CSS2C08

COMPUTER NETWORKS

MODULE 3

1. Network layer services
2. Routing
3. IP
4. Routing in internet
5. Router
6. IPV6
7. Multicast routing
8. Mobility

Routing In Internet(Routing)

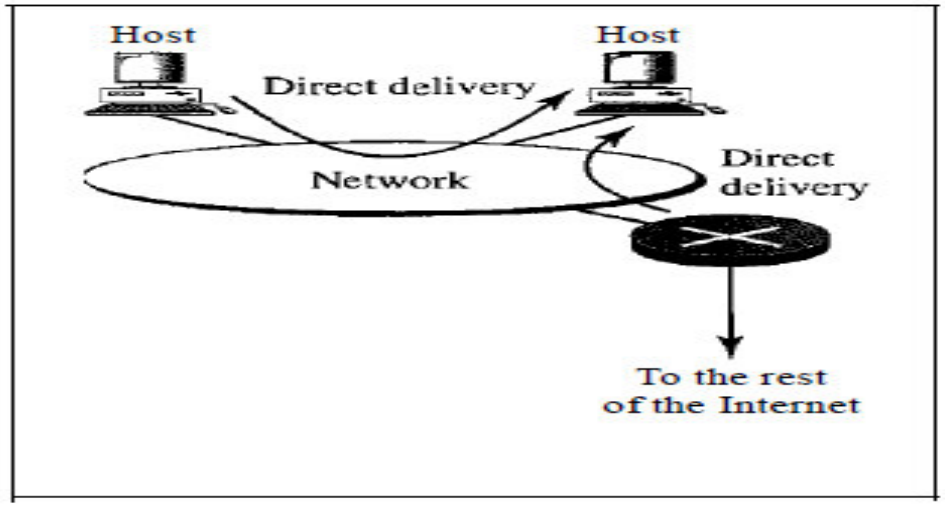
1. Delivery
2. Forwarding
3. Routing
4. Routing protocols

DELIVERY

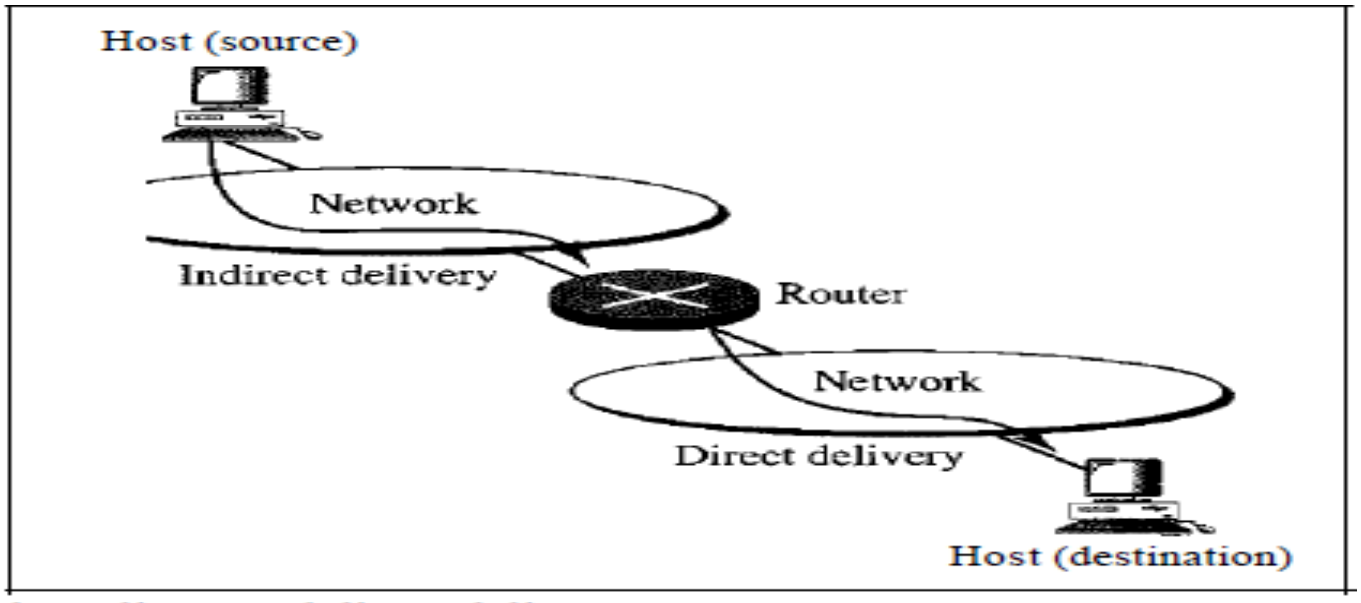
- Delivery refers to the way a packet is handled by the underlying networks under the control of the network layer.
- The network layer supervises the handling of the packets by the underlying physical networks.

➢ Different methods of delivery:

❖ **Direct Delivery:** The delivery of a packet is called direct if the deliverer (host or router) and the destination are on the same network.



❖ **Indirect Delivery:** the delivery of a packet is called indirect if the deliverer (host or router) and the destination are on different networks.



FORWARDING

- Forwarding refers to the way a packet is delivered to the next station.
- Forwarding means to place the packet in its route to its destination.
- Forwarding requires a host or a router to have a routing table.
- When a host has a packet to send or when a router has received a packet to be forwarded, it looks at this table to find the route to the final destination.

- **Forwarding Techniques:**

1. Next-Hop Method Versus Route Method
2. Network-Specific Method Versus Host-Specific Method
3. Default Method

1. Next-Hop Method Versus Route Method:

One technique to reduce the contents of a routing table is called the next-hop method. In this technique, the routing table holds only the address of the next hop instead of information about the complete route (route method). The entries of a routing table must be consistent with one another.

a. Routing tables based on route

Destination	Route
Host B	R1, R2, host B

Routing table for host A

Destination	Route
Host B	R2, host B

Routing table for R1

Destination	Route
Host B	Host B

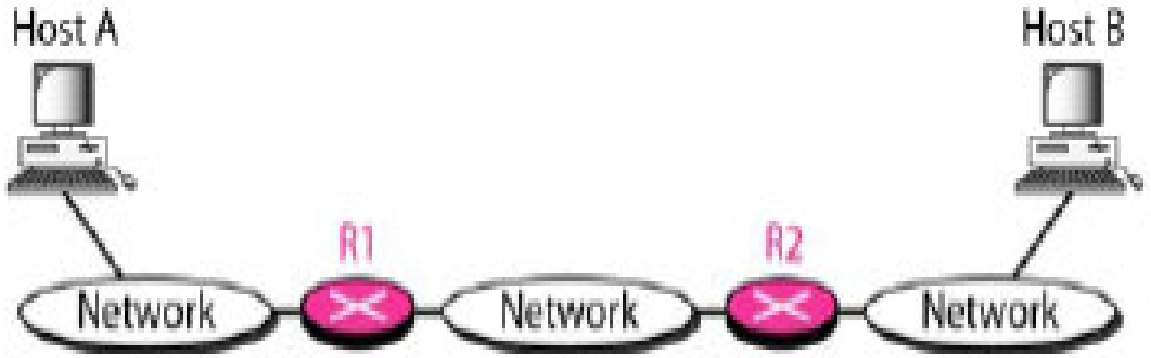
Routing table for R2

b. Routing tables based on next hop

Destination	Next hop
Host B	R1

Destination	Next hop
Host B	R2

Destination	Next hop
Host B	---



2. Network-Specific Method Versus Host-Specific Method

A second technique to reduce the routing table and simplify the searching process is called the network-specific method. Here, instead of having an entry for every destination host connected to the same physical network (host-specific method), we have only one entry that defines the address of the destination network itself.

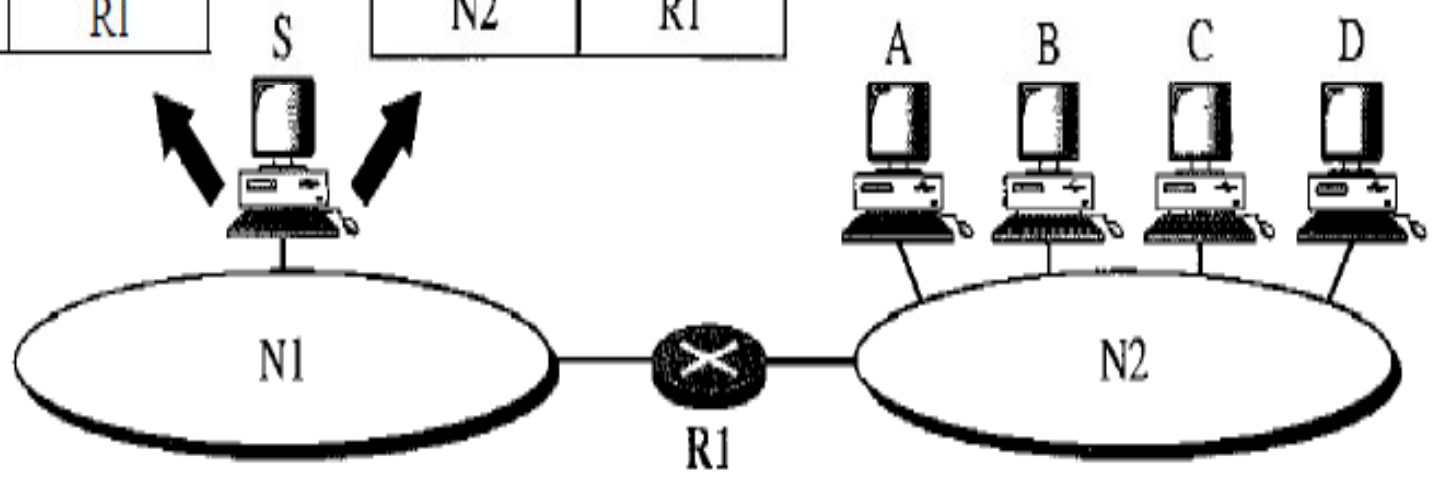
Host-specific routing is used for purposes such as checking the route or providing security measures.

Routing table for host S based on host-specific method

Destination	Next hop
A	R1
B	R1
C	R1
D	R1

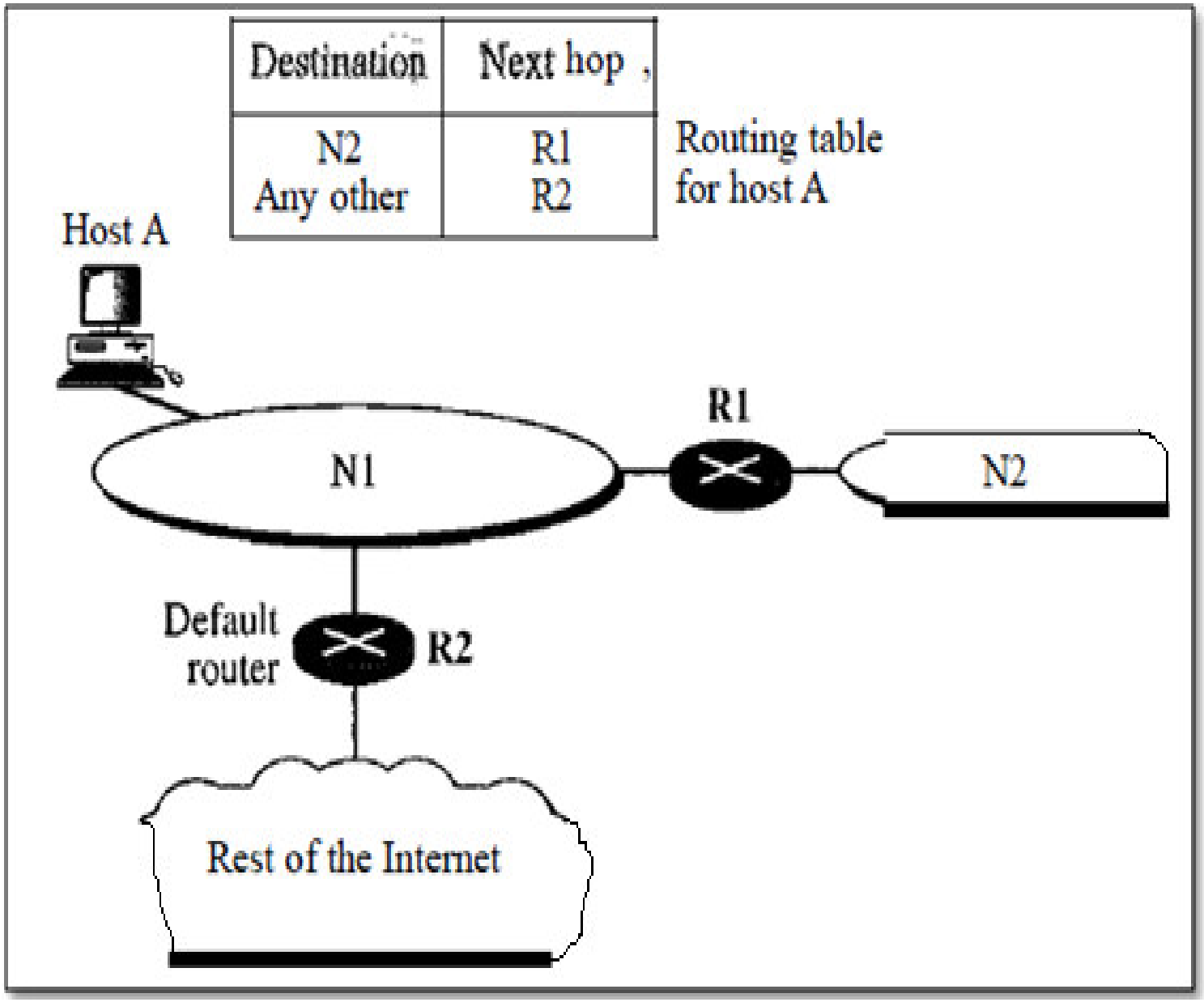
Routing table for host S based on network-specific method

Destination	Next hop
N2	R1



3. Default Method

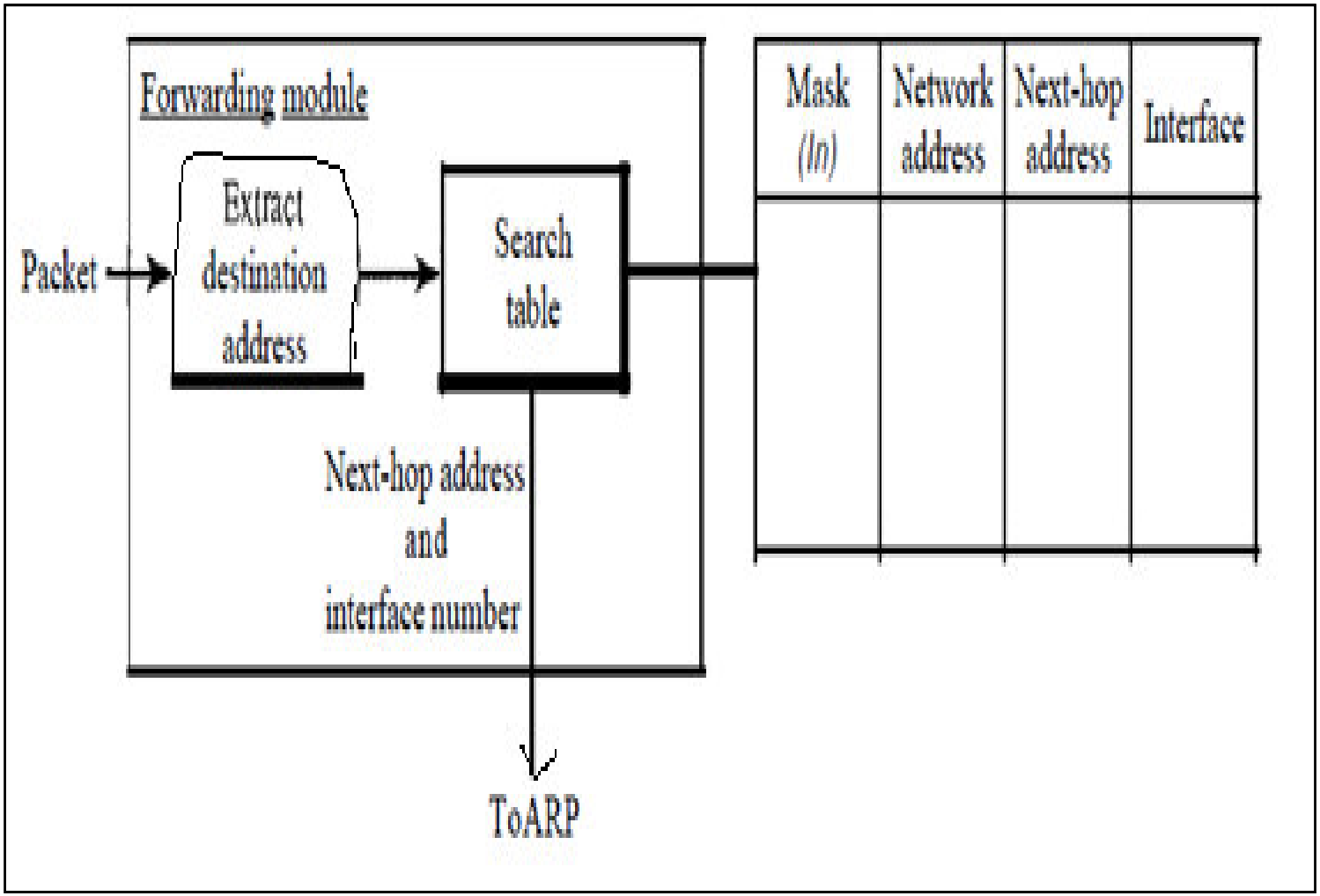
Another technique to simplify routing is called the default method. Host A is connected to a network with two routers. Router R1 routes the packets to hosts connected to network N2. However, for the rest of the Internet, router R2 is used. So instead of listing all networks in the entire Internet, host A can just have one entry called the default (normally defined as network address 0.0.0.0).


Destination
Next hop
N2
R1
Any other
R2
Routing table
for host A
Host A
R1
N1
N2
Default
router
R2
Rest of the Internet

- **Forwarding Process:**

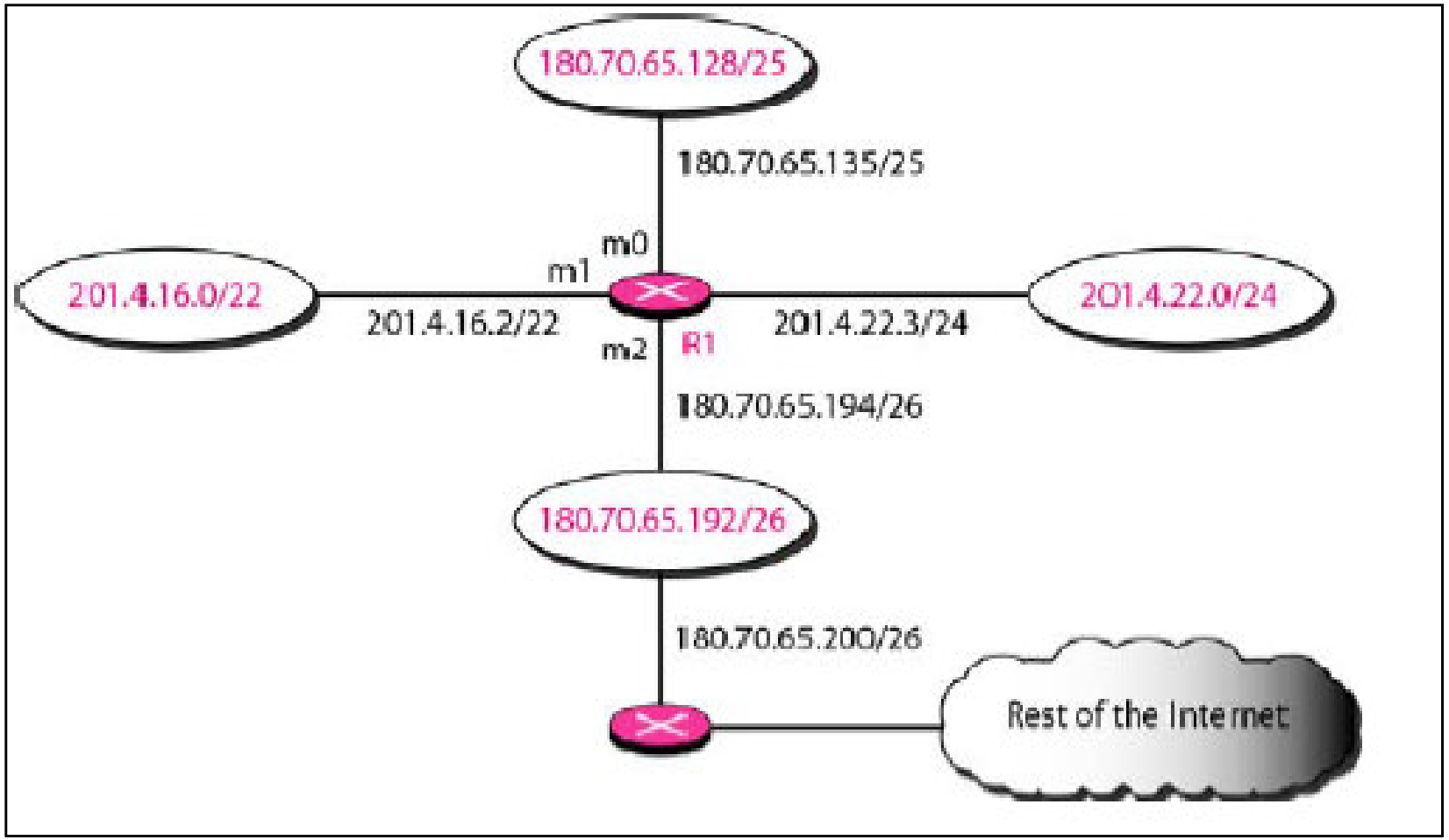
  We assume that hosts and routers use classless addressing .In classless addressing, the routing table needs to have one row of information for each block involved. The table needs to be searched based on the network address (first address in the block). Unfortunately, the destination address in the packet gives no clue about the network address. To solve the problem, we need to include the mask *(In)* in the table; we need to have an extra column that includes the mask for the corresponding block.

  In classless addressing, we need at least four columns in a routing table.


Forwarding module
Packet
Extract destination address
Search table
Next-hop address and interface number
ToARP
Mask (/n)
Network address
Next-hop address
Interface

➢ Example:

Make a routing table for router R1, using the configuration in Figure below:



Routing table for router R1:

Mask	*Network Address*	*Next Hop*	*Interface*
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

Case 1:

Show the forwarding process if a packet arrives at R1with the destination address 180.70.65.140.

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 180.70.65.128, which does not match the corresponding network address.
2. The second mask (/25) is applied to the destination address. The result is 180.70.65.128, which matches the corresponding network address. The next-hop address (the destination address of the packet in this case) and the interface number m0 are passed to ARP for further processing.

Case 2:

Show the forwarding process if a packet arrives at R1 with the destination address201.4.22.35

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 1).
2. The second mask (/25) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 2).
3. The third mask (/24) is applied to the destination address. The result is 201.4.22.0, which matches the corresponding network address. The destination address of the packet and the interface number m3 are passed to ARP.

Case 3:

Show the forwarding process if a packet arrives at R1 with the destination address 18.24.32.78.

This time all masks are applied, one by one, to the destination address, but no matching network address is found. When it reaches the end of the table, the module gives the next-hop address 180.70.65.200 and interface number m2 to ARP. This is probably an outgoing package that needs to be sent, via the default router, to someplace else in the Internet.

ROUTING TABLE

A host or a router has a routing table with an entry for each destination, or a combination of destinations, to route IP packets. The routing table can be either:

1. Static or
2. Dynamic.

1. Static Routing Table:

A static routing table contains information entered manually. The administrator enters the route for each destination into the table. When a table is created, it cannot update automatically when there is a change in the Internet. The table must be manually altered by the administrator.

2. Dynamic Routing Table;

A dynamic routing table is updated periodically by using one of the dynamic routing protocols such as RIP, OSPF, or BGP. Whenever there is a change in the Internet, such as a shutdown of a router or breaking of a link, the dynamic routing protocols update all the tables in the routers (and eventually in the host) automatically.

The routers in a big internet such as the Internet need to be updated dynamically for efficient delivery of the IP packets.

- **Common fields in a routing table:**

Mask	Network address	Next-hop address	Interface	Flags	Reference count	Use

- **Mask**: This field defines the mask applied for the entry.
- **Network address**: This field defines the network address to which the packet is finally delivered. In the case of host-specific routing, this field defines the address of the destination host.

- **Next-hop address**: This field defines the address of the next-hop router to which the packet is delivered.
- **Interface**: This field shows the name of the interface.
- **Flags:** This field defines up to five flags. Flags are on/off switches that signify either presence or absence. The five flags are U (up), G (gateway), H (host-specific), D (added by redirection), and M (modified by redirection).
    - ❖**U (up).** The U flag indicates the router is up and running. If this flag is not present, it means that the router is down. The packet cannot be forwarded and is discarded.

- **G (gateway).** The G flag means that the destination is in another network. The packet is delivered to the next-hop router for delivery (indirect delivery). When this flag is missing, it means the destination is in this network (direct delivery).
- **H (host-specific).** The H flag indicates that the entry in the network address field is a host-specific address. When it is missing, it means that the address is only the network address of the destination.
- **D (added by redirection).** The D flag indicates that routing information for this destination has been added to the host routing table by a redirection message from ICMP.
- **M (modified by redirection).** The M flag indicates that the routing information for this destination has been modified by a redirection message from ICMP.

- **Reference count:** This field gives the number of users of this route at the moment. For example, if five people at the same time are connecting to the same host from this router, the value of this column is 5.
- **Use:** This field shows the number of packets transmitted through this router for the corresponding destination.

ROUTING PROTOCOLS(unicast)

- Routing protocols are used to continuously update the routing tables that are consulted for forwarding and routing.
- A routing protocol is a combination of rules and a procedure that lets routers in the internet inform each other of changes. It allows routers to share whatever they know about the internet or their neighborhood.
- The routing protocols also include procedures for combining information received from other routers.

- **Optimization:**
  - A router receives a packet from a network and passes it to another network. A router is usually attached to several networks. When it receives a packet, to which network should it pass the packet? The decision is based on optimization: Which of the available pathways is the optimum pathway?

- One approach is to assign a cost for passing through a network. We call this cost a metric. However, the metric assigned to each network depends on the type of protocol. Some simple protocols, such as the Routing Information Protocol (RIP), treat all networks as equals. The cost of passing through a network is the same; it is one hop count. So if a packet passes through 10 networks to reach the destination, the total cost is 10 hop counts.

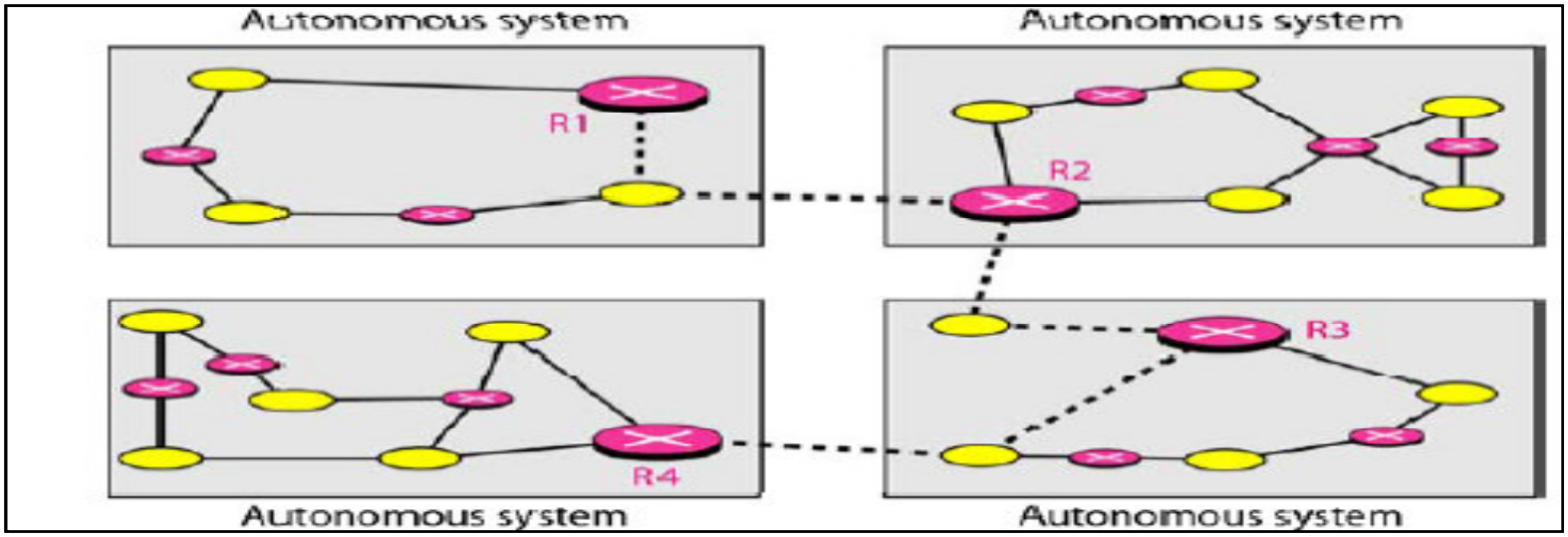
- Other protocols, such as Open Shortest Path First (OSPF), allow the administrator to assign a cost for passing through a network based on the type of service required. A route through a network can have different costs (metrics). For example, if maximum throughput is the desired type of service, a satellite link has a lower metric than a fiber-optic line. Routers use routing tables to help decide the best route. OSPF protocol allows each router to have several routing tables based on the required type of service.

- In the Border Gateway Protocol (BGP), the criterion is the policy, which can be set by the administrator. The policy defines what paths should be chosen.

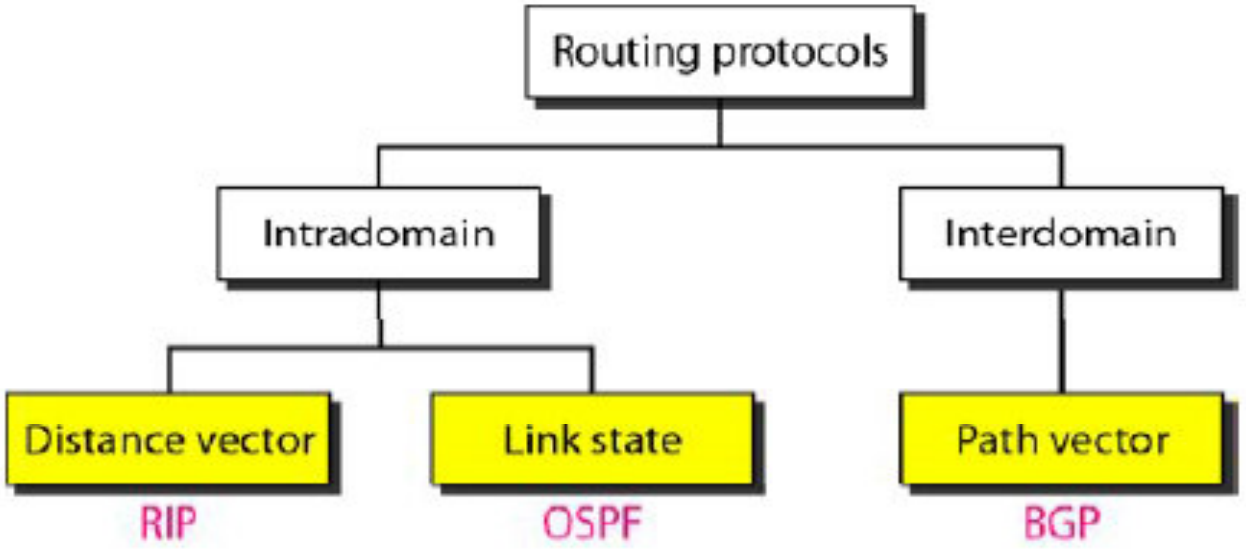
➢ Intra- and Interdomain Routing:

- ❖ An internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers. For this reason, an internet is divided into autonomous systems.
- ❖ An autonomous system (AS) is a group of networks and routers under the authority of a single administration.
- ❖ Routing inside an autonomous system is referred to as intradomain routing.
- ❖ Routing between autonomous systems is referred to as interdomain routing

- Several intradomain and interdomain routing protocols are in use:
  - Two intradomain routing protocols: Distance vector and link state.
  - One interdomain routing protocol: path vector.



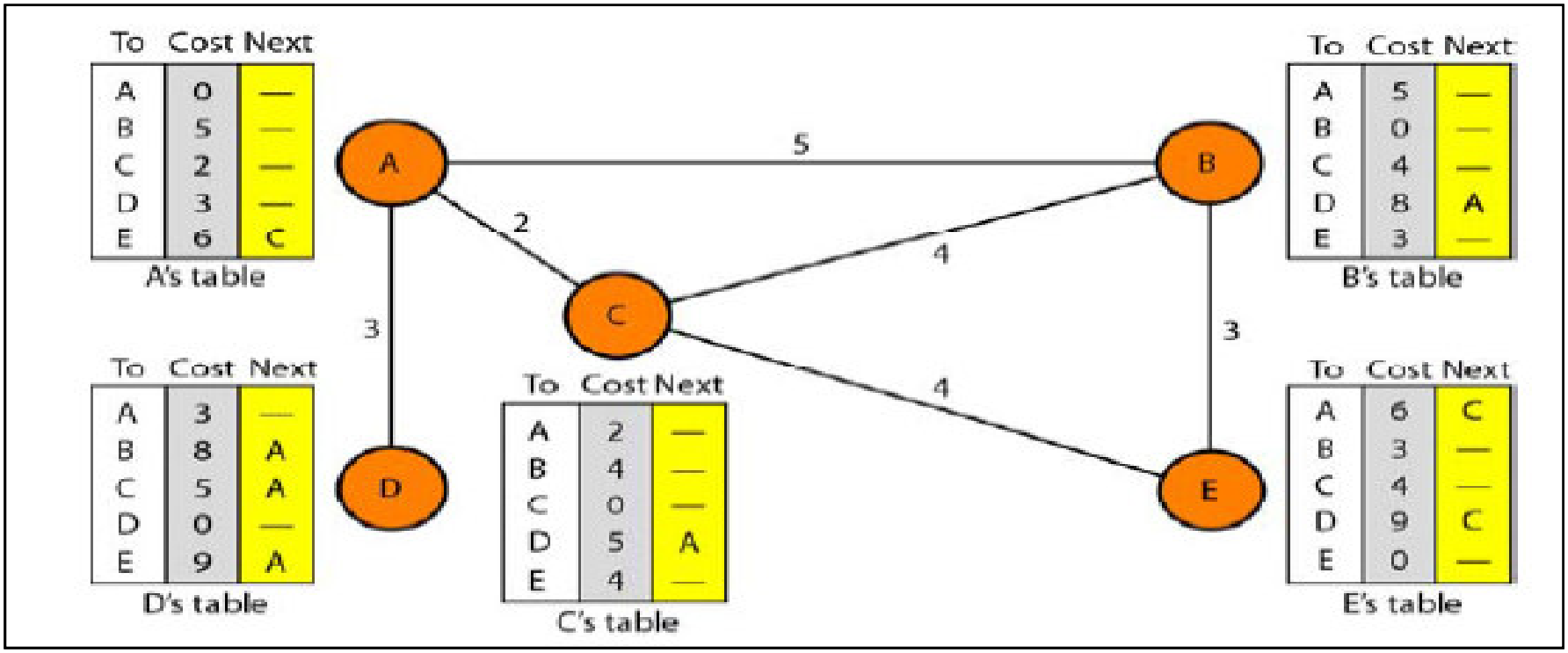
- **The routing protocols are:**
  - Routing Information Protocol (RIP) is an implementation of the distance vector protocol.
  - Open Shortest Path First (OSPF) is an implementation of the link state protocol.
  - Border Gateway Protocol (BGP) is an implementation of the path vector protocol.



1. Distance Vector Routing:

- In distance vector routing, the least-cost route between any two nodes is the route with minimum distance.
- In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node.
- The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).
- We can think of nodes as the cities in an area and the lines as the roads connecting them. A table can show a tourist the minimum distance between cities.

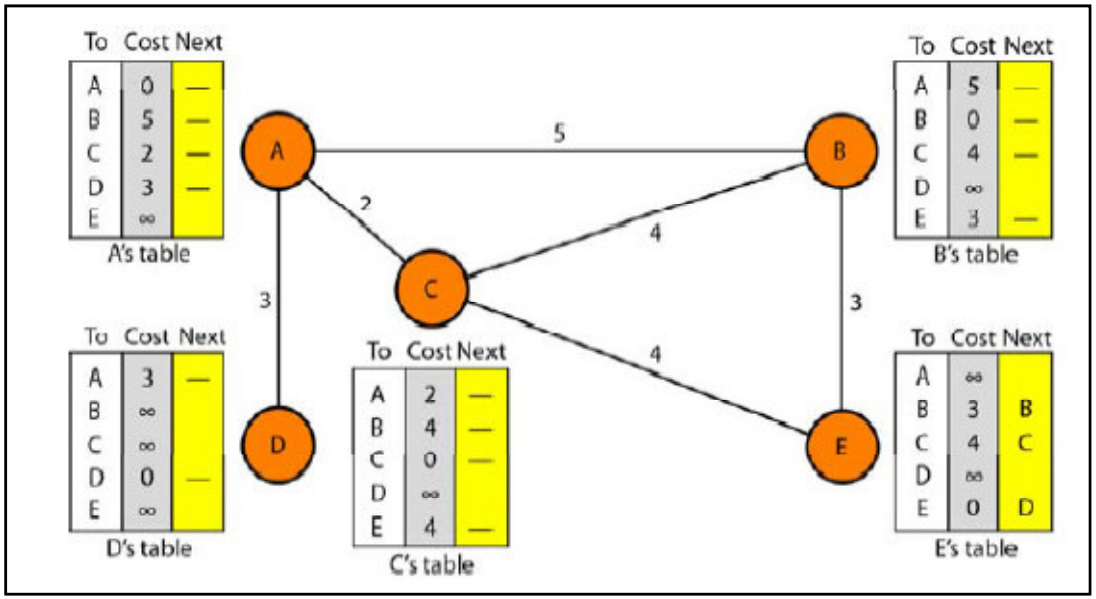
We show a system of five nodes with their corresponding tables.



The table for node A shows how we can reach any node from this node. For example, our least cost to reach node E is 6. The route passes through C.

INITIALIZATION :

Each node can know only the distance between itself and its immediate neighbors, those directly connected to it. So for the moment, we assume that each node can send a message to the immediate neighbors and find the distance between itself and these neighbors. The distance for any entry that is not a neighbor is marked as infinite (unreachable).



SHARING:

The whole idea of distance vector routing is the sharing of information between neighbors. Although node A does not know about node E, node C does. So if node C shares its routing table with A, node A can also know how to reach node E. On the other hand, node C does not know how to reach node D, but node A does. If node A shares its routing table with node C, node C also knows how to reach node D. In other words, nodes A and C, as immediate neighbors, can improve their routing tables if they help each other.

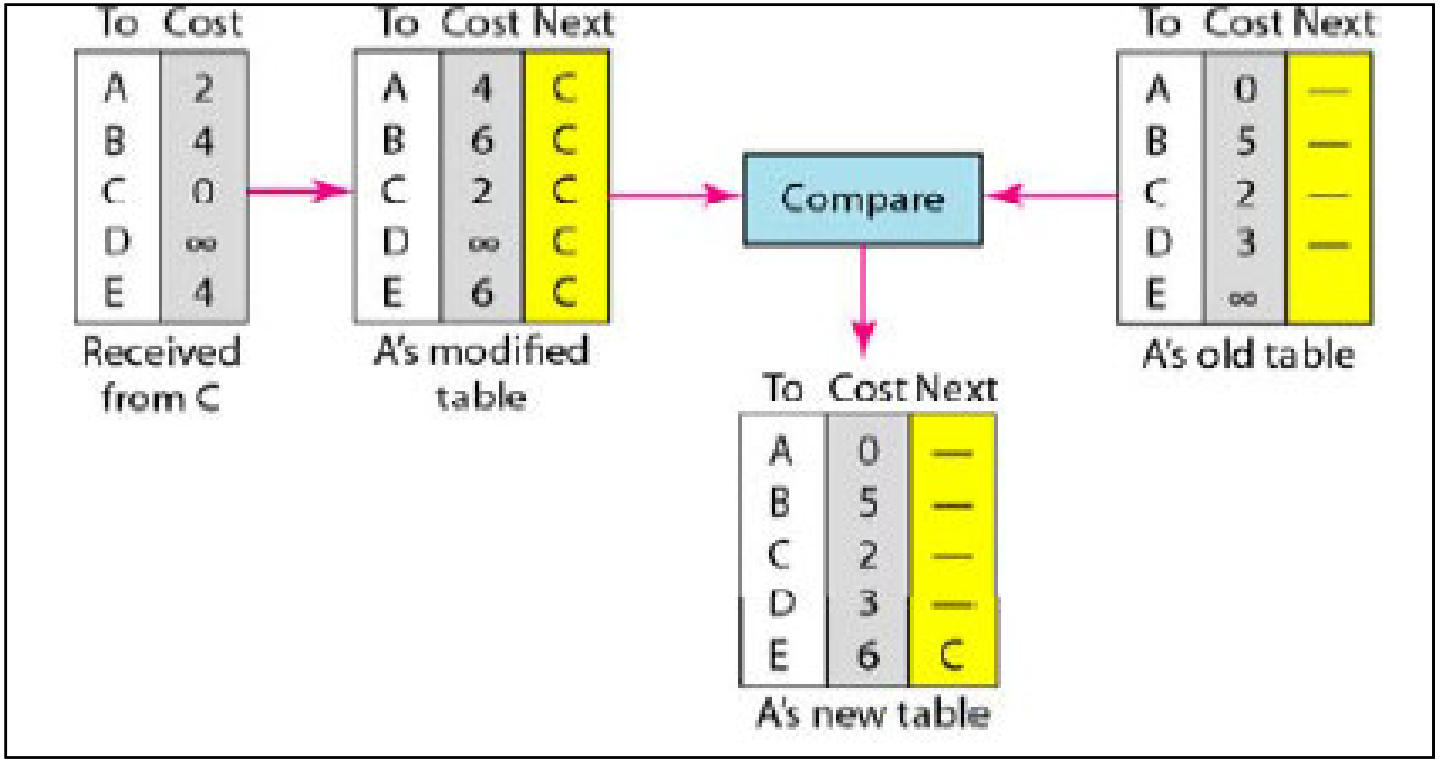
Each node is to send its entire table to the neighbor and let the neighbor decide what part to use and what part to discard. However, the third column of a table (next stop) is not useful for the neighbor. When the neighbor receives a table, this column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table. A node therefore can send only the first two columns of its table to any neighbor. In other words, sharing here means sharing only the first two columns.

UPDATING:

When a node receives a two-column table from a neighbor, it needs to update its routing table. Updating takes three steps:

1. The receiving node needs to add the cost between itself and the sending node to each value in the second column. The logic is clear. If node C claims that its distance to a destination is x mi, and the distance between A and C is y mi, then the distance between A and that destination, via C, is $x + y$ mi.
2. The receiving node needs to add the name of the sending node to each row as the third column if the receiving node uses information from any row. The sending node is the next node in the route.

3. The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table.
    a) If the next-node entry is different, the receiving node chooses the row with the smaller cost. If there is a tie, the old one is kept.
    b) If the next-node entry is the same, the receiving node chooses the new row. For example, suppose node C has previously advertised a route to node X with distance 3.



Each node can update its table by using the tables received from other nodes. In a short time, if there is no change in the network itself, such as a failure in a link, each node reaches a stable condition in which the contents of its table remains the same.

- **When to Share**

  The table is sent both periodically and when there is a change in the table.

  - **Periodic Update**: A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing.
  - **Triggered Update:** A node sends its two-column routing table to its neighbors anytime there is a change in its routing table. This is called a triggered update.

    The change can result from the following.

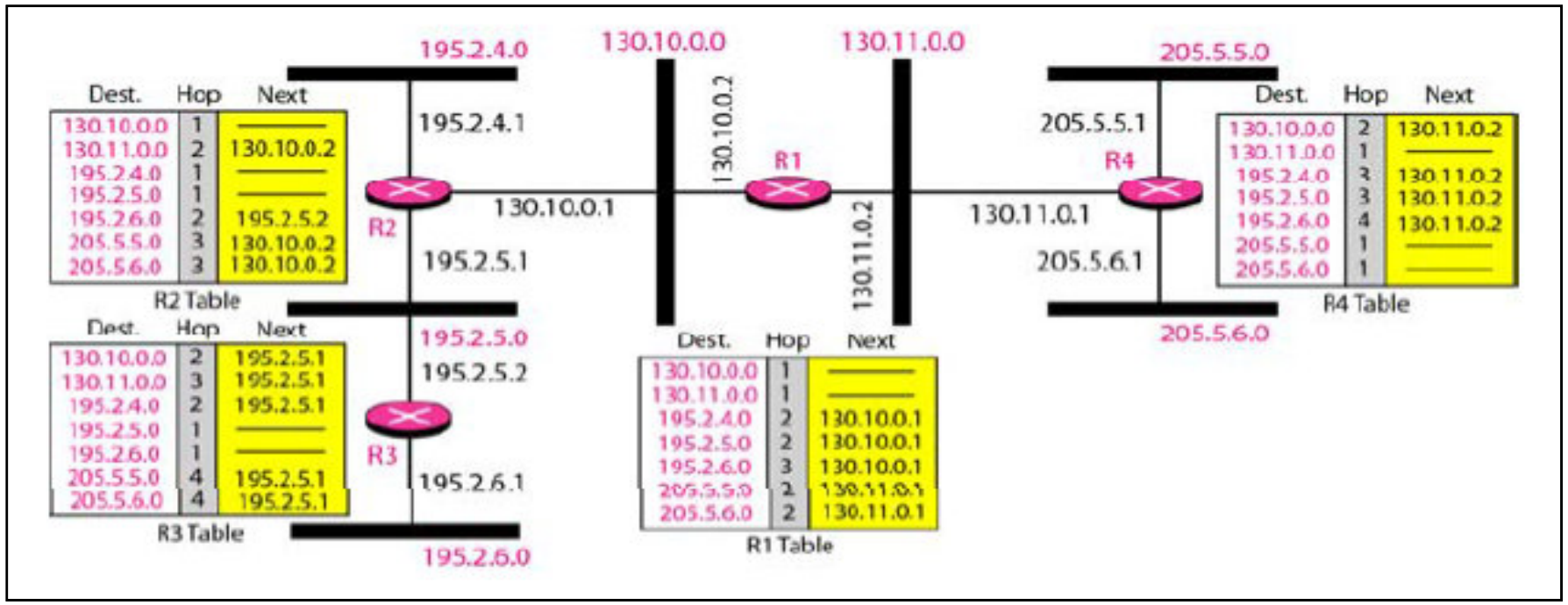
    - node receives a table from a neighbor, resulting in changes in its own table after updating.
    - A node detects some failure in the neighboring links which results in a distance change to infinity.

- **Problem of distance vector routing:**

  A problem with distance vector routing is instability, which means that a network using this protocol can become unstable.

RIP(Routing Information Protocol)

- The Routing Information Protocol (RIP) is an intradomain routing protocol used inside an autonomous system. It is a very simple protocol based on distance vector routing.
- RIP implements distance vector routing directly with some considerations:
  1. In an autonomous system, we are dealing with routers and networks (links). The routers have routing tables; networks do not.
  2. The destination in a routing table is a network, which means the first column defines a network address.
  3. The metric used by RIP is very simple; the distance is defined as the number of links (networks) to reach the destination. For this reason, the metric in RIP is called a hop count.
  4. Infinity is defined as 16, which means that any route in an autonomous system using RIP cannot have more than 15 hops.
  5. The next-node column defines the address of the router to which the packet is to be sent to reach its destination.

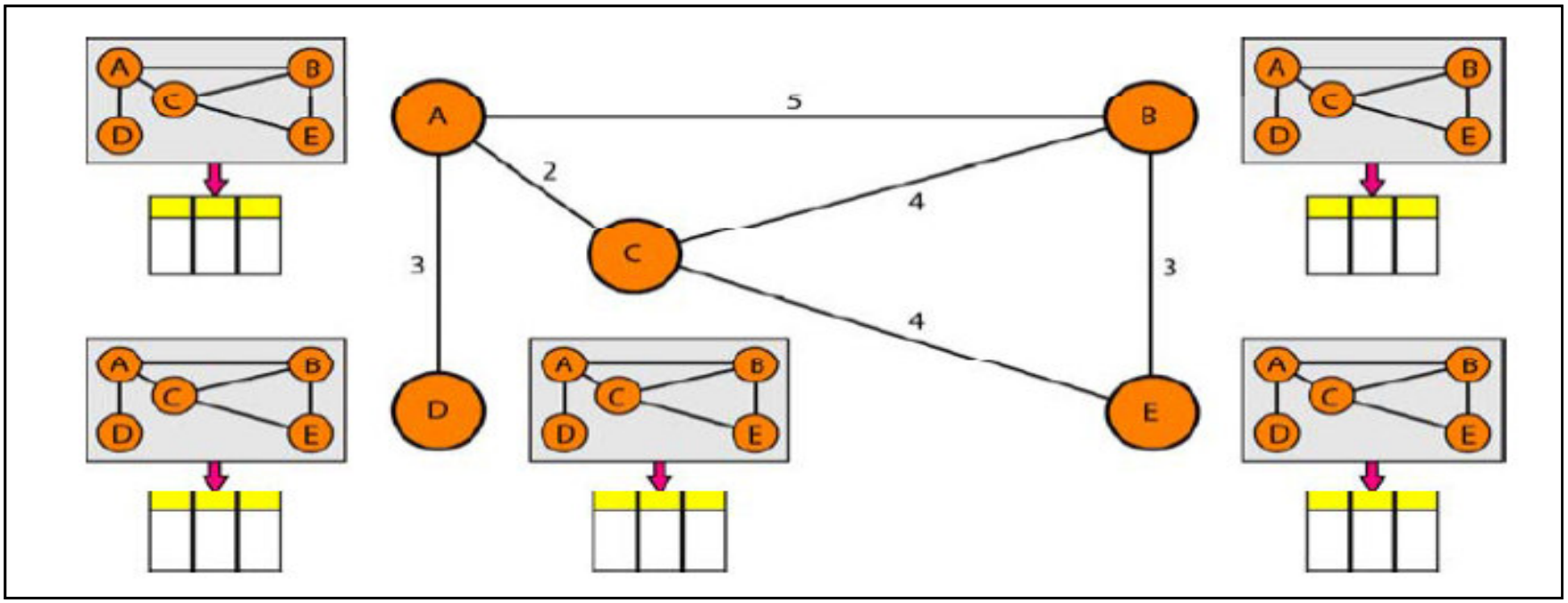


The above figure shows an autonomous system with seven networks and four routers. The table of each router is also shown. Let us look at the routing table for R1. The table has seven entries to show how to reach each network in the autonomous system. Router R1 is directly connected to networks 130.10.0.0 and 130.11.0.0, which means that there are no next-hop entries for these two networks. To send a packet to one of the three networks at the far left, router R1 needs to deliver the packet to R2. The next-node entry for these three networks is the interface of router R2 with IP address 130.10.0.1. To send a packet to the two networks at the far right, router R1 needs to send the packet to the interface of router R4 with IP address 130.11.0.1. The other tables can be explained similarly.

2. Link State Routing:

In link state routing, if each node in the domain has the entire topology of the domain the list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down)-the node can use Dijkstra's algorithm to build a routing table.

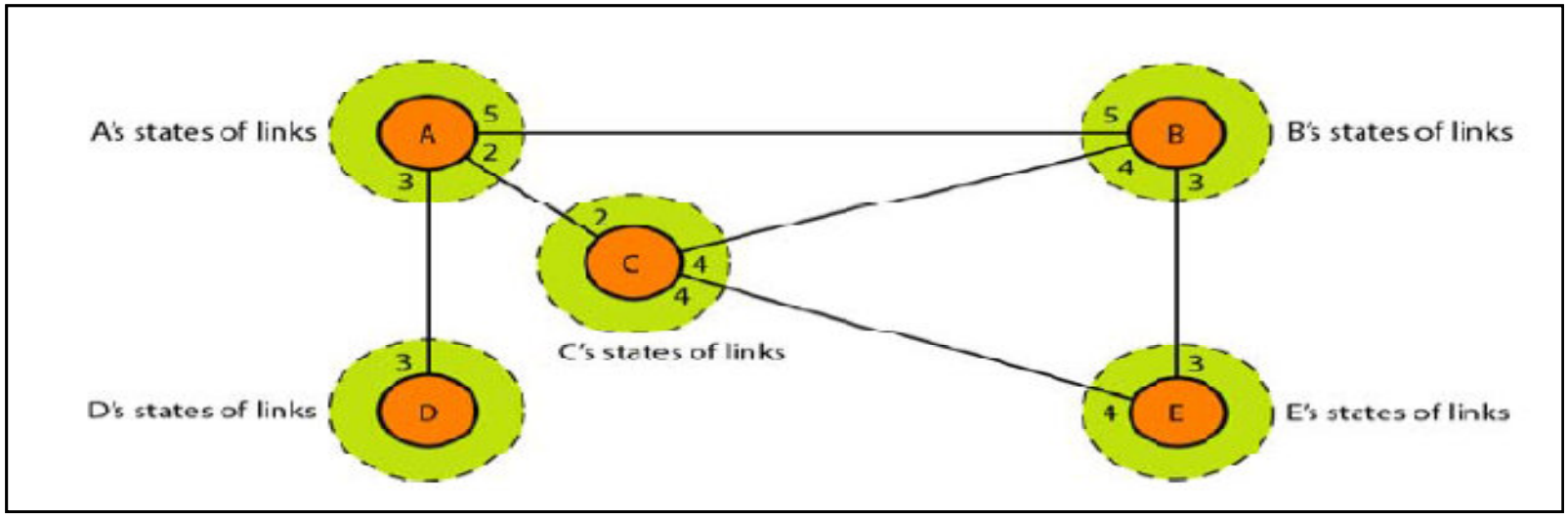
Concept of link state routing:



The figure shows a simple domain with five nodes. Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology. This is analogous to a city map. While each person may have the same map, each needs to take a different route to reach her specific destination

The topology must be dynamic, representing the latest state of each node and each link. If there are changes in any point in the network (a link is down), the topology must be updated for each node.

Link state routing is based on the assumption that, although the global knowledge about the topology is not clear, each node has partial knowledge: it knows the state (type, condition, and cost) of its links. In other words, the whole topology can be compiled from the partial knowledge of each node.



Node A knows that it is connected to node B with metric 5, to node C with metric 2, and to node D with metric 3. Node C knows that it is connected to node A with metric 2, to node B with metric 4, and to node E with metric 4. Node D knows that it is connected only to node A with metric 3. And so on. Although there is an overlap in the knowledge, the overlap guarantees the creation of a common topology-a picture of the whole domain for each node.

- **Building Routing Tables:**

  In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.

1. Creation of the states of the links by each node, called the link state packet (LSP).
2. Dissemination of LSPs to every other router, called **flooding,** in an efficient and reliable way.
3. Formation of a shortest path tree for each node.
4. Calculation of a routing table based on the shortest path tree.

1. Creation of Link State Packet (LSP):

- A link state packet can carry a large amount of information. For the moment, however, we assume that it carries a minimum amount of data: the node identity, the list of links, a sequence number, and age.
- The first two, node identity and the list of links, are needed to make the topology.
- The third, sequence number, facilitates flooding and distinguishes new LSPs from old ones.
- The fourth, age, prevents old LSPs from remaining in the domain for a long time.

❖ LSPs are generated on two occasions:

- When there is a change in the topology of the domain. Triggering of LSP dissemination is the main way of quickly informing any node in the domain to update its topology.
- On a periodic basis. The period in this case is much longer compared to distance vector routing. As a matter of fact, there is no actual need for this type of LSP dissemination. It is done to ensure that old information is removed from the domain. The timer set for periodic dissemination is normally in the range of 60 min or 2 h based on the implementation. A longer period ensures that flooding does not create too much traffic on the network.

2. Flooding of LSPs

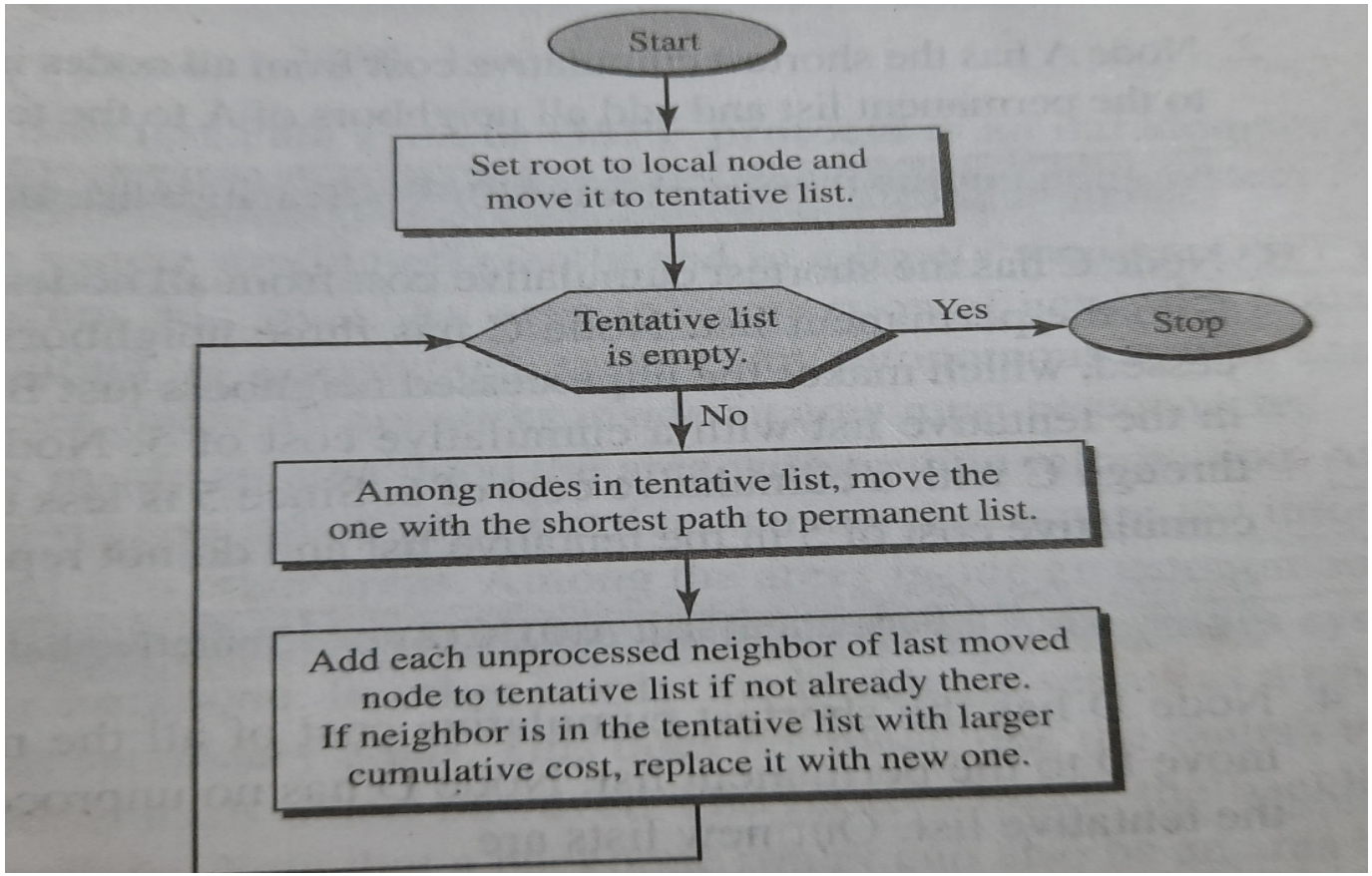
After a node has prepared an LSP, it must be disseminated to all other nodes, not only to its neighbors. The process is called flooding and based on the following:

- The creating node sends a copy of the LSP out of each interface.
- A node that receives an LSP compares it with the copy it may already have. If the newly arrived LSP is older than the one it has (found by checking the sequence number), it discards the LSP. If it is newer, the node does the following:
  - ✓It discards the old LSP and keeps the new one.
  - ✓It sends a copy of it out of each interface except the one from which the packet arrived. This guarantees that flooding stops somewhere in the domain (where a node has only one interface).

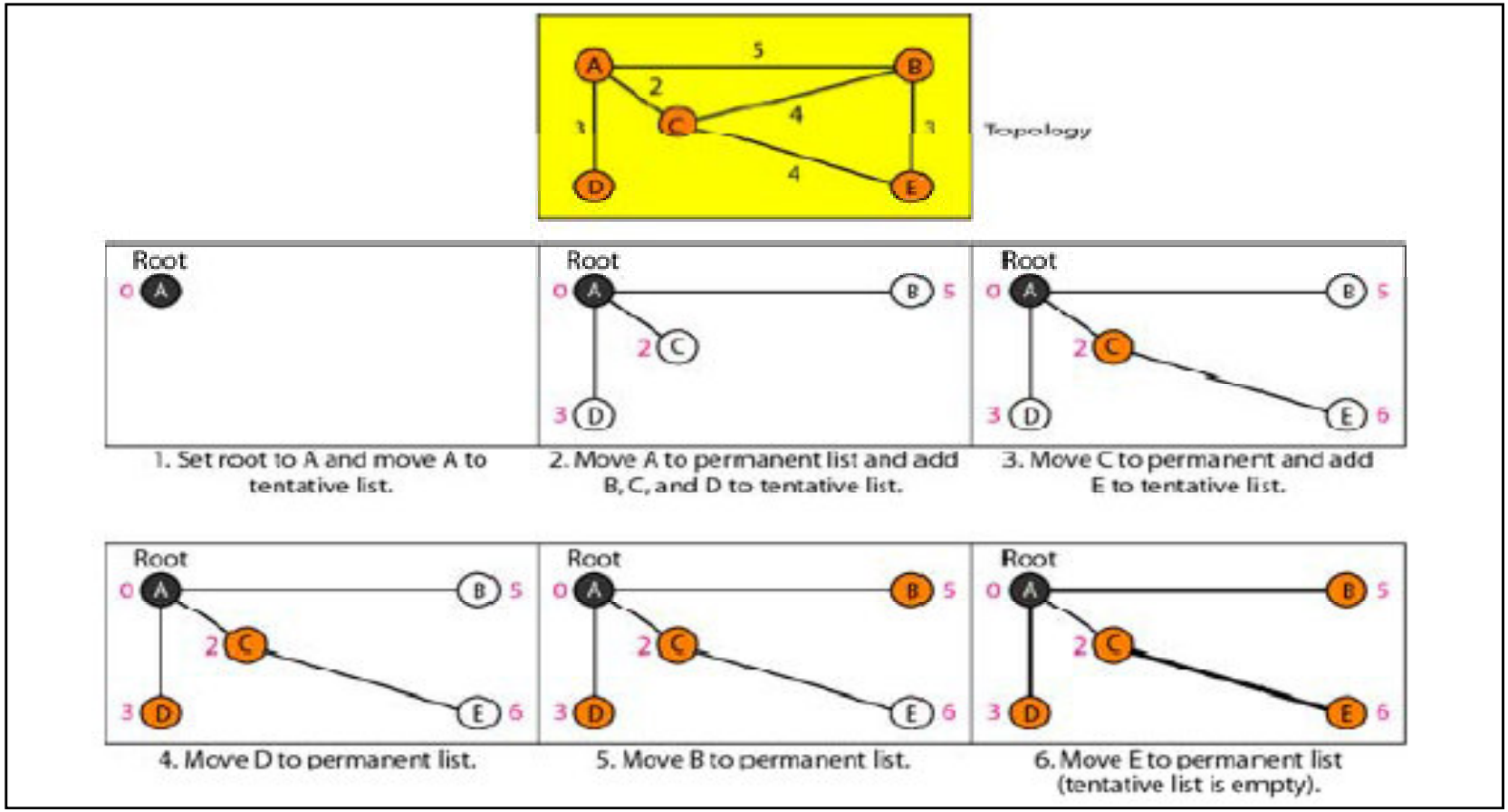
3. Formation of a shortest path tree: Dijkstra Algorithm:

- After receiving all LSPs, each node will have a copy of the whole topology. However, the topology is not sufficient to find the shortest path to every other node; a shortest path tree is needed.
- A tree is a graph of nodes and links; one node is called the root. All other nodes can be reached from the root through only one single route. A shortest path tree is a tree in which the path between the root and every other node is the shortest. What we need for each node is a shortest path tree with that node as the root.

- The Dijkstra algorithm creates a shortest path tree from a graph. The algorithm divides the nodes into two sets: tentative and permanent. It finds the neighbors of a current node, makes them tentative, examines them, and if they pass the criteria, makes them permanent



Let us apply the algorithm to node A of our sample graph in below figure. To find the shortest path in each step, we need the cumulative cost from the root to each node, which is shown next to the node.



The following shows the steps. At the end of each step, we show the permanent (filled circles) and the tentative (open circles) nodes and lists with the cumulative costs.

1. We make node A the root of the tree and move it to the tentative list. Our two lists are

   Permanent list: empty Tentative list: A(0)

2. Node A has the shortest cumulative cost from all nodes in the tentative list. We move A to the permanent list and add all neighbors of A to the tentative list. Our new lists are

   Permanent list: A(0) Tentative list: B(5), C(2), D(3)

3. Node C has the shortest cumulative cost from all nodes in the tentative list. We move C to the permanent list. Node C has three neighbors, but node A is already processed, which makes the unprocessed neighbors just B and E. However, B is already in the tentative list with a cumulative cost of 5. Node A could also reach node B through C with a cumulative cost of 6. Since 5 is less than 6, we keep node B with a cumulative cost of 5 in the tentative list and do not replace it. Our new lists are

   Permanent list: A(0), C(2) Tentative list: B(5), D(3), E(6)

4. Node D has the shortest cumulative cost of all the nodes in the tentative list. We move D to the permanent list. Node D has no unprocessed neighbor to be added to the tentative list. Our new lists are

Permanent list: A(0), C(2), D(3) Tentative list: B(5), E(6)

5. Node B has the shortest cumulative cost of all the nodes in the tentative list. We move B to the permanent list. We need to add all unprocessed neighbors of B to the tentative list (this is just node E). However, E(6) is already in the list with a smaller cumulative cost. The cumulative cost to node E, as the neighbor of B, is 8. We keep node E(6) in the tentative list. Our new lists are

Permanent list: A(0), B(5), C(2), D(3) Tentative list: E(6)

6. Node E has the shortest cumulative cost from all nodes in the tentative list. We move E to the permanent list. Node E has no neighbor. Now the tentative list is empty. We stop; our shortest path tree is ready. The final lists are

Permanent list: A(0), B(5), C(2), D(3), E(6) Tentative list: empty

4. Calculation of Routing Table from Shortest Path Tree:

Each node uses the shortest path tree protocol to construct its routing table. The routing table shows the cost of reaching each node from the root.

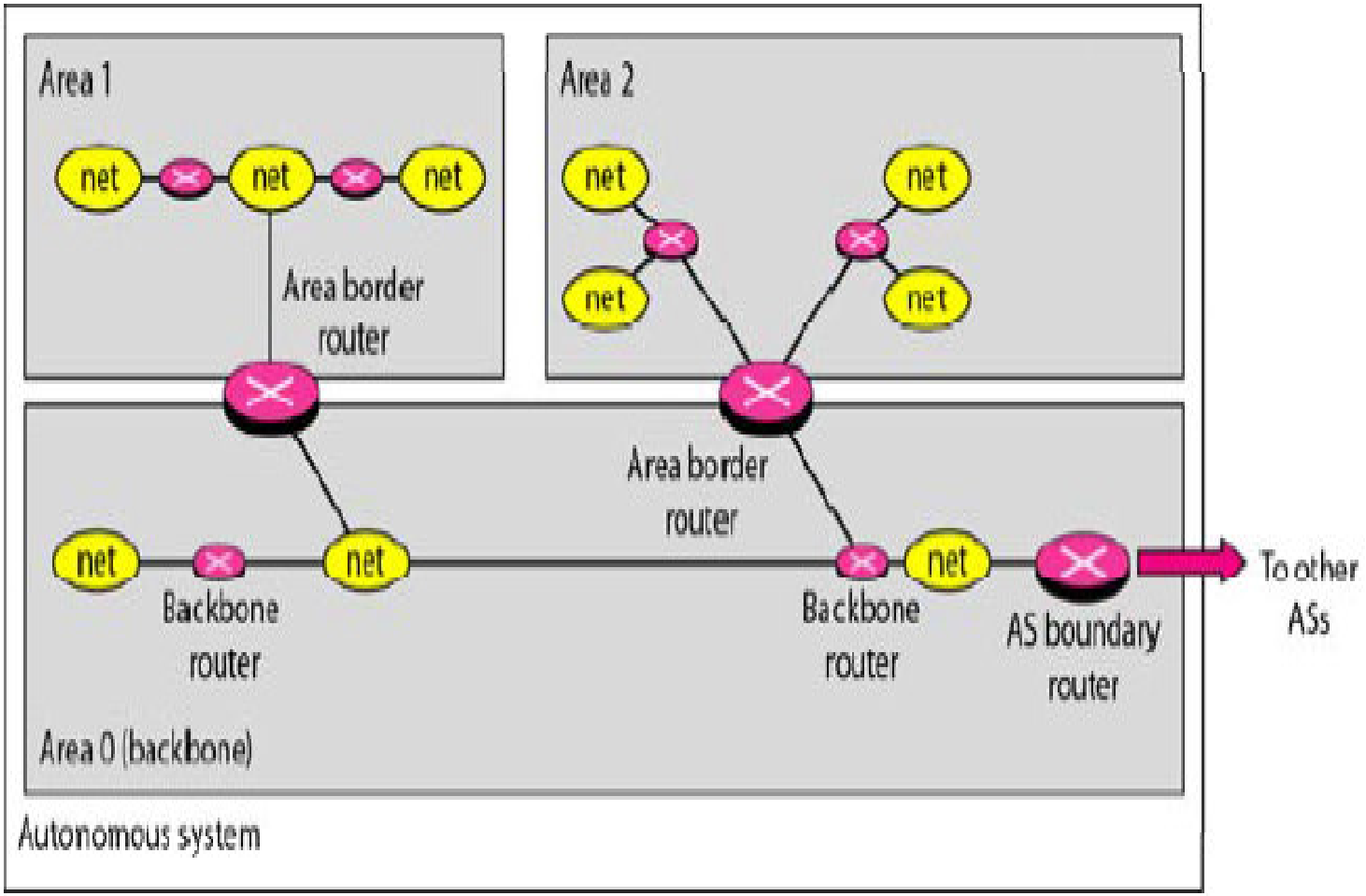
Routing table for node A

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

Open Shortest Path First or OSPF

- The Open Shortest Path First or OSPF protocol is an intradomain routing protocol based on link state routing. Its domain is also an autonomous system.
- **Areas in an autonomous system:**
  - To handle routing efficiently and in a timely manner, OSPF divides an autonomous system into areas.
  - An area is a collection of networks, hosts, and routers all contained within an autonomous system. An autonomous system can be divided into many different areas. All networks inside an area must be connected.

- At the border of an area, special routers called area border routers summarize the information about the area and send it to other areas.
- Among the areas inside an autonomous system is a special area called the backbone; all the areas inside an autonomous system must be connected to the backbone.
- The routers inside the backbone are called the backbone routers. Note that a backbone router can also be an area border router.
- If, because of some problem, the connectivity between a backbone and an area is broken, a virtual link between routers must be created by an administrator to allow continuity of the functions of the backbone as the primary area.
- Each area has area identification. The area identification of the backbone is zero.


Area 1
net
net
net
Area border
router
Area 2
net
net
net
net
Area border
router
net
Backbone
router
net
Backbone
router
net
AS boundary
router
To other
ASs
Area 0 (backbone)
Autonomous system

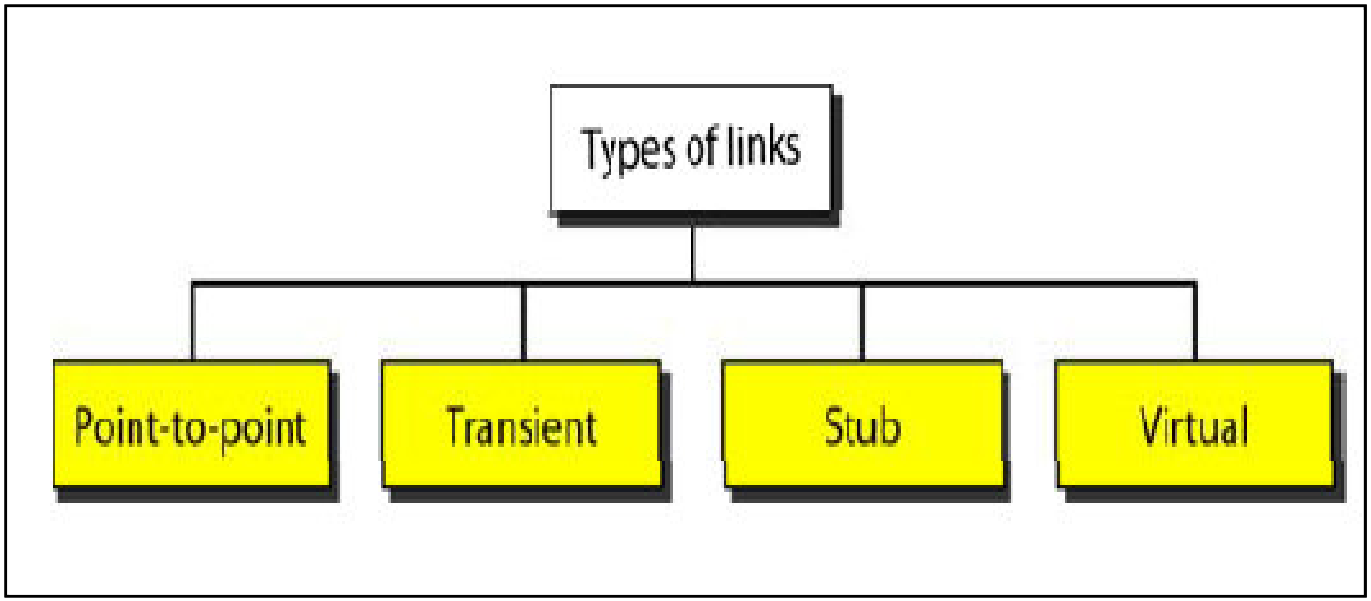
❖ Metric

The OSPF protocol allows the administrator to assign a cost, called the metric, to each route. The metric can be based on a type of service (minimum delay, maximum throughput, and so on). As a matter of fact, a router can have multiple routing tables, each based on a different type of service.

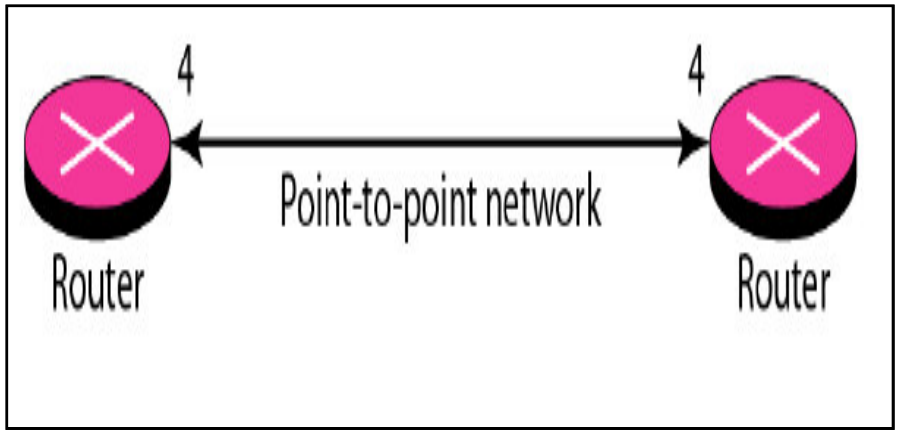
❖ Types of Links:

In OSPF terminology, a connection is called a link. Four types of links have been defined:

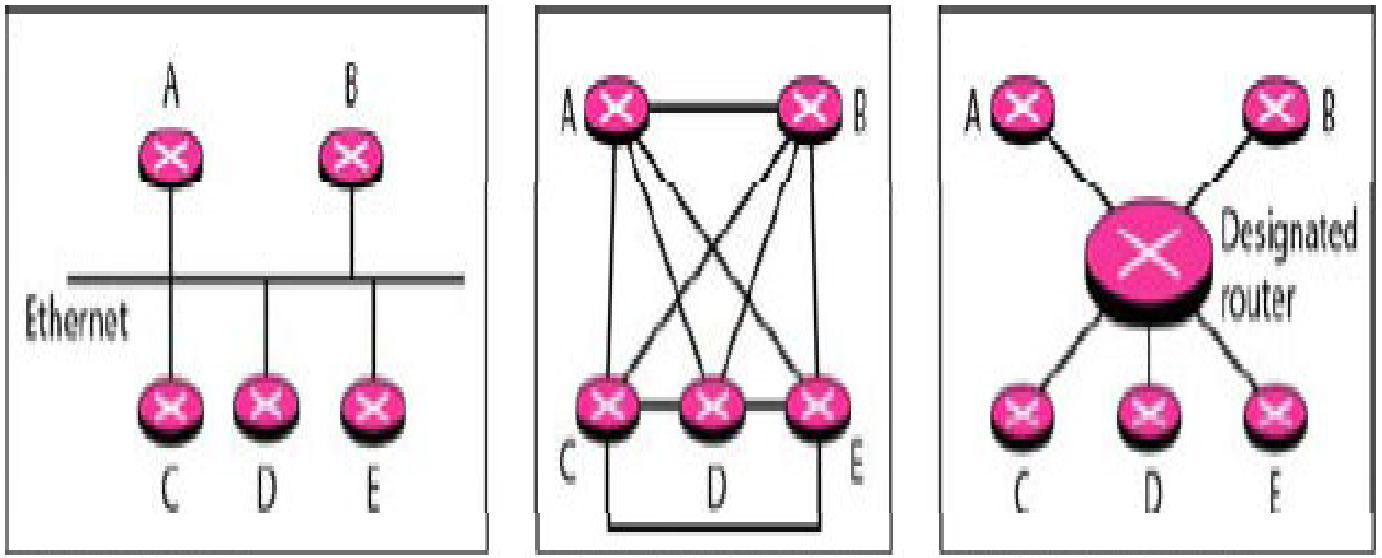
- point-to-point,
- transient,
- stub, and
- virtual.


Types of links
Point-to-point
Transient
Stub
Virtual

1. **A point-to-point link** connects two routers without any other host or router in between. In other words, the purpose of the link (network) is just to connect the two routers. An example of this type of link is two routers connected by a telephone line or a T line. There is no need to assign a network address to this type of link.

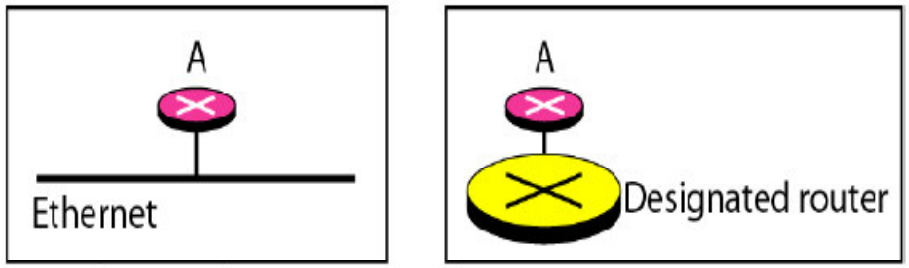


2. **A transient link** is a network with several routers attached to it. The data can enter through any of the routers and leave through any router. All LANs and some WANs with two or more routers are of this type. In this case, each router has many neighbors.



a. Transient network b. Unrealistic representation c. Realistic representation

3. **A stub link** is a network that is connected to only one router. The data packets enter the network through this single router and leave the network through this same router. This is a special case of the transient network. We can show this situation using the router as a node and using the designated router for the network. However, the link is only one-directional, from the router to the network.



a. Stub network b. Representation

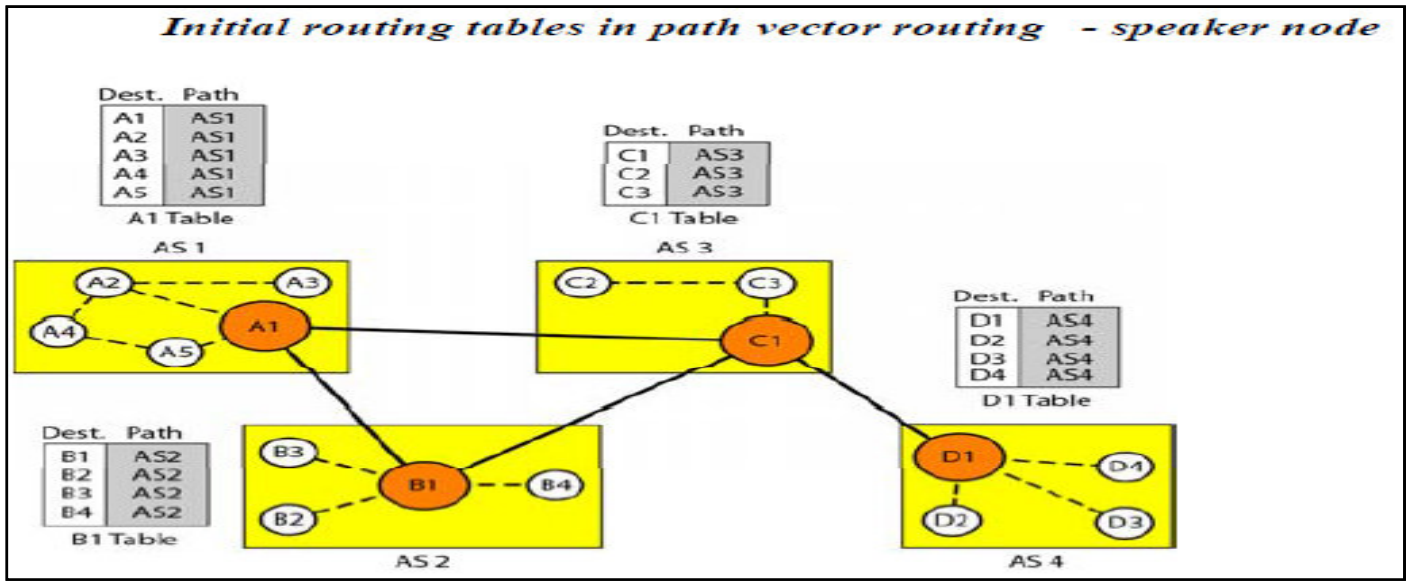
4. When the link between two routers is broken, the administration may create a **virtual link** between them, using a longer path that probably goes through several routers.

3. Path Vector Routing

- Path vector routing proved to be useful for interdomain routing. The principle of path vector routing is similar to that of distance vector routing. In path vector routing, we assume that there is one node in each autonomous system that acts on behalf of the entire autonomous system.

Initialization:

At the beginning, each speaker node can know only the reachability of nodes inside its autonomous system.



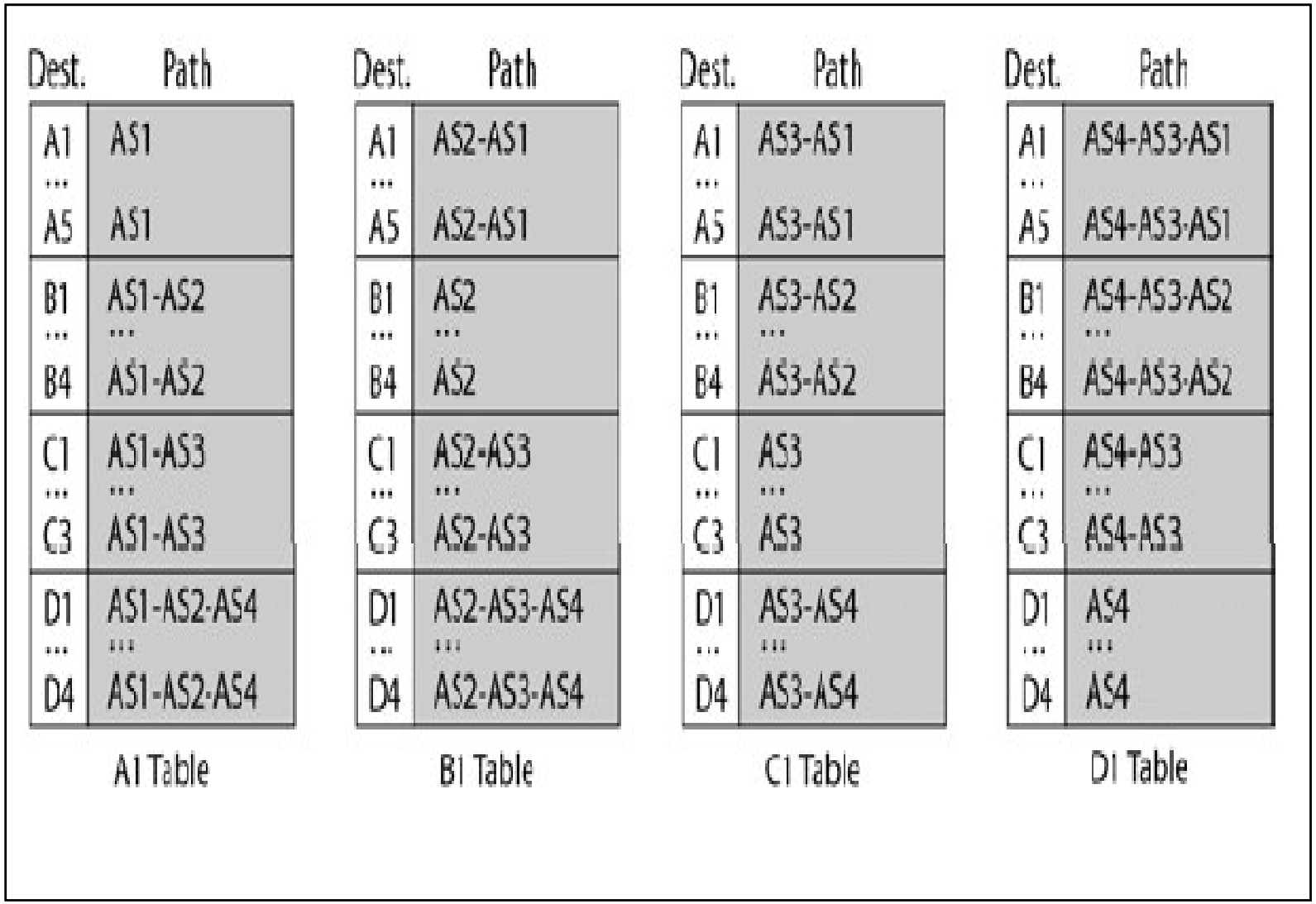
Node A1 is the speaker node for AS1, B1 for AS2, C1 for AS3 and D1 for AS4. Node A1 creates an initial table that shows A1 to A5 are located in AS1 and can be reached through it. Node B1 advertises that B1 to B4 are located in AS2 and can be reached through Bl. And so on.

Sharing:

Just as in distance vector routing, in path vector routing, a speaker in an autonomous system shares its table with immediate neighbors. In node A1 shares its table with nodes B1 and C1. Node C1 shares its table with nodes D1, B1, andA1. Node B1 shares its table with C1 and A1. Node D1 shares its table with C1.

Updating:

When a speaker node receives a two-column table from a neighbor, it updates its own table by adding the nodes that are not in its routing table and adding its own autonomous system and the autonomous system that sent the table. After a while each speaker has a table and knows how to reach each node in other Ass.The below figure shows the tables for each speaker node after the system is stabilized.


Dest.
Path
A1
...
A5
AS1
B1
...
B4
AS1-AS2
C1
...
C3
AS1-AS3
D1
...
D4
AS1-AS2-AS4
A1 Table
Dest.
Path
A1
...
A5
AS2-AS1
B1
...
B4
AS2
C1
...
C3
AS2-AS3
D1
...
D4
AS2-AS3-AS4
B1 Table
Dest.
Path
A1
...
A5
AS3-AS1
B1
...
B4
AS3-AS2
C1
...
C3
AS3
D1
...
D4
AS3-AS4
C1 Table
Dest.
Path
A1
...
A5
AS4-AS3-AS1
B1
...
B4
AS4-AS3-AS2
C1
...
C3
AS4-AS3
D1
...
D4
AS4
D1 Table

❖ **Advantage:**

- **Loop prevention.** The instability of distance vector routing and the creation of loops can be avoided in path vector routing. When a router receives a message, it checks to see if its autonomous system is in the path list to the destination. If it is, looping is involved and the message is ignored.
- **Policy routing**. Policy routing can be easily implemented through path vector routing. When a router receives a message, it can check the path. If one of the autonomous systems listed in the path is against its policy, it can ignore that path and that destination. It does not update its routing table with this path, and it does not send this message to its neighbors.
- **Optimum path**. We are looking for a path to a destination that is the best for the organization that runs the autonomous system. We definitely cannot include metrics in this route because each autonomous system that is included in the path may use a different criterion for the metric. One system may use, internally, RIP, which defines hop count as the metric; another may use OSPF with minimum delay defined as the metric. The optimum path is the path that fits the organization.

Border Gateway Protocol (BGP)

- Border Gateway Protocol (BGP) is an interdomain routing protocol using path vector routing.
- **Path Attributes:**
  - The path was presented as a list of autonomous systems, but is, in fact, a list of attributes. Each attribute gives some information about the path. The list of attributes helps the receiving router make a more-informed decision when applying its policy.
  - Attributes are divided into two broad categories: well known and optional. A well known attribute is one that every BGP router must recognize. An optional attribute is one that needs not be recognized by every router.

- Well-known attributes are themselves divided into two categories: mandatory and discretionary.
- A well-known mandatory attribute is one that must appear in the description of a route.
- A well-known discretionary attribute is one that must be recognized by each router, but is not required to be included in every update message.
- One well known mandatory attribute is ORIGIN. This defines the source of the routing information (RIP, OSPF, and so on).
- Another well-known mandatory attribute is AS_PATH. This defines the list of autonomous systems through which the destination can be reached.
- Still another well-known mandatory attribute is NEXT-HOP, which defines the next router to which the data packet should be sent.

- The optional attributes can also be subdivided into two categories: transitive and non transitive.
- An optional transitive attribute is one that must be passed to the next router by the router that has not implemented this attribute.
- An optional non transitive attribute is one that must be discarded if the receiving router has not implemented it.

❖ **BGP Sessions:**

The exchange of routing information between two routers using BGP takes place in a session. A session is a connection that is established between two BGP routers only for the sake of exchanging routing information. To create a reliable environment, BGP uses the services of TCP. In other words, a session at the BGP level, as an application program, is a connection at the TCP level. However, there is a subtle difference between a connection in TCP made for BGP and other application programs. When a TCP connection is created for BGP, it can last for a long time, until something unusual happens. For this reason, BGP sessions are sometimes referred to as semipermanent connections.

- **External and Internal BGP** If we want to be precise, BGP can have two types of sessions: external BGP (E-BGP) and internal BGP (I-BGP) sessions.
  - The E-BGP session is used to exchange information between two speaker nodes belonging to two different autonomous systems.
  - The I-BGP session, on the other hand, is used to exchange routing information between two routers inside an autonomous system.