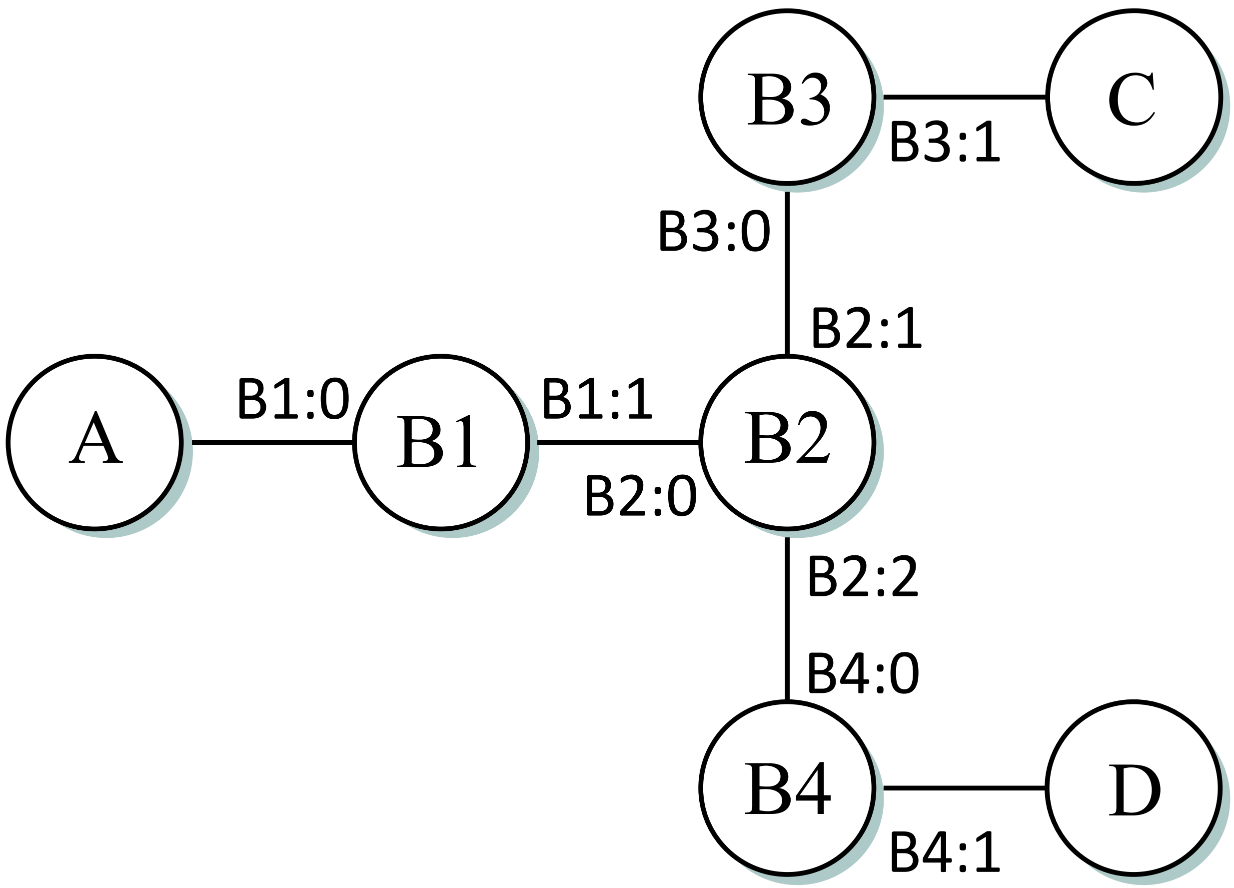
Review notes

March 8, 2018

Learning switches

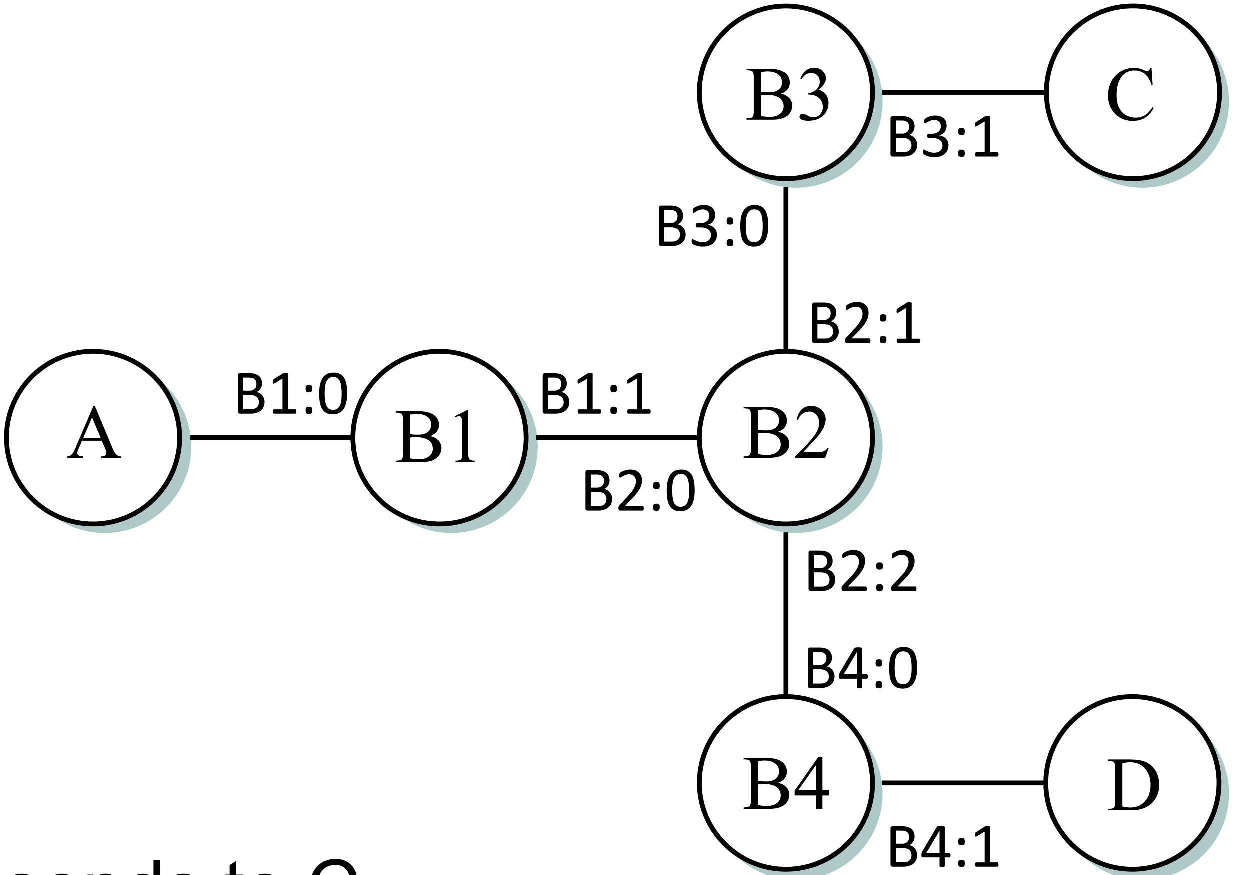


Learning switches

Consider the arrangement of learning switches shown figure above. Assuming all have empty switching tables, initially, indicate what the switching tables of each switch (B1-B4) contain after each of the following transmissions.

- D sends to C.
- C sends to D.
- A sends to C.

Learning switches

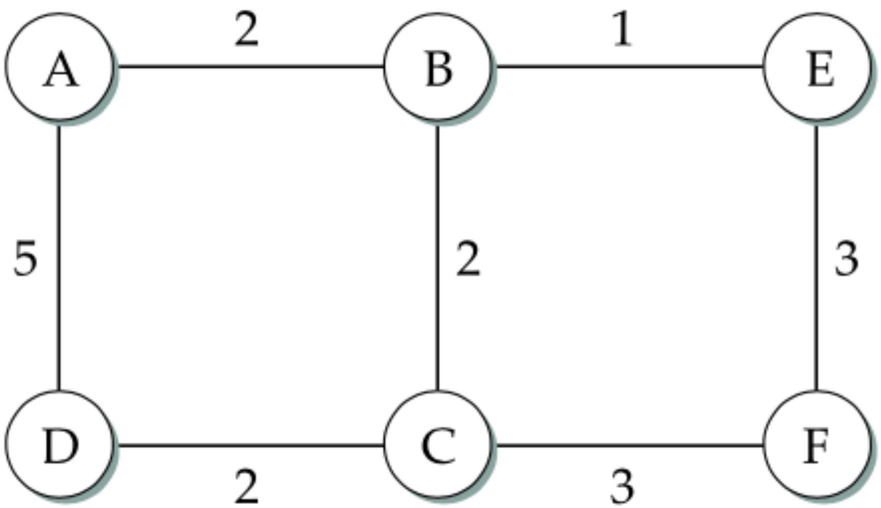


- D sends to C.
- C sends to D.
- A sends to C.

Solution

- All bridges see the packet from D to C. Only B3, B2, and B4 see the packet from C to D. Only B1, B2, and B3 see the packet from A to C.
- B1 B1-0: A B1-1: D
- B2 B2-0: A B2-1: C B2-2: D
- B3 B3-1: C B3-0: A, D
- B4 B4-1: D B4-0: C

Distance vector routing



For the network given in the figure, give global distance-vector when

(a) Each node knows only the distances to its immediate neighbors.

(b) Each node has reported the information it had in the preceding step to its immediate neighbors.

(c) Step (b) happens a second time.

Solution

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	2	∞	5	∞	∞
B	2	0	2	∞	1	∞
C	∞	2	0	2	∞	3
D	5	∞	2	0	∞	∞
E	∞	1	∞	∞	0	3
F	∞	∞	3	∞	3	0

(a)

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	2	4	5	3	∞
B	2	0	2	4	1	4
C	4	2	0	2	3	3
D	5	4	2	0	∞	5
E	3	1	3	∞	0	3
F	∞	4	3	5	3	0

(b)

Solution

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	2	4	5	3	6
B	2	0	2	4	1	4
C	4	2	0	2	3	3
D	5	4	2	0	5	5
E	3	1	3	5	0	3
F	6	4	3	5	3	0

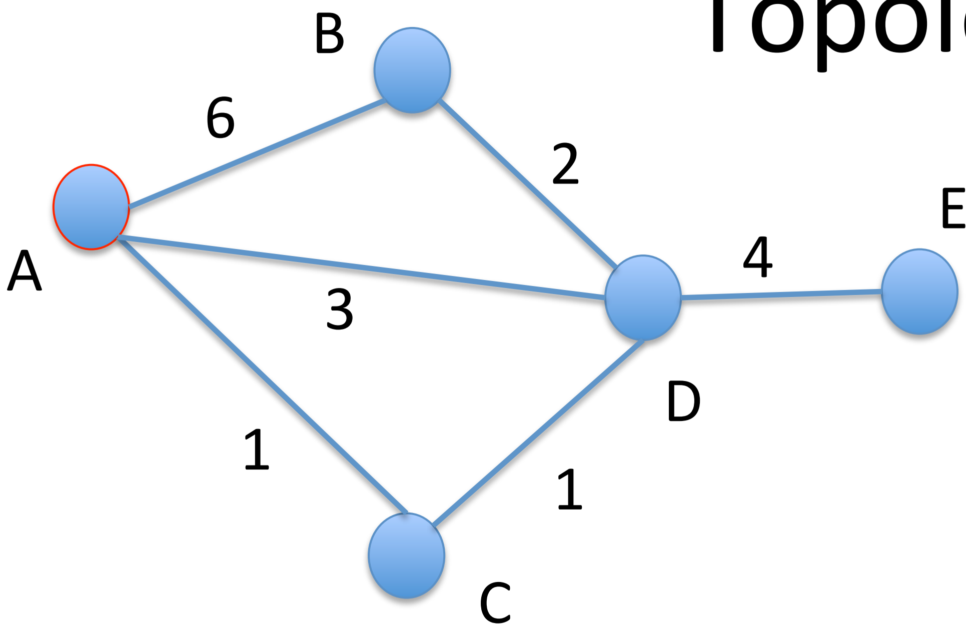
(c)

Execute Dijkstra on a graph

- Dijkstra's Algorithm - Assume non-negative link weights
  - N: set of nodes in the graph
  - l((i, j): the non-negative cost associated with the edge between nodes i, j ∈N and l(i, j) = ∝ if no edge connects i and j
  - Let s ∈N be the starting node which executes the algorithm to find shortest paths to all other nodes in N
  - Two variables used by the algorithm
    - M: set of nodes incorporated so far by the algorithm
    - C(n) : the cost of the path from s to each node n
    - The algorithm

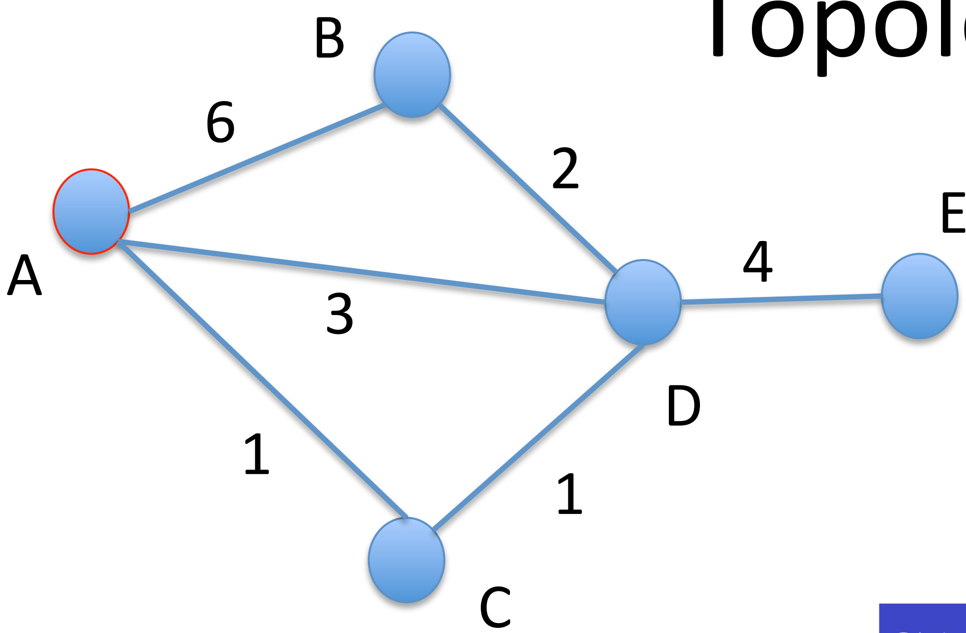
```
M = {s}
For each n in N - {s}
    C(n) = l(s, n)  /* costs of directly connected nodes */
while ( N ≠ M)
  M = M U {w} such that C(w) is the minimum
                for all w in (N-M)  /* add a node */
  For each n in (N-M)         /* recalculate costs */
          C(n) = MIN (C(n), C(w) + l(w, n))
```

Topology



At Node A

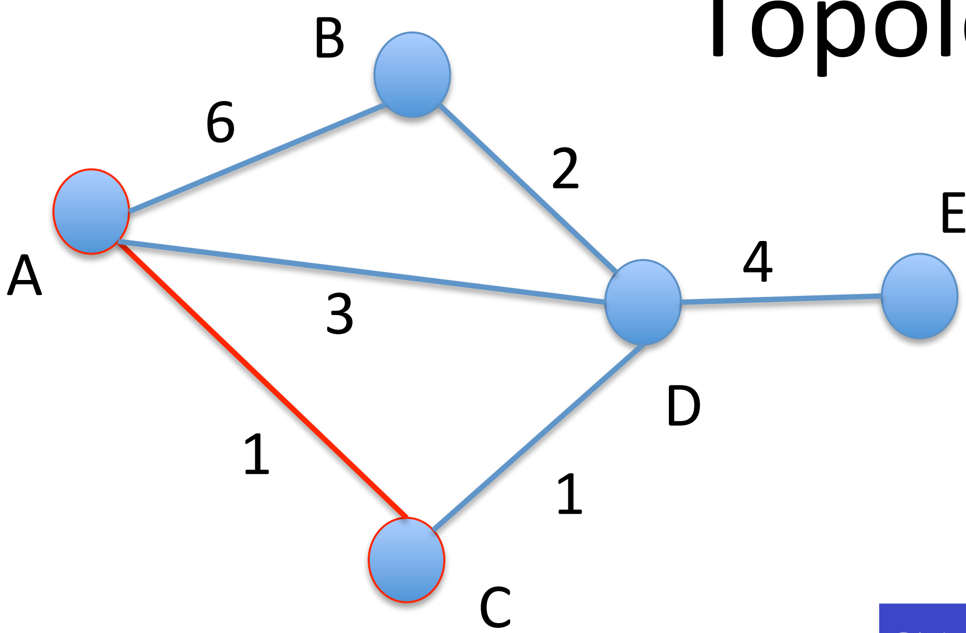
Topology



At Node A

C(N)	B	C	D	E
M={A}	6	1	3	INF

Topology



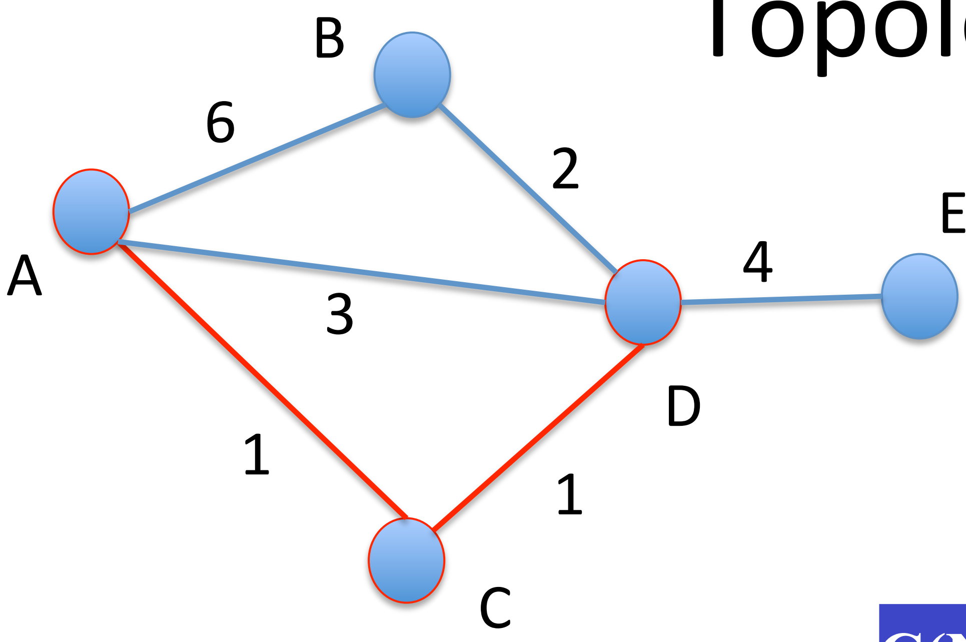
At Node A

ParentC) = A

Cost(D) > Cost(C) + l(C,D)

C(N)	B	C	D	E
M={A}	6	1	3	INF
M={A, C}	6	1	2 ~~3~~	INF

Topology



At Node A

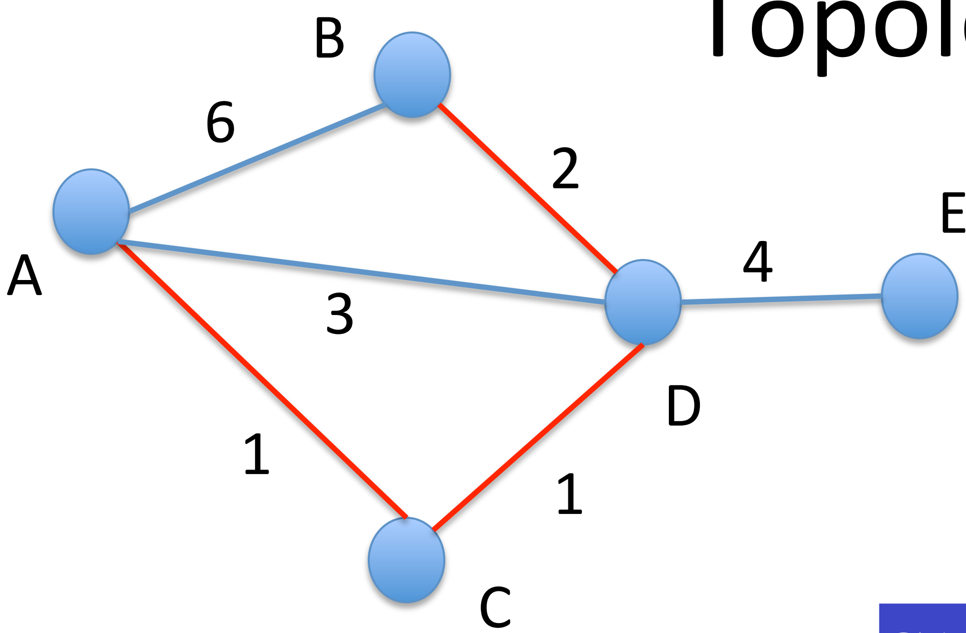
C(N)	B	C	D	E
M={A}	6	1	3	INF
M={A, C}	6	1	2	INF
M={A,C,D}	4 ~~6~~	1	2	6 ~~INF~~

ParentD) = C

Cost(B) > Cost(D) + l(D,B)

Cost(E) > Cost(D) + l(D,E)

Topology

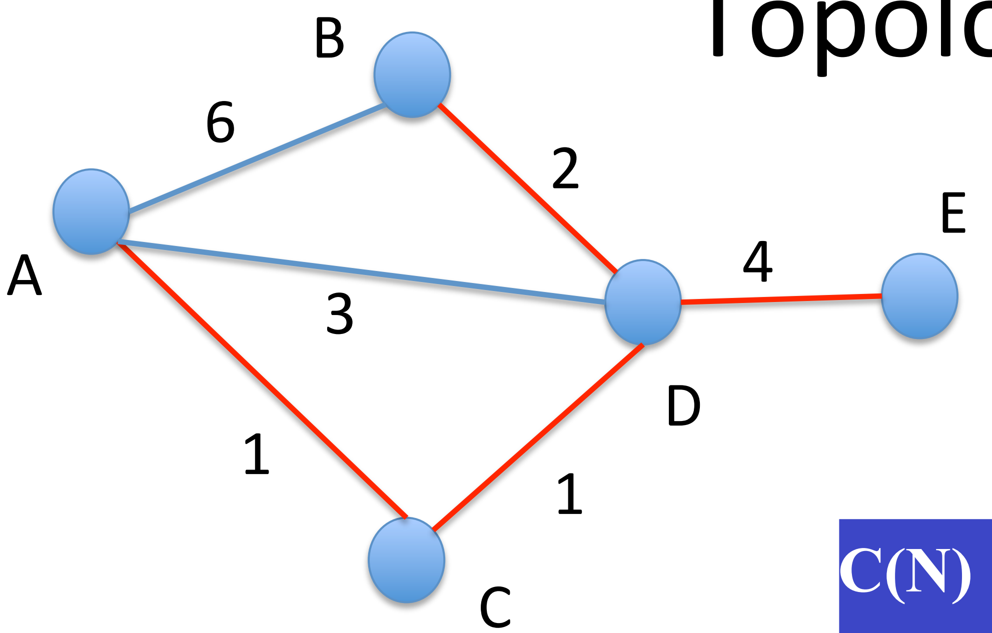


At Node A

Parent(B) = D

C(N)	B	C	D	E
M={A}	6	1	3	INF
M={A, C}	6	1	2	INF
M={A,C,D}	4	1	2	6
M={A,C,D,B}	4	1	2	6

Topology

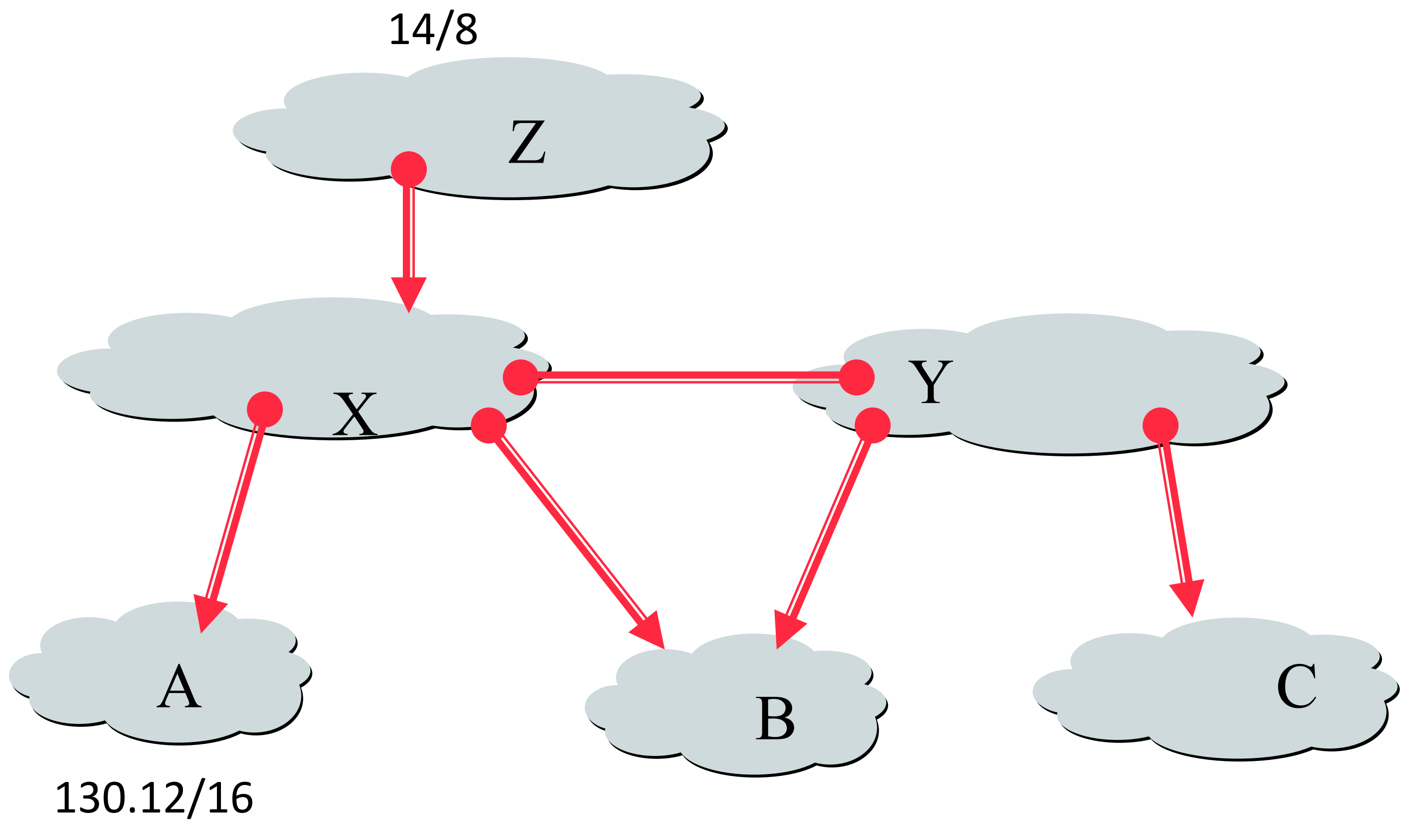


At Node A

Parent(E) = D

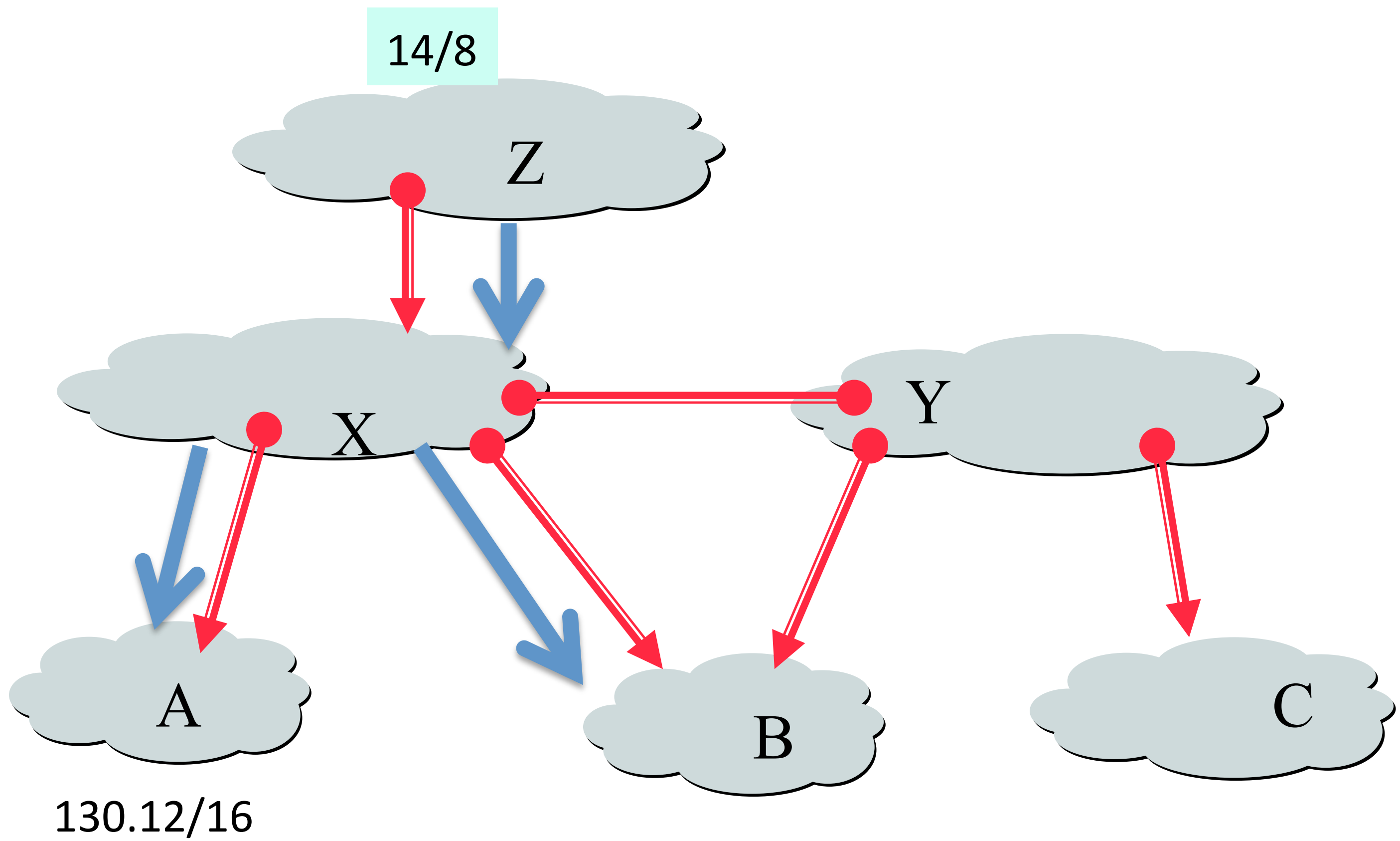
C(N)	B	C	D	E
M={A}	6	1	3	INF
M={A, C}	6	1	2	INF
M={A,C,D}	4	1	2	6
M={A,C,D,B}	4	1	2	6
M={A,C,D,B,E}	4	1	2	6

Along what paths will each of the IP address blocks below be exported in the topology below?
Recall arrow points from provider to customer, line without arrow connects peers



Along what paths will each of the IP address blocks be exported in the topology below?

For 14/8



Along what paths will each of the IP address blocks be exported in the topology below?

For 130.12/16