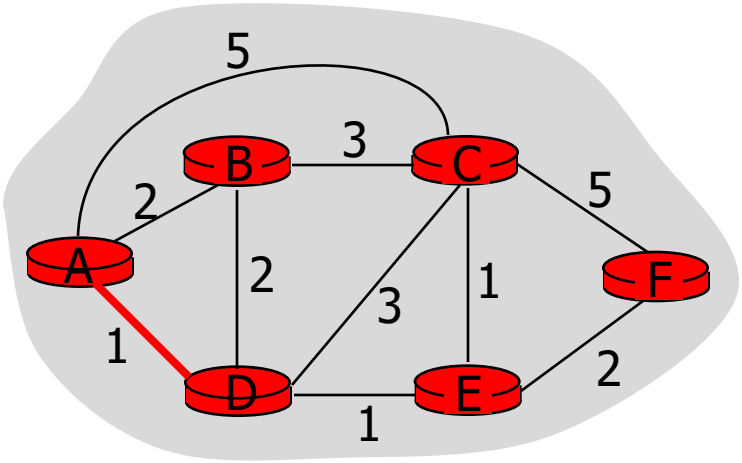
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1						
2						
3						
4						
5						




```
1  Initialization:
2    S = {A};
3    for all nodes v
4      if v adjacent to A
5        then D(v) = c(A,v);
6        else D(v) = ∞ ;
…
```

Example: Dijkstra's Algorithm

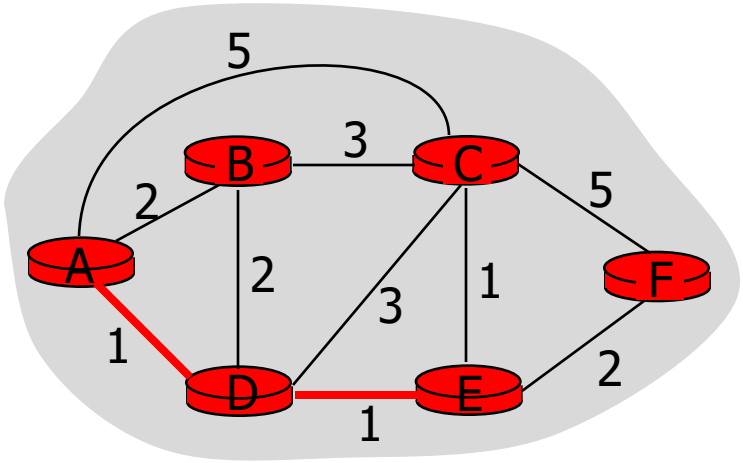
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	∞
2						
3						
4						
5						




```
...
8   Loop
9     find w not in S s.t. D(w) is a
      minimum;
10    add w to S;
11    update D(v) for all v adjacent
        to w and not in S:
12      D(v) = min( D(v), D(w) + c(w,v) );
13    until all nodes in S;
```

Example: Dijkstra's Algorithm

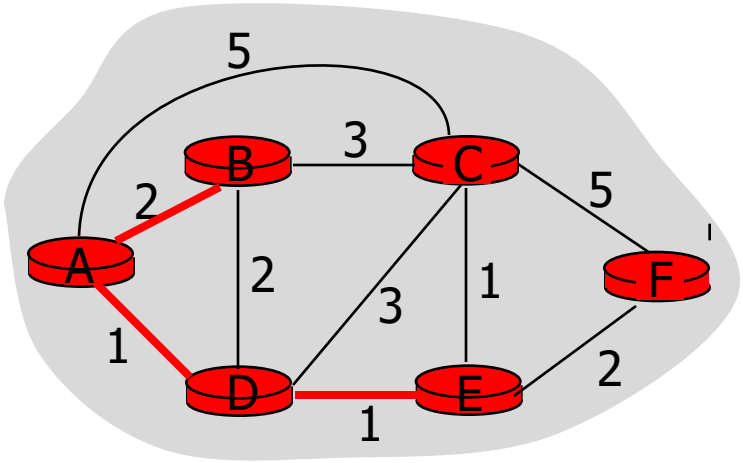
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	∞
2	ADE		3,E			4,E
3						
4						
5						




```
...
8   Loop
9     find w not in S s.t. D(w) is a
      minimum;
10    add w to S;
11    update D(v) for all v adjacent
        to w and not in S:
12      D(v) = min( D(v), D(w) + c(w,v) );
13    until all nodes in S;
```

Example: Dijkstra's Algorithm

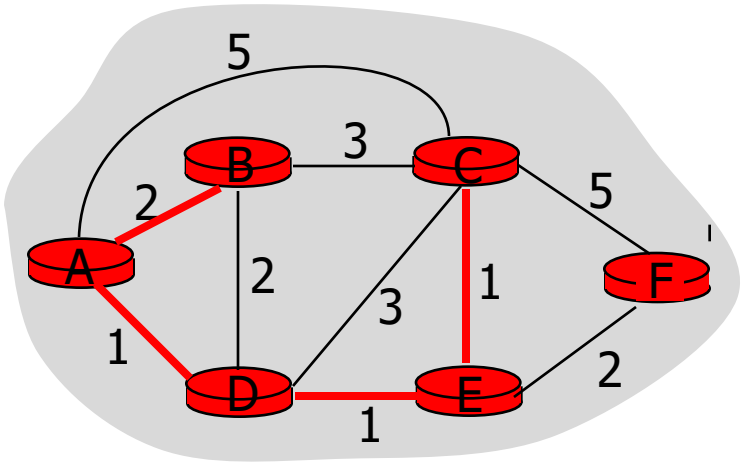
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	∞
2	ADE		3,E			4,E
3	ADEB					
4						
5						




```
...
8   Loop
9     find w not in S s.t. D(w) is a
      minimum;
10    add w to S;
11    update D(v) for all v adjacent
        to w and not in S:
12      D(v) = min( D(v), D(w) + c(w,v) );
13    until all nodes in S;
```

Example: Dijkstra's Algorithm

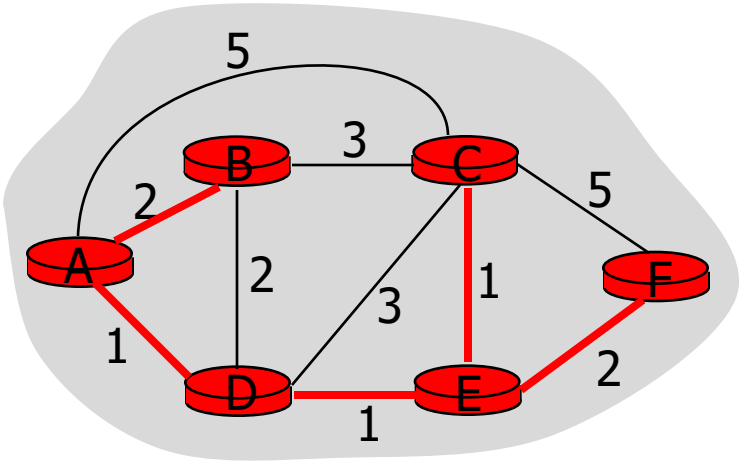
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	∞
2	ADE		3,E			4,E
3	ADEB					
4	ADEBC					
5						




```
...
8   Loop
9     find w not in S s.t. D(w) is a
      minimum;
10    add w to S;
11    update D(v) for all v adjacent
        to w and not in S:
12      D(v) = min( D(v), D(w) + c(w,v) );
13    until all nodes in S;
```

Example: Dijkstra's Algorithm

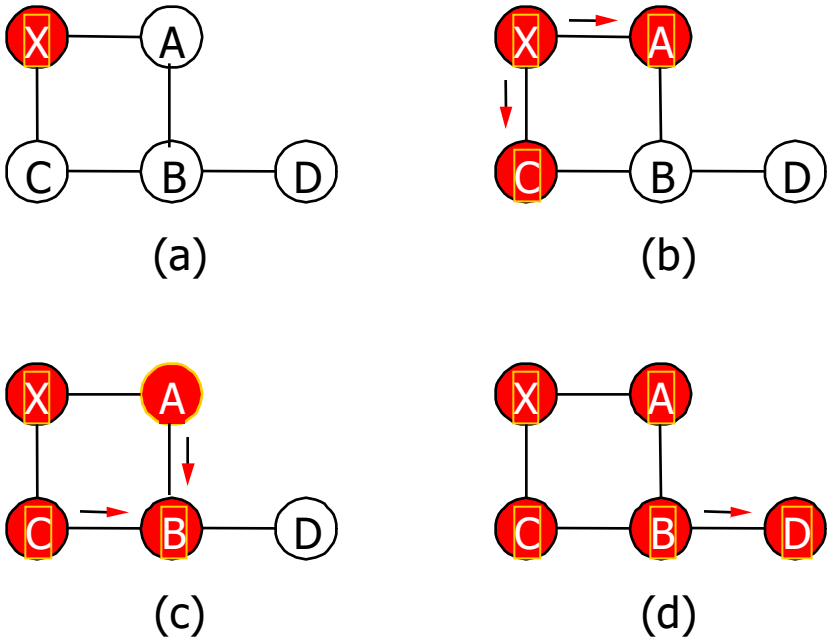
Step	start S	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	∞
2	ADE		3,E			4,E
3	ADEB					
4	ADEBC					
5	ADEBCF					




```
...
8   Loop
9     find w not in S s.t. D(w) is a
      minimum;
10    add w to S;
11    update D(v) for all v adjacent
        to w and not in S:
12      D(v) = min( D(v), D(w) + c(w,v) );
13    until all nodes in S;
```

Flooding the Link State

- Node sends link-state information out its links
- Similar to broadcasting, but not quite! Why?



Flooding the Link State

- Challenges
  - Loops
  - Packet loss
  - Out-of-order arrival
- Solutions
  - Sequence numbers
  - Acknowledgments and retransmissions
  - Time-to-live for each packet

Link State vs. Distance Vector

- Overhead:
  - LS: Store all links (entire network). Also, bandwidth consumption, computation.
  - DV: Entry for each possible destination/next-hop
- Convergence:
  - LS: Reacts more quickly, in bounded time to connectivity changes
  - DV: Count-to-infinity problem. Slower convergence. Bounded path length (16)
- Global policies:
  - LS: Able to impose global policies in a globally consistent way
  - DV: Harder, since do not have complete network topology. Next week: see a variant called Path Vector used in BGP

Agenda

- Intradomain Routing ✓
  - Link State ✓
- Layer-3 Checkpoint NEXT
- Interdomain Routing
  - Structure of the Internet
  - Policy
  - BGP
  - Issues with BGP