SRM Institute of Science and Technology

21CSC302J-COMPUTER NETWORKS

Unit- III

Routing

Routing In An Internet

- **In a packet switching system,** ***routing refers to the process of choosing a path over which to send packets***

- Routing occurs at several levels.

- **The goal of IP is to**
  - ***Provide a virtual network that encompasses multiple physical networks***
  - ***Offers a connectionless datagram delivery service.***

- IP forwarding is also called internet routing or IP routing.

- The information used to make routing decisions is known as IP routing information.

- Like routing within a single physical network,
  - ***IP routing chooses a path over which a datagram should be sent.***

- Unlike routing within a single network,
  - ***IP routing algorithm must choose how to send a datagram across multiple physical networks.***

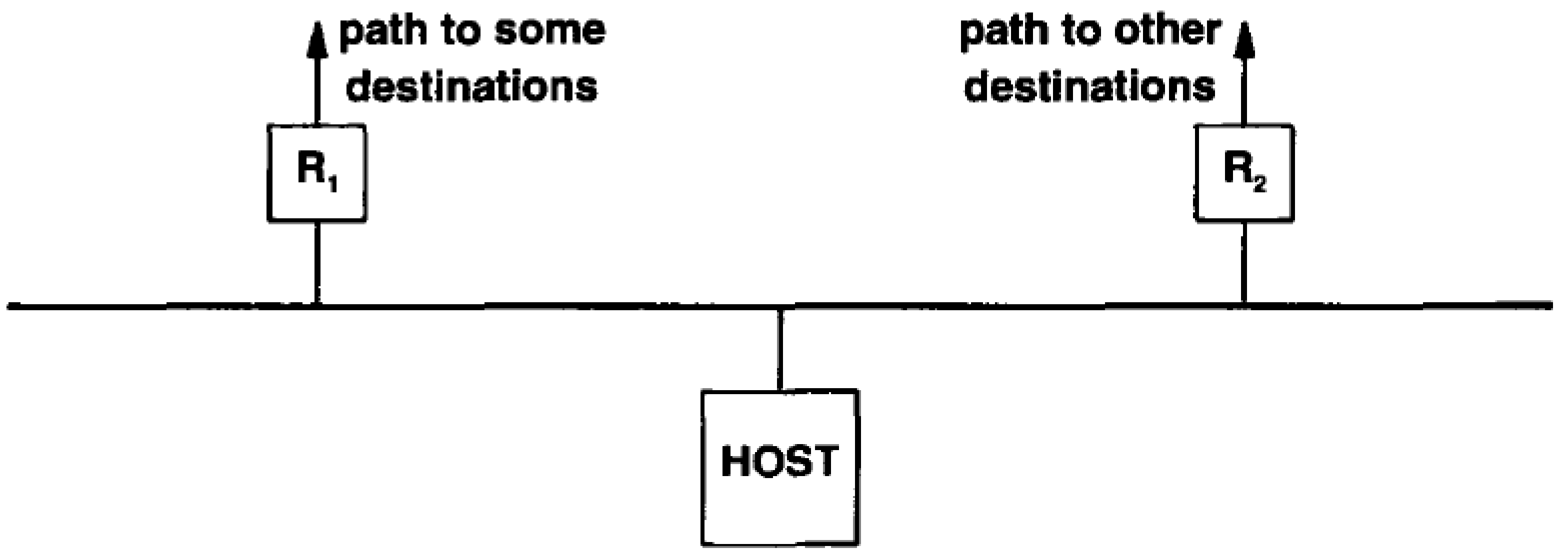
- Routing in an internet can be difficult,
  - ***multiple physical network connections.***

- Both hosts and routers participate in routing an IP datagram to its destination.

- When an application program on a host attempts to communicate, the TCP/IP protocols eventually generate one or more IP datagram.

- The host must make an initial routing decision when it chooses where to send the datagrams.

- Figure shows, hosts must make routing decisions even if they have only one network connection.



An example of a singly-homed host that must route datagram.

- The primary purpose of routers is to make IP routing decisions.
- What about multi-homed hosts?
- Any computer with multiple network connections can act as a router
- Multi-homed hosts running TCP/IP have all the software needed for routing.

Direct And Indirect Delivery

- we can divide routing into two forms:
  - ***Direct delivery and indirect delivery.***

- Direct delivery,
  - ***Transmission of a datagram from one machine across a single physical network directly to another***
  - ***Two machines can engage in direct delivery only if they both attach directly to the same underlying physical transmission system***

- Indirect delivery
  - **the destination is not on a directly attached network,**
  - **forcing the sender to pass the datagram to a router for delivery.**

Datagram Delivery Over A Single Network

- One machine on a given physical network can send a physical frame directly to another machine on the same network.

- To transfer an IP datagram,
  - ***The sender encapsulates the datagram in a physical frame,***
  - ***Maps the destination IP address into a physical address, and***
  - ***Uses the network hardware to deliver it.***

Datagram Delivery Over A Single Network

Transmission of an IP datagram between two machines on a single physical network does not involve routers. The sender encapsulates the datagram in a physical frame, binds the destination IP address to a physical hardware address, and sends the resulting frame directly to the destination.

Datagram Delivery Over A Single Network

How does the sender know whether the destination lies on a directly connected network?

Because the internet addresses of all machines on a single network include a common network prefix and extracting that prefix requires only a few machine instructions, testing whether a machine can be reached directly is extremely efficient.

Datagram Delivery Over A Single Network

direct route the datagram does not happen to pass through any intervening routers.

Indirect Delivery

- more difficult than direct delivery
- The sender must identify a router to which the datagram can be sent.
- The router must forward the datagram on toward its destination network.
- To visualize how indirect routing works, imagine a large internet with many networks interconnected by routers but with only two hosts at the far ends.

Indirect Delivery

- **When one host wants to send to the other, *it encapsulates the datagram* and sends it to the nearest router.**

- The host can reach a router because all physical networks are interconnected, so there must be a router attached to each network.

- The originating host can reach a router using a single physical network.

- **Once the frame reaches the router,**
  - ***software extracts the encapsulated datagram, and***
  - ***the IP software selects the next router along the path towards the destination.***

Indirect Delivery

- **The datagram is again placed in a frame *and sent over the next physical network to a second router,* and so on, until it can be delivered directly.**

Routers in a TCP/IP internet form a cooperative, interconnected structure. Datagrams pass from router to router until they reach a router that can deliver the datagram directly.

How can a router know where to send each datagram? How can a host know which router to use for a given destination?

Table-Driven IP Routing

- The usual IP routing algorithm employs an Internet routing table (sometimes called an IP routing table)
  - ***Each machine stores information about possible destinations and how to reach them.***

- Both hosts and routers route datagrams, both have IP routing tables.

- Whenever the IP routing software in a host or router needs to transmit a datagram, it consults the routing table to decide where to send the datagram.

What information should be kept in routing tables?

Table-Driven IP Routing

- If every routing table ***contained information about every possible destination address,*** it would be impossible to keep the tables current.

- The number of possible destinations is large, machines would have insufficient space to store the information.

Next-Hop Routing

- Using the network portion of a destination address instead of the complete host address makes routing efficient and keeps routing tables small.

- More important, it helps hide information, keeping the details of specific hosts confined to the local environment in which those hosts operate.

- **Typically, a routing table contains pairs (N, R)**
  - ***N is the IP address of a destination network,***
  - ***R is the IP address of the "next" router along the path to network N.***

Next-Hop Routing

- Router R is called the next hop, and **the idea of using a routing table to store a next hop for each destination** is ***called next-hop routing.***

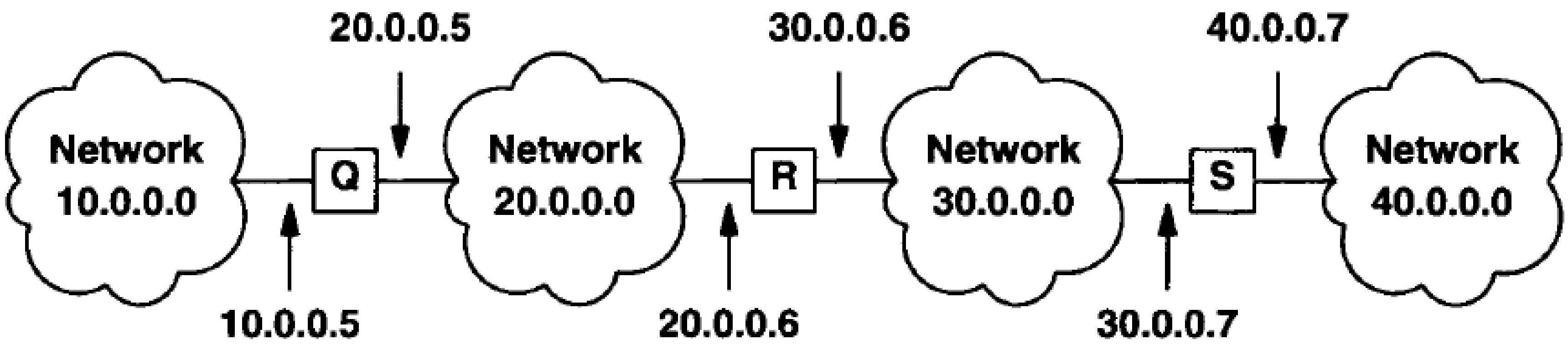
- The routing table in a router R only specifies one step along the path from R to a destination network - the router does not know the complete path to a destination.

- each entry in a routing table points to a router that can be reached across a single network.

- all routers listed in machine M’s routing table must lie on networks to which M connects directly.

Next-Hop Routing

- When a datagram is ready to leave M, IP software locates the destination IP address and extracts the network portion.
- M then uses the network portion to make a routing decision, selecting a router that can be reached directly.
- In practice, we apply the principle of information hiding to hosts as well.
- We insist that although hosts have IP routing tables, they must keep minimal information in their tables.
- The idea is to force hosts to rely on routers for most routing.



TO REACH HOSTS ON NETWORK	ROUTE TO THIS ADDRESS
20.0.0.0	DELIVER DIRECTLY
30.0.0.0	DELIVER DIRECTLY
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

The routing table in R.

Next-Hop Routing

- the size of the routing table depends on the number of networks in the internet;

- it only grows when new networks are added.

- The table size and contents are independent of the number of individual hosts connected to the networks.

Default Routes

- ***Another technique used to hide information and keep routing table sizes small***

- The idea is to have the IP routing software first look in the routing table for the destination network.

- **If no route appears in the table,** ***the routing routines send the datagram to a default router.***

- **Default routing is especially useful** ***when a site has a small set of local addresses*** and only one connection to the rest of the internet.

Default Routes

- For example, default routes work well in host computers that attach to a single physical network and reach only one router leading to the remainder of the internet.

- The routing decision consists of two tests: one for the local net and a default that points to the only router.

- Even if the site contains a few local networks, the routing is simple because it consists of a few tests for the local networks plus a default for all other destinations.

Host Specific Routes

- All routing is based on networks and not on individual hosts, most IP routing software allows per-host routes to be specified as a special case.

- Having per-host routes gives the local network administrator more control over network use, permits testing, and can also be used to control access for security purposes.

- When debugging network connections or routing tables, the ability to specify a special route to one individual machine turns out to be especially useful.

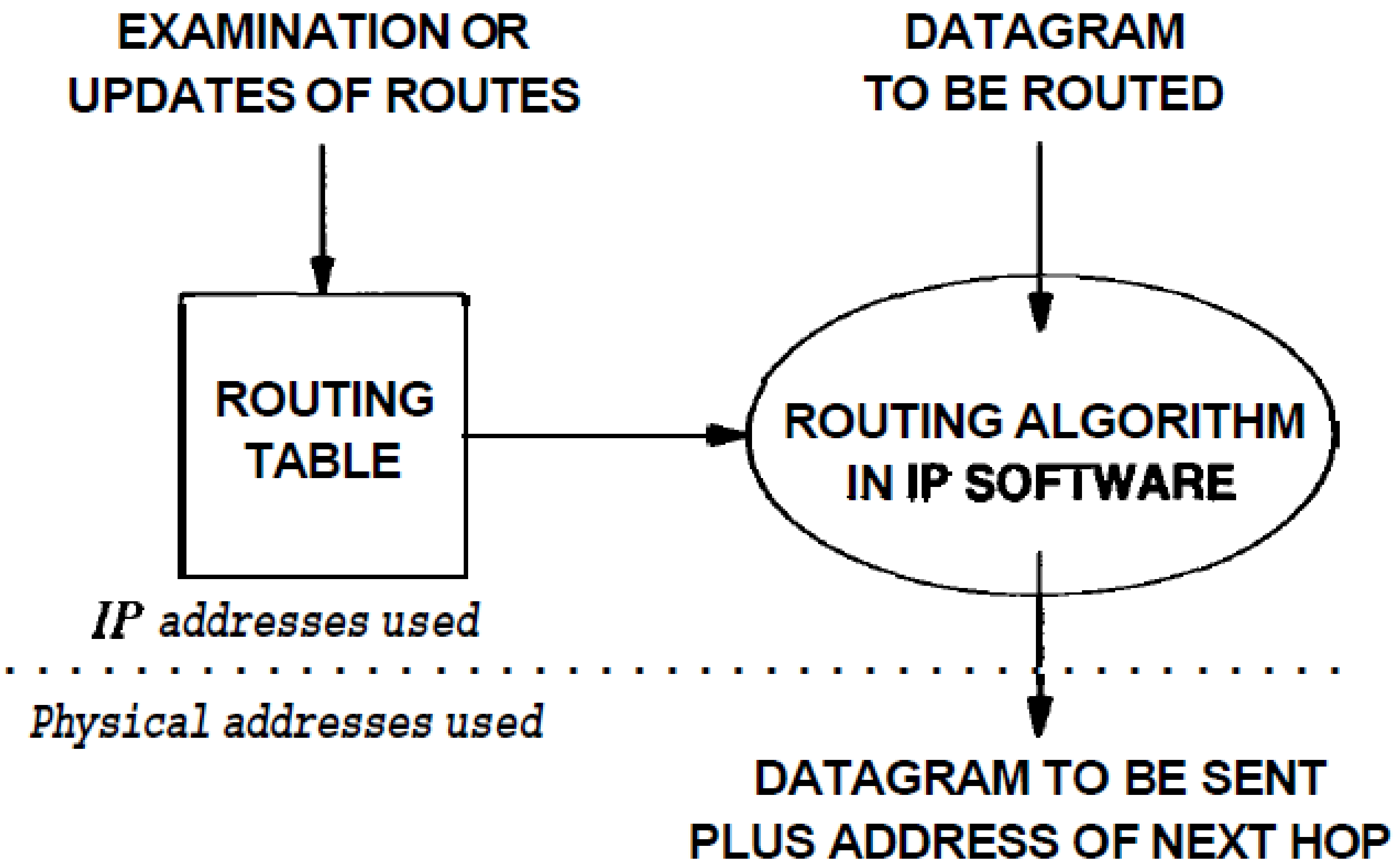
The IP Routing Algorithm

Algorithm:

RouteDatagram (Datagram, **RoutingTable**)

Extract destination **IP** address, D, from the datagram
    and compute the network prefix, N;
if N matches any directly connected network address
    deliver datagram to destination D over that network
    (This involves resolving D to a physical address,
    encapsulating the datagram, and sending the frame.)
else if the table contains a host-specific route for D
    send datagram to next-hop specified in table
else if the table contains a route for network N
    send datagram to next-hop specified in table
else if the table contains a default route
    send datagram to the default router specified in table
else declare a routing error;

The IP Routing Algorithm



Static Routing and Dynamic Routing

Static Routing

- Static routing and dynamic routing are two methods used
  - *To determine how to send a packet toward its destination.*

- **Static routing**
  - **uses preconfigured routes to send traffic to its destination**
  - **Network administrators use static routing, or nonadaptive routing,**
  - **to define a route when there is a single route or a preferred route for traffic to reach a destination.**

Static Routing

- **Uses small routing tables** ***with only one entry for each destination.***

- ***requires less computation time than dynamic routing*** **because each route is preconfigured.**

- **Static routes are preconfigured,** ***administrators must manually reconfigure routes to adapt to changes*** in the network when they occur.

- **Static routes are generally used in networks** ***where administrators don't expect any changes.***

Dynamic routing

- Uses algorithms to determine the best path
- **Requires routers to exchange information with other routers** ***to learn about paths through the network.***
- Dynamic routing, sometimes called adaptive routing
- **More complex than static routing** because ***it creates more possible routes to send packets across a network.***
- Dynamic routes are typically used in larger, fluid networks where static routes would be cumbersome to maintain and frequently reconfigure.

Dynamic routing

- Consumes more bandwidth than static routing.

- Uses algorithms
  - *To compute multiple possible routes and*
  - *Determine the best path for traffic to travel through the network.*

- It uses two types of complex algorithms:
  - *distance vector protocols and link state protocols.*

- Both distance vector and link state protocols create a routing table within the router that includes an entry for each possible destination of a network, group of networks or specific subnet.

Dynamic routing

- Each entry specifies which network connection to use to send out a received packet.

Static routing vs. dynamic routing

	Static routing	Dynamic routing
PATH SELECTION	One pre-configured route to destination	Multiple available routes to destination
ROUTE UPDATES	Engineers must reconfigure to make route changes	Algorithms automatically update with preferred route changes
ROUTING TABLES	Smaller routing table with only one entry for each destination	Routers send out entire routing tables to identify route availability
PROTOCOLS AND ALGORITHMS	Does not use protocols or algorithms for pre-configured route	Distance vector algorithms (RIP, IGRP) and link state algorithms (OSPF, IS-IS) adjust routes
COMPUTATION AND BANDWIDTH	Requires less computation time and bandwidth	Requires more computation and bandwidth
SECURITY	Better security	Less security
USE CASES	Used in smaller networks with fewer routers and unchanging network architecture	Used in larger networks and in networks that change frequently

Distance Vector Routing

- Dynamic Routing protocol
- Also called as Bellman-Ford algorithm or Ford Fulkerson algorithm
- Used to calculate the shortest path
- Calculates the distance and direction of the next hop from the information obtained by the neighboring router.
- It is necessary to keep track of the topology and inform neighboring devices if any changes occur in the topology.

- It is mainly used in ARPANET, and RIP.
- Each router maintains a distance table known as Vector

few key points

- **Network Information**
  - *Every node in the network should have information about its neighboring node.*
  - *Each node in the network is designed to share information with all the nodes in the network.*

- **Routing Pattern**
  - *In DVR the data shared by the nodes are transmitted only to that node that is linked directly to one or more nodes in the network.*

- **Data sharing**
  - *The nodes share the information with the neighboring node from time to time as there is a change in network topology.*

few key points

- The Distance vector algorithm is ***iterative, asynchronous and distributed.***
  - ***Distributed***
    - ***each node receives information from one or more of its directly attached neighbors,***
    - ***performs calculation and then distributes the result back to its neighbors.***
  - ***Iterative***
    - ***its process continues until no more information is available to be exchanged between neighbors.***
  - ***Asynchronous***
    - ***It does not require that all of its nodes operate in the lock step with each other.***

few key points

- **Knowledge about the whole network**
  - ***Each router shares its knowledge through the entire network.***
  - ***The Router sends its collected knowledge about the network to its neighbors.***

- **Routing only to neighbors**
  - ***The router sends its knowledge about the network to only those routers which have direct links.***
  - ***The router sends whatever it has about the network through the ports.***
  - ***The information is received by the router and uses the information to update its own routing table.***

few key points

- **Information sharing at regular intervals**
  - ***Within 30 seconds, the router sends the information to the neighboring routers.***

How the DVR Protocol Works

In DVR, each router maintains a routing table. It contains only one entry for each router. It contains two parts – a preferred outgoing line to use for that destination and an estimate of time (delay). Tables are updated by exchanging the information with the neighbor's nodes

Each router knows the delay in reaching its neighbors

Routers periodically exchange routing tables with each of their neighbors.

How the DVR Protocol Works

It compares the delay in its local table with the delay in the neighbor's table and the cost of reaching that neighbor.

If the path via the neighbor has a lower cost, then the router updates its local table to forward packets to the neighbor.

How the DVR Protocol Works

Information kept by DV router -

- Each router has an ID
- Associated with each link connected to a router, there is a link cost (static or dynamic).
- Intermediate hops

Distance Vector Table Initialization -

- Distance to itself = 0
- Distance to ALL other routers = infinity number.

```
Dx(y) = Estimate of least cost from x to y
C(x,v) =  Node x knows cost to each neighbor v
Dx   =  [Dx(y): y ∈ N ] = Node x maintains distance vector
Node x also maintains its neighbors' distance vectors
- For each neighbor v, x maintains Dv = [Dv(y): y ∈ N ]
```

How the DVR Protocol Works

When a node x receives new DV estimate from any neighbor v, it saves v's distance vector and it updates its own DV using

$$d_x(y) = \min_v\{c(x,v) + d_v(y)\}$$

At each node x,

Initialization

for all destinations y in N:

$D_x(y) = c(x,y)$ // If y is not a neighbor then $c(x,y) = \infty$

for each neighbor w

$D_w(y) = ?$ for all destination y in N.

for each neighbor w

send distance vector $D_x = [\ D_x(y) : y \text{ in } N\ ]$ to w

loop

wait(until I receive any distance vector from some neighbor w)

for each y in N:

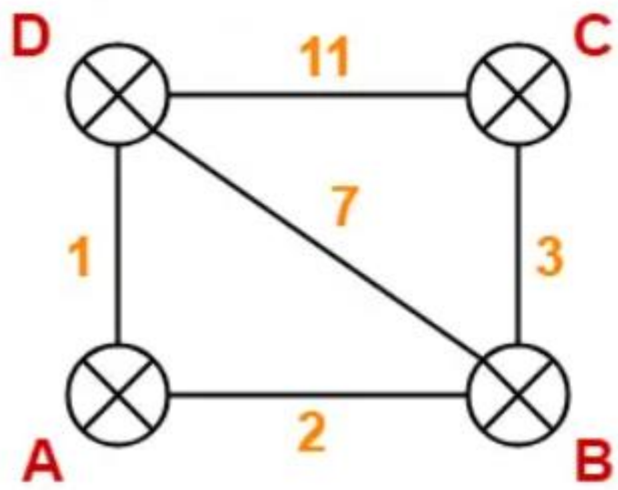
$D_x(y) = \text{minv}\{c(x,v)+D_v(y)\}$

If $D_x(y)$ is changed for any destination y

Send distance vector $D_x = [\ D_x(y) : y \text{ in } N\ ]$ to all neighbors

forever

How the DVR Protocol Works – STEP 01



At Router A-

Destination	Distance	Next Hop
A	0	A
B	2	B
C	∞	–
D	1	D

At Router B-

Destination	Distance	Next Hop
A	2	A
B	0	B
C	3	C
D	7	D

How the DVR Protocol Works - STEP 01

At Router C-

Destination	Distance	Next Hop
A	∞	–
B	3	B
C	0	C
D	11	D

At Router D -

Destination	Distance	Next Hop
A	1	A
B	7	B
C	11	C
D	0	D

STEP 02

At Router A- *Router A receives distance vectors from its neighbors B and D.*

From B
2
0
3
7

Cost(A→B) = 2

From D
1
7
11
0

Cost(A→D) = 1

Destination	Distance	Next hop
A	0	A
B		
C		
D		

New Routing Table at Router A

Cost of reaching destination B from router A = min { 2+0 , 1+7 } = 2 via B.

Cost of reaching destination C from router A = min { 2+3 , 1+11 } = 5 via B.

Cost of reaching destination D from router A = min { 2+7 , 1+0 } = 1 via D

STEP 02

At Router A-

- Router A can reach the destination router B via its neighbor B or neighbor D.
- It chooses the path which gives the minimum cost.
- Cost of reaching router B from router A via neighbor B = Cost (A→B) + Cost (B→B)= **2 + 0** = 2
- Cost of reaching router B from router A via neighbor D = Cost (A→D) + Cost (D→B) = **1 + 7** = 8
- Since the cost is minimum via neighbor B, so router A chooses the path via B.
- It creates an entry (2, B) for destination B in its new routing table.
- Similarly, we calculate the shortest path distance to each destination router at every router.

Destination	Distance	Next Hop
A	0	A
B	2	B
C	5	B
D	1	D

STEP 02

At Router B- *Router B receives distance vectors from its neighbors A, C and D*

From A	From C	From D
0	∞	1
2	3	7
∞	0	11
1	11	0
Cost (B→A) = 2	Cost (B→C) = 3	Cost (B→D) = 7

Destination	Distance	Next hop
A		
B	0	B
C		
D		

New Routing Table at Router B

Cost of reaching destination A from router B = min { 2+0 , 3+∞ , 7+1 } = 2 via A

Cost of reaching destination C from router B = min { 2+∞ , 3+0 , 7+11 } = 3 via C

Cost of reaching destination D from router B = min { 2+1 , 3+11 , 7+0 } = 3 via A

STEP 02

At Router B-

Destination	Distance	Next Hop
A	2	A
B	0	B
C	3	C
D	3	A

STEP 02

At Router C- *Router C receives distance vectors from its neighbors B and D*

From B
2
0
3
7

Cost (C→B) = 3

From D
1
7
11
0

Cost (C→D) = 11

Destination	Distance	Next hop
A		
B		
C	0	C
D		

New Routing Table at Router C

Cost of reaching destination A from router C = min { 3+2 , 11+1 } = 5 via B

Cost of reaching destination B from router C = min { 3+0 , 11+7 } = 3 via B

Cost of reaching destination D from router C = min { 3+7 , 11+0 } = 10 via B

STEP 02

At Router C-

Destination	Distance	Next Hop
A	5	B
B	3	B
C	0	C
D	10	B

STEP 02

At Router D- *Router D receives distance vectors from its neighbors A, B and C*

From A	From B	From C
0	2	∞
2	0	3
∞	3	0
1	7	11
Cost (D→A) = 1	Cost (D→B) = 7	Cost (D→C) = 11

Destination	Distance	Next hop
A		
B		
C		
D	0	D

New Routing Table at Router D

Cost of reaching destination A from router D = min { 1+0 , 7+2 , 11+∞ } = 1 via A

Cost of reaching destination B from router D = min { 1+2 , 7+0 , 11+3 } = 3 via A

Cost of reaching destination C from router D = min { 1+∞ , 7+3 , 11+0 } = 10 via B

STEP 02

At Router D-

Destination	Distance	Next Hop
A	1	A
B	3	A
C	10	B
D	0	D

STEP 03

Each router exchanges its distance vector obtained in Step-02 with its neighboring routers

After exchanging the distance vectors, each router prepares a new routing table

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

STEP 03

At Router A- *Router A receives distance vectors from its neighbors B and D.*

From B

2
0
3
3

Cost(A→B) = 2

From D

1
3
10
0

Cost(A→D) = 1

Destination	Distance	Next hop
A	0	A
B		
C		
D		

New Routing Table at Router A

Cost of reaching destination B from router A = min { 2+0 , 1+3 } = 2 via B

Cost of reaching destination C from router A = min { 2+3 , 1+10 } = 5 via B

Cost of reaching destination D from router A = min { 2+3 , 1+0 } = 1 via D

STEP 03

At Router A-

Destination	Distance	Next Hop
A	0	A
B	2	B
C	5	B
D	1	D

STEP 03

At Router B- *Router B receives distance vectors from its neighbors A, C and D*

From A
0
2
5
1

Cost (B→A) = 2

From C
5
3
0
10

Cost (B→C) = 3

From D
1
3
10
0

Cost (B→D) = 3

Destination	Distance	Next hop
A		
B	0	B
C		
D		

New Routing Table at Router B

Cost of reaching destination A from router B = min { 2+0 , 3+5 , 3+1 } = 2 via A

Cost of reaching destination C from router B = min { 2+5 , 3+0 , 3+10 } = 3 via C

Cost of reaching destination D from router B = min { 2+1 , 3+10 , 3+0 } = 3 via A

STEP 03

At Router B-

Destination	Distance	Next Hop
A	2	A
B	0	B
C	3	C
D	3	A

STEP 03

At Router C- *Router C receives distance vectors from its neighbors B and D*

From B

2
0
3
3

Cost (C→B) = 3

From D

1
3
10
0

Cost (C→D) = 10

Destination	Distance	Next hop
A		
B		
C	0	C
D		

New Routing Table at Router C

Cost of reaching destination A from router C = min { 3+2 , 10+1 } = 5 via B

Cost of reaching destination B from router C = min { 3+0 , 10+3 } = 3 via B

Cost of reaching destination D from router C = min { 3+3 , 10+0 } = 6 via B

STEP 03

At Router C-

Destination	Distance	Next Hop
A	5	B
B	3	B
C	0	C
D	6	B

STEP 03

At Router D- *Router D receives distance vectors from its neighbors A, B and C*

From A

0
2
5
1

Cost (D→A) = 1

From B

2
0
3
3

Cost (D→B) = 3

From C

5
3
0
10

Cost (D→C) = 10

Destination	Distance	Next hop
A		
B		
C		
D	0	D

New Routing Table at Router D

Cost of reaching destination A from router D = min { 1+0 , 3+2 , 10+5 } = 1 via A

Cost of reaching destination B from router D = min { 1+2 , 3+0 , 10+3 } = 3 via A

Cost of reaching destination C from router D = min { 1+5 , 3+3 , 10+0 } = 6 via A

STEP 02

At Router D-

Destination	Distance	Next Hop
A	1	A
B	3	A
C	6	A
D	0	D

Thank You