Network Layer Routing - I

Yatindra Nath Singh

ynsingh@ieee.org

Dept. Of Electrical Engineering

IIT Kanpur-208016

20 August 2001

What is a network Layer?

In a network, nodes are connected by point-to-point links.

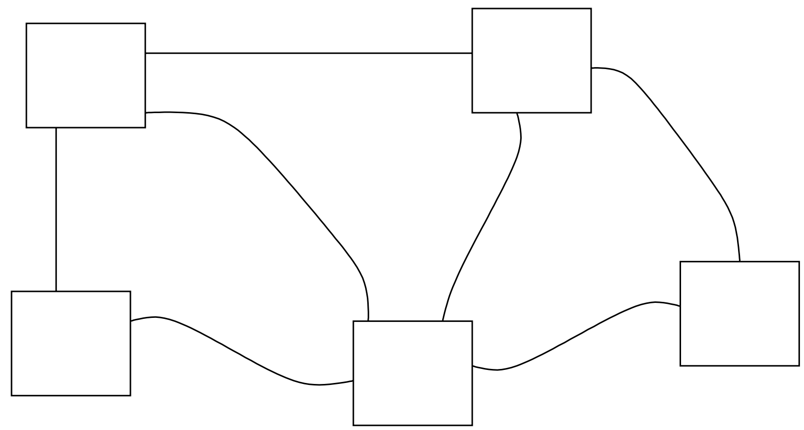

Fig.1

Network is build using layered architectures

- Each layer uses the services provided by lower layer.
- Each layer provides the services to upper layer.

Physical Layer

specifies

- physical specifications,
- signals levels and format,
- modulation formats

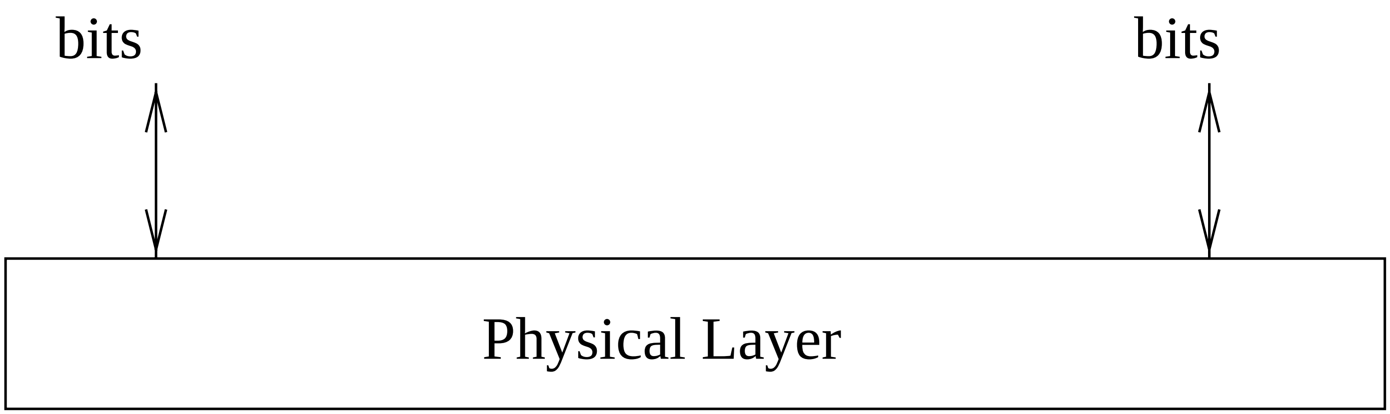

Fig.2

Provides the bit pipe to DLC (Data link control) layer

Data Link Control Layer

Another abstraction layer over Physical layer

Provides reliable transport of information over point-to-point link.

Transmission - via *frames*

frame has

- mechanism to delimit the frame boundries
- recovery of lost frames
- mechanism for detection of damaged frames

Network Layer

Using the DLC layer functionality, *Network Layer* is built

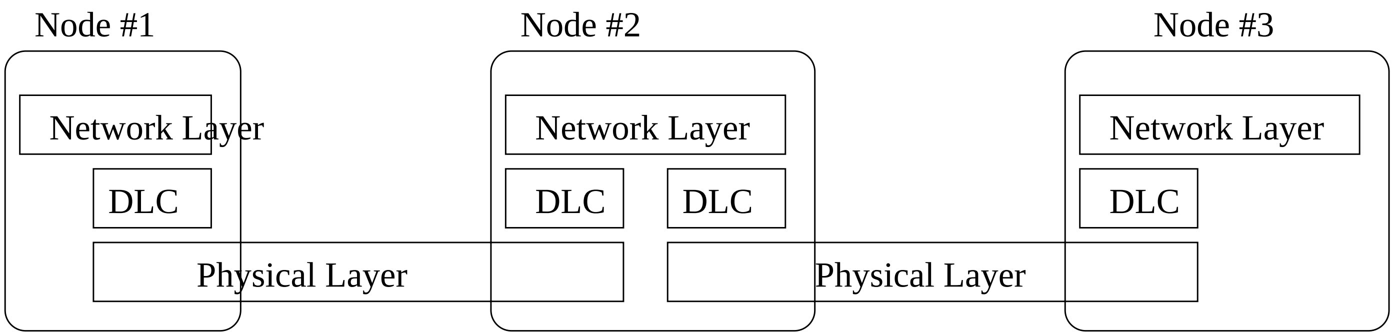

Fig.3

main function of network layer

- to route the packets incomming packet to appropriate interface or upper layer based on the destination host address in packet.

If network is circuit switched

- to determine the route while setting up the path

Function of Network Layer

- to maintain and periodically update information about network topology.
- All incomming packets if not destined for the machine are routed to appropriate interface.
- All the packets which are invalid are removed from network.
- Buffer management to minimize queueing delay and avoiding deadlocks and congestion.

In Fig.1, each node is a router (mesh topology).

Each node has multiple interfaces.

To facilitate the packet forwarding, each node has a forwarding table.

Each forwarding table entry has

- Destination address
- interface to packet should be forwarded

E.g. - forwarding table at node 1 is as follows

Destination	interface
1	upper layer
2	a
3	c
4	b
5	a or b

These forwarding table can be *static* or *dynamic*.

Static Routing Tables

- created once manually.
- Do not change on their own.
- Administrator knows the topology.
- Decides the routing table for each node.
- These tables are entered manually in each node.

Dyamic Routing

Static routing cannot take care of changes in topology which happens

- due to upgradation
- due to faults

Dynamic routing constantly updates the routing table after getting the information about the network topology.

Hierarchical routing

when large no. of nodes - present in a network, routing tables will become large. *Each routing table has entry for every node.*

Hierarchical routing can be used to reduce the size of routing table.

Consider network in Fig.4. Network partitioned in four subnets.

For routing,

- each node knows routes for reaching all node in its own subset.
- knows to reach other subnets.
- total entries in routing table of a node = number of nodes in subnets + number of other subnets. (for node 1A, 4+3=7 entries).

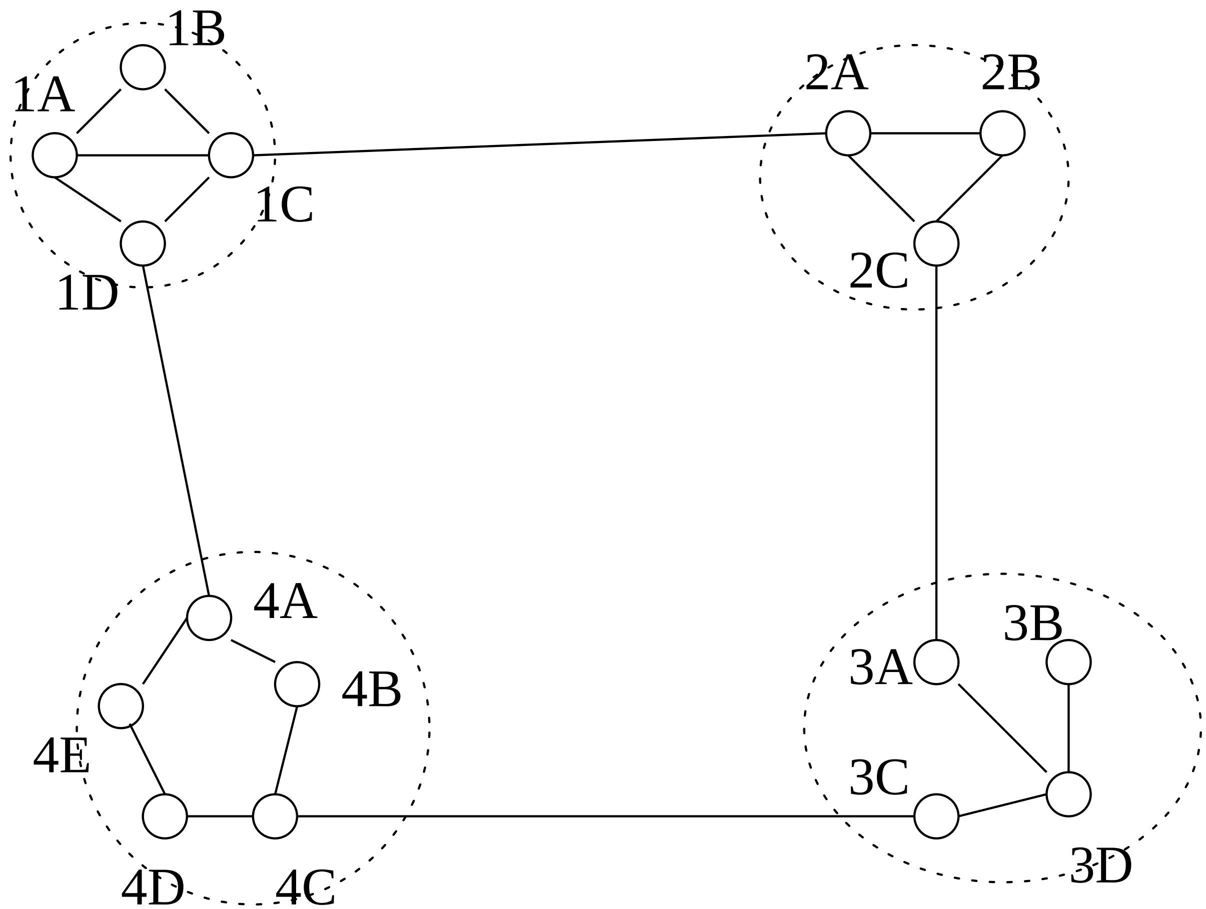

Fig.4

If hierarchical routing not used, number of entries in each routing table = number of nodes in the network. (For every node, 16 entries)

Routing tables for node 1A

Hierarchical routing		Non hierarchical routing	
Destination	Next host	Destination	next host
1A	-	1A	-
1B	1B	1B	1B
1C	1C	1C	1C
1D	1D	1D	1D
2	1C	2A	1C
3	1C	2B	1C
4	1D	2C	1C

Hierarchical routing		Non hierarchical routing	
Destination	Next host	Destination	next host
		3A	1C
		3B	1C
		3C	1D
		3D	1C
		4A	1D
		4B	1D
		4C	1D
		4D	1D
		4E	1D

routing table space - saved drastically using multiple hierarchies.

Optimal number of levels - $\ln N$

Routing Algorithms

Basic idea

- Cost is assigned to each link. The route is decided so as to minimize the cost from source to destination.
- From each node one can create spanning tree (There are no closed loops.)
- The movement along the path in spanning tree gives the shortest path (minimum cost path) from the node to every other node.

To build the shortest path spanning tree for a node

- If node j is falling on shortest path from node i to node k.

Then the same path is also the shortest path from node i to node j or from node j to node k.

Consider the network in Fig.5

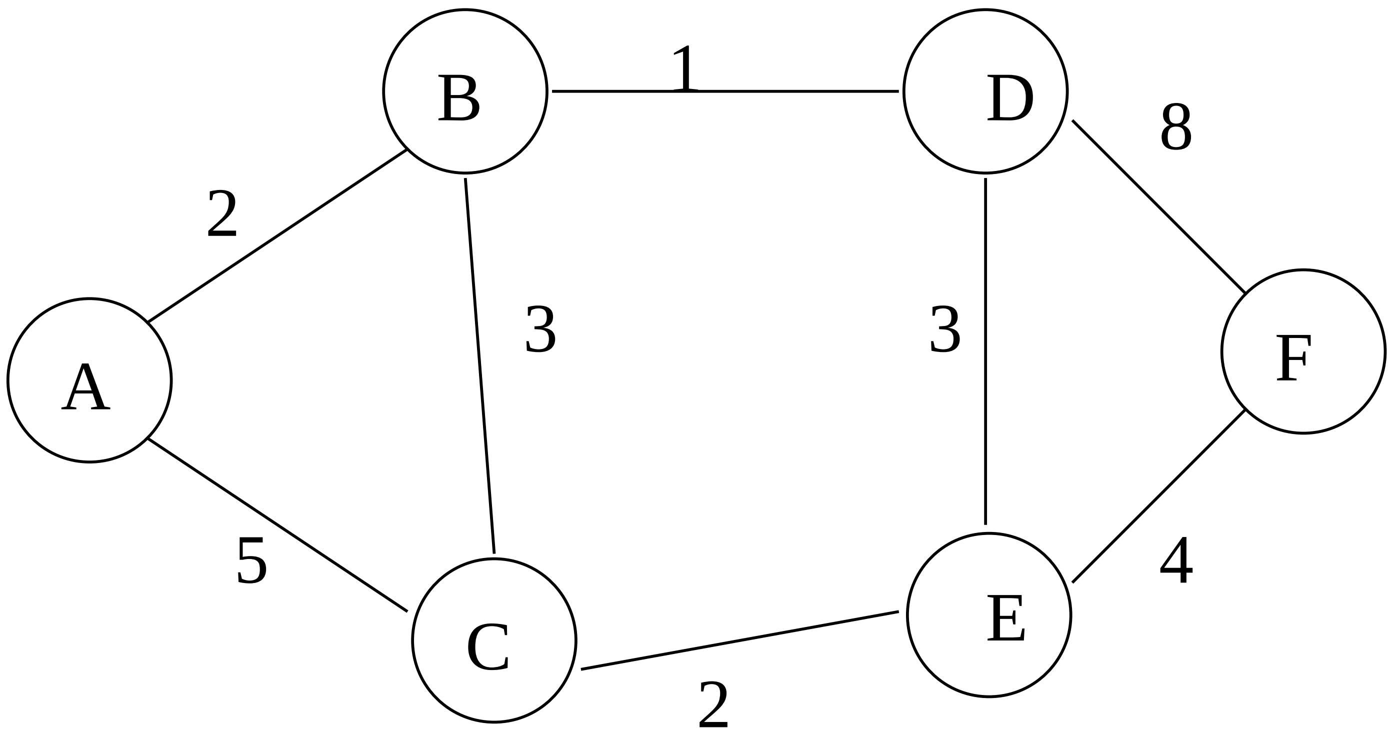

Fig.5

- the number on the links are costs.

Building spanning tree for node A

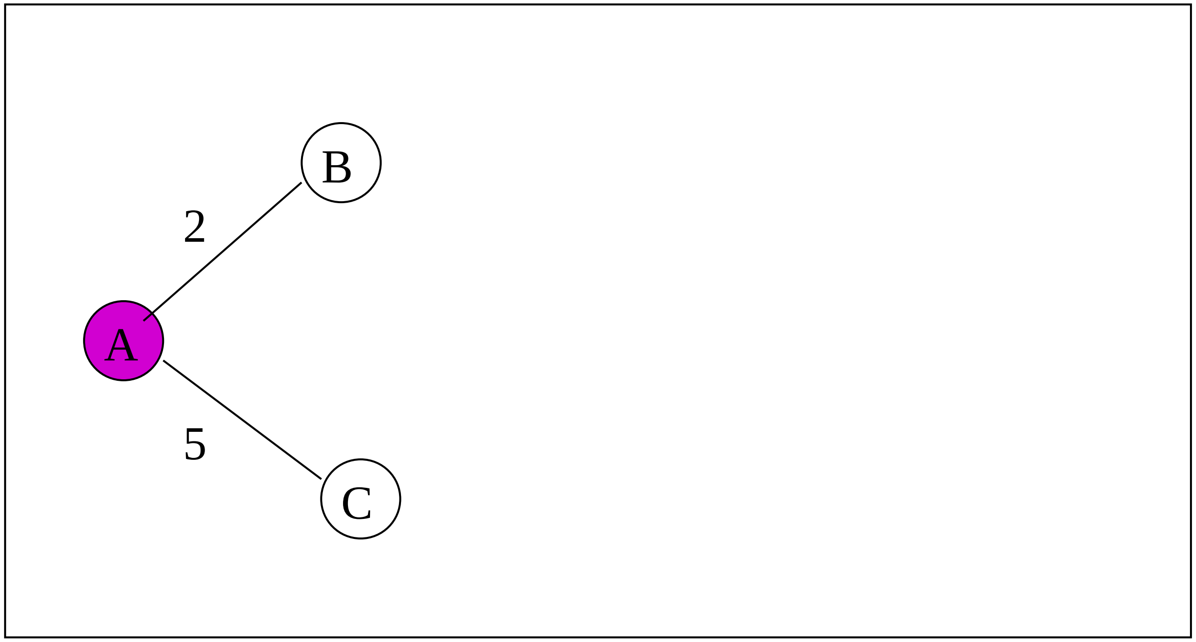

Fig.6a

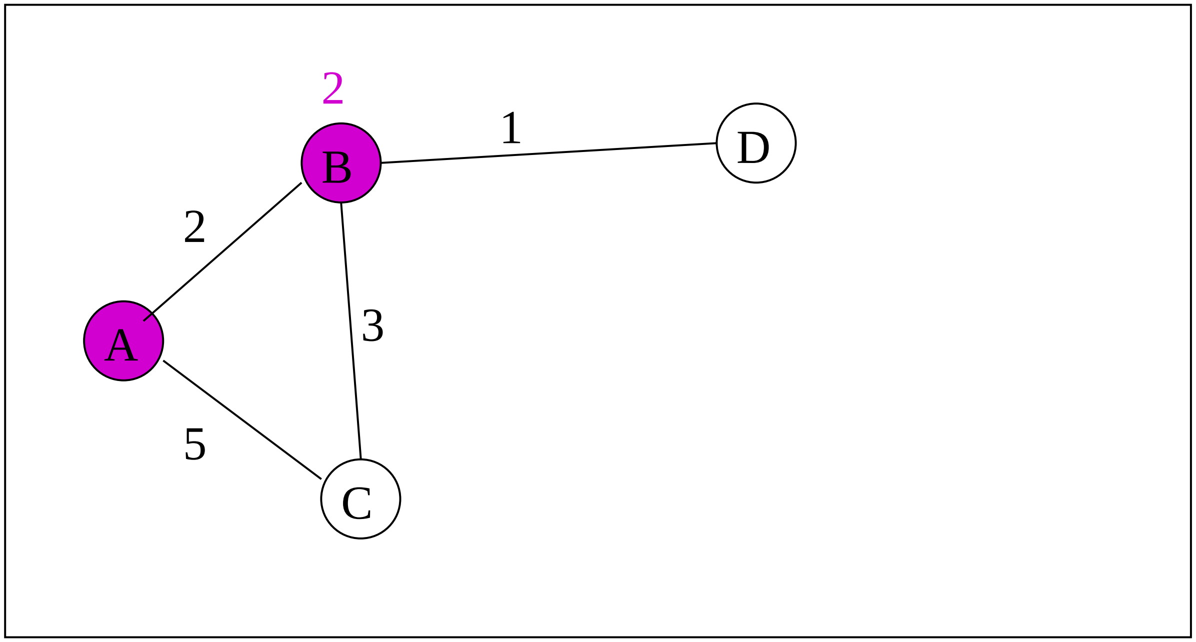

Fig.6b

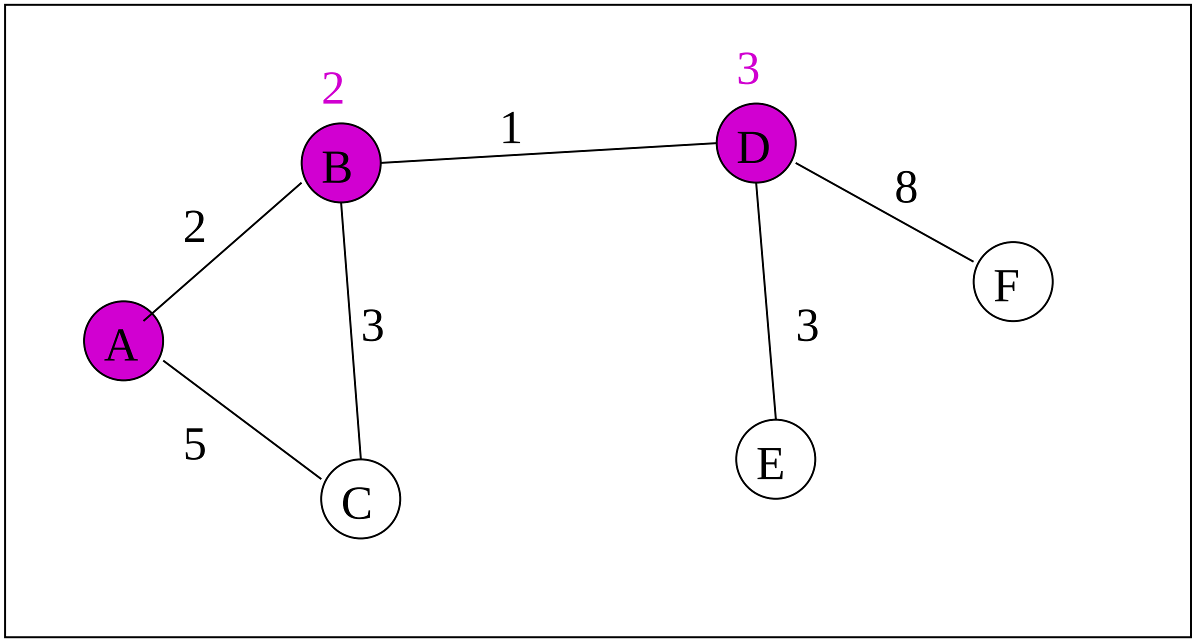

Fig.6c

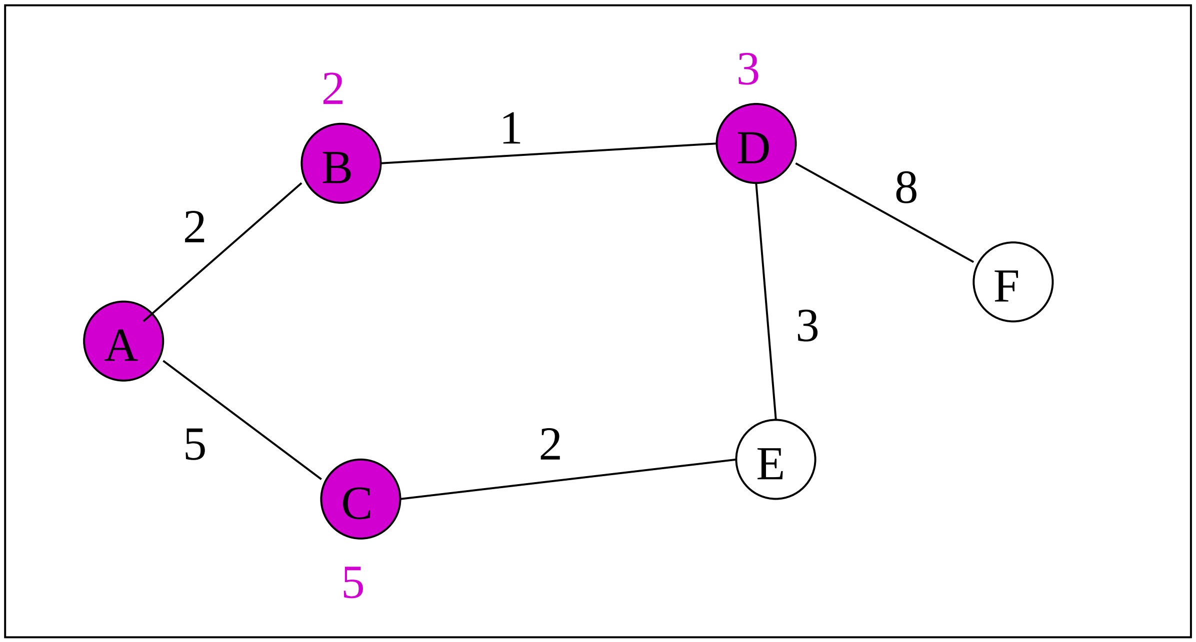

Fig.6d

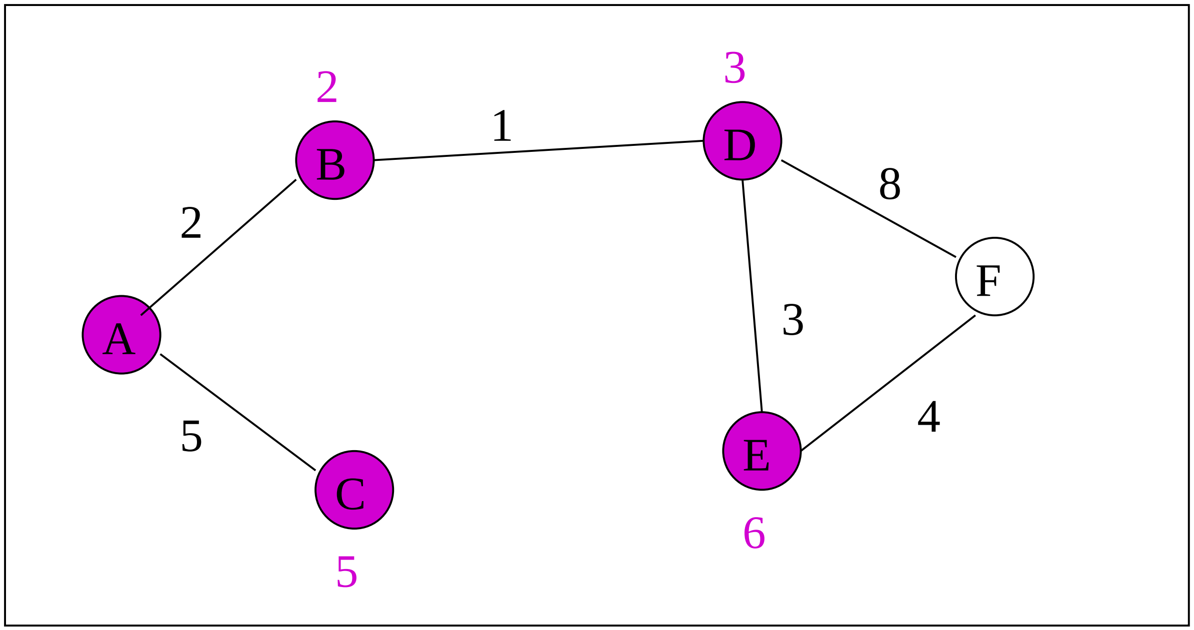

Fig.6e

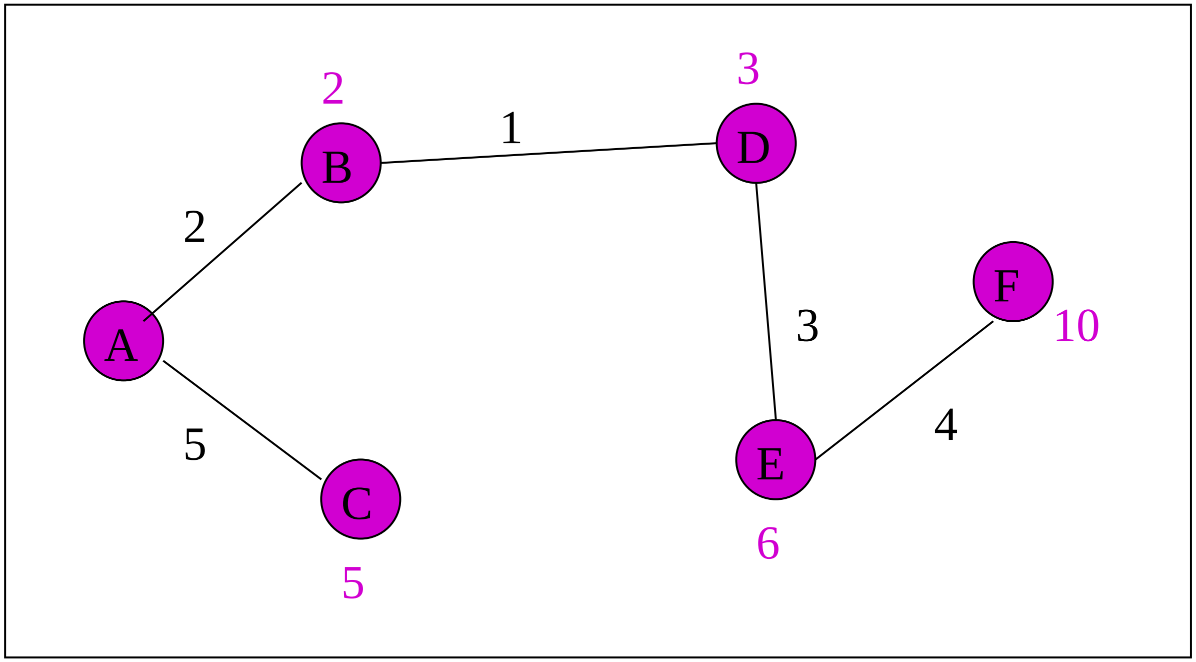

Fig.6f

The above algorithm is Djiktra's Algorithm.

It partions the network into number of partitions equal to links attached to the node.

E.g. - for node A two partitions are created

Partition 1 - B, D, E, F
Partition 2 - C

Routing table will be

Destination hosts	next host
B, D, E, F	B
C	C

There are two popular algorithms for maintainence of routing tables.

Both the algorithms converge to routing tables corresponding to minimum cost spanning tree.

Distance Vector Routing

- In this algorithm, neighbouring nodes exchange routing tables. The exchange happens periodically.
- neighbouring nodes also estimate the cost of link connecting them.
  - The cost may be number of hops.
  - It may the queue length at the node of the packets to be forwarded to second node.
- using the links cost and neighbouring routing tables, each node updates its own routing tables.
  - If in received routing table, neighbouring node k is at distance (cost) $d(k, j)$ from node j (destination).

- Let the current node be i.
- Node i computes for all neighbouring nodes distance $d(i,j) = d(k,j) + Cost(i,k)$. The Routing table forwards the packets for j to the neighbour node through which the computed distance is minimum.

- If exchange packet is not received from a neighbouring node, the distance is assumed ∞ to it.

- *Once exchange packets are received, all routing table entries are computed afresh from received information.* Old routing tables are removed.

Actual implementation of algorithm is different. The above rules are for illustration of concept only.

Example

consider the network shown in Fig.7.

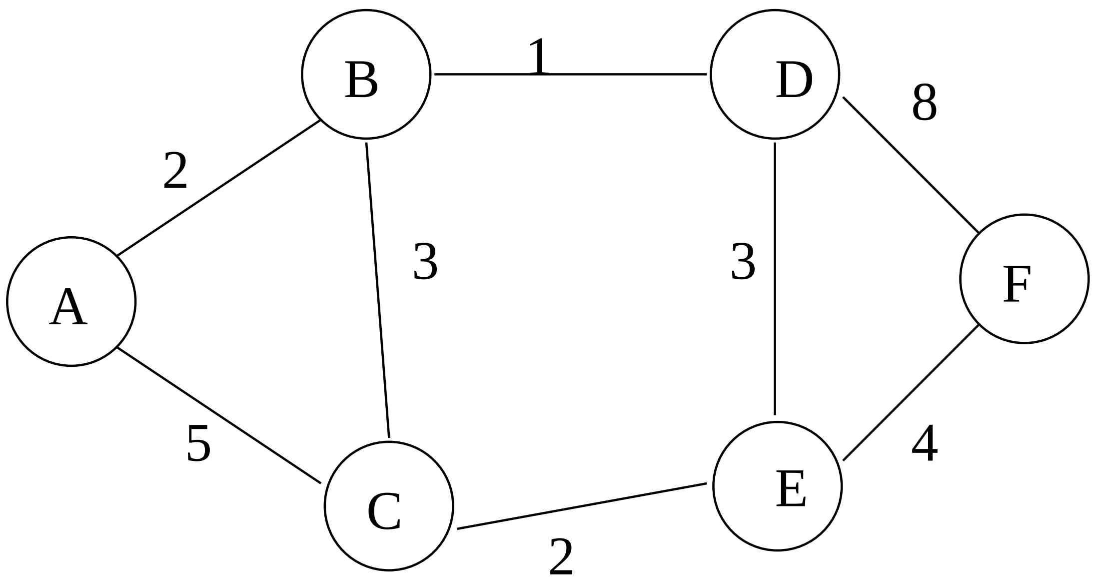

Fig.7

We will look at the routing table entries for destination F only. Entries for other destination also propagate similarly.

After bootup, none of the node have entry for dest F.

	A		B		C		D		E		F	
0	-	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	F	8	F	4	-	-
2	-	-	D	9	E	6	E	7	F	4	-	-
3	B	11	D	9	E	6	E	7	F	4	-	-
4	B	11	D	9	E	6	E	7	F	4	-	-

At bootup each node is not aware of other nodes, so routing tables are empty.

Each routing table entry is (cost, next node). Destination is F as assumed.

From routing table entries for Destination F, tracing minimum cost past from A-F : ABDEF

- same as given by Djikshtra's Algo.

Let Link AB fails. A & B will not exchange the packets

$\rightarrow$ Cost of AB will become ∞.

Tracing the changes for entries for node F from stablized routing tables.

	A		B		C		D		E		F	
4	B	11	D	9	E	6	E	7	F	4	-	-
5	C	11	D	9	E	6	E	7	F	4	-	-
5	C	11	D	9	E	6	E	7	F	4	-	-

Even after fault, the routing tables adjust to new optimal solution.