Module 5
Routing Protocols

Routing Protocols

- Routing Protocols: Generally in routing, a packet is routed, hop by hop, from its source to its destination by the help of forwarding tables.
- The source host needs no forwarding table because it delivers its packet to the default router in its local network.
- The destination host also needs no forwarding table because it receives the packet from its default router in its local network.
- This means that only the routers that glue together the networks in the internet need forwarding tables.

Routing Protocols

- An Internet as a Graph: To find the best route, an internet can be modeled as a graph.
- A graph is a set of nodes and edges (lines) that connect the nodes.
- To model an internet as a graph, let be each router as a node and each network between a pair of routers as an edge.
- In fact, internet can be modeled as a weighted graph, in which each edge is associated with a cost.

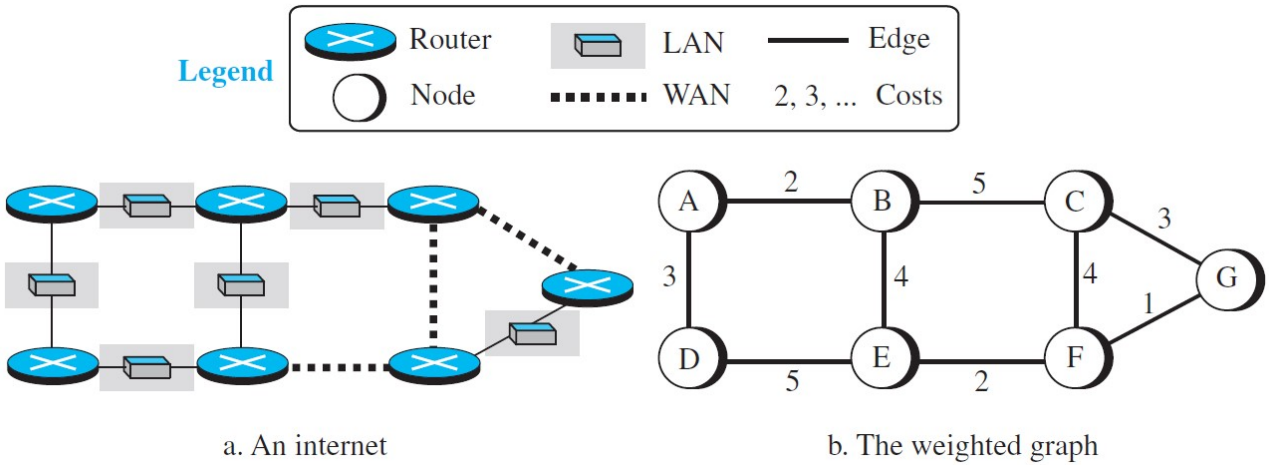


Figure 1 : An internet and its graphical representation

Routing Protocols

- Least-Cost Trees: If there are N routers in an internet, there are $(N-1)$ least-cost paths from each router to any other router.
- This means, $N \times (N-1)$ least-cost paths for the whole internet.
  If we have only 10 routers in an internet, we need 90 least-cost paths.
- A better way to see all of these paths is to combine them in a least-cost tree.
- A least-cost tree is a tree with the source router as the root that spans the whole graph (visits all other nodes) and in which the path between the root and any other node is the shortest.
- In this way, we can have only one shortest-path tree for each node;
  N least-cost trees for the whole internet.

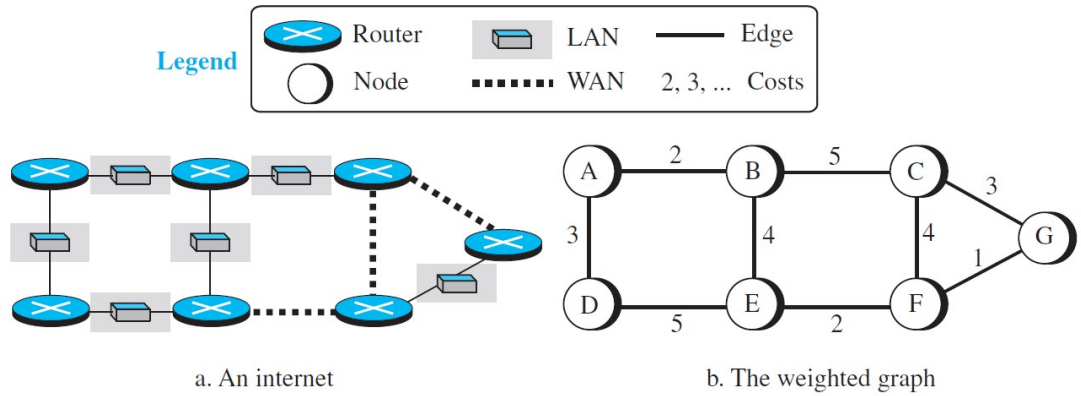


Figure 2 : An internet and its graphical representation

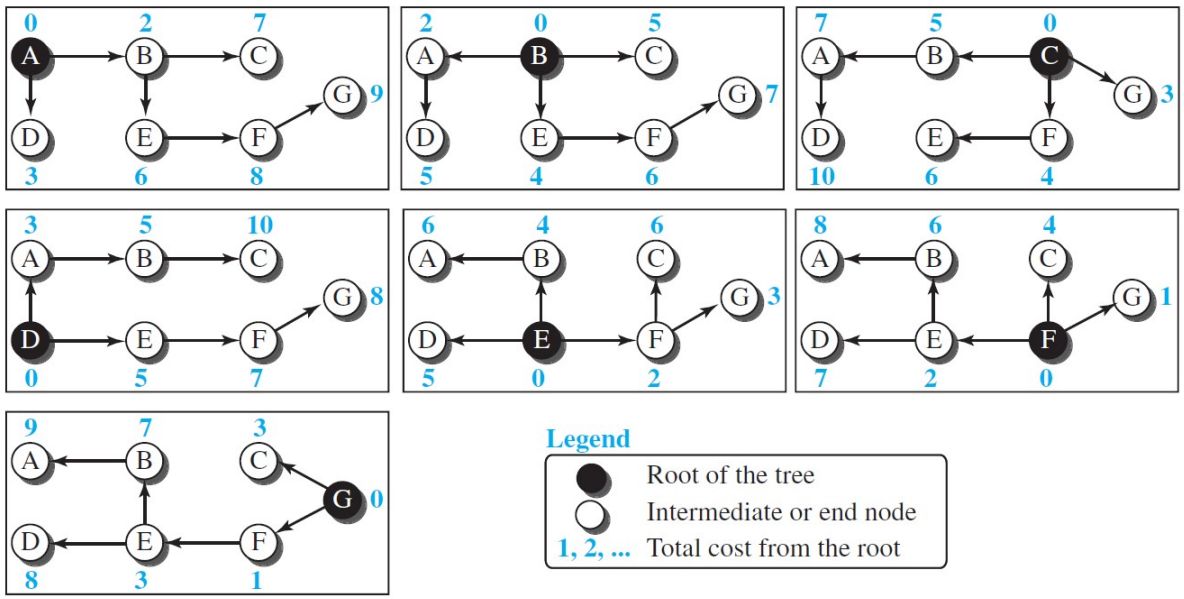
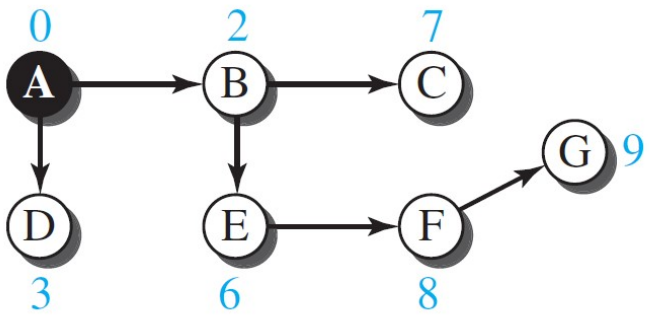


Figure 3 : Least-cost trees for nodes in the above internet

Routing Protocols

- Distance-Vector Routing:
  - A least-cost tree is a combination of least-cost paths from the root of the tree to all destinations.
    These paths are graphically glued together to form the tree.
  - Distance-vector routing unglues these paths and creates a distance vector, a one-dimensional array to represent the tree.



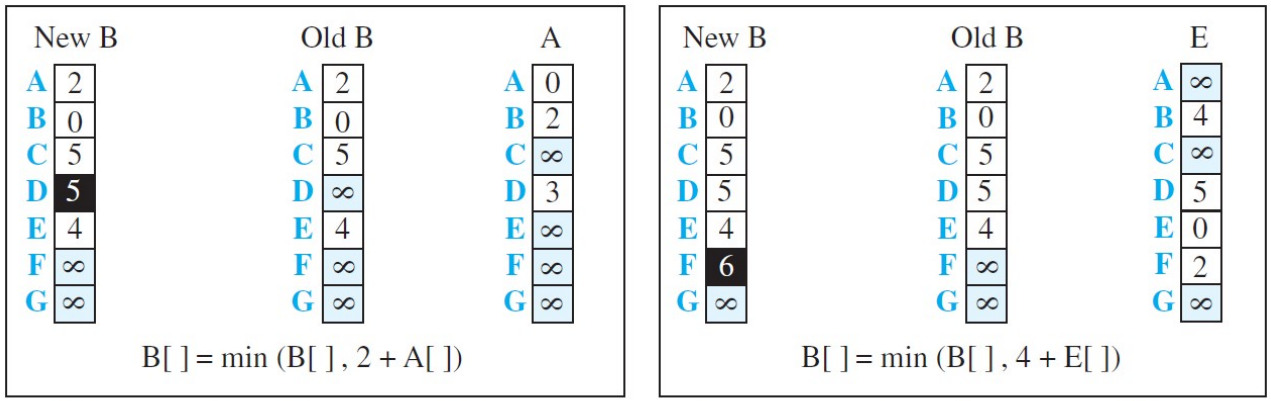
a. Tree for node A

	A
A	0
B	2
C	7
D	3
E	6
F	8
G	9

b. Distance vector for node A

Figure 4 : The distance vector corresponding to a tree

- The rudimentary vectors cannot help the internet to effectively forward a packet.
  For example, node A thinks that it is not connected to node G because the corresponding cell shows the least cost of infinity.
- To improve these vectors, the nodes in the internet need to help each other by exchanging information.
- After each node has created its vector, it sends a copy of the vector to all its immediate neighbors.
- After a node receives a distance vector from a neighbor, it updates its distance vector using the Bellman-Ford equation.



a. First event: B receives a copy of A's vector.

b. Second event: B receives a copy of E's vector.

Figure 5 : Updating distance vectors

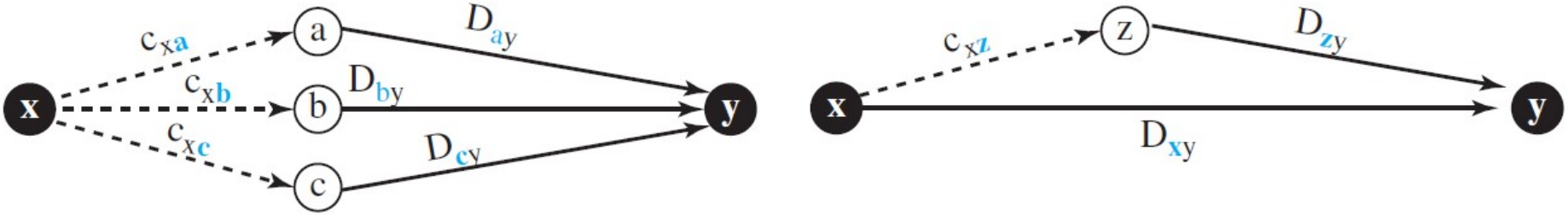
- Bellman-Ford equation: The heart of distance-vector routing is the famous Bellman-Ford equation.
- This equation is used to find the least cost (shortest distance) between a source node, x, and a destination node, y, through some intermediary nodes $(a, b, c, \dots)$ when the costs between the source and the intermediary nodes and the least costs between the intermediary nodes and the destination are given.
- If D_{ij} is the shortest distance and c_{ij} is the cost between nodes i and j

$$D_{xy} = \min\{(c_{xa} + D_{ay}), (c_{xb} + D_{by}), (c_{xc} + D_{cy}), \dots\}$$

- In distance-vector routing, normally we want to update an existing least cost with a least cost through an intermediary node, such as z, if the latter is shorter.
  In this case, the equation becomes simpler, as shown below;

$$D_{xy} = \min\{D_{xy}, (c_{xz} + D_{zy})\}$$

Routing Protocols



a. General case with three intermediate nodes

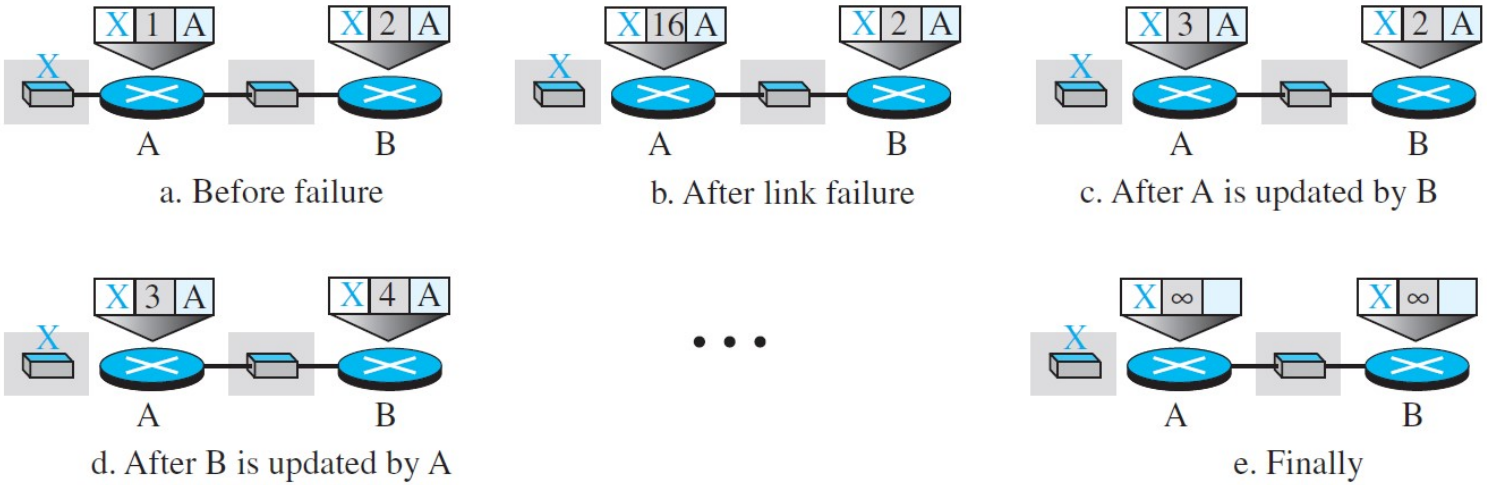
b. Updating a path with a new route

Figure 6 : Graphical idea behind Bellman-Ford equation

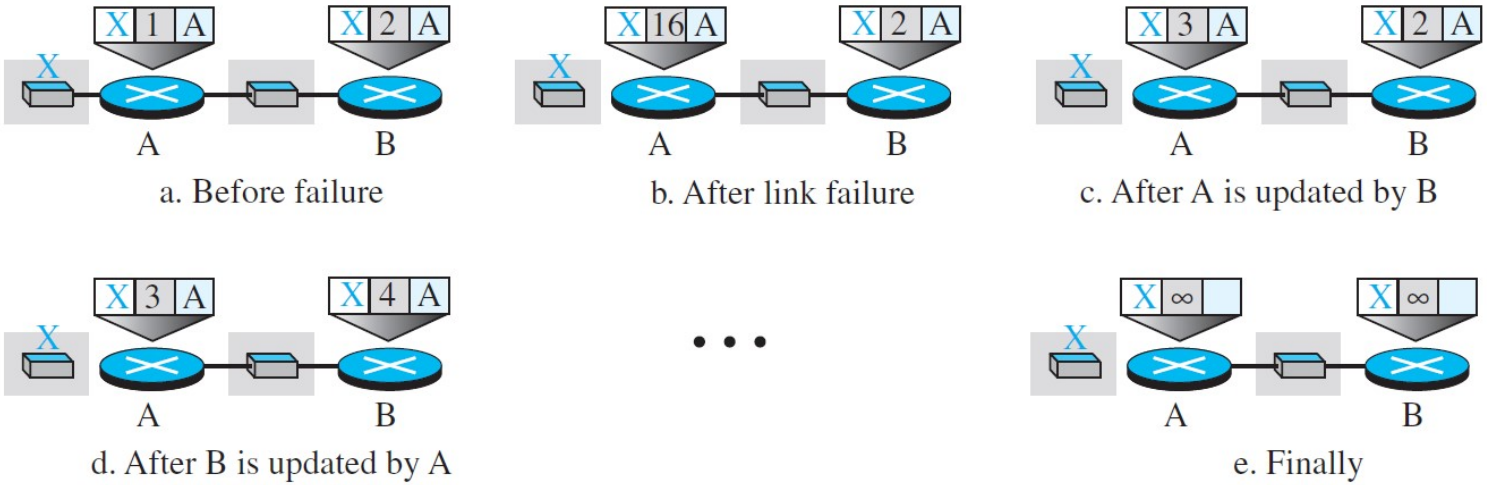
Routing Protocols

- Count to Infinity:
  - A problem with distance-vector routing is that any decrease in cost (good news) propagates quickly, but any increase in cost (bad news) will propagate slowly.
  - For a routing protocol to work properly, if a link is broken (cost becomes infinity), every other router should be aware of it immediately.
  - But in distance-vector routing, this takes some time. The problem is referred to as count to infinity.

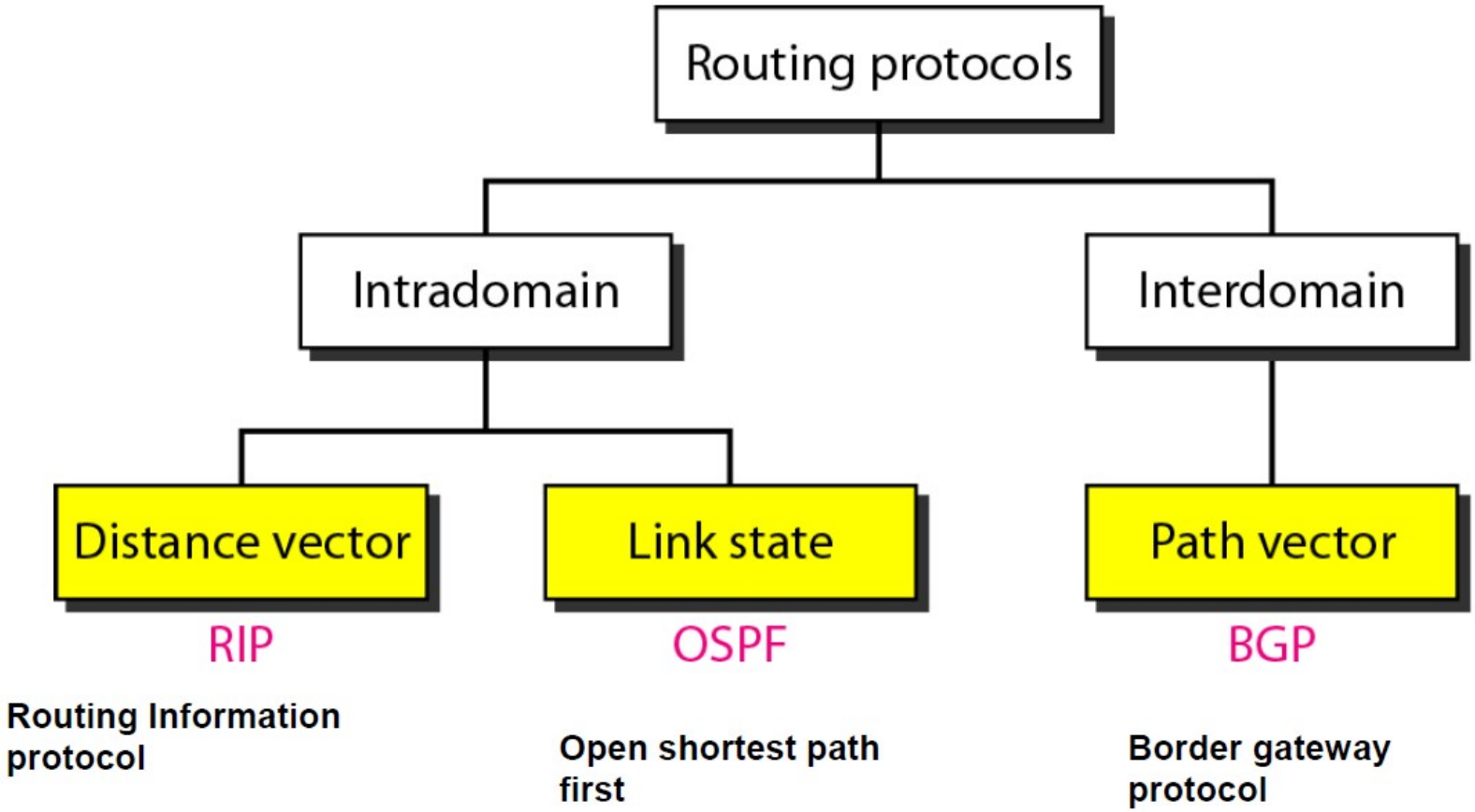
- One example of count to infinity is the two-node loop problem;
  - At the beginning, both nodes A and B know how to reach node X.
  - But suddenly, the link between A and X fails. Then, Node A changes its table.
  - If A can send its table to B immediately, everything is fine.
    However, the system becomes unstable if B sends its forwarding table to A before receiving As forwarding table.
  - Node A receives the update and, assuming that B has found a way to reach X, immediately updates its forwarding table
  - Now A sends its new update to B.
  - Now B thinks that something has been changed around A and updates its forwarding table.



- One example of count to infinity is the two-node loop problem;
  - The cost of reaching X increases gradually until it reaches infinity. At this moment, both A and B know that X cannot be reached.
  - However, during this time the system is not stable.
  - Node A thinks that the route to X is via B;
    node B thinks that the route to X is via A.
  - If A receives a packet destined for X, the packet goes to B and then comes back to A.
    Similarly, if B receives a packet destined for X, it goes to A and comes back to B.
  - Packets bounce between A and B, creating a two-node loop problem.



Routing Protocols



Routing Protocols

- Routing Protocols: A routing protocol is a set of rules that specify how routers identify and forward packets along a network path.
- Routing protocols are grouped into two distinct categories;
  - Intra-domain routing protocol
  - Inter-domain routing protocol

Routing Protocols

- Routing Protocols: A routing protocol is a set of rules that specify how routers identify and forward packets along a network path.
- Routing protocols are grouped into two distinct categories;
  - Intra-domain routing protocol
  - Inter-domain routing protocol
- In intra-domain routing protocol, the routers find the best path to route traffic within a network.
  These protocols are used to exchange routing information about destinations that are reachable within a domain.

Routing Protocols

- Routing Protocols: A routing protocol is a set of rules that specify how routers identify and forward packets along a network path.
- Routing protocols are grouped into two distinct categories;
  - Intra-domain routing protocol
  - Inter-domain routing protocol
- In intra-domain routing protocol, the routers find the best path to route traffic within a network.
  These protocols are used to exchange routing information about destinations that are reachable within a domain.
- Inter-domain routing is the process of directing data traffic between different autonomous systems on the internet.

Network layer

- Routing Information Protocol (RIP) is a distance-vector routing protocol. Routers running the distance-vector protocol send all or a portion of their routing tables in routing-update messages to their neighbors.
- Open Shortest Path First (OSPF) is a link state routing protocols that use the shortest path first algorithm to find the best path for data packets to travel between nodes in a network.

Routing Information protocol(RIP)

Routing Information protocol(RIP)

- Routing Information protocol(RIP) is one of the intradomain routing protocol used inside an autonomous system.
  It is used to update routing tables inside the autonomous system.

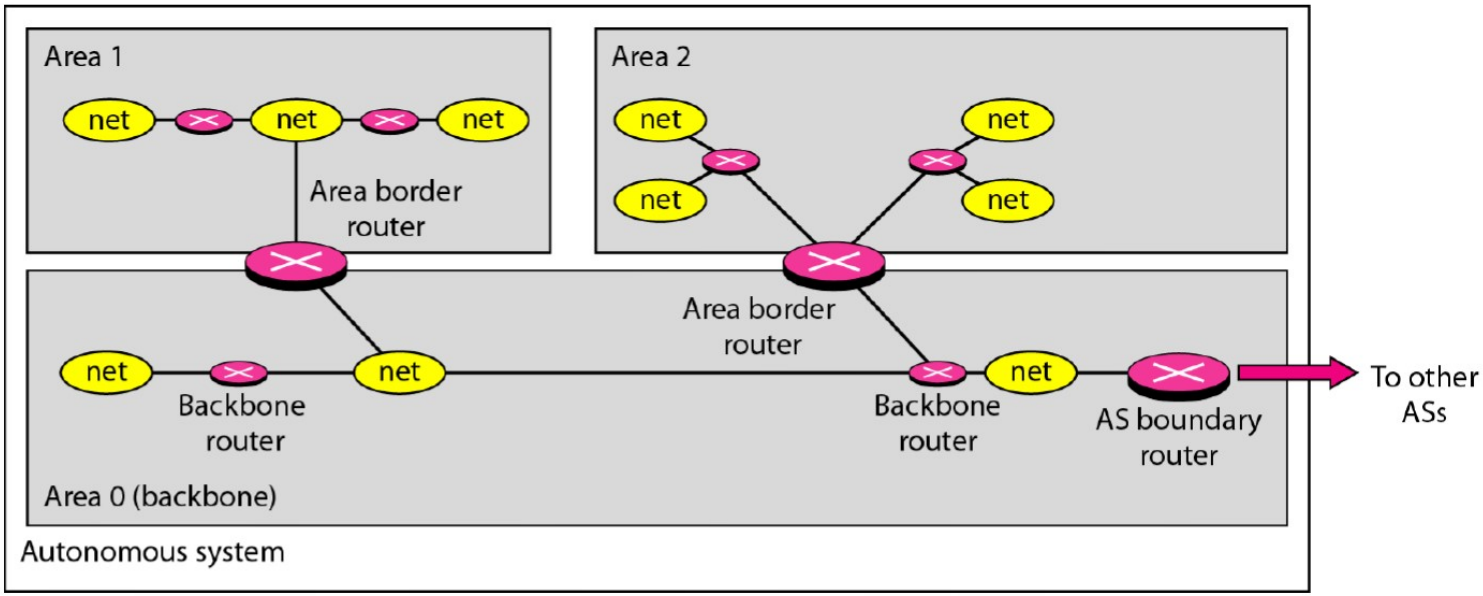


Figure 7 : Areas in an autonomous system

Routing Information protocol(RIP)

- Every router keeps a routing table that has
  - one entry for each destination network address
  - the shortest distance to reach the destination in hop count
  - the next router to which the packet should be delivered to reach its final destination
- The hop count is the number of networks that a packet encounters to reach its final destination.

Routing Information protocol(RIP)

- Hop Count: A router in this protocol basically implements the distance vector routing algorithm.

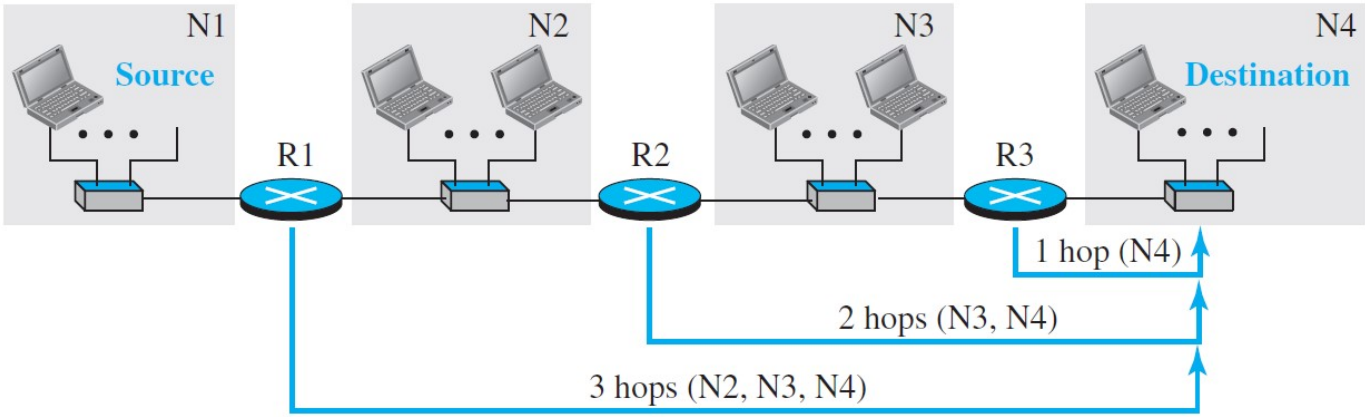


Figure 8 : Hop counts in RIP

- Figure shows the concept of hop count advertised by three routers from a source host to a destination host.
- In RIP, the maximum cost of a path can be 15, which means 16 is considered as infinity (no connection).
- For this reason, RIP can be used only in autonomous systems in which the diameter of the AS is not more than 15 hops.

- Forwarding Tables: A forwarding table in RIP is a three column table in which
  - the first column is the address of the destination network
  - the second column is the address of the next router to which the packet should be forwarded
  - the third column is the cost (the number of hops) to reach the destination network

Forwarding table for R1

Destination network	Next router	Cost in hops
N1	——	1
N2	——	1
N3	R2	2
N4	R2	3

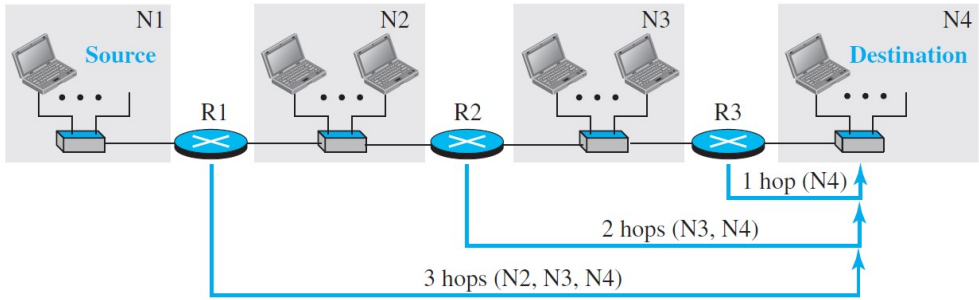
Forwarding table for R2

Destination network	Next router	Cost in hops
N1	R1	2
N2	——	1
N3	——	1
N4	R3	2

Forwarding table for R3

Destination network	Next router	Cost in hops
N1	R2	3
N2	R2	2
N3	——	1
N4	——	1

Figure 9 : Forwarding tables



Routing Information protocol(RIP)

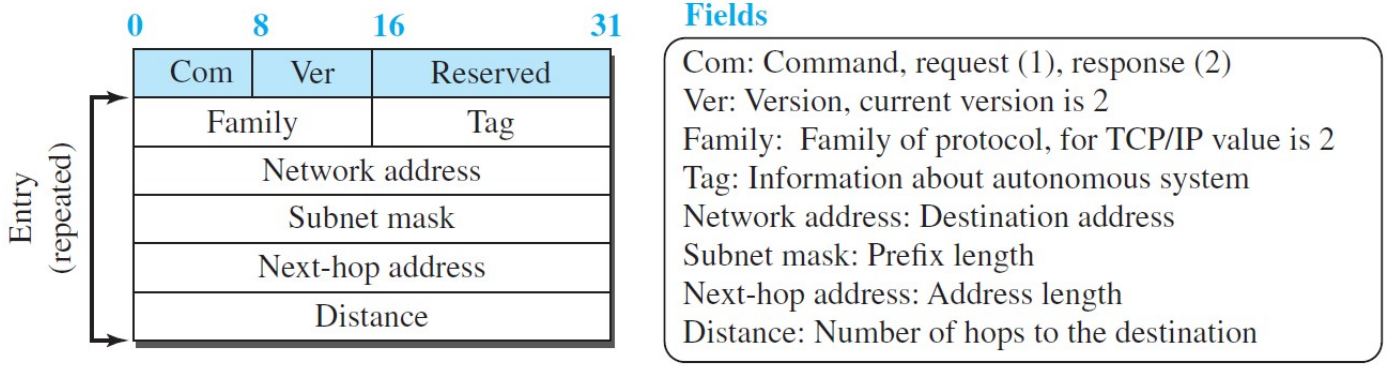


Figure 10 : RIP message format

RIP updating Algorithm

Receive: a response RIP message

1. Add one hop to the hop count for each advertised destination.
2. Repeat the following steps for each advertised destination:
    a. If (destination not in the routing table)
        I. Add the advertised information to the table.
    b. Else
        I. If (next-hop field is the same)
            i. Replace entry in the table with the advertised one.
        II. Else
            i. If (advertised hop count smaller than one in the table)
                - Replace entry in the routing table.
3. Return.

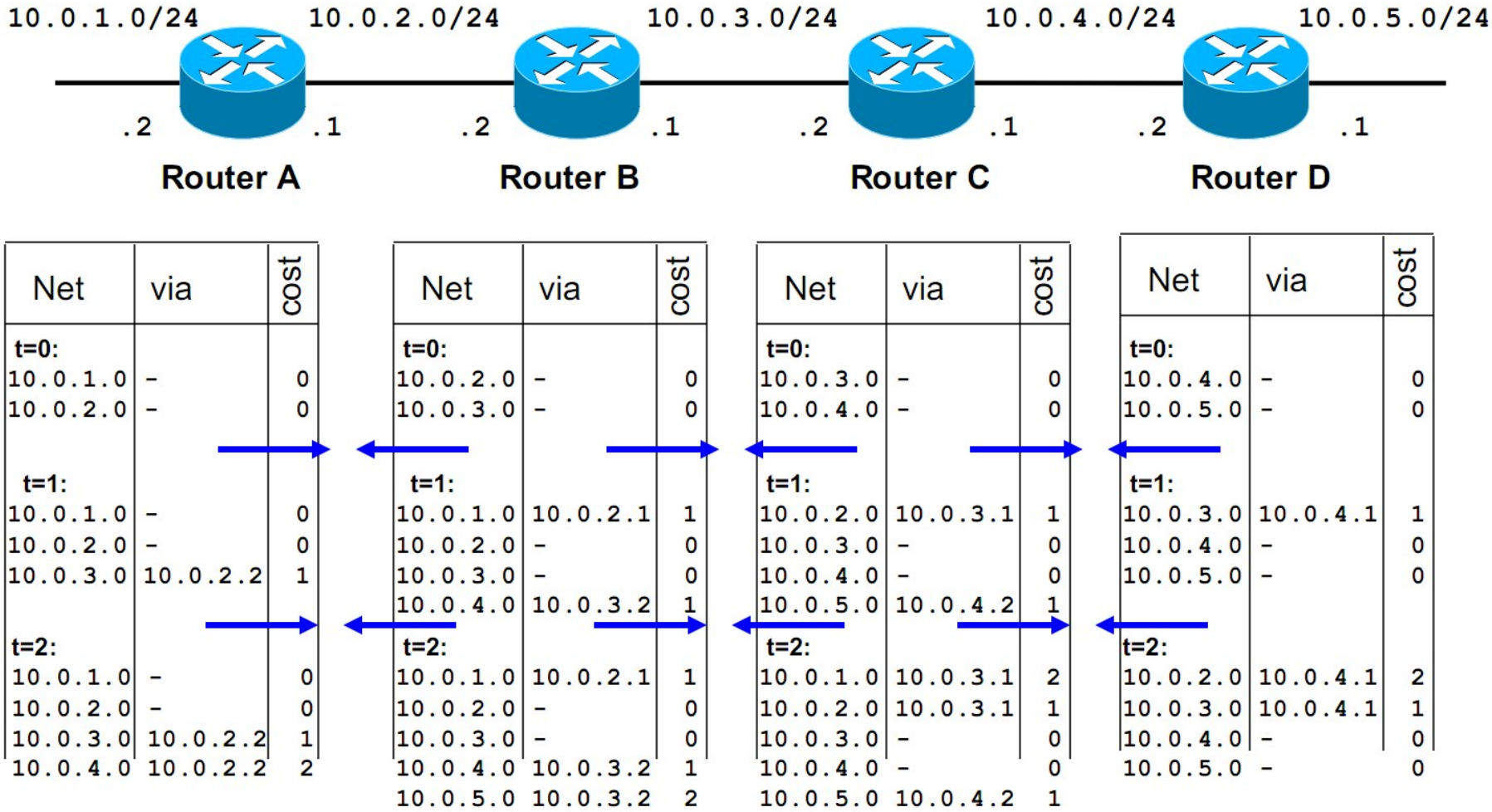
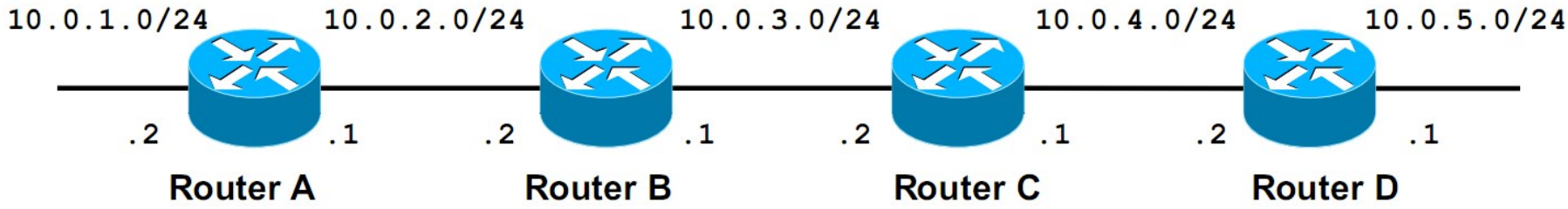


Figure 11 : RIP process



Net	via	cost
t=2:		
10.0.1.0	*-*	*0*
10.0.2.0	*-*	*0*
10.0.3.0	*10.0.2.2*	*1*
10.0.4.0	*10.0.2.2*	*2*
t=3:		
10.0.1.0	-	0
10.0.2.0	-	0
10.0.3.0	10.0.2.2	1
10.0.4.0	10.0.2.2	2
10.0.5.0	10.0.2.2	3

Net	via	cost
t=2:		
10.0.1.0	*10.0.2.1*	*1*
10.0.2.0	*-*	*0*
10.0.3.0	*-*	*0*
10.0.4.0	*10.0.3.2*	*1*
10.0.5.0	*10.0.3.2*	*2*
t=3:		
10.0.1.0	10.0.2.1	1
10.0.2.0	-	0
10.0.3.0	-	0
10.0.4.0	10.0.3.2	1
10.0.5.0	10.0.3.2	2

Net	via	cost
t=2:		
10.0.1.0	*10.0.3.1*	*2*
10.0.2.0	*10.0.3.1*	*1*
10.0.3.0	*-*	*0*
10.0.4.0	*-*	*0*
10.0.5.0	*10.0.4.2*	*1*
t=3:		
10.0.1.0	10.0.3.1	2
10.0.2.0	10.0.3.1	1
10.0.3.0	-	0
10.0.4.0	-	0
10.0.5.0	10.0.4.2	1

Net	via	cost
t=2:		
10.0.2.0	*10.0.4.1*	*2*
10.0.3.0	*10.0.4.1*	*1*
10.0.4.0	*-*	*0*
10.0.5.0	*-*	*0*
t=3:		
10.0.1.0	10.0.4.1	3
10.0.2.0	10.0.4.1	2
10.0.3.0	10.0.4.1	1
10.0.4.0	-	0
10.0.5.0	-	0

Figure 12 : RIP process

Routing Information protocol(RIP)

- A router receives a RIP message from neighboring router.
  The messages lists destination networks and their corresponding hop counts.
- The first step according to the updating algorithm is to increase the hop count by 1.
- Next, this updated RIP packet and the old routing table are compared.
- The result is a routing table with an up-to-date hop count for each destination.

Link State Routing

Link State Routing

- Link state routing is a technique in which each router shares the knowledge of its neighborhood with every other router in the internetwork.
- Link State Routing algorithm involves;
  - Knowledge about the neighborhood: Instead of sending its routing table, a router sends the information about its neighborhood only.
    A router broadcast its identities and cost of the directly attached links to other routers.
  - Flooding: Each router sends the information to every other router on the internetwork. This process is known as Flooding.
    Every router that receives the packet sends the copies to all its neighbors.
    Finally, each and every router receives a copy of the same information.
  - Information sharing: A router sends the information to every other router only when the change occurs in the information.

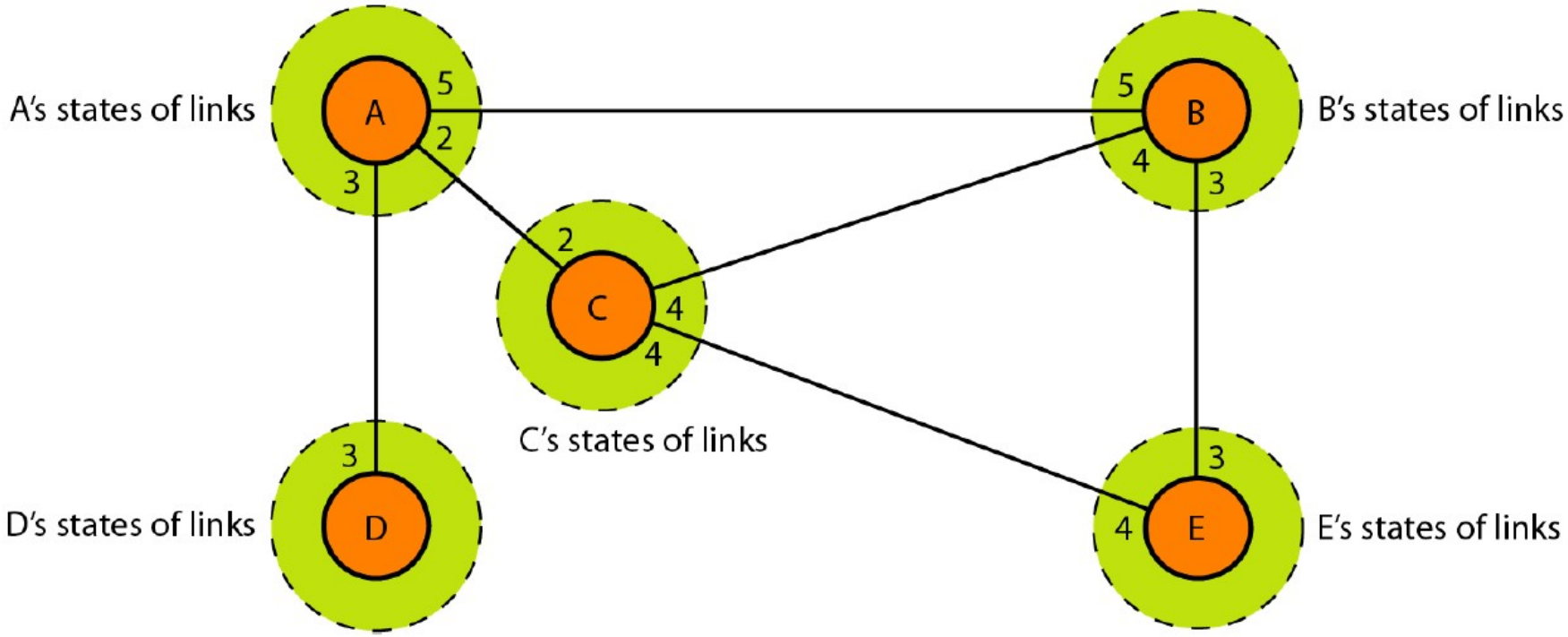
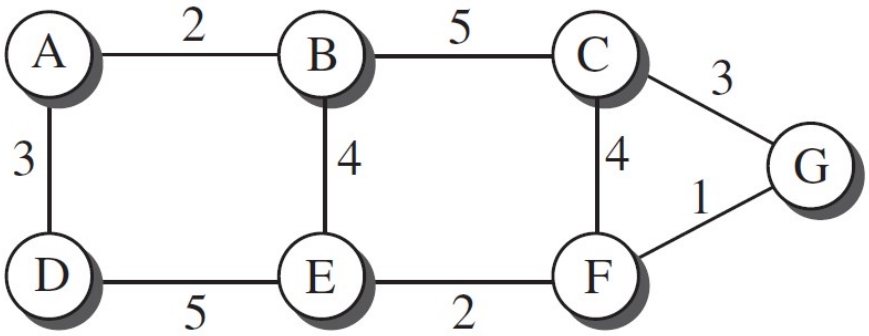


Figure 13 : Link state knowledge

Link State Routing

- Link State Database(LSDB): In Link state routing, the cost associated with an edge defines the state of the link.
- Links with lower costs are preferred to links with higher costs;
  if the cost of a link is infinity, it means that the link does not exist or has been broken.
- To create a least-cost tree with this method, each node needs to have a complete map of the network, which means it needs to know the state of each link.
- The collection of states for all links is called the link-state database (LSDB).
- There is only one LSDB for the whole internet; each node needs to have a duplicate of it to be able to create the least-cost tree.



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

Figure 14 : Example of Link state database

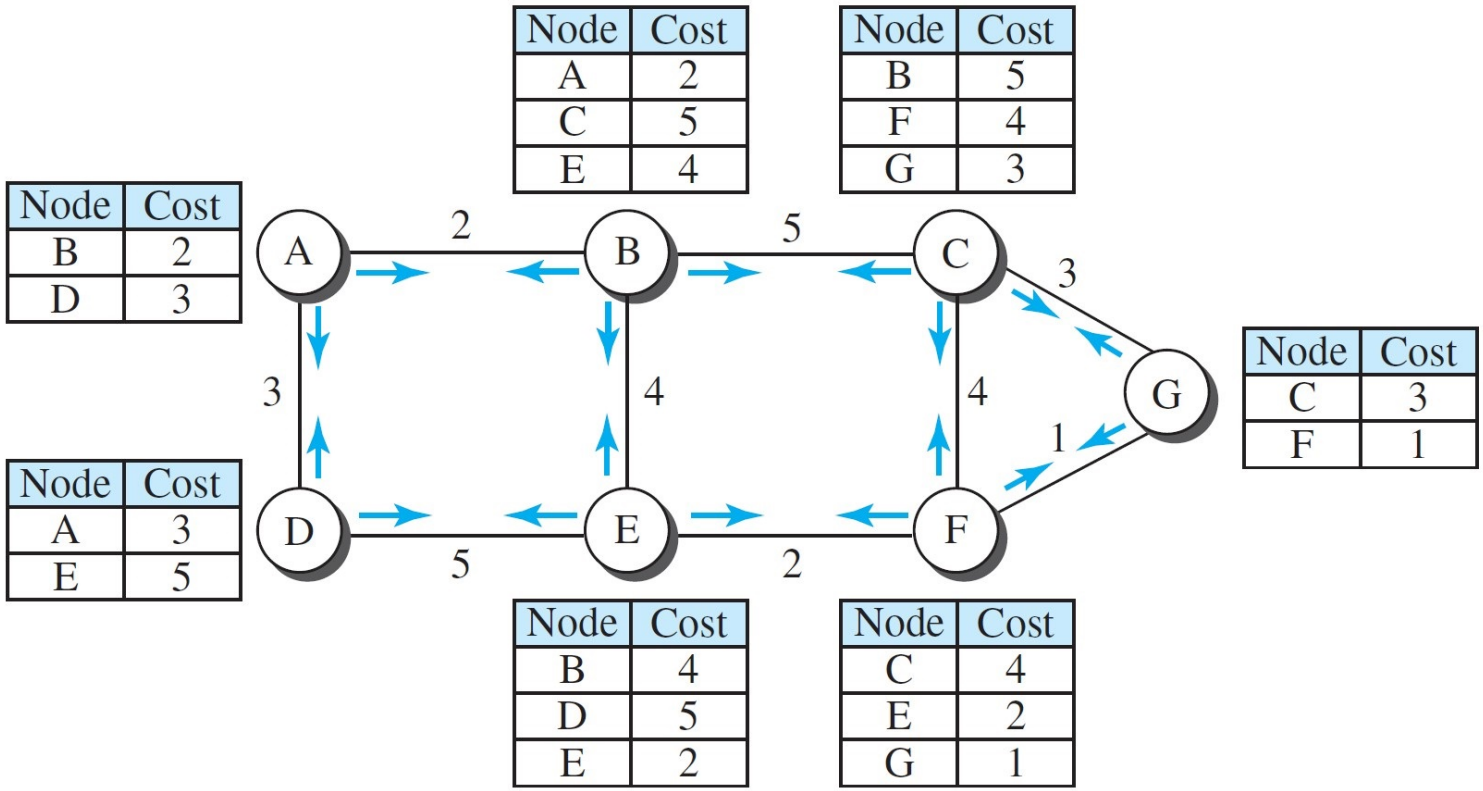


Figure 15 : Link state packet(LSPs) created and sent out by each node to build LSDB

- The Link state routing algorithm is also known as Dijkstra's algorithm which is used to find the shortest path from one node to every other node in the network.

- The Link state routing algorithm is also known as Dijkstra's algorithm which is used to find the shortest path from one node to every other node in the network.

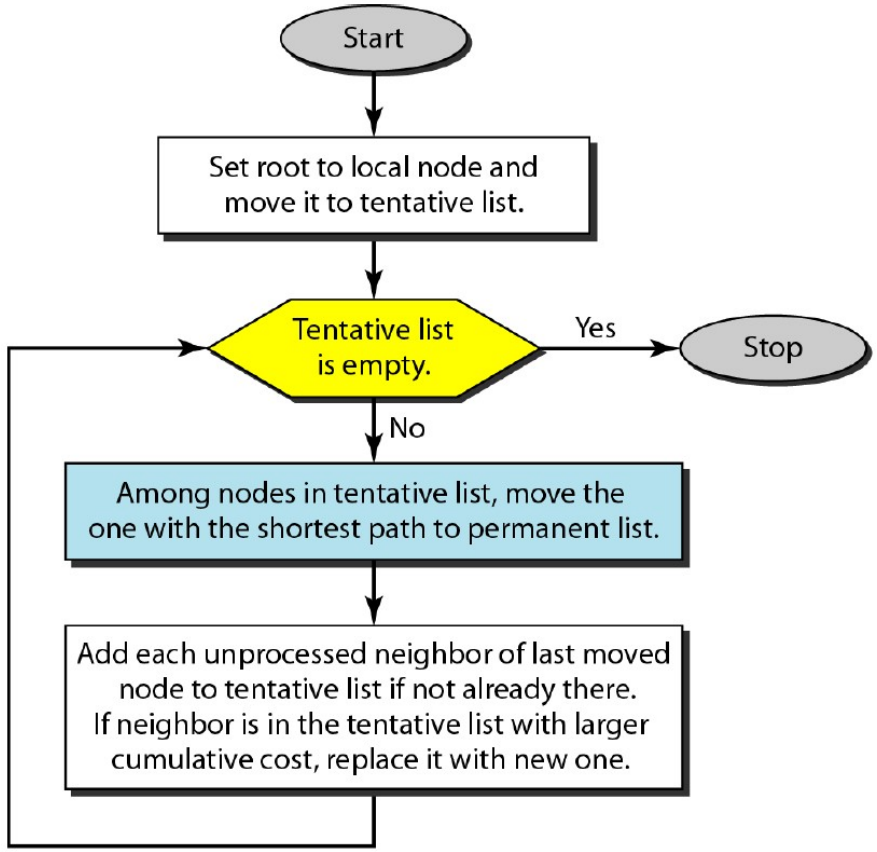


Figure 16 : Dijkstra's algorithm

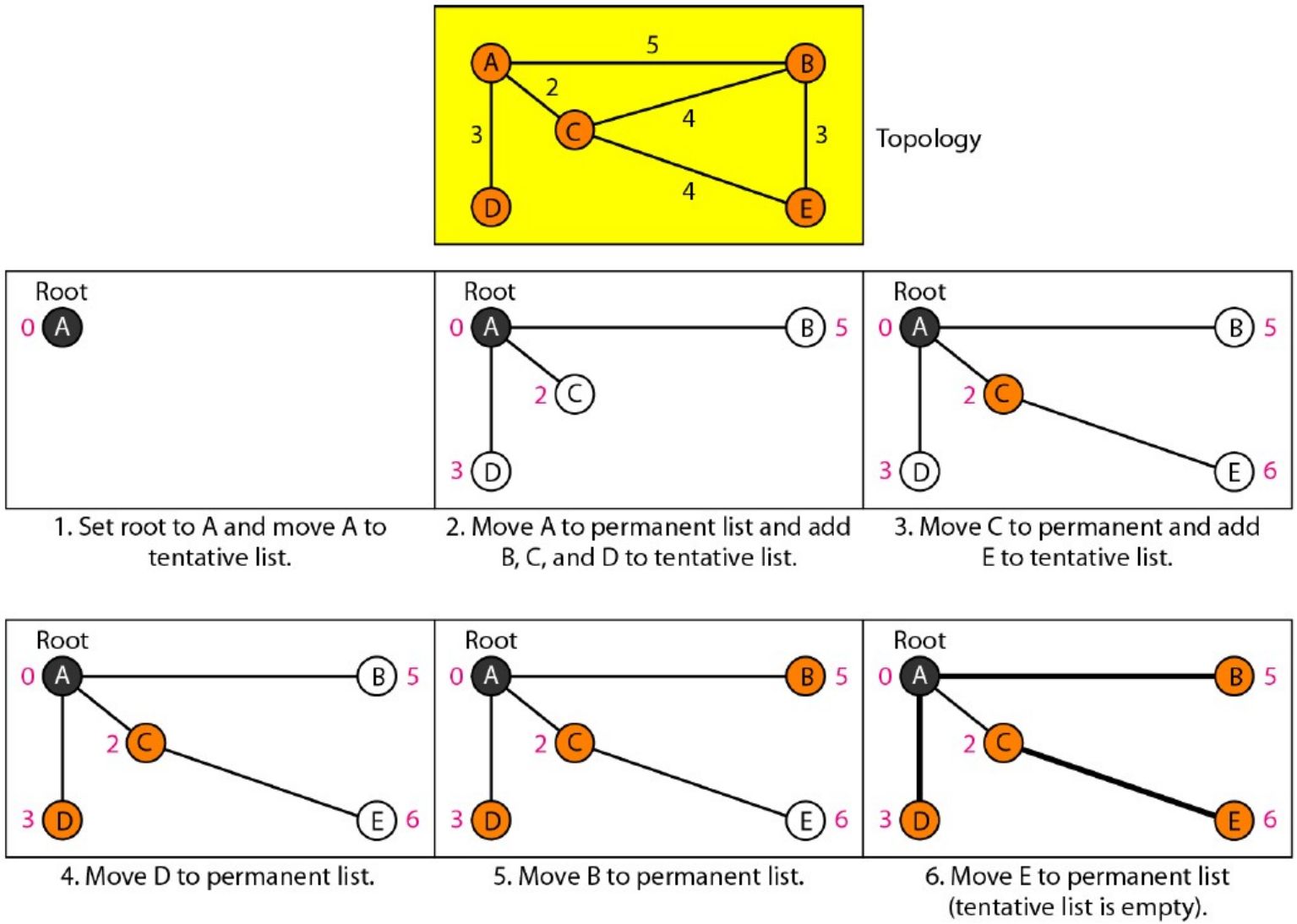


Figure 17 : Example of formation of shortest path tree

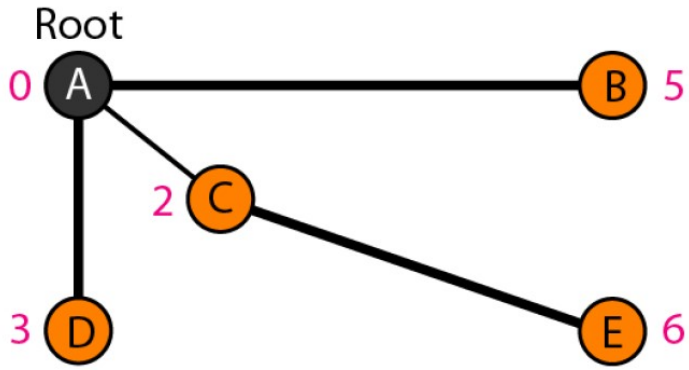


Figure 18 : Example of formation of shortest path tree

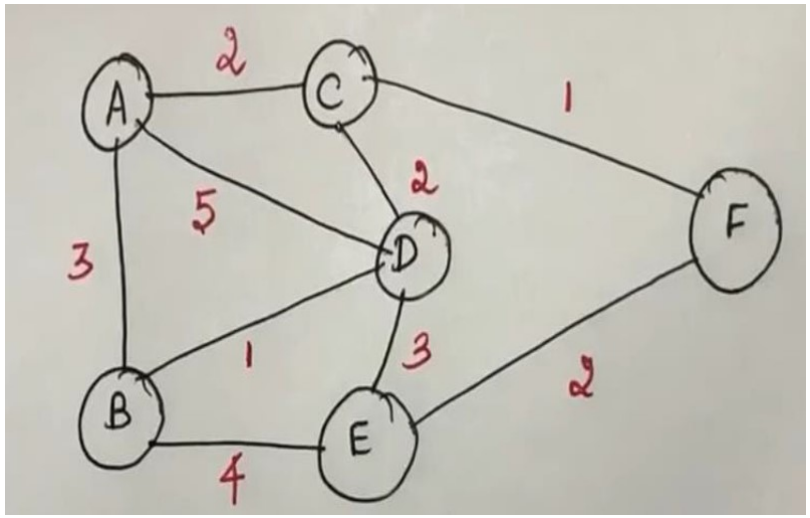
Node	*Cost*	*Next Router*
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

Figure 19 : Routing Table for Node A

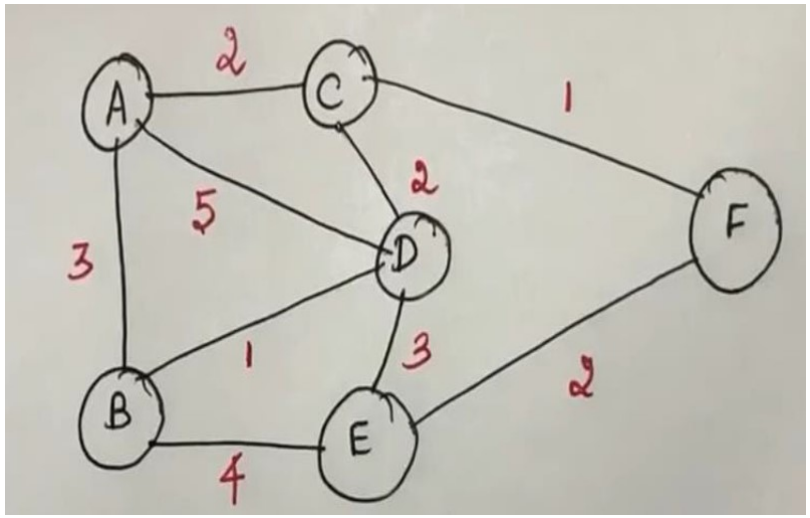
- Dijkstras algorithm:
  - net topology, link costs known to all nodes
    accomplished via link state broadcast; all nodes have same information
  - computes least cost paths from one node to all other nodes
    gives routing table for that node
  - after k iterations, know least cost path to k destination
- Notations;
  - $c(i, j)$: link cost from node i to j.
    cost is infinite if not direct neighbors
  - $D(v)$: current value of cost of path from source to destination 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

```
1   Dijkstra's Algorithm ( )
2   {
3       // Initialization
4       Tree = {root}                    // Tree is made only of the root
5       for (y = 1 to N)                 // N is the number of nodes
6       {
7           if (y is the root)
8               D[y] = 0                 // D[y] is shortest distance from root to node y
9           else if (y is a neighbor)
10              D[y] = c[root][y]        // c[x][y] is cost between nodes x and y in LSDB
11          else
12              D[y] = ∞
13      }
14      // Calculation
15      repeat
16      {
17          find a node w, with D[w] minimum among all nodes not in the Tree
18          Tree = Tree ∪ {w}            // Add w to tree
19          // Update distances for all neighbors of w
20          for (every node x, which is a neighbor of w and not in the Tree)
21          {
22              D[x] = min{D[x], (D[w] + c[w][x])}
23          }
24      } until (all nodes included in the Tree)
25  } // End of Dijkstra
```

- Example1: Create a routing table for the given graph using Dijkstra algorithm



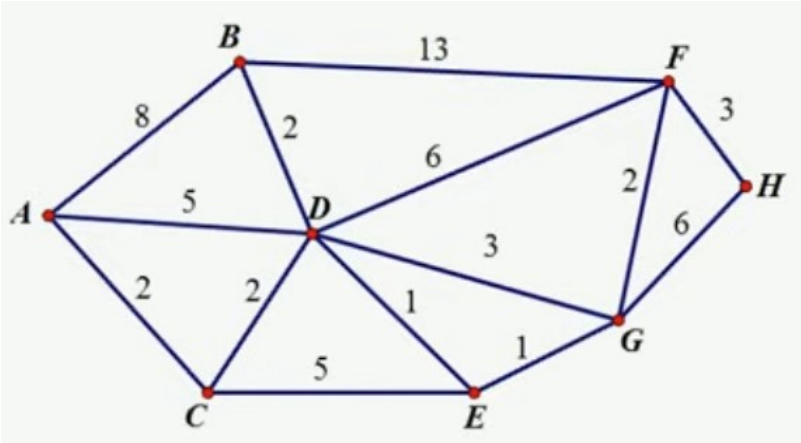
- Example1: Create a routing table for the given graph using Dijkstra algorithm



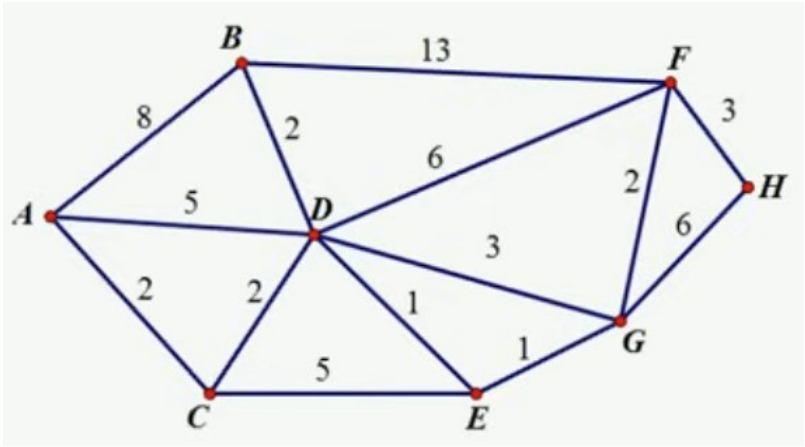
- Solution:

Iteration	True	B	C	D	E	F
Initial	{A}	3	(2)	5	∞	∞
1	{A, C}	(3)	-	4	∞	3
2	{A, B, C}	-	-	4	7	(3)
3	{A, B, C, F}	-	-	(4)	5	-
4	{A, B, C, D, F}	-	-	-	5	-
-	{A, B, C, D, E, F}	-	-	-	-	-

- Example2: Create a routing table for the given graph using Dijkstra algorithm



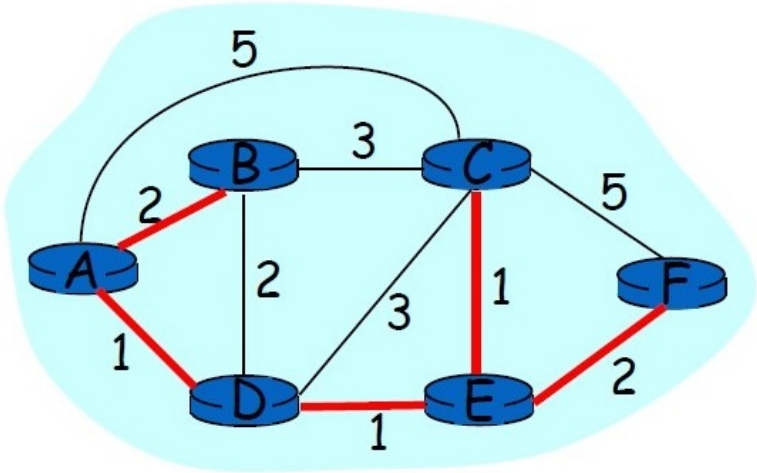
- Example2: Create a routing table for the given graph using Dijkstra algorithm



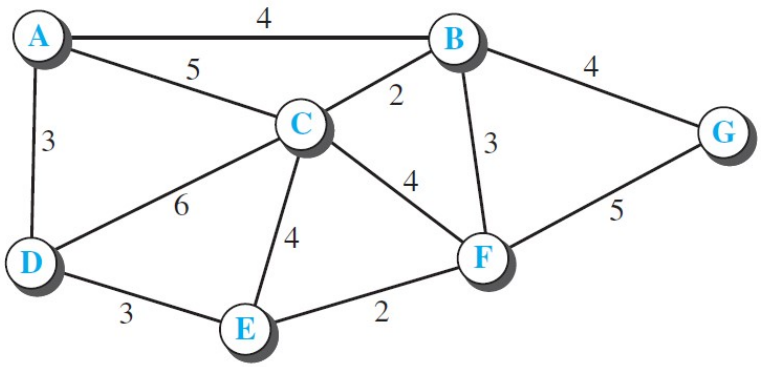
- Solution:

V	A	B	C	D	E	F	G	H
A	$\boxed{0_A}$	8_A	2_A	5_A	∞	∞	∞	∞
C		8_A	$\boxed{2_A}$	4_C	7_C	∞	∞	∞
D		6_D		$\boxed{4_C}$	5_D	10_D	7_D	∞
E		6_D			$\boxed{5_D}$	10_D	6_E	∞
B		$\boxed{6_D}$				10_D	6_E	∞
G						8_G	$\boxed{6_E}$	12_G
F						$\boxed{8_G}$		11_F
H								$\boxed{11_F}$

- Example3: Create a routing table for the given graph using Dijkstra algorithm



- Example4: Create a routing table for the given graph using Dijkstra algorithm



Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF)

- Open Shortest Path First (OSPF) is also an intradomain routing protocol like RIP, but it is based on the link-state routing protocol.
- In OSPF, like RIP, the cost of reaching a destination from the host is calculated from the source router to the destination network.
- However, each link (network) can be assigned a weight based on the throughput, round-trip time, reliability, and so on.

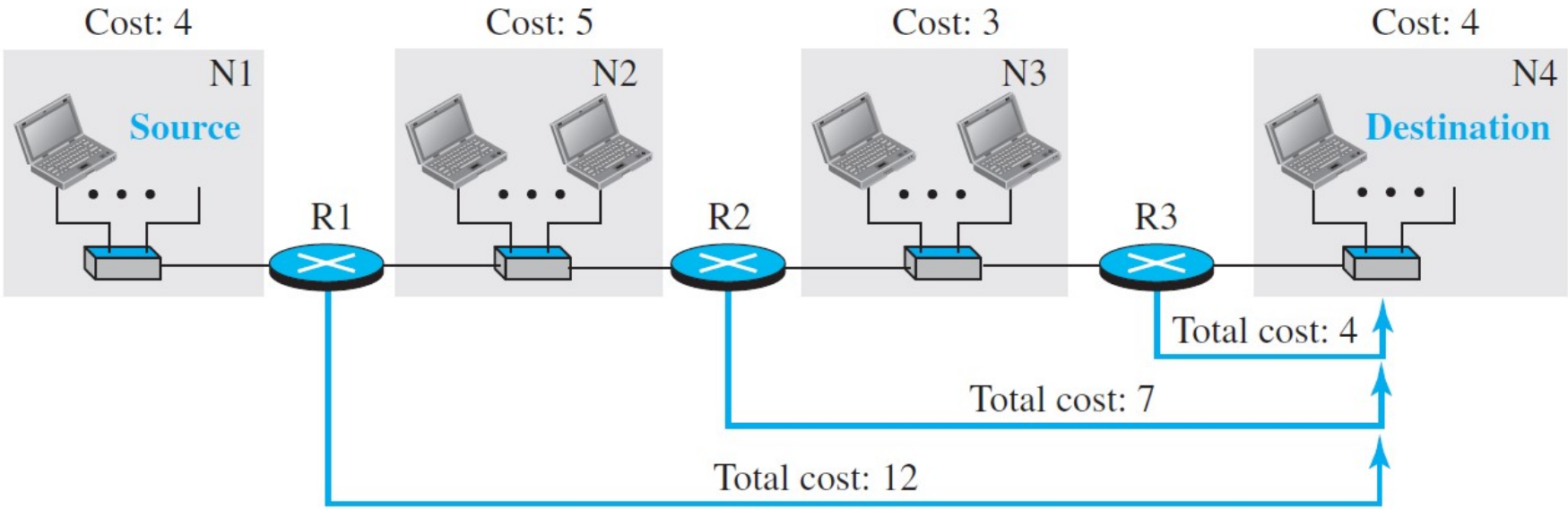


Figure 20 : Metric in OSPF

- Forwarding Tables: Each OSPF router can create a forwarding table after finding the shortest-path tree between itself and the destination using Dijkstra's algorithm.

Forwarding table for R1

Destination network	Next router	Cost
N1	——	4
N2	——	5
N3	R2	8
N4	R2	12

Forwarding table for R2

Destination network	Next router	Cost
N1	R1	9
N2	——	5
N3	——	3
N4	R3	7

Forwarding table for R3

Destination network	Next router	Cost
N1	R2	12
N2	R2	8
N3	——	3
N4	——	4

Figure 21 : Forwarding Tables in OSPF

- However, each router in an area needs to know the information about the link states not only in its area but also in other areas.
- For this reason, one of the areas in the AS is designated as the backbone area, responsible for gluing the areas together.
- The routers in the backbone area are responsible for passing the information collected by each area to all other areas.

Open Shortest Path First (OSPF)

- In this way, a router in an area can receive all LSPs generated in other areas.
- For the purpose of communication, each area has an area identification.
- The areaidentification of the backbone is zero.
- Figure shows an autonomous system and its areas.

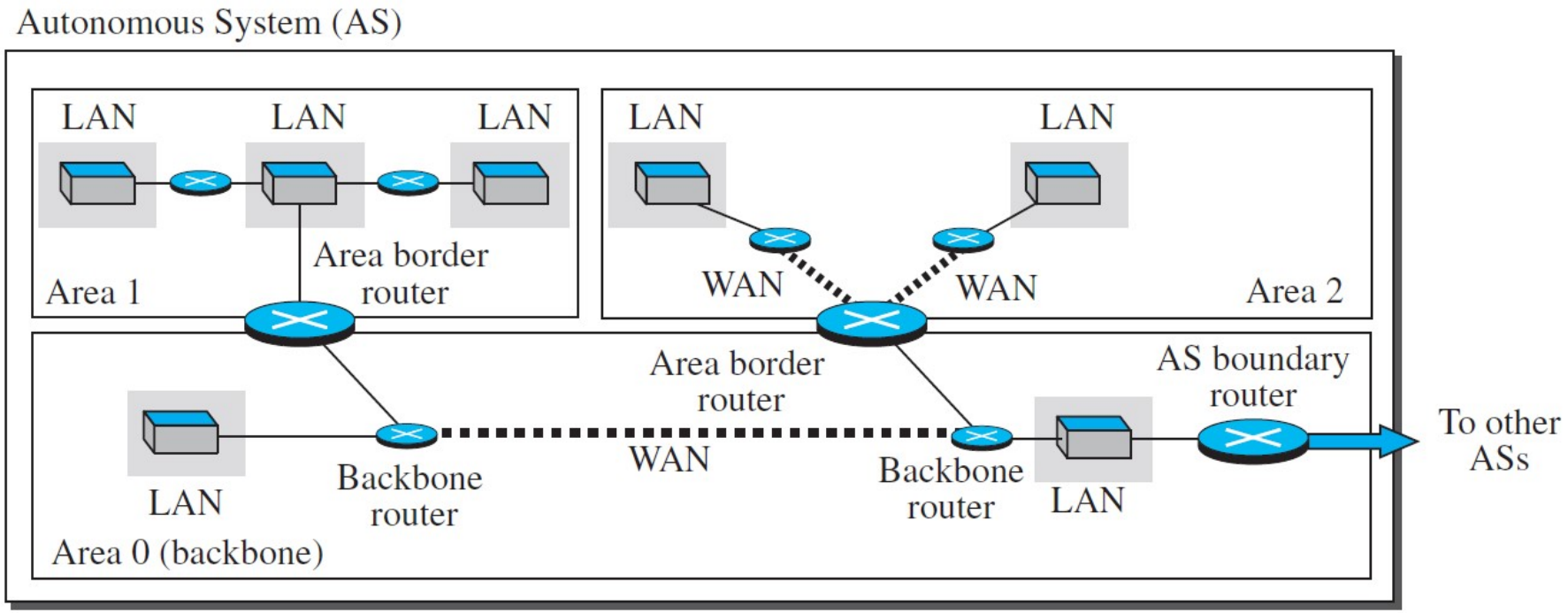


Figure 22 : Areas in an autonomous system

Open Shortest Path First (OSPF)

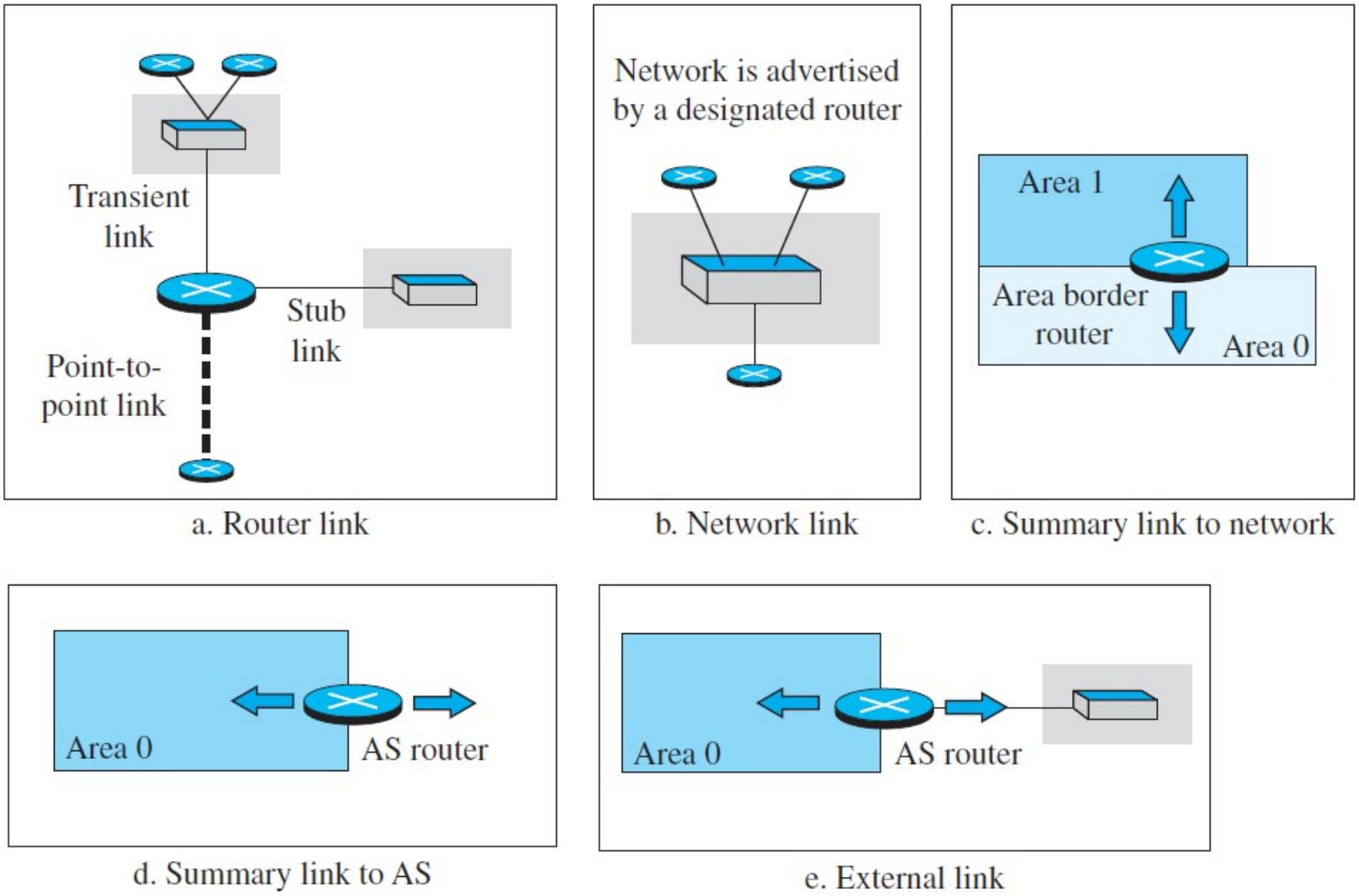


Figure 23 : Five different LSP's

- Router link: A router link advertises the existence of a router as a node.
- In addition to giving the address of the announcing router, this type of advertisement can define one or more types of links that connect the advertising router to other entities.
- A transient link announces a link to a transient network, a network that is connected to the rest of the networks by one or more routers.
- A stub link advertises a link to a stub network, a network that is not a through network.
- A point-to-point link should define the address of the router at the end of the point-to-point line and the cost to get there

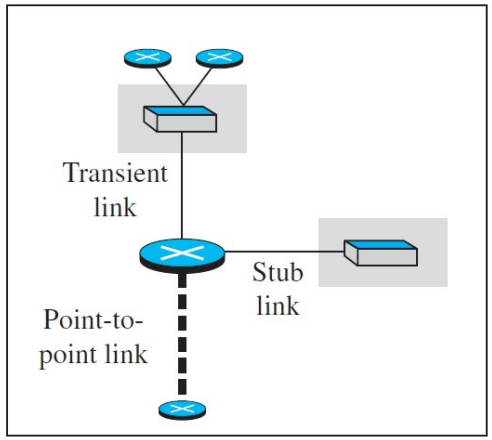


Figure 24 : Router link

- Network link: A network link advertises the network as a node.
- However, since a network cannot do announcements itself (it is a passive entity), one of the routers is assigned as the designated router and does the advertising.
- n addition to the address of the designated router, this type of LSP announces the IP address of all routers.

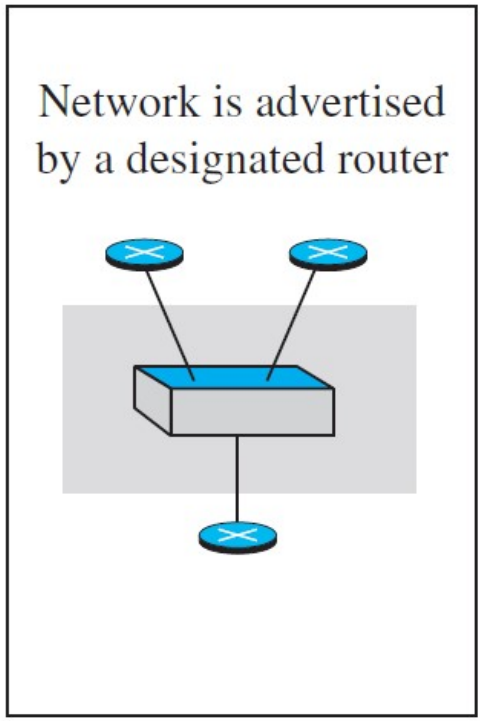


Figure 25 : Network link

- Summary link to network: This is done by an area border router; it advertises the summary of links collected by the backbone to an area or the summary of links collected by the area to the backbone.

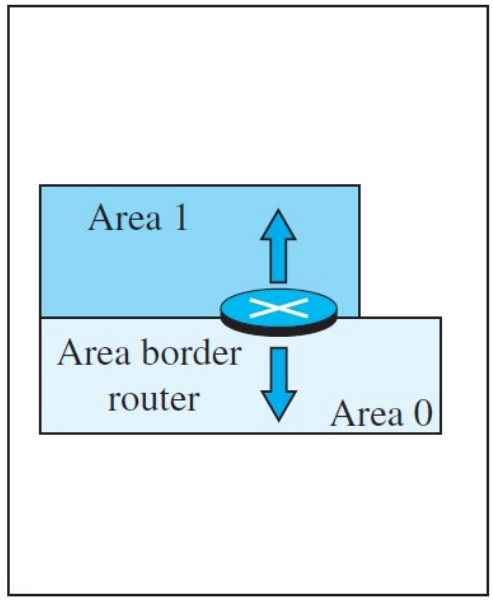


Figure 26 : Summary link to network

- Summary link to AS: This is done by an AS router that advertises the summary links from other ASs to the backbone area of the current AS, information which later can be disseminated to the areas so that they will know about the networks in other ASs.

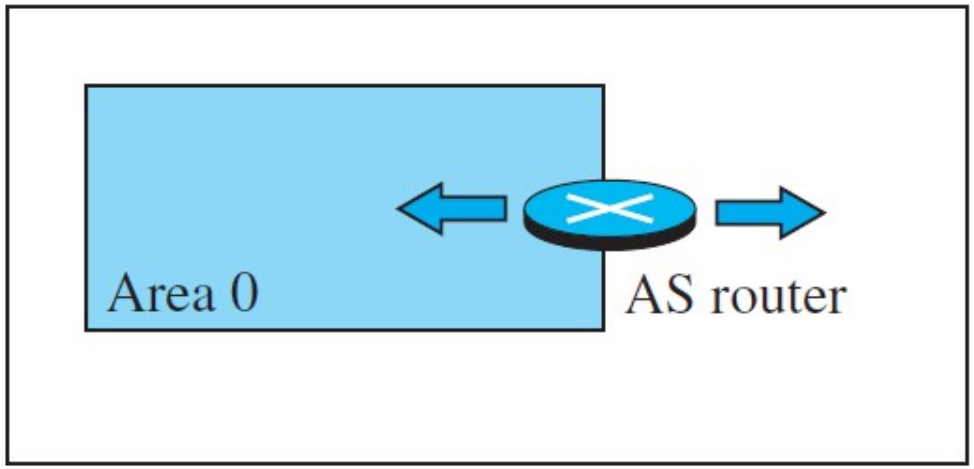


Figure 27 : Summary link to AS

- External link: This is also done by an AS router to announce the existence of a single network outside the AS to the backbone area to be disseminated into the areas.

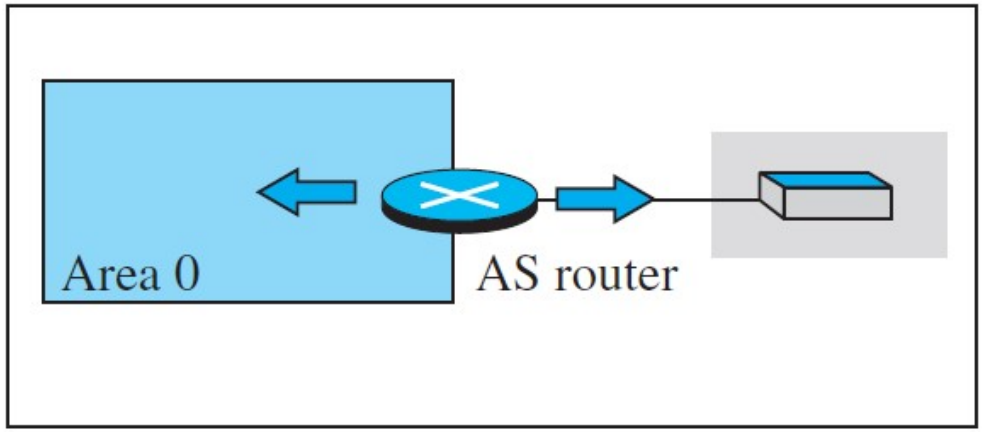


Figure 28 : External link

Thank you

&

Queries