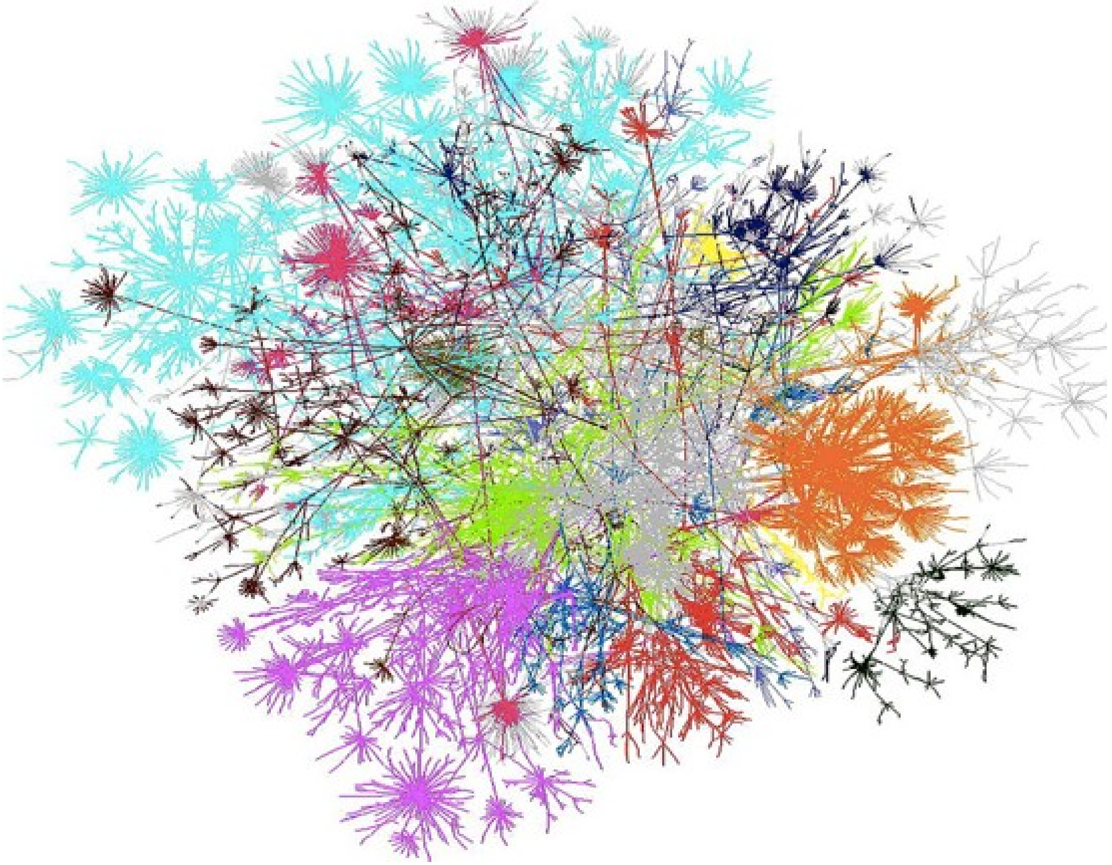
CS 31006: Computer Networks – Internet Routing



Department of Computer Science and Engineering

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Rajat Subhra Chakraborty
rschakraborty@cse.iitkgp.ac.in

Sandip Chakraborty
sandipc@cse.iitkgp.ac.in

A Practice Problem

IP	Netmask	Next hop
128.96.170.0	255.255.254.0	Eth0
128.96.168.0	255.255.254.0	Eth1
128.96.166.0	255.255.254.0	R2
128.96.164.0	255.255.252.0	R3
0.0.0.0	0.0.0.0	R4

- What would be next hop for the following destination IPs?
  - 128.96.171.92
  - 128.96.167.151
  - 128.96.163.151

Problem courtesy: Computer Networks, Larry L Peterson and Bruce S Davie

A Practice Problem

IP	Netmask	Next hop
128.96.170.0	255.255.254.0	Eth0
128.96.168.0	255.255.254.0	Eth1
128.96.166.0	255.255.254.0	R2
128.96.164.0	255.255.252.0	R3
0.0.0.0	0.0.0.0	R4

- What would be next hop for the following destination IPs?
  - 128.96.171.92

171 -- 10101011
254 -- 11111110

Logical AND -- 10101010 -- 170

Destination is Eth0

Problem courtesy: Computer Networks, Larry L Peterson and Bruce S Davie

A Practice Problem

IP	Netmask	Next hop
128.96.170.0	255.255.254.0	Eth0
128.96.168.0	255.255.254.0	Eth1
128.96.166.0	255.255.254.0	R2
128.96.164.0	255.255.252.0	R3
0.0.0.0	0.0.0.0	R4

- What would be next hop for the following destination IPs?
  - 128.96.167.151

  167 -- 10100111
  254 -- 11111110

  Logical AND -- 10100110 -- 166

Problem courtesy: Computer Networks, Larry L Peterson and Bruce S Davie

A Practice Problem

IP	Netmask	Next hop
128.96.170.0	255.255.254.0	Eth0
128.96.168.0	255.255.254.0	Eth1
128.96.166.0	255.255.254.0	R2
128.96.164.0	255.255.252.0	R3
0.0.0.0	0.0.0.0	R4

- What would be next hop for the following destination IPs?
  - 128.96.167.151

  167 -- 10100111

  252 -- 11111100

  Logical AND -- 10100100 -- 164

  Longest Match is 254

  Next hop Eth1

Problem courtesy: Computer Networks, Larry L Peterson and Bruce S Davie

Routing in the Internet

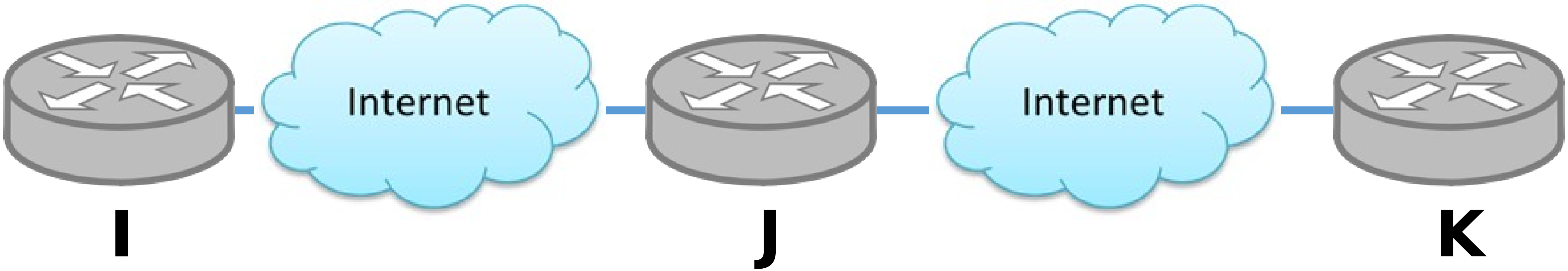
- Two different variants of routing protocols – (a) **Intra-AS routing protocols**, and (b) **Inter-AS routing protocols**

- **Intra-AS (Intra Domain) Routing Protocols:** Routing within an AS; Example- Routing Information Protocol (RIP), Open Shortest Path First (OSPF). Sometimes, they are called as **Interior Gateway Protocols (IGP)**

- **Inter-AS (Inter Domain) Routing Protocols:** Routing among the various ASes based on peering relationship; Example: Border Gateway Protocol (BGP). These are known as **Exterior Gateway Protocols (EGP)**

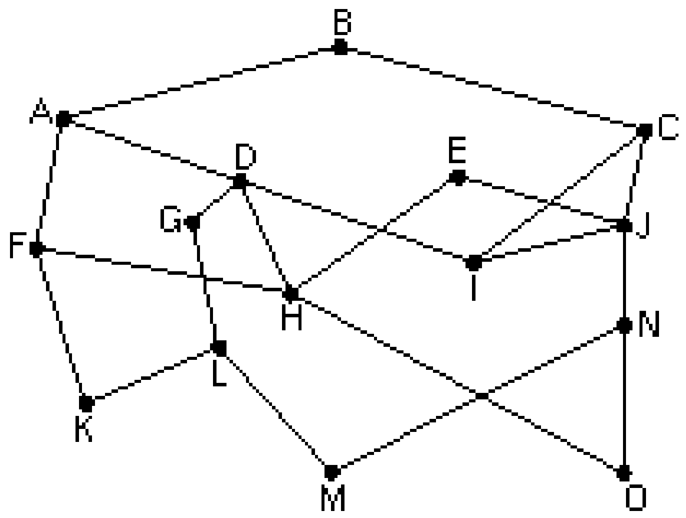
The Optimality Principle (Bellman 1957)

If Router J is on the optimal path from Router I to Router K, the the optimal path from J to K also falls among the same route.

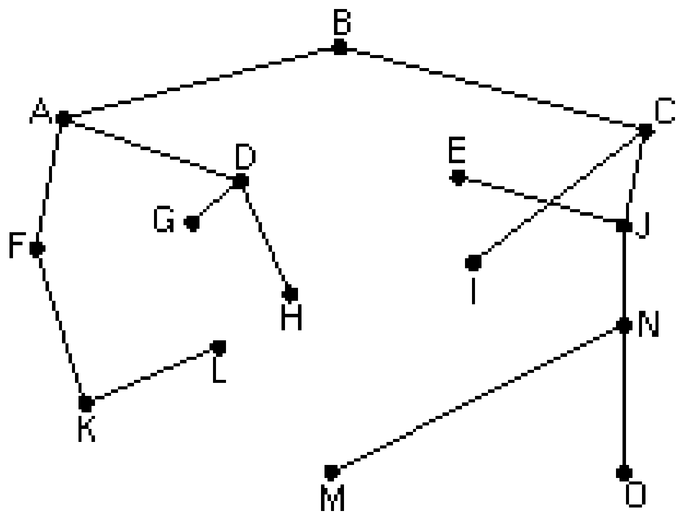


Sink Tree

The set of optimal routes from all sources to a given destination form a tree rooted at the destination. Such a tree is called a sink tree.



(a)



(b)

Can you tell, why we are more focused to a destination centric tree, rather than a

Sink Tree

- Sink tree for a destination may not be unique – there can be an alternate path between two routers with the same path length.
- We may use a more generic data structure – a **Directed Acyclic Graph (DAG),** denoting all possible paths from all the source to a destination.
- Both a tree or a DAG is okay – our objective here is to ensure **loop-free routing**

Distance Vector Routing

- Operates by having each router maintain a table (i.e., a vector) giving the best known distance to each destination and which link to use to get there.
- A distributed version of the **Bellman-Ford Algorithm for Shortest Path.**
- Used in **Routing Information Protocol (RIP) –** the original ARPANET routing algorithm.

Distance Vector Routing

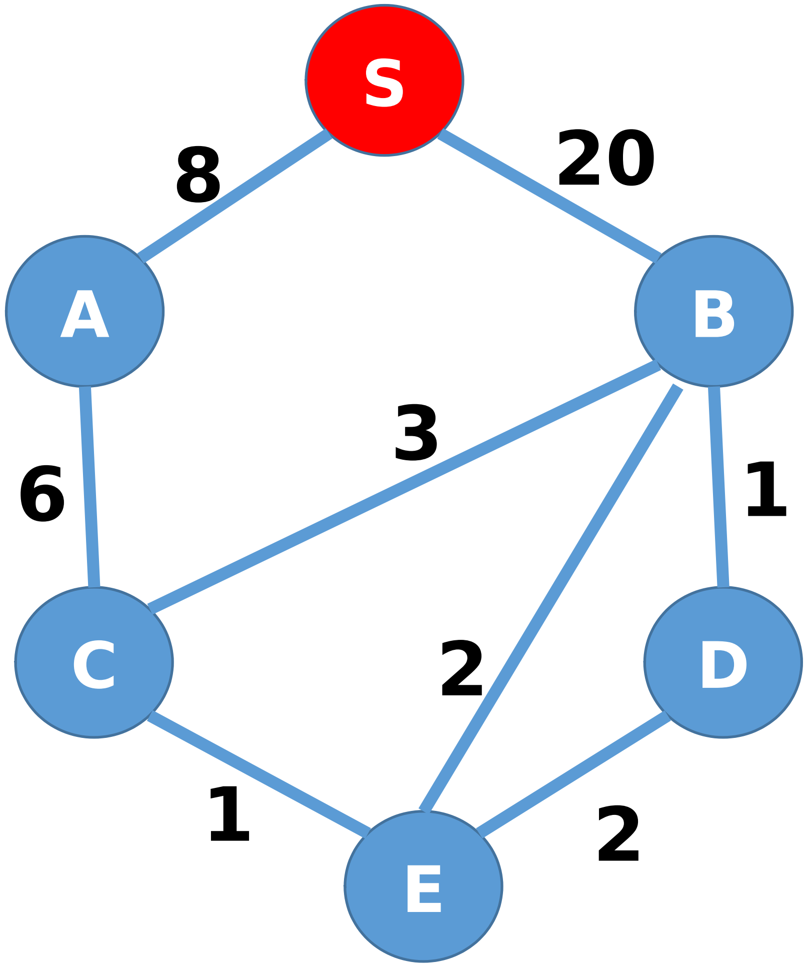
- **Bellman Ford Equation (Dynamic Programming)**

: Cost of least cost path from node to node

: Cost from node to node where , the set of neighbors for node

- We start from the sink (the destination), and recursively construct the shortest path from all the nodes in the subnet.

Bellman Ford in Centralized Network



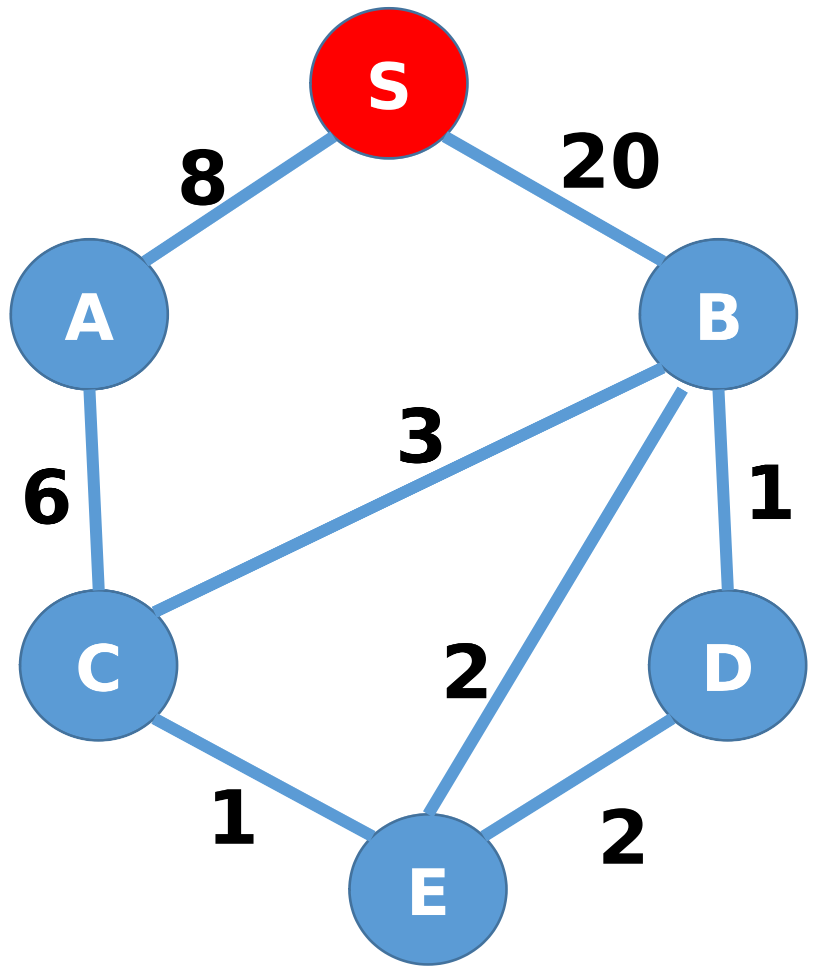
Initialize

S	A	B	C	D	E
0	8	20	∞	∞	∞
-	S	S	-	-	-

Iteration 1

S	A	B	C	D	E
0	8	20	14	21	22
-	S	S	A	B	B

Bellman Ford in Centralized Network



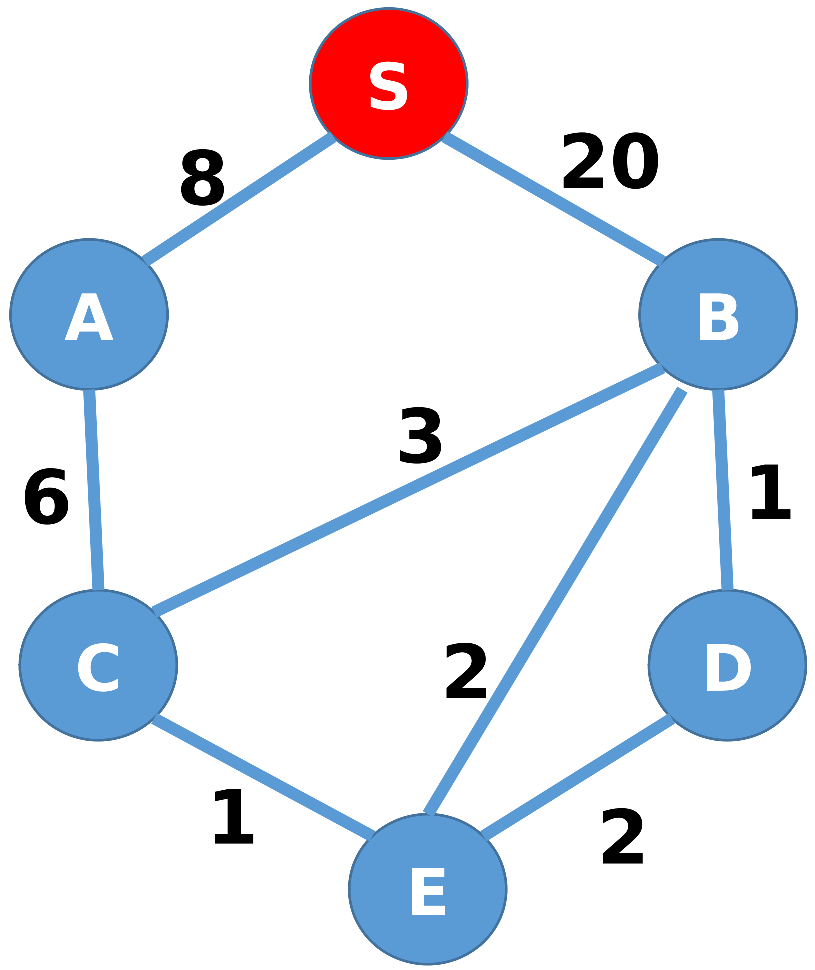
Iteration 1

S	A	B	C	D	E
0	8	20	14	21	22
-	S	S	A	B	B

Iteration 2

S	A	B	C	D	E
0	8	17	14	21	15
-	S	C	A	B	C

Bellman Ford in Centralized Network



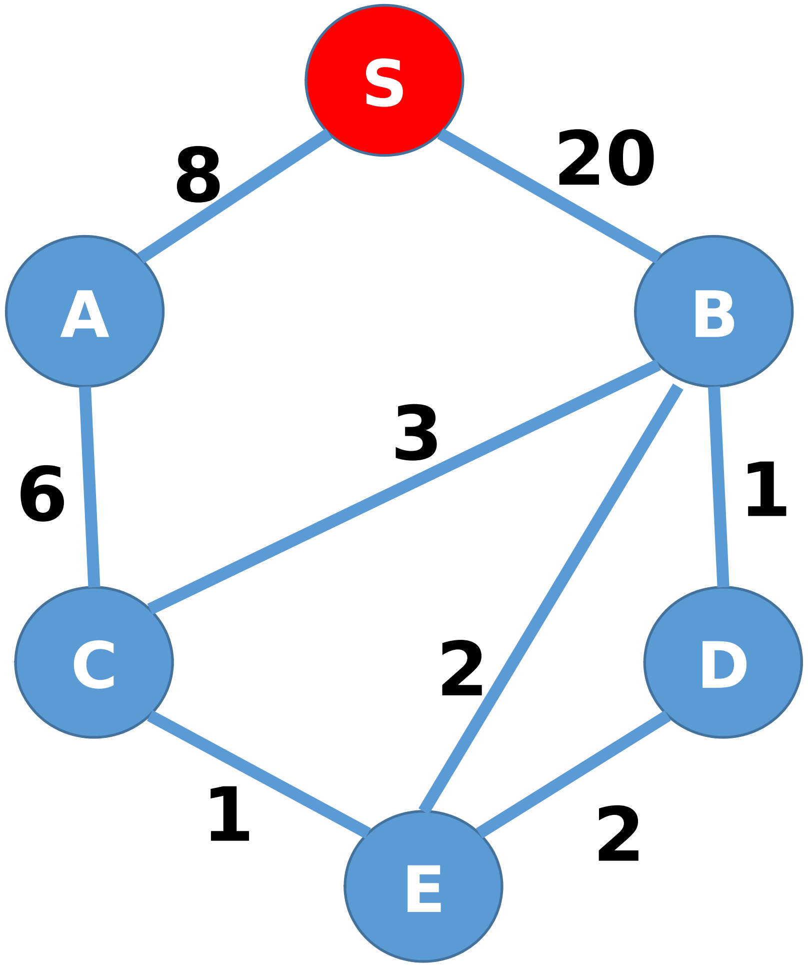
Iteration 2

S	A	B	C	D	E
0	8	17	14	21	15
-	S	C	A	B	C

Iteration 3

S	A	B	C	D	E
0	8	17	14	17	15
-	S	C	A	E	C

Bellman Ford in Centralized Network



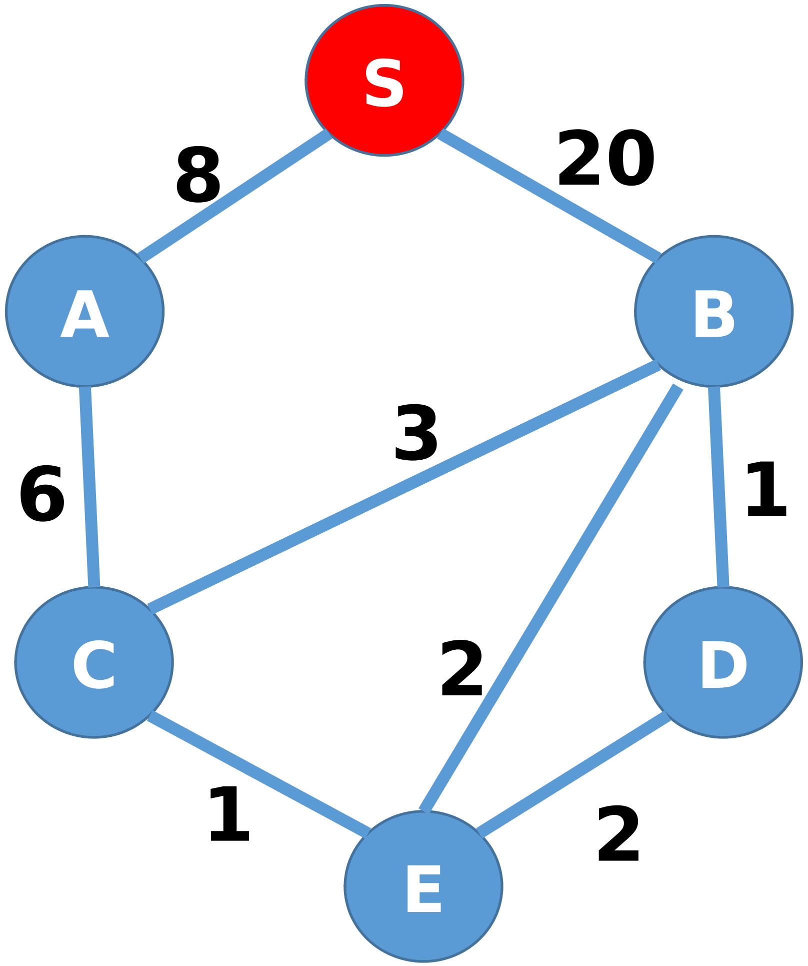
Iteration 3

S	A	B	C	D	E
0	8	17	14	17	15
-	S	C	A	E	C

Iteration 4

S	A	B	C	D	E
0	8	17	14	17	15
-	S	C	A	E	C

Distributed Bellman Ford To Distance Vector Routing



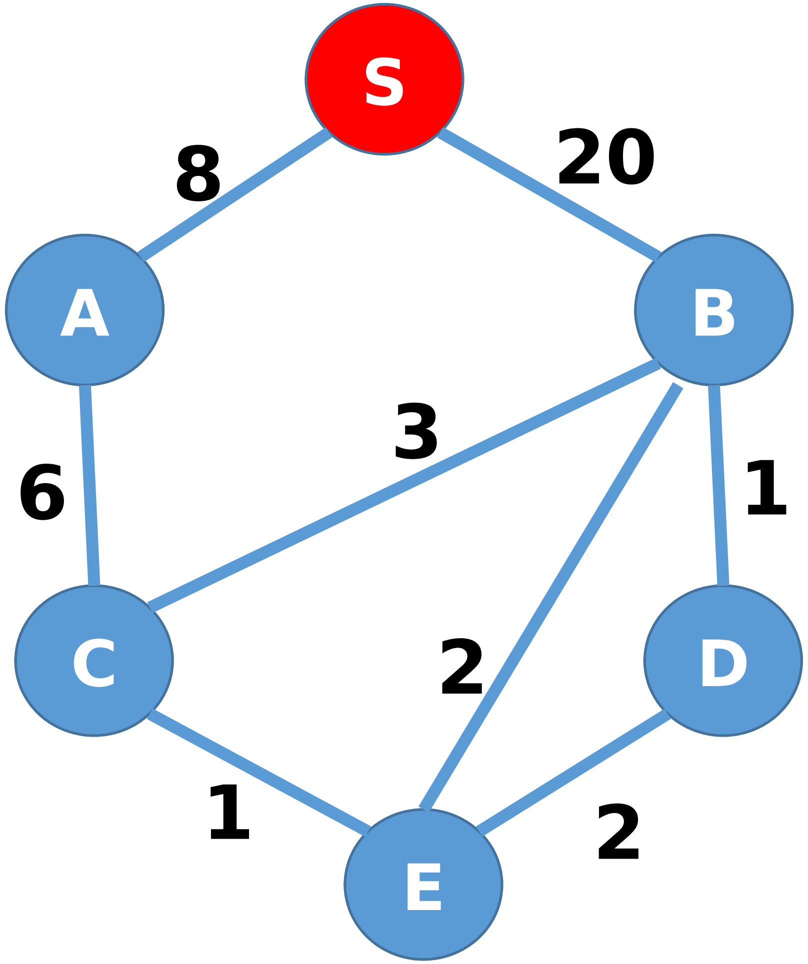
- How do we move from one iteration to the next iteration for Bellman Ford algorithm?

- **We need the currently computed distance information from each node to others in the network – this is called the distance vector**

- **Used to find out the routes between any two nodes**

- In RIP with CIDR, the routing table

Distributed Bellman Ford To Distance Vector Routing



	S	A
S	00,-	08,S
A	08,A	00,-
B	20,B	9,C
C	23,B	6,C
D	21,B	9,C
E	15,A	7,C

	S
S	00,-
A	08,A
B	17,A
C	14,A
D	17,A
E	15,A

Centralized Bellman Ford Versus Distance Vector Routing

- Note the differences between **centralized Bellman Ford algorithm and** its distributed version used in **distance vector routing.**

- We are destination oriented (path to sink) for both the variants

- **Centralized Bellman Ford:** We find out the shortest path from all other nodes to the sink

- **Distance Vector Routing:** We find out the shortest path from the current node to all other nodes that can work as the sink

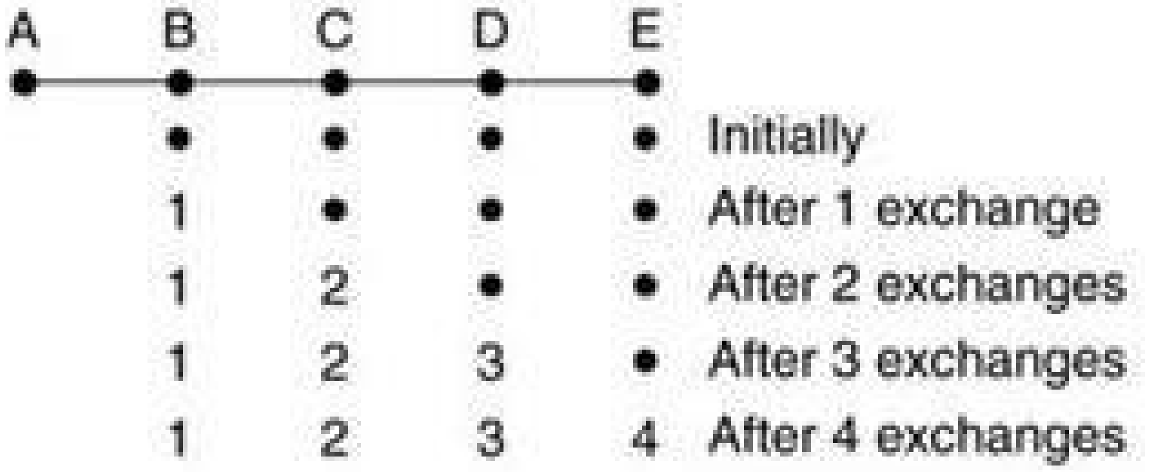
Centralized Bellman Ford Versus Distance Vector Routing

- **Can you guess the reason for this? – Think of the convergence criteria**

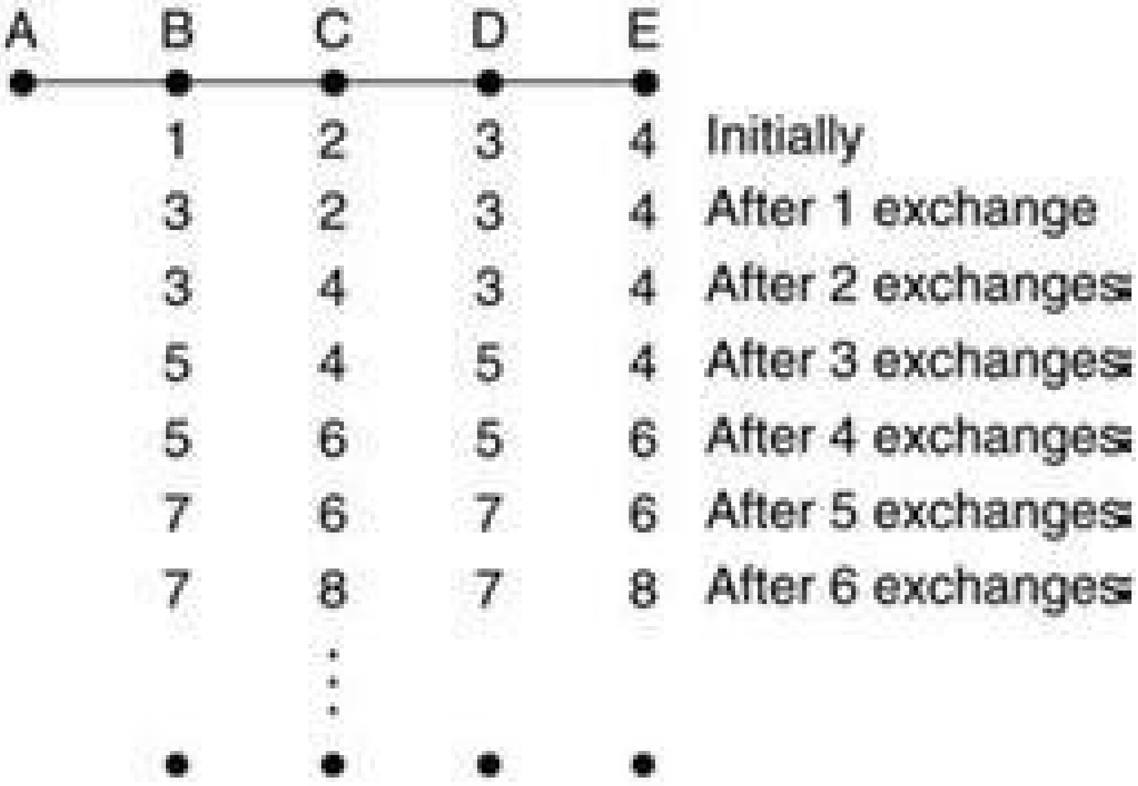
- **Centralized Bellman Ford:** You know the number of iterations required. For n number of nodes, you need n-1 iterations.
  - **Requires synchronous updates of distance tables**

- **Distance Vector Routing:** You do not know the number of iterations required, number of nodes (n) changes dynamically, distance vector updates are asynchronous
  - **Update the local routing table asynchronously as and when you receive an update from a neighbor**

The Count to Infinity Problem



(a)



(b)

Link A-B comes up

Link A-B gets down

Solving Count to Infinity

- **Split Horizon:** Routing information is prevented from exiting the router on an interface through which the information was received.
  - A router needs to wait for a timeout to remove an entry from a routing table

- **Poisson Reverse:** All known networks are advertised in each routing update. However, those networks learned through a specific interface are advertised as **unreachable** in the routing announcements sent out to that interface.
  - Routes are immediately removed when they are marked as unreachable from all the routing updates from all the neighbors.

- However, the solutions adds up cost – increases the size of

Limitations of Distance Vector Routing / RIP

- The resolution to the counting to infinity problem enforces a maximum cost for a network path (generally 15 in RIP). This limits the diameter of a AS to a maximum of 15 hops.
- High signaling overhead - Periodic broadcasting of the distance vector table can result in increased utilization of the network resources for signaling.
- The algorithm is relatively slow to converge; you require information from all the nodes in the AS.