Network Layer

* The network layer is concerned with getting the packets from the source all the way to destination.

* This may require many routers and intermediate hops along the way from source to destination.

* Network layer is the lowest layer that deals with end-to-end transmission.

→ Functions of Network layer :-

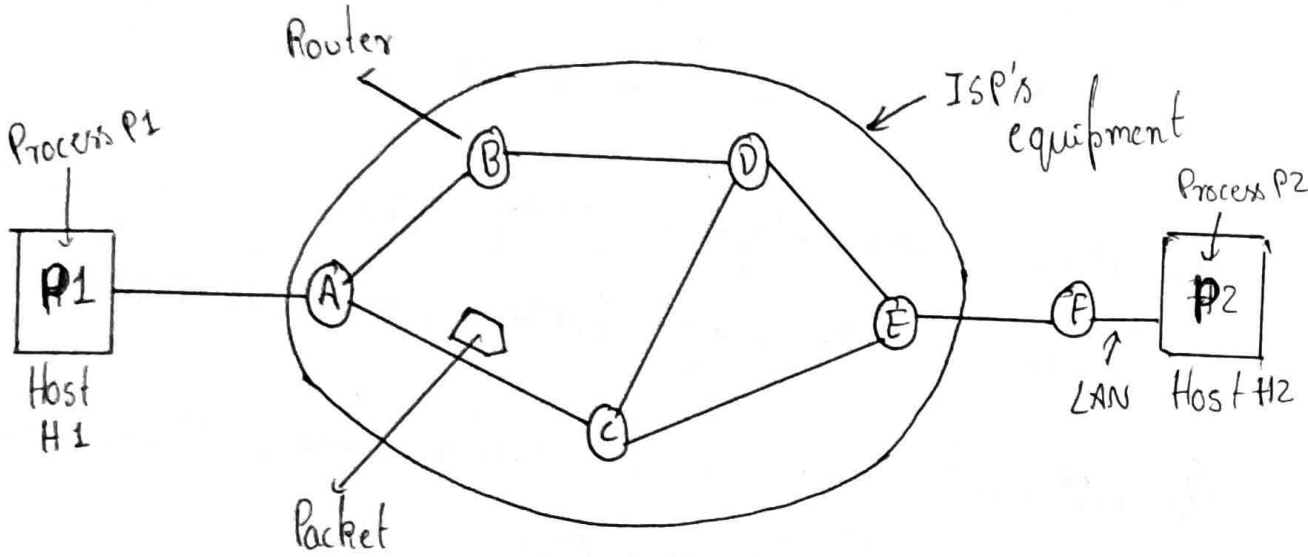
- Learn about Topology of Network
- Compute appropriate path for packet
- Chosing routes avoiding overloading of channels
- Coordinating traffic flow accross networks and managing network utilization.

→ Autonomous Systems :-

Source and destination are in two different independently operated networks.

⇒ Network Layer Design Issues :-

(i) Store and Forward Packet Switching :-



- The host with a packet to send transmits it to the nearest router, either on its own LAN or a Point-to-Point link to ISP.
- The Packet is stored there until it has fully arrived and the link has finished its processing by verifying the Checksum.
- Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered.

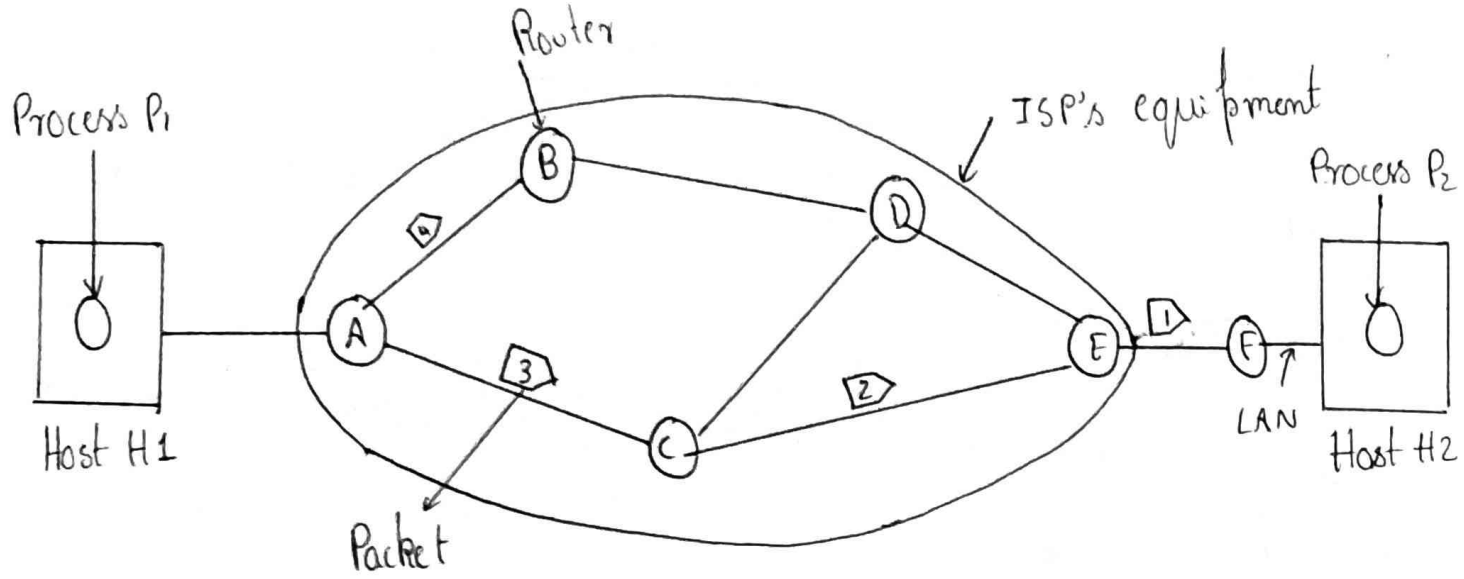
(ii) Services Provided to Transport Layer:-

The services should be carefully designed with the following goals:-

* The services should be independent of router technology
* The transport layer should be sheilded from number, type and topology of routers present.
* The network layer address made available to transport layer should use a uniform numbering plan, even across LAN's and WAN's.

(iii) Implementation of Connectionless service:-

* The packets are injected into network individually and they are routed independently of each other.
* No advance setup is needed.
* The packets are frequently called Datagrams and the network is called Datagram Network.
* Each packet carries the source and destination addresses.



A's Table

Dest	Line
A	–
B	B
C	C
D	B
E	C
F	C

A's Table (later)

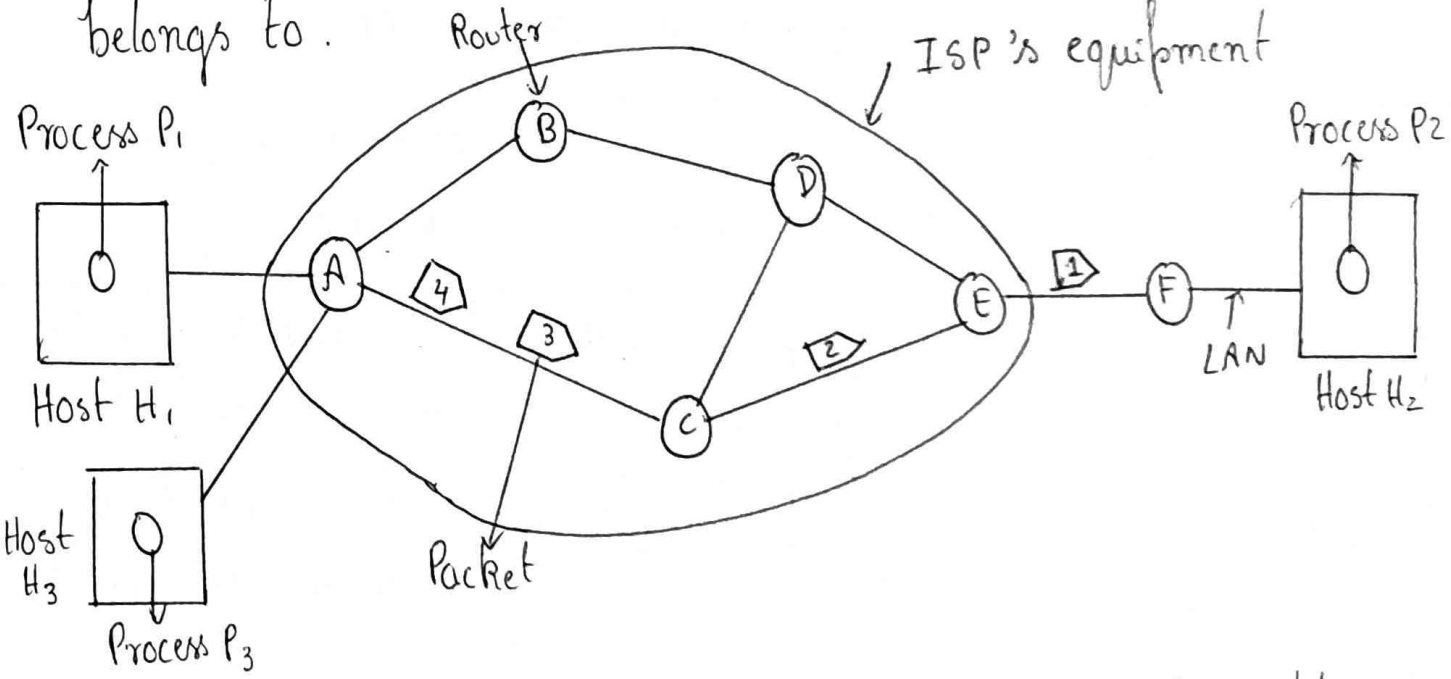
Dest	Line
A	-
B	B
C	C
D	B
E	B
F	B

(*) The first three packets are sent through route ACE but during transmission of 4th packet 'A' got to know of a traffic jam in ACE so it decided to change the route via B, so there is a change in the Routing Table of A.

(*) The algorithm that manages these tables and takes routing decision is called Routing Algorithm.

(iv) Implementation of Connection Oriented Service:- (Virtual Circuit Networks)

* The main idea is to avoid having to chose a new route for every packet sent.
* Instead, when a connection is established, a route from source to destination machine is chosen as part of connection setup and stored in table of routers.
* This route is used for all trafting flowing over the connection.
* Each packet carries an identifier telling where it belongs to.



A's Table

H1	1
H3	1

In

C	1
C	2

Out

C's Table

A	1
A	2

E	1
E	2

E's Table

C	1
C	2

E	1
E	2

⊛ When both H1 and H3 want to transmit at the same time they both want to use connection identifier 1

⊛ 'A' can differentiate among the two but 'c' cannot so A gives connection identifier 2 in the outgoing traffic.

⇒ Comparision of Virtual Circuit and Datagram Network

Issue	Datagram Network	Virtual Circuit Network
Circuit Setup	Not Needed	Required
Addressing	Each packet contains full source & dest address	Each packet contains a short VC Number
State Information	Routers do not hold state information	Each VC requires router table space per connection
Routing	Each packet routed independently	Route chosen when VC is setup and all packets follow
Effect of Router failures	None, except for packet loss during crash	All VC's that passed through that router terminated
Quality of Service and Congestion Control	Difficult	Easy if enough resources can be allocated in advance for each VC

Some other differences :-

(i) Virtual Circuit requires more setup time while Datagram requires more Parsing time.

(ii) Destination addresses in Datagram are longer than circuit numbers in Virtual-circuit networks.

(iii) Datagram network requires more table space in router memory because each router needs entry for all possible destinations.

(iv) Datagrams also allow the routers to balance traffic throughout the network, since the routes can be changed partway through a long series of transmission

⇒ Routing Algorithms :-

The routing algorithm is that part of network layer software responsible for deciding which output line an incoming packet should be transmitted on.

→ Desirable Properties in Routing algorithm :-

- Correctness
- Simplicity
- Robustness
- Stability
- Fairness
- Efficiency.

⇒ Two types of Routing Algorithm :-

(i) Adaptive (or) Dynamic :-

Change their routing decisions to reflect the changes in topology and sometimes changes in traffic as well

(ii) Non-adaptive (or) Static :-

Do not change their routing decisions on any measurements (or) estimates of current topology and traffic.

⇒ Optimality Principle :-

⊛ It states that " If a router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along same route.

→ Sink Tree :-

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

⊛ Note :- Sink Tree is not Unique, other trees with the same path length may exist.

⊛ If we allow all of the possible paths to be chosen, the tree becomes a Directed Acyclic Graph.

⊛ Since Sink Tree is indeed a tree, it does not contain any loops, so each packet will be delivered within a finite and bounded number of hops.

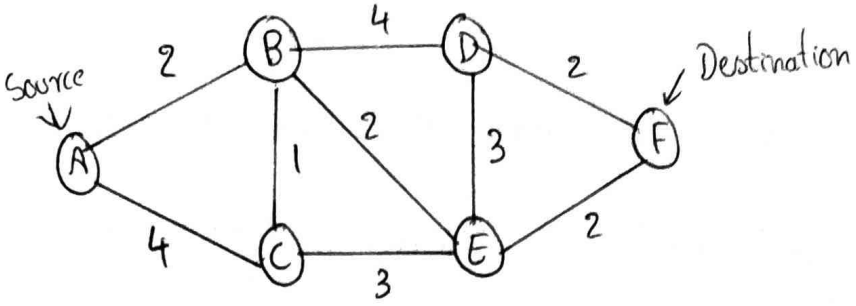
⇒ Shortest Path Algorithm :-

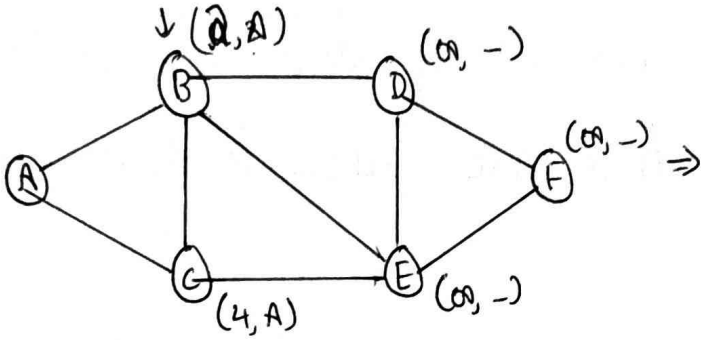
→ Different metrics for shortest path :-

Number of hops, geographic distance in km, delay blw each hop, bandwidth, average traffic, communication cost etc...

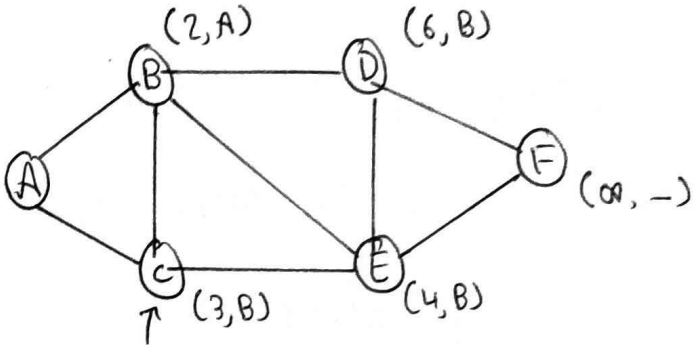
⇒ Dijkstra's Algorithm :-

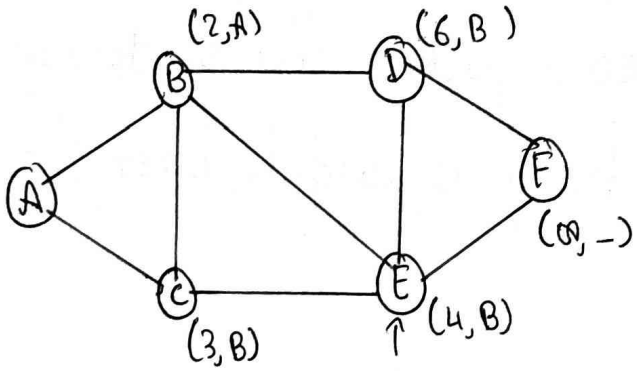
* Dijkstra's algorithm finds the shortest path blw a source and destination in network.
* Each node is labelled with distance from source node along best path node
* The distance or weights of edges must be non-negative
* Initially no paths are known to all paths are labelled with ∞.
* As algo proceeds the optimal path are found changing the labels of node with better path
* Applicable for both directed and undirected graph


Source
Destination
A
B
C
D
E
F
2
4
1
2
3
2
4
3
2

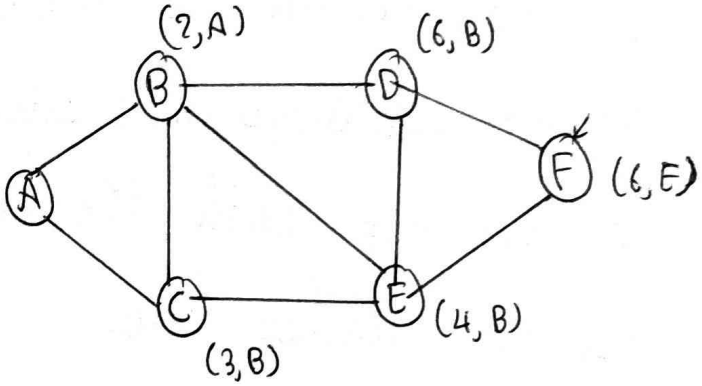

(2,A)
(∞, –)
(∞, –)
(4, A)
(∞, –)

⇒


(2,A)
(6, B)
(∞, –)
(3,B)
(4, B)


(2,A)
(6, B)
(∞, –)
(3, B)
(4, B)

⇒


(2,A)
(6, B)
(6, E)
(4, B)
(3, B)

⇒ Flooding Algorithm :-

⊛ Flooding is a simple local routing technique in which every incoming packets is set out on every outgoing adjacent line except one it arrived from.

⊛ Flooding generates a vast number of duplicate packets there must be some measures that are taken to eliminate the duplication.

⊛ One measure to reduce duplication is to have a hop counter contained in header of each packet that is decremented at each hop, with the packet being discarded when the counter reaches zero.

⊛ Ideally, the hop counter should be intialized to length of the path from source to destination @ the worst case (full diameter of network)

⊛ But, this technique is not ideal because the hop count grows and routers duplicate packets.

⊛ A better technique is to have the routers keep track of which packets have been flooded, to avoid sending them again the second time.

(*) Each packet may be assigned with a sequence number and the source router to check if the packet have already been flooded.

(*) Each router then needs a list per source router telling which sequence numbers originating at source have already been seen.

(*) If incoming packet is on list, then it is not flooded.

→ Important uses of flooding :-

(i) Broadcasting :- Because it ensures that packet is delivered to every node in network.

(ii) Military Network Located in War-Zone :-

Since, flooding is tremondously robust i.e even if large number of routers are blown to pieces, flooding will find a path if one exists to get packet to destination.

(iii) Used as a building block for other routing algorithms since it requires only little way of setup.

(iv) It can also be used in Distributed Systems.

⇒ Distance Vector Routing :-

⊛ A distance vector routing algorithm is sometimes called the Bellman - Ford Algorithm.

⊛ It operates by having each router maintain a table (i.e a vector) giving the best known distance to each destination and which link to use to get there.

⊛ These tables are updated by exchanging information with the neighbours. Eventually, every router knows the best link to reach each destination.

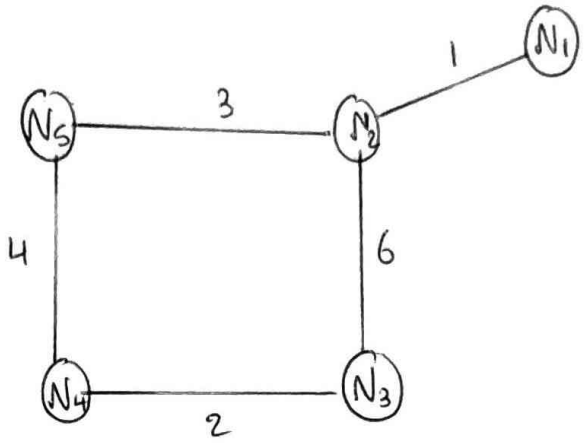
⊛ This algorithm was the original ARPANET routing algorithm and was also used in Internet under the name RIP.

⊛ Initially, each of the router within the network must know how many router participants are there in network's topology.

⊛ The router participants are identified by sending hello packets to all the intermediate routers and in turn acknowledgement is given by active users who are ready to take part in communication.

(*) Each routing table has 3 standard entries namely Destination, Distance and next hop.

Ex:-



Initially :-

N_1

Dest	Dist	Next Hop
N_1	0	N_1
N_2	1	N_2
N_3	∞	-
N_4	∞	-
N_5	∞	-

N_2

Dest	Dist	Next Hop
N_1	1	N_1
N_2	0	N_2
N_3	6	N_3
N_4	∞	-
N_5	3	N_5

N_3

Dest	Dist	Next Hop
N_1	∞	-
N_2	6	N_2
N_3	0	N_3
N_4	2	N_4
N_5	∞	-

N_4

Dest	Dist	Next Hop
N_1	∞	-
N_2	∞	-
N_3	2	N_3
N_4	0	N_4
N_5	4	N_5

N_5

Dest	Dist	Next Hop
N_1	∞	-
N_2	3	N_2
N_3	∞	-
N_4	4	N_4
N_5	0	N_5

Finally after all iterations :-

N_1

Dest	Dist	Next Hop
N_1	0	N_1
N_2	1	N_2
N_3	7	N_2
N_4	8	N_2
N_5	4	N_2

N_2

Dest	Dist	Next Hop
N_1	1	N_1
N_2	0	N_2
N_3	6	N_3
N_4	7	N_5
N_5	3	N_5

N_3

Dest	Dist	Next Hop
N_1	7	N_2
N_2	6	N_2
N_3	0	N_3
N_4	2	N_4
N_5	6	N_4

N_4

Dest	Dist	Next Hop
N_1	8	N_5
N_2	7	N_5
N_3	2	N_3
N_4	0	N_4
N_5	4	N_5

N_5

Dest	Dist	Next Hop
N_1	4	N_2
N_2	3	N_2
N_3	6	N_4
N_4	4	N_4
N_5	0	N_5

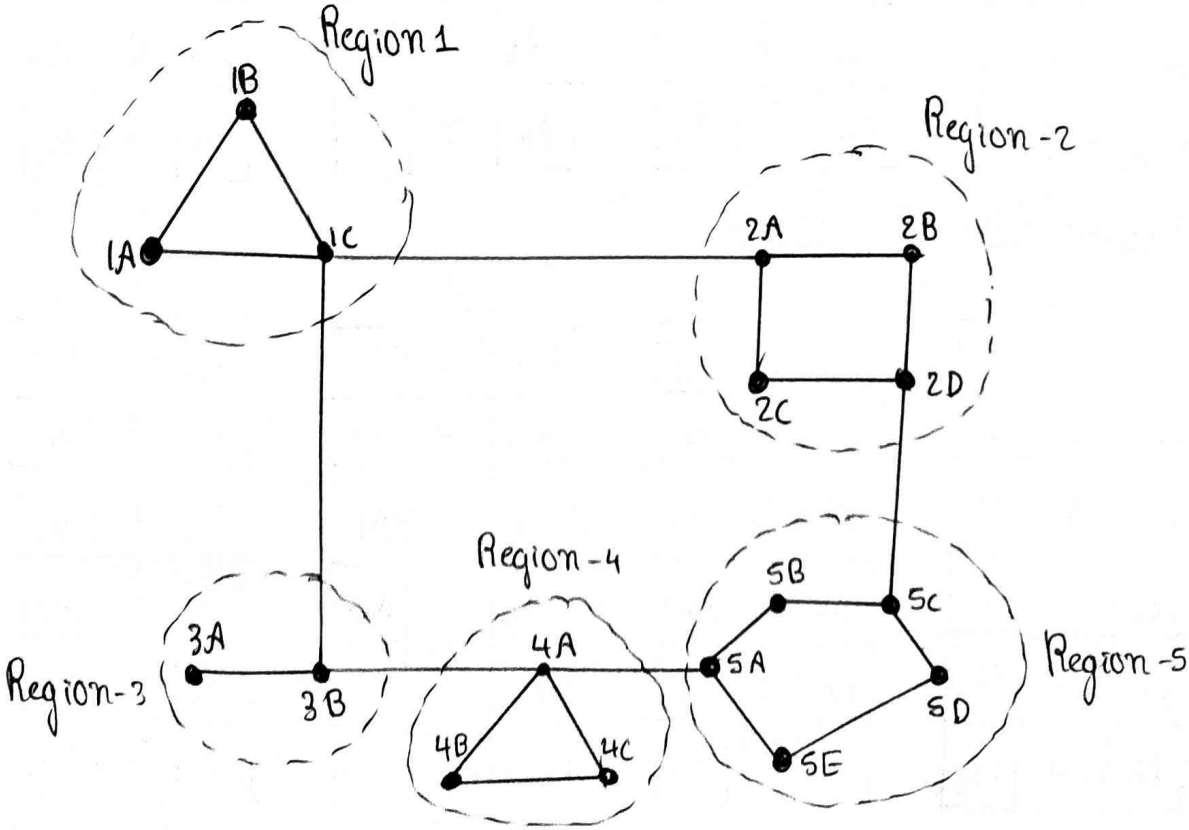
⇒ Hierarchial Routing within a Network:-

* As the networks grow in size, the router routing tables grow proportionally. Not only is router memory is consumed but also more CPU time is needed to scan them and more bandwidth is required to send status report.

* Hierarchial routing is used to overcome this issue.

* In hierarchial routing, the routers are divided into what we will call regions or areas.

* Each router knows how to route packets to destination within its own region but does not know anything about internal structure of other regions.



Full Table for 1A

Dest.	Line	Hop
1A	-	-
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

Heirarchial Table for 1A

Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

* The full routing table of 1A has 17 entries, but in heirarchial routing it has only 7 entries.
* As the ratio of number of regions to the number of router per region grows, the saving in table space increases.

* Unfortunately, these gains are not free. There is a penalty to paid : increased path length.
* For example, the best route from 1A to 5C is via region 2, but with heirarchial routing, all traffic to region 5 goes via region 3, because that is better for most routers in region 5.

⊛ For huge networks, a two-level hierarchy may be insufficient; it may be necessary to group the regions into clusters, the clusters into zones, the zones into groups and so on.

⊛ Kamoun and Kleinrock Formula:-

For N routers, optimal number of levels

$$= \ln N$$

∴ Total number of entries per router

$$= e \ln N$$

→ They have also shown that the increase in effective mean path length caused by hierarchical routing is sufficiently small that is usually acceptable.

→ Link - State Routing (Dijkstra's Algorithm):-

* The idea behind Link state is that when a router is initialized, it determines the link cost on each of its network interfaces.

* The router then advertises this set of link costs to all other routers in the internet Topology.

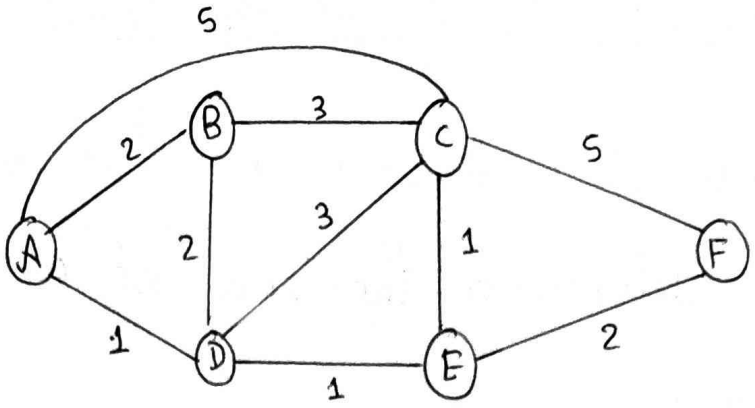
* Because each router recieves the link cost of all routers in configuration each router can construct the topology of entire configuration and then calculate shortest path to destination.

* Dijkstra's Algorithm is used to find the shortest path from a single source to all vertices.

* It is called Link- State routing because every router shares its info and knowledge about rest of the other connected routers.

* This algorithm gives optimal solution to graphs having positive edges only.

Ex:-



Step	Node Set 'N'	D(B) P(B)	D(C) P(C)	D(D) P(D)	D(E) P(E)	D(F) P(F)
1	A	2, A	5, A	(1, A)	∞, –	∞, –
2	A, D	(2, A)	4, D	–	2, D	∞
3	A, D, B	–	4, D	–	(2, D)	∞
4	A, D, B, E	–	(3, E)	–	–	4, E
5	A, D, B, E, C	–	–	–	–	(4, E)
6	A, D, B, E, C, F	–	–	–	–	–

Shortest Paths :-

A → B = 2

A → C = 3

A → D = 1

A → E = 2

A → F = 4

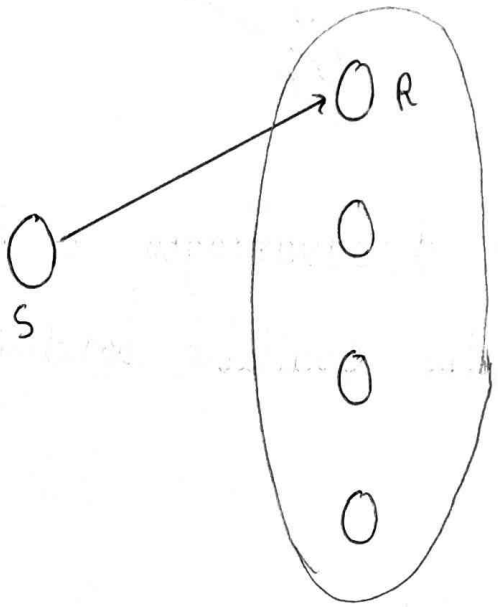
D → Distance from source to destination in a particular iteration

P → Previous node traversed to reach destination.

→ Some other types of Routing :-

① Unicast Routing :- (Ex :- Sending an email)

It is the one to one communication adopted between a single sender and single reciever.

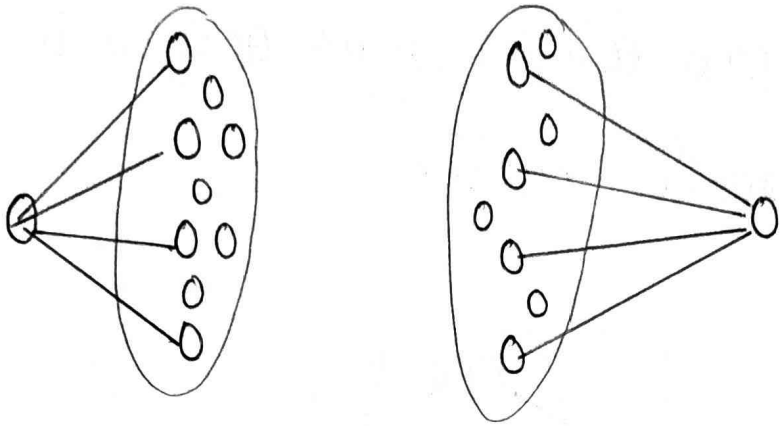


(ii) Multicast Routing :-

(*) Sending messages to groups that are numerically large in size but smaller compared to Network Size is called Multicasting and the algorithm used is Multicast Routing.

(*) There may be another situation where Multiple senders are sending messages to a single recepient.

(*) Multicast lets servers direct single copies of data st that are then simulated and routed to hosts that request



(*) One of the main disadvantages is that as the number users increases, the available bandwidth for each user decreases

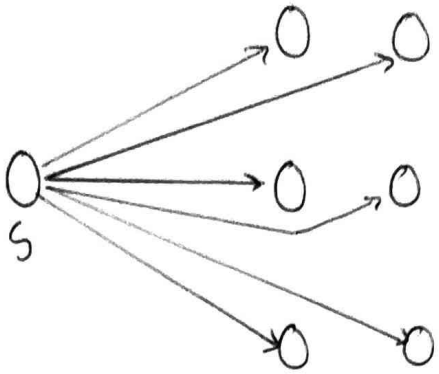
Ex:- Online Streaming

(3) Broadcasting :- (Ex:- Radio @ FM)

(*) The transmission of data from one host to all other nodes connected in a network is called Broadcasting.

(*) There is maximum bandwidth utilization in case of broadcast since packets are given to all nodes even if some nodes are not interested.

(*) They may create large amount of network Traffic so it is slower.



Broadcast :- One to all Mapping

④ Anycast Routing :-

(*) In anycast routing, a packet is delivered to the nearest member of a group.

(*) Sometimes these nodes provide a service such as time of day (or) content distribution for which it is getting the right information all that matters. It does not matter which node is contacted.