Module-3

The Network Layer.

Introduction.

The nlw layer gets segments from the transport layer, encapsulates each segment into datagram (nlw layer packet) & then sends it to nearby router, in the receiving side the nlw layer gets datagram from a router, extracts transport layer segments & delivers to transport Layer.

What's Inside a Router?

Router Components

1. I/p Ports

It performs physical layer fun' of terminating an incoming physical link at a router, & performs link-layer functions needed to incorporate with the link layer.

2. Switching fabric → Connects i/p ports with o/p ports.

3. O/p ports → receives packets from S. fabric & for transmit these packets by performing physical & link-layer functions.

4. Routing Processor.

i) Executes routing protocols

ii) Maintains routing tables & link state info'.

iii) Computes forwarding table for the router.

Router forwarding plane:

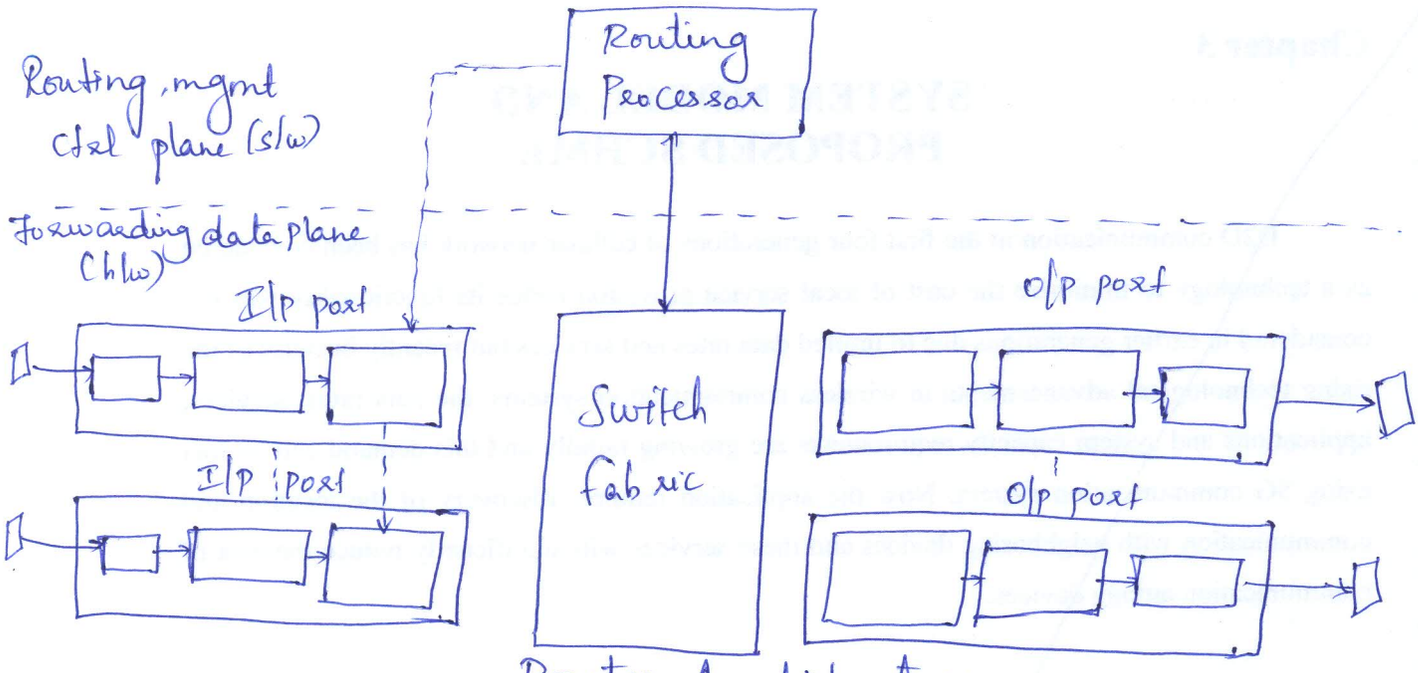
I/P port + Switching fabric + o/p port =

Router forwarding plane (hlw)

↓

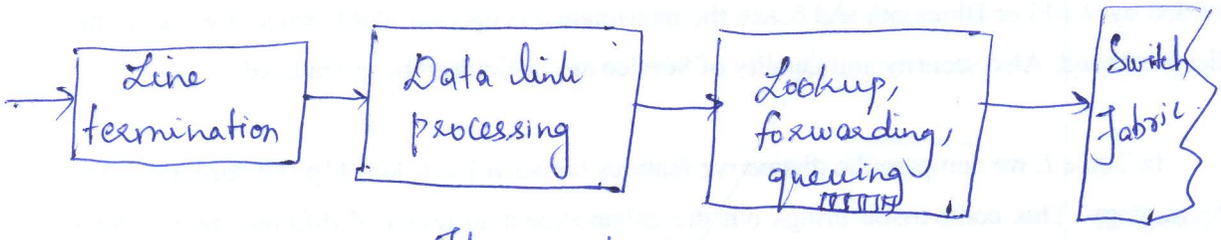
Purchased merchant-silicon chips / own hlw designs.

Router ctrl plane - Routing Processor (s/w).



Router Architecture

Input Processing



I/p port processing.

The forwarding table is computed & updated by the routing processor, with a shadow copy typically stored at each i/p port, thus avoiding a centralized processing bottleneck.

Switching -

Switching Fabric → heart of the router, through this fabric packets are actually switched from i/p port to the o/p port.

Switching Techniques:

i) Switching via m/y

ii) Switching via a bus

iii) Switching via an interconnection - n/w.

i) Switching via m/y:

Used in traditional computers, where i/p & o/p ports communicate through CPU (routing processor).

The i/p port will 1st send a packet to the processor & from the processor it will be directed to the o/p port.

Only one packet can be send at a time.

ii) Switching via a bus

I/p port transfers packets directly to the o/p port over a shared bus, by attaching a label to the packet, indicating the o/p port.

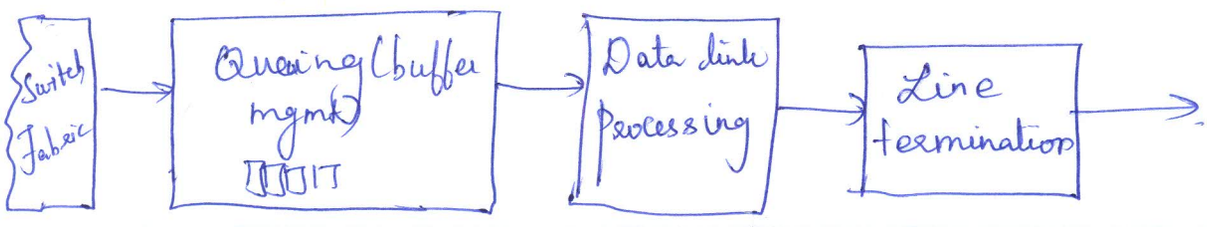
If multiple packets arrive to the router at the same time each at a different i/p port, all but one must wait since only one packet can cross the bus at a time.

iii) Switching via an interconnection n/w.

A crossbar switch is an interconnection n/w consisting of 2N buses that connect N i/p ports to N o/p ports.

Crossbar n/ws are capable of forwarding multiple packets in parallel.

o/p Processing



o/p port processing.

Packets have been stored in the o/p port's m/y & transmitted over the o/p link.

Packet Scheduler → at the o/p port must choose one packet among those queued for transmission, based on first-come-first-served (FCFS) scheduling / weighted Fair queuing (WFQ) etc. Packet scheduling plays a crucial role in providing quality-of-service guarantees. link speed is also considered.

If there is not enough m/y to buffer an incoming packet, decision must be made to either drop the arriving packet (drop-tail) or remove one or more already-queued packets.

It is better to drop a packet before the buffer is full (active queue mgmt (AQM) algo) → Random Early Detection (RED)

Head-of-the-line (HOL) Blocking → i/p queued switch - a queued packet in an i/p queue must wait for transfer through the fabric (even though its o/p port is free) bcoz it is blocked by another packet at the head-of-the line.

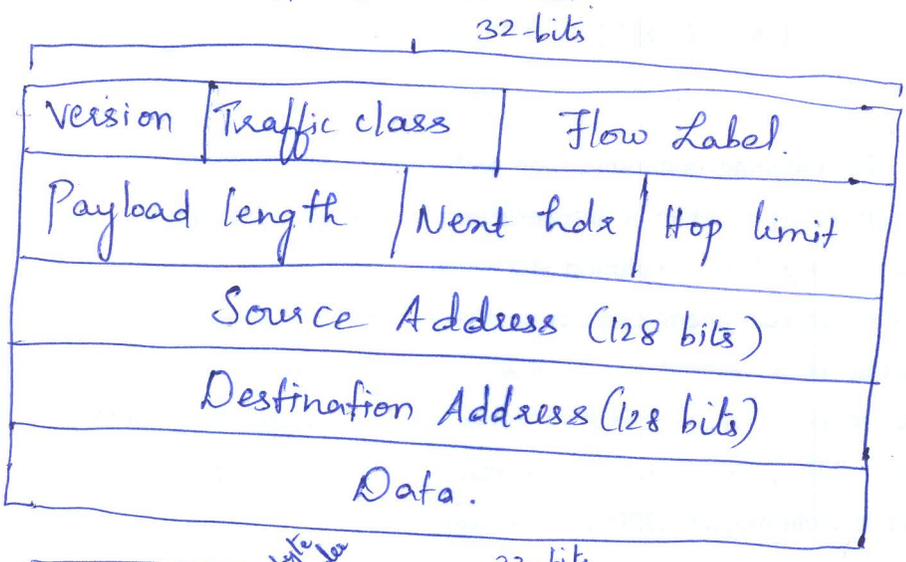
The Routing ctrl plane.

→ Fully resides & executes in a routing processor within the router.

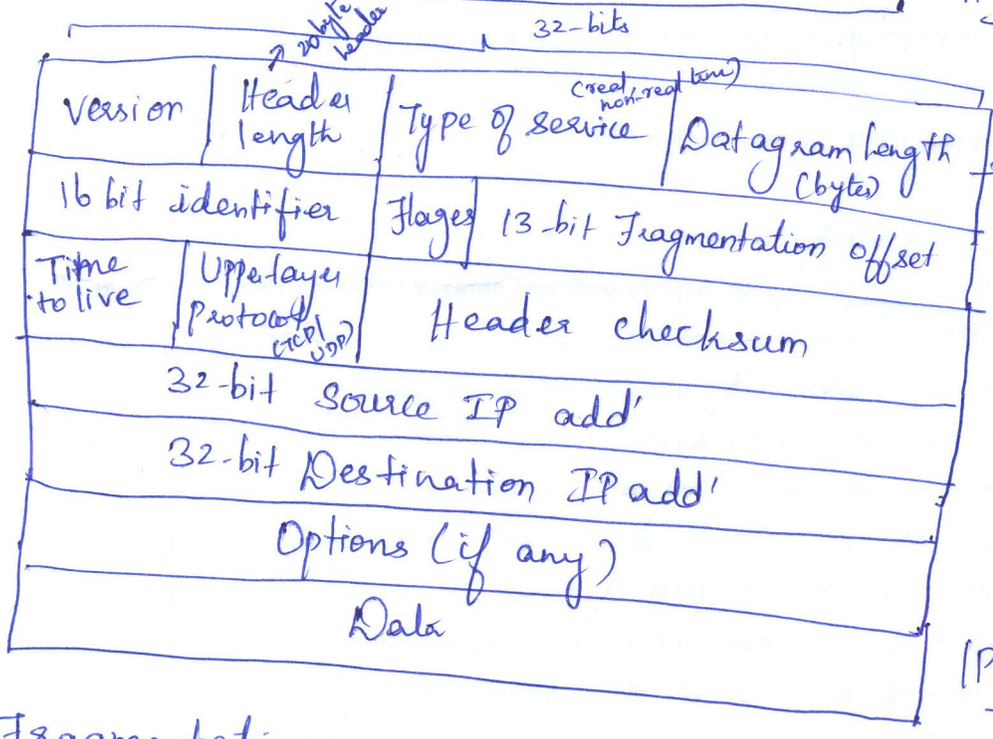
The nlw-wide routing ctrl plane is thus decentralized with different pieces executing at different routers & interacting by sending ctrl msgs to each other.

IPv6

Inorder to fulfill the need for large address space, IPv6 has been introduced.



IPv6 Datagram format



IPv4 Datagram format

Fragmentation:

Converting IP datagram, (i.e. splitting & adding header) into frame for link-layer is

Reassembly: Fragments need to be reassembled before they reach the transport layer at the destination.

IP address - 32 bit long. 2^{32} → possible IP add' (ie) ≈ 4 billion

IPv6 Datagram format.

i) Expanded addressing capabilities: The IP address size has been increased from 32 to 128 bits, inorder not to ran out of IP add'. In addition to unicast & multicast add', anycast add' has been introducts to deliver datagram to any one of a group of hosts.

ii) A streamlined 40-byte header: Many IPV4 fields have been dropped or made optional. (IPV4 - 20 byte header)

iii) Flow labeling & priority: Sender may request special handling of packets by labeling them eg: audio & video. Priority. ICMP msgs.

Fields:

1. Version - IP version no'.
2. Traffic class - Provides type of service.
3. Flow label - Identify a flow of datagrams.
4. Payload length - Data + header.
5. Next hdr - Gives the protocol to which the datagrams have to be transmitted.
6. Hop limit - The count will get decrement once the datagram reaches a router, ..
7. Source & destination address - 128 bit.
8. Data - Payload portion that needs to be transmitted to the destination.

Not in IPV6

i) Fragmentation & Reassembly: Doesn't allow in intermediate routers, rather sends ICMP msg saying that "Packet size is too big" to the sender.

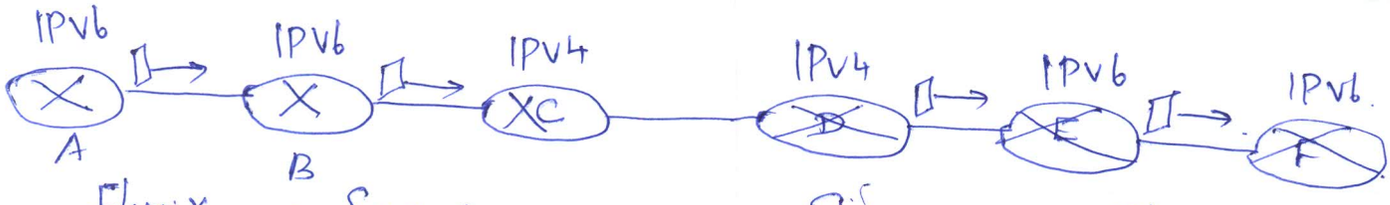
ii) Header checksum: As transport layer is providing checksum, it is not needed in nlw layer.

iii) Options: Instead next hdr field is present.

Transitioning from IPV4 to IPV6.

Pblm: How will the public Internet, which is based on IPV4, be transitioned to IPV6. i.e. how IPV6-capable s/m's communicate with IPV4 s/m's.

1. Dual-stack approach 2. Tunneling.



A dual-stack approach.

Data field of IPV6 datagram is copied to datafield of IPV4 datagram. The add' info in IPV6 will be lost.

Physical View

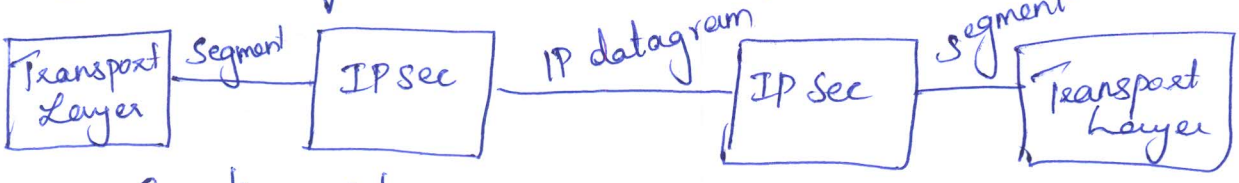
Flow: X	Source: B	Source: B	Flow: X
Source: A	Dest: E	Dest: E	Source: A
Dest: F			Dest: F
data	Flow: X	Flow: X	data
A to B: IPV6	Source: A	Source: A	E to F: IPV6
	Dest: F	Dest: F	
	data	data	
	B to C: IPV4	D to E: IPV4	
	(encapsulating IPV6)	(encapsulating IPV6)	

Logical view.

Tunneling

The IPV6 node in the sending side of the tunnel will add the entire IPV6 datagram into the payload (data) field of the IPV4 datagram. So, add info fields in IPV6 will remain unlost.

IP Security → Secure N/w layer protocol.



Sending side.

Receiving side.

IPSec in sending side - encrypts the segment & encapsulates resulting payload into IP datagram.

IPSec in receiving side - decrypts IP datagram & send the segment to the transport layer.

IPsec is compatible with IPv4 & IPv6.

Services

i) Cryptographic agreement: mechanisms that allow the 2 communicating hosts to agree on cryptographic alg's & keys

ii) Encryption of IP datagram payloads.

iii) Data integrity: → allows the receiver to verify the received payload have not been modified.

iv) Origin authentication → done by source IP add'.

Routing Algorithms

Given a set of routers, with links connecting the routers, a routing alg' finds the optimal path from source to the destination.

I. Classification of Routing Algorithms

1. Global routing algorithm: computes the least-cost path blw source to destination using global knowledge about the nlw, referred to as Link-State algorithms.

2. Decentralized routing algorithm: the shortest-path is calculated in an iterative, distributed manner, called as Distance Vector (DV) algorithms.

II. 1. Static Routing algorithms. Routes change very slowly (a human manually edit' a router's forwarding table)

2. Dynamic Routing algorithms → Responsive to nlw changes.

III. 1. Load-Sensitive algo' → link costs vary dynamically to reflect the congestion level.

2. Load-insensitive algo' - link costs does not explicitly reflect the current level of congestion.

The link-state (LS) Routing Algorithm

Dijkstra's algorithm

Computes the least-cost path from one node to all other nodes in the n/w.

D(v): Cost of the least-cost path from the source node to destination v as of this iteration of the alg'.

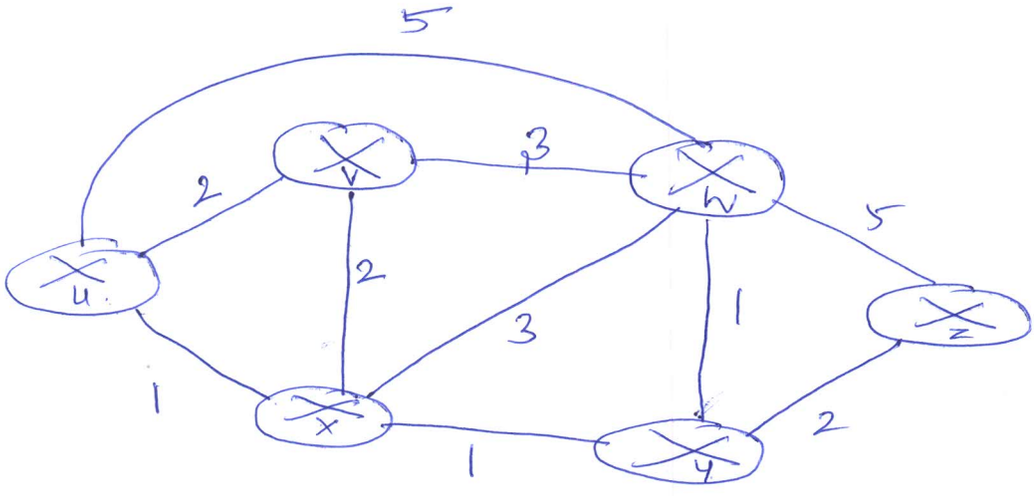
p(v): previous node (neighbor of v) along the current least-cost path from the source to v.

N': subset of nodes.

link-state (LS) alg' for Source Node u.

```
1  Initialization:
2  N' = {u}
3  for all nodes v
4  if v is a neighbor of u
5  then D(v) = c(u,v)        c(u,v) - cost of the path
6  else D(v) = ∞
7
8  Loop
9  find w not in N' such that D(w) is a minimum
10 add w to N'
```

11 update D(v) for each neighbor v of w & not in N':
12 D(v) = min(D(v), D(w) + c(w,v))
13 /* new cost to v is either old cost to v or known
14 least path cost to w plus cost from w to v */
15 until N' = N



Initialization step.

u	v	x	w	y	z.
	2	1	5	∞	∞ → bcoz no direct path

Step	N'	v	w	x	y	z.
0	u	2,u	5,u	1,u	∞	∞.
1	ux	2,u	~~5~~ 4,x		2,x	∞
2	uxy	2,u	3,y			4,y.
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz.					

Complexity — $O(n^2)$.

The Distance-Vector Routing Algorithm

DV alg' is iterative, asychronous & distributed.

Iterative - The process continues until no more info' is enchanged blw neighbors.

Asynchronous - It doesn't require all of the nodes to operate in lockstep with each other.

Distributed - Each node receives some info' from one or more of its directly attached neighbors, performs a calculation & then distributes the result to the neighbors.

Bellman-Ford equation:

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

$d_x(y)$ → shortest path from node x to nod y.

$\min_v$ - is taken over all of x's neighbors.

With DV alg; each node x maintains the following routing info':

→ For each neighbor v, the cost $c(x,v)$ from x to directly attached neighbor, v.

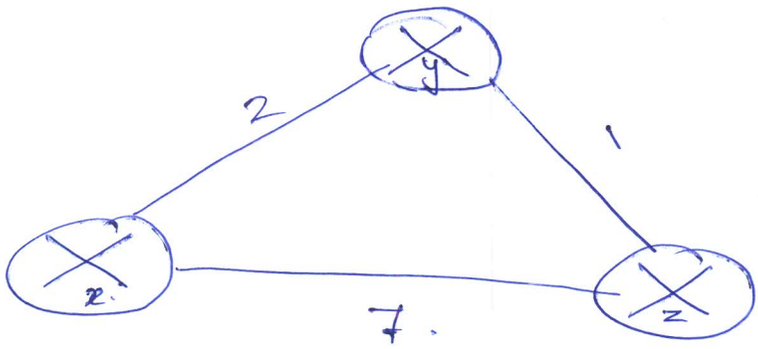
→ Node x's nei. distance vector, that is, $D_x = [D_x(y): y \text{ in } N]$, containing x's estimate of its cost to all destinations, y in N

→ The distance vectors of each of its neighbors, that is, $D_v = [D_v(y): y \text{ in } N]$ for each neighbor v of x.

Bellman-Ford-equation to update distance vector

$$D_x(y) = \min_v \{c(x,v) + D_v(y)\} \text{ for each node } y \text{ in } N.$$

1 Initialization:
2 for all destinations y in N:
3 $D_x(y) = c(x,y)$ /* if y is not a neighbor then $c(x,y) = \infty$ */
4 for each neighbor w
5 $D_w(y) = ?$ for all destinations y in N
6 for each neighbor w
7 send distance vector $D_x = [D_x(y): y \text{ in } N]$ to w
8
9 loop
10 wait (until I see a link cost change to some neighbor w or
11 until I receive a distance vector from some neighbor w)
12
13 for each y in N:
14 $D_x(y) = \min_v \{ c(x,v) + D_v(y) \}$
15
16 if $D_x(y)$ changed for any destination y
17 send distance vector $D_x = [D_x(y): y \text{ in } N]$ to all neighbors
18
19 forever



Node x table.

Cost to

from	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

Cost to

	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

Cost to

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Node y table

	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Node z table

	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	~~∞~~ 7	~~∞~~ 1	~~∞~~ 0

	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

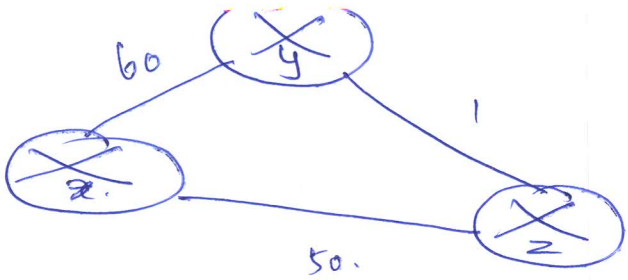
	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Distance - Vector Algorithm.

$D_x(x) = 0$

$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} = \min\{2+0, 7+1\} = 2.$

$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} = \min\{2+1, 7+0\} = 3.$



At t1 the cost blw x & y, x & z changes.

y send sends packets through z, but z will send back to z

As y & z didn't update this. This is called as routing loop.

At pow time t2 y & z will update each others new cost.

Complexity $O(|V|.|E|)$ Bellman . $O(V^{12})$

Comparison:

LS	DV.
Each node need to know the cost of each link in the nlw. (Msg Complexity).	Need to know only when there is least path
2. Speed of Convergence $O(\|N\|)^2$	Suffers from count-to-infinity pblm. $O(\|V\|.\|E\|)$ → Bellman-Ford)
3. Robustness If a router fails, then every node have to calculate its own routing table (forwarding table)	A node can advertise incorrect least-cost paths to all of its destinations

Hierarchical Routing.

Issues in Routing.

i) Scale → if the routers grow.

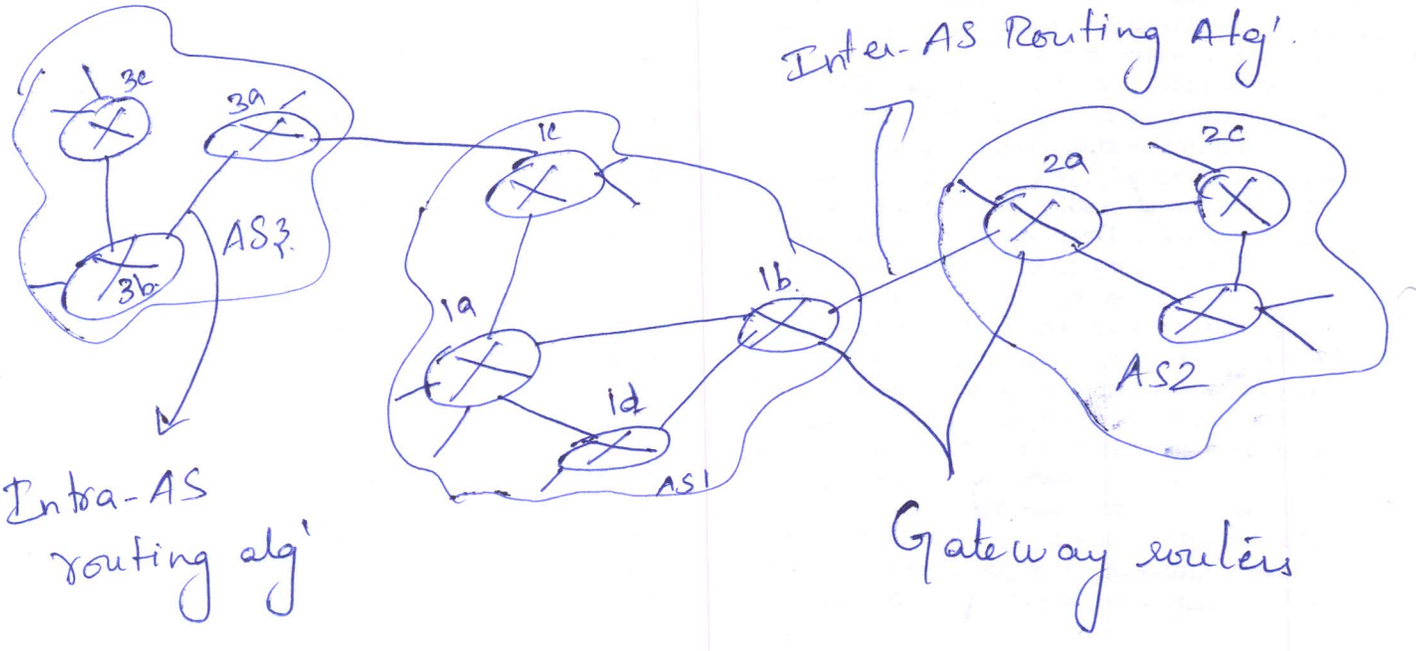
ii) Administrative autonomy → by an organization

→ these issues have been solved by organizing routers into autonomous s/m's (ASs), with each AS consisting of a group of routers that are typically under the same administrative ctrl.

→ All routers within a AS will run same routing alg: referred as intra-autonomous s/m routing protocol.

Gateway routers - forwards packets from one AS to other.

→ Inter-AS routing Protocol → Communication blw 2 AS.



An Eg' of interconnected autonomous s/m's.

Id have to send one packet to subnet x.

2 possibilities:

i) 1c

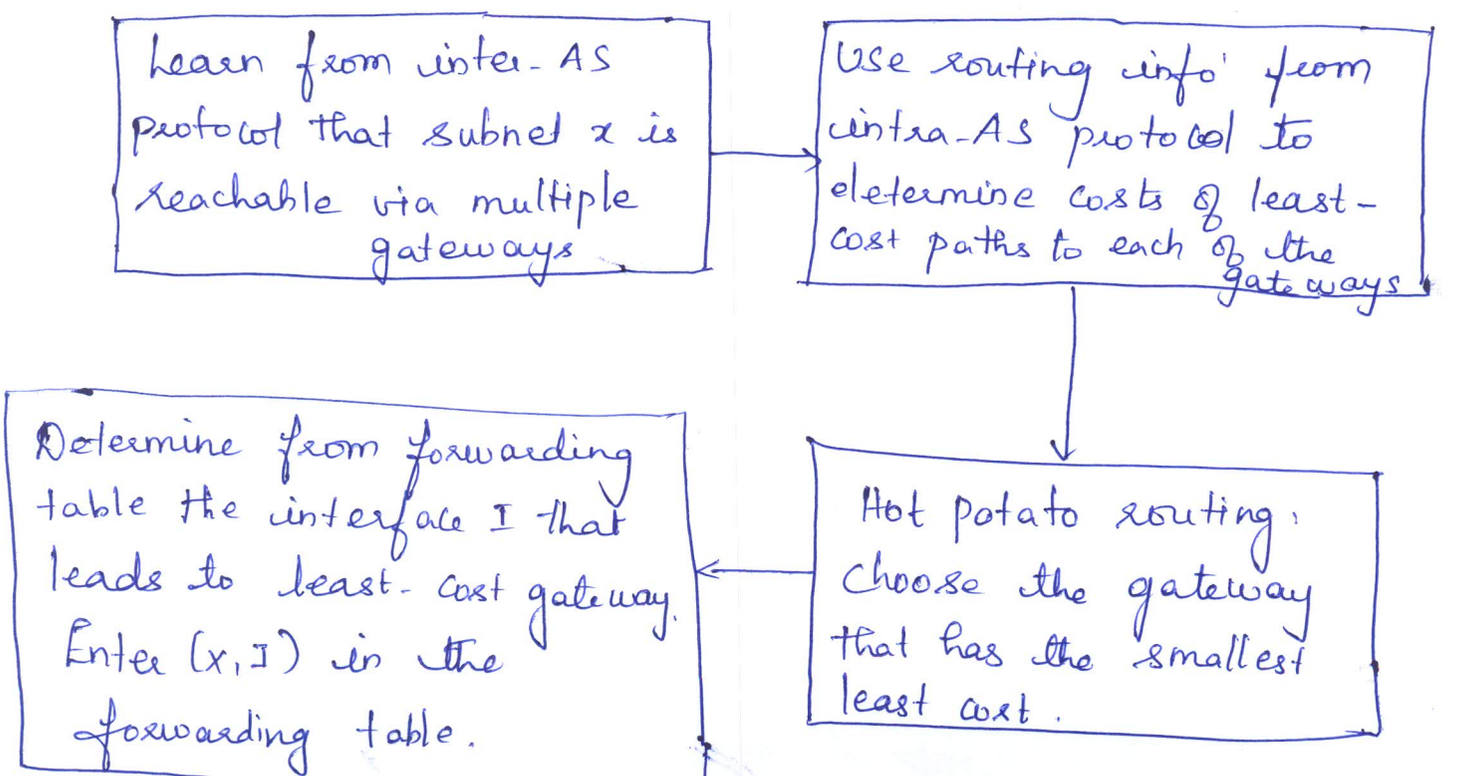
ii) 1b.

Approach -> Hot-potato Routing.

Id will check using intra-AS routing protocol, whether 1c / 1b is having shortest path from Id.

Once the path is chosen, router Id adds an entry for subnet x in its forwarding table.

Now AS1 will learn whether by AS2 / AS3 subnet x is reachable. If AS1 finds through AS2 x is reachable, then it forward this info to AS3, such that if AS3 wants to send anything to A X it will send via AS1.



Steps in adding an outside-AS destination in a router's forwarding table

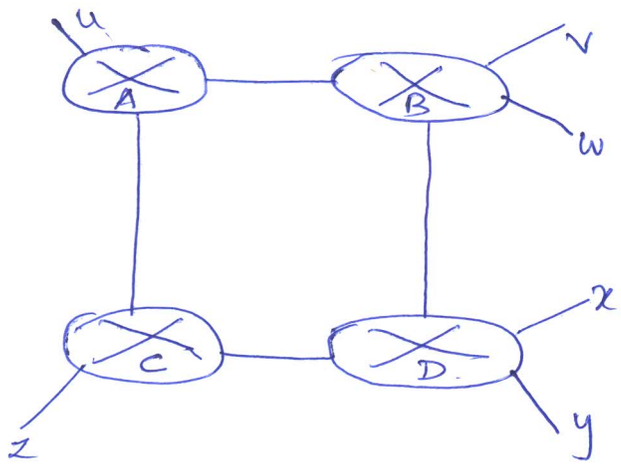
Routing in the Internet

Intra-AS Routing in the Internet: RIP [Routing info' Protocol]

Used to determine how routing is performed within an autonomous system (AS) → also called as Interior gateway protocols.

RIP is a DV protocol.

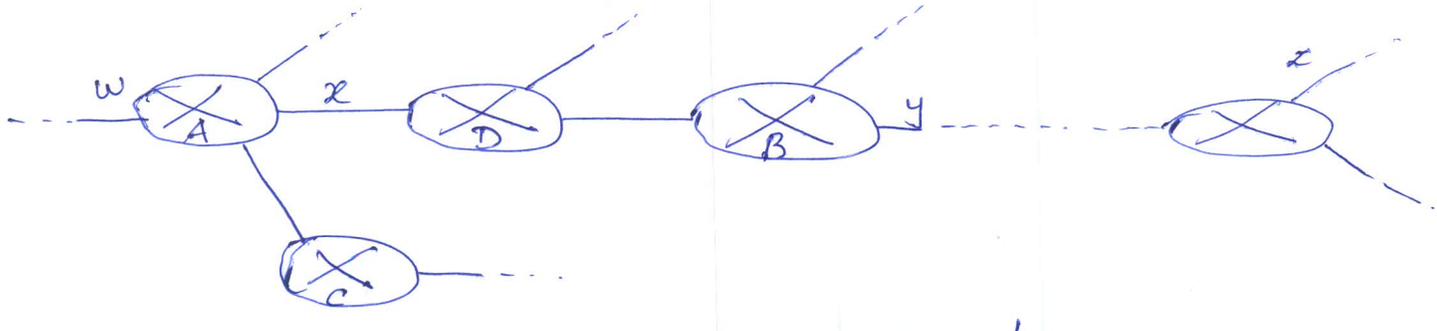
Hop - no'. of subnets traversed along the shortest path from source router to destination subnet, including the destination subnet.



Destination	Hops
u	1
v	2
w	2
x	3
y	3
z	2

No' of hops from source router A to various subnets

Routing updates are exchanged blw neighbors every 30sec using RIP response msg. (RIP advertisements).



A portion of an autonomous sln.

Destination Subnet	Next Router	No' of hops to Destination
w	A	2
y	B	2
z	B	7
x	—	1
...	...	...

Routing table in router D before receiving ad' from A.

Destination Subnet	Next Router	No' of hops to Dest'
z	C	4
w	—	1
x	—	1
...	...	...

Advertisement from router A.

Destination Subnet	Next Router	No' of hops to Dest'
w	A	2
y	B	2
z	B	5
...	...	...

Routing table in router D after receiving advertisement from router A.

Intra-AS Routing in the Internet: OSPF.

OSPF - used in upper-tier ISPs.
RIP - used in lower-tier ISPs.

OSPF - uses Dijkstra least cost alg (LS alg) to determine shortest path to the subnets.

A router broadcasts routing info to all other routers (not only with neighbors in case of DV).

It also broadcasts a link's state (every 30 min), even if there is not change also. It checks whether the links are operational by sending HELLO msg.

Responsible for reliable msg transfer & link-state broadcast.

Advances added in OSPF:

1) Security: Authentication - only trusted routers can participates in the OSPF protocol.

OSPF packets are not authenticated.

Authentication can be provided by i) Password.
ii) MD5 (Msg Digest 5).

In MD5, in sending side hash value of the msg is computed & send along with the msg, in the receiver side again the hash value is computed & compared with the one received.

iii) Multiple same-cost paths: Allows multiple paths with same cost to the destination.

iii). Integrated support for unicast & multicast routing.

iv). Support for hierarchy within a single routing domain.

An OSPF AS can be configured hierarchically into areas.

Area Border routers - routing packets outside the area.

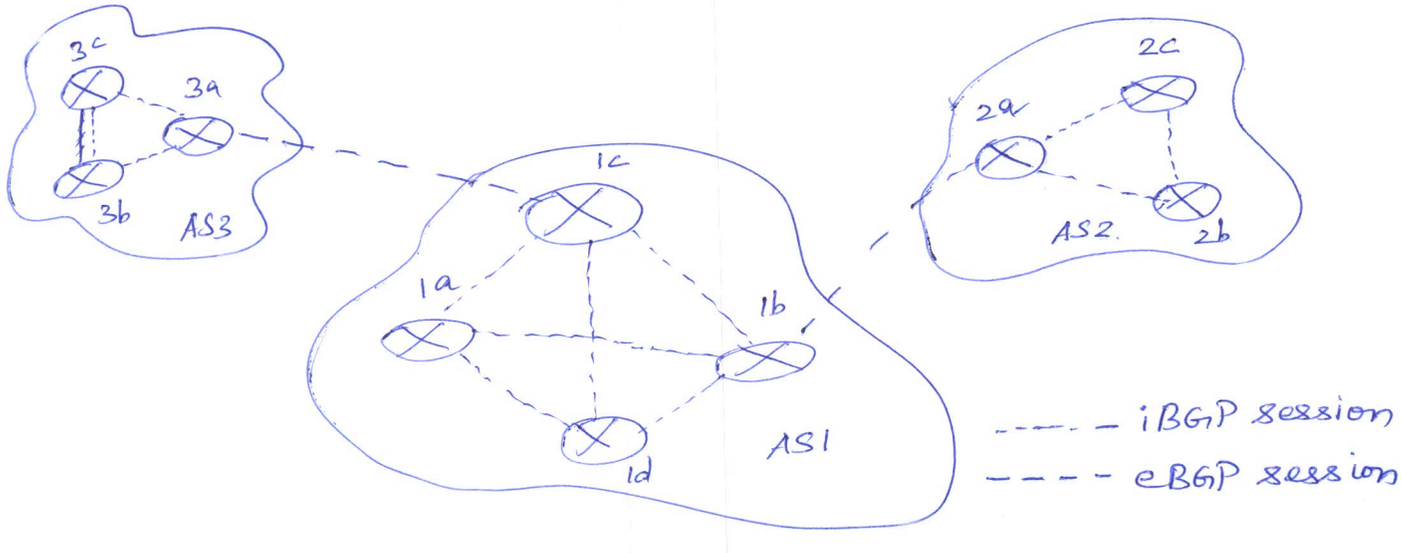
Backbone area - route traffic blw other areas in the AS.

Packet → Area Border Router → Backbone → Area Border Router → (destination area)
(Source area) Intra-area Routing. area

Inter-AS Routing: BGP (Border Gateway Protocol) version 4.

BGP provides each AS a means to:

1. Obtain subnet reachability info' from neighboring ASs.
2. Propagate the reachability info' to all routers internal to the AS.
3. Determine "good" routes to subnets based on the reachability info' & AS policy.



eBGP & iBGP sessions

BGP Peers

→ The 2 routers at the end of the connection are called BGP peers.

→ The TCP connection along with all the BGP msgs sent over the connection is called a BGP session.

→ external BGP (eBGP) session - BGP session that spans 2 ASs.

→ internal BGP (iBGP) session - BGP session blw routers in the same AS.

Path Attributes & BGP Routes

In BGP, an autonomous s/m is identified by its globally unique autonomous s/m no' (ASN), like IP add' are assigned by regional registries.

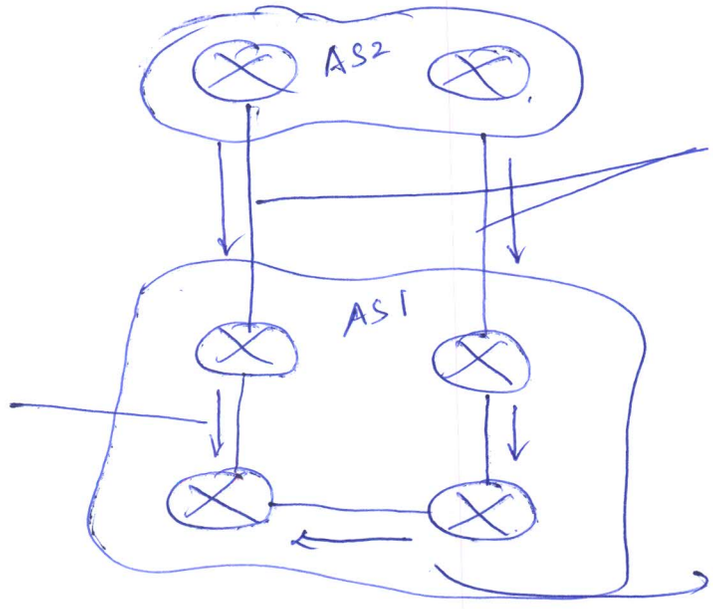
Route: BGP attributes + prefix.

BGP peers advertise routes with each other across a BGP session.

Important attributes:

AS-PATH: Contains the ASs through which the advertisement for the prefix has passed. When a prefix is passed into an AS, the AS adds its ASN to the AS-PATH attribute.

NEXT-HOP: Router interface that begins the AS-PATH. NEXT-HOP is used by routers to properly configure the forwarding tables.



Using the NEXT-HOP values & the intra-AS routing alg', the router can determine the cost of the path to each peering link, & then apply hot-potato routing to determine the appropriate interface.

Import policy: Decides whether to accept a route or not.

BGP Route Selection

I/p to Route Selection process: set of all routes that have been learned & accepted by the router.

Elimination rules:

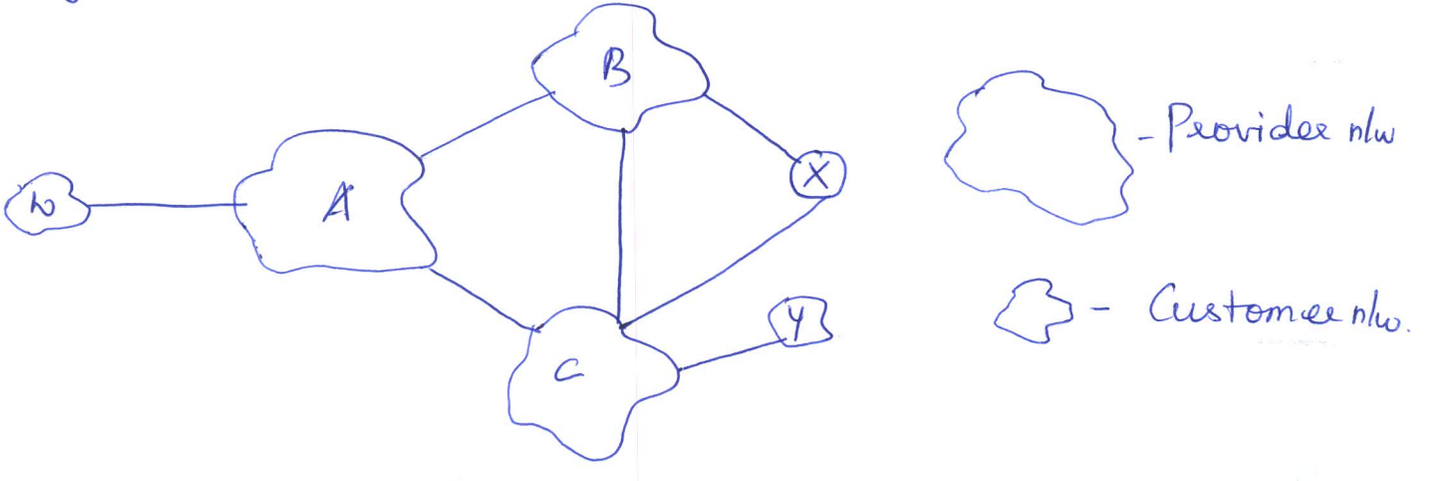
i) Local preference values are assigned to the routes. The routes with highest local preference values will be selected.

ii) From the selected routes (with same local preference value) the route with shortest AS-PATH is selected.

iii) From the selected routes (with same AS-PATH length), the router with the closest NEXT-HOP router is selected.

iv) If more than one route still remains, the router uses BGP identifiers to select the route

Routing Policy



A Simple BGP scenario

Stub nlw:

All traffic entering a stub nlw must be destined for that nlw & all traffic leaving the stub nlw must have originated in that nlw.

W & Y - stub nlw's, X - multi-homed stub nlw.

X may know of a path (XCY), that reaches nlw Y, it will not advertise this path to B. Since B is unaware that X has a path to Y, B would never forward traffic destinated to Y via X.

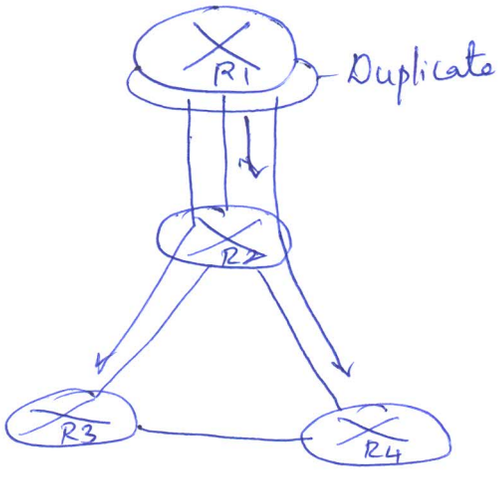
Broadcast & Multicast Routing

Unicast - point-to-point comm'

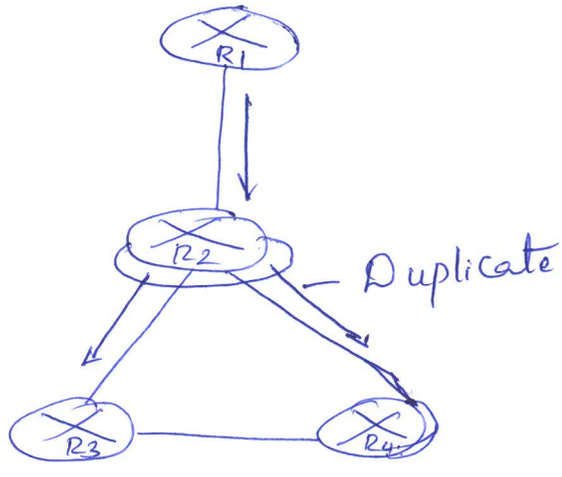
Broadcast - the nlw layer provides a service of delivering a packet sent from a source node to all other nodes in the nlw.

Multicast: Source node to a subnet of other nodes.

Broadcast Routing Alg':



Source duplication.



In-network duplication.

In order to broadcast a packet, multiple packets (duplicate copies) of a packet need to be created & send to every destination. which leads to unnecessary overhead.

Uncontrolled Flooding.

Flooding: Source node sends a copy of the packet to all of its neighbors. When a node receives a broadcast packet, it duplicates it & forwards it to all of its neighbors. This broadcast storm, will result in so many broadcast packets.

Controlled Flooding

Avoids unnecessary flooding (if a node has already received & flooded an earlier copy of a packet).

In Types

i) Sequence-no' controlled flooding.

A source node puts its add' as well as a broadcast seq' no' into a broadcast packet, & sends the packet to all of its neighbors.

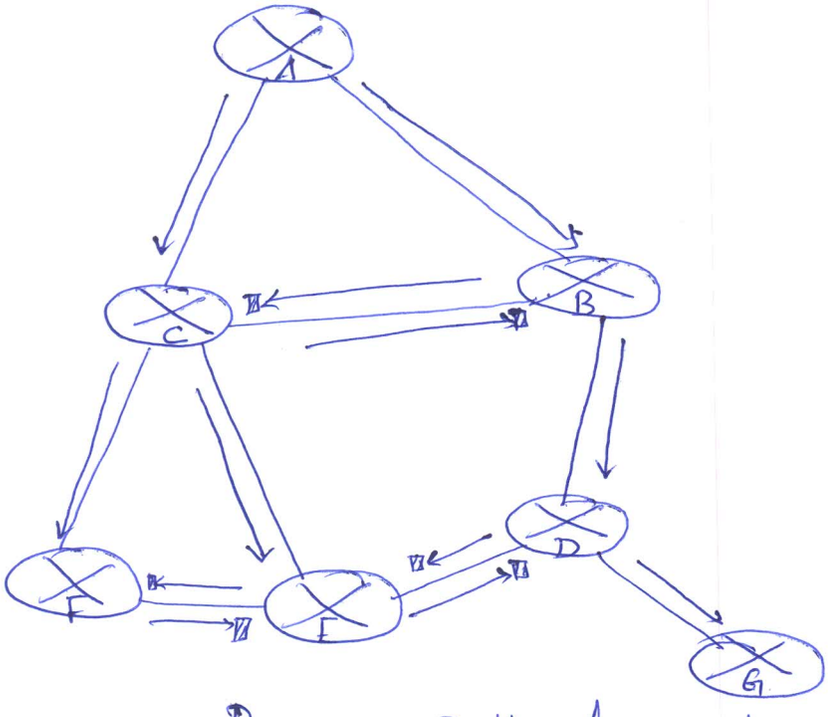
Each node maintains a list of the source add' & seq. no' of each broadcast packet it has already received, duplicated & forwarded.

When a node receives a broadcast packet it will check that packet in the list.

If the packet is there in the list, then it is dropped, if else it is duplicated & forwarded.

Reverse Path forwarding (RPF).

If a router receives a broadcast packet from a source, with source address, it transmits the packet to all of its outgoing links if it receives the packet from the shortest path back to the source, else will discard the packet. (bcz it will consider that it has received or will receive the packet by shortest path).



→ - pkt will be forwarded

→|| - pkt not forwarded beyond receiving router.

Reverse Path forwarding

Spanning-Tree Broadcast

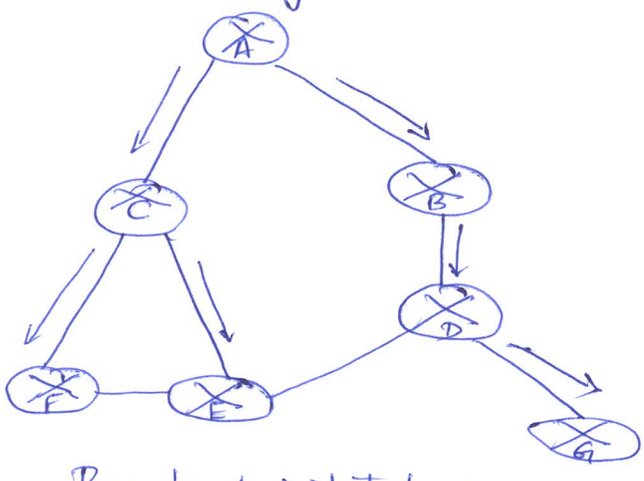
Every node will receive exactly only one copy of the broadcast packet.

Spanning tree - a tree that contains each & every node in a graph.

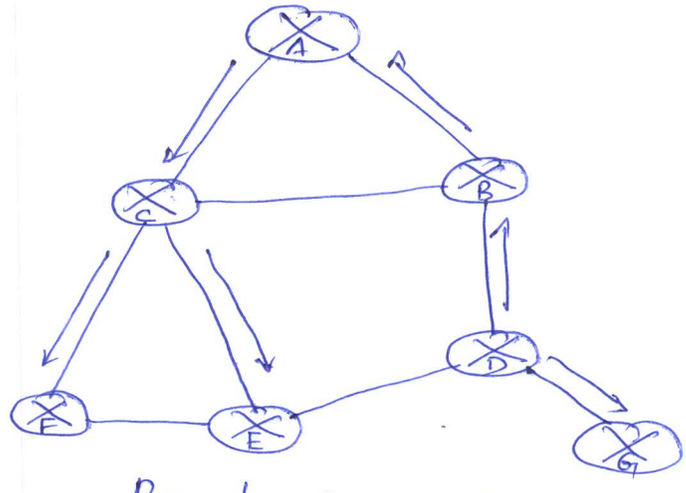
A spanning tree of a graph $G=(N,E)$ is a graph $G'=(N,E')$ such that E' is a subset of E, G' is connected, G' contains no cycles, & G' contains all the original nodes in G.

If each link has an associated cost & the cost of a tree is the sum of the link costs, then a spanning tree whose cost is the min' of all of the graph's spanning trees is called a min' spanning tree.

Constructing spanning tree is an app' to provide broadcast.



Broadcast initated at A.



Broadcast initiated at D.

Broadcast along a spanning tree

When a source node wants to send a broadcast packet, it first sends the packet to all of its incident links of the spanning tree.

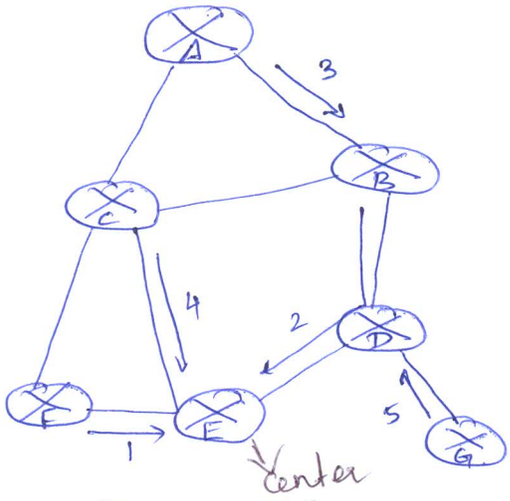
After receiving the broadcast packet, a node will send the packet to its neighbors in the spanning tree.

Complexity:

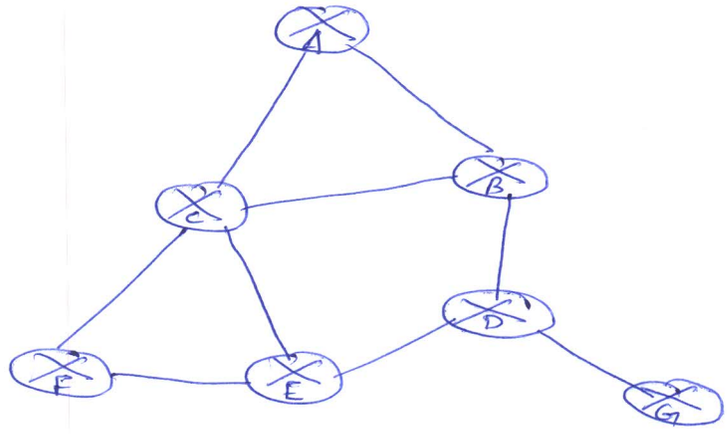
Creating & maintaining spanning tree.

Approach:

Center-based approach to build a spanning tree: a center node (rendezvous point or core) is defined.



Stepwise construction of spanning tree.



Constructed spanning tree

F joins the tree by sending route tree-joining msg to E. The link EF has become as initial spanning tree. like that, all the nodes will join the spanning tree & the links have been grafted to the spanning.

If A wants to send a tree-joining msg, it is send to B, as B has already joined the spanning tree.

Multicast

Packet is sent only to a subset of nlw nodes.

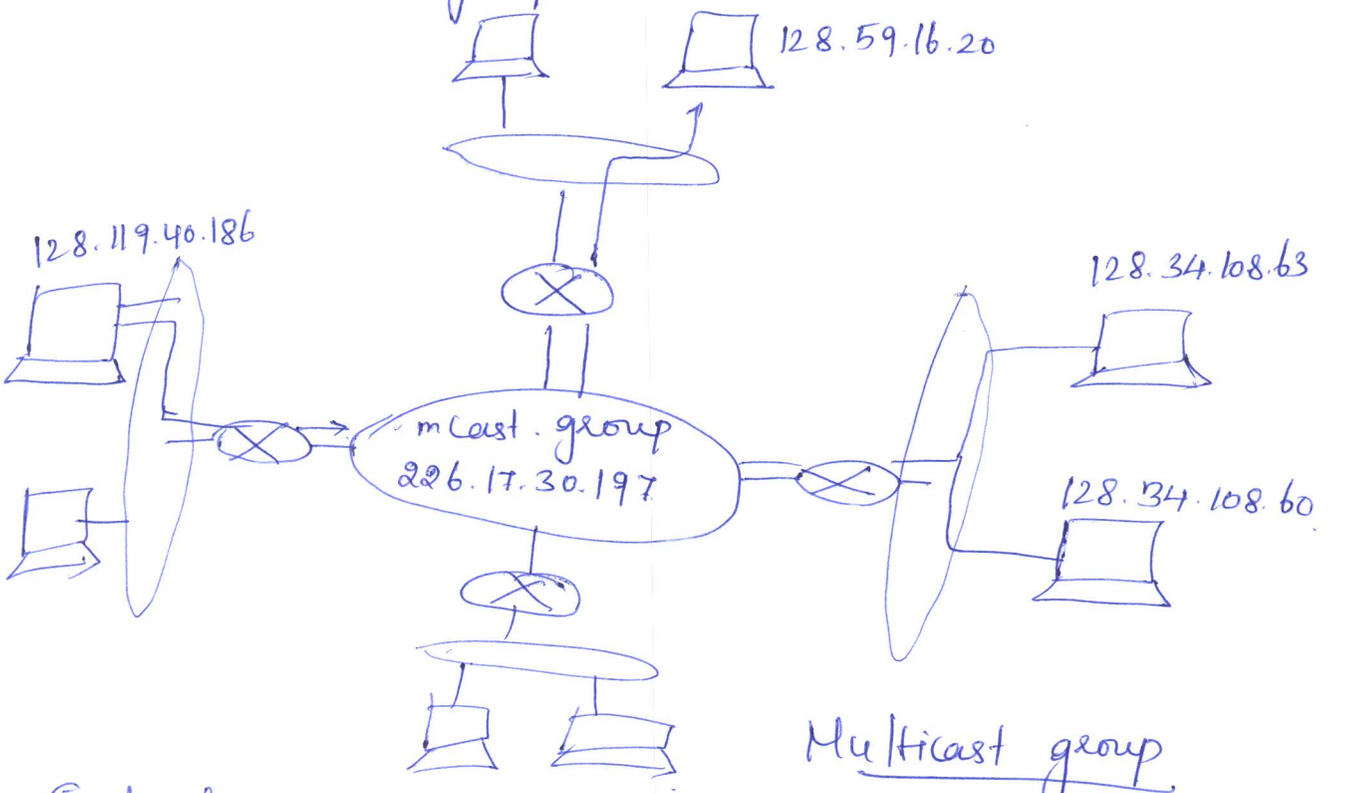
eg: bulk data transfer, transfer of audio, video, live lecture, teleconferencing.

Multicast packets are addressed using address indirection.

A single identifier is used for the group of receivers & a copy of the packet is addressed to the group.

Single add' to group of receivers - class D multicast IP add'.

The group of receivers associated with class D add' are is called as multicast group.



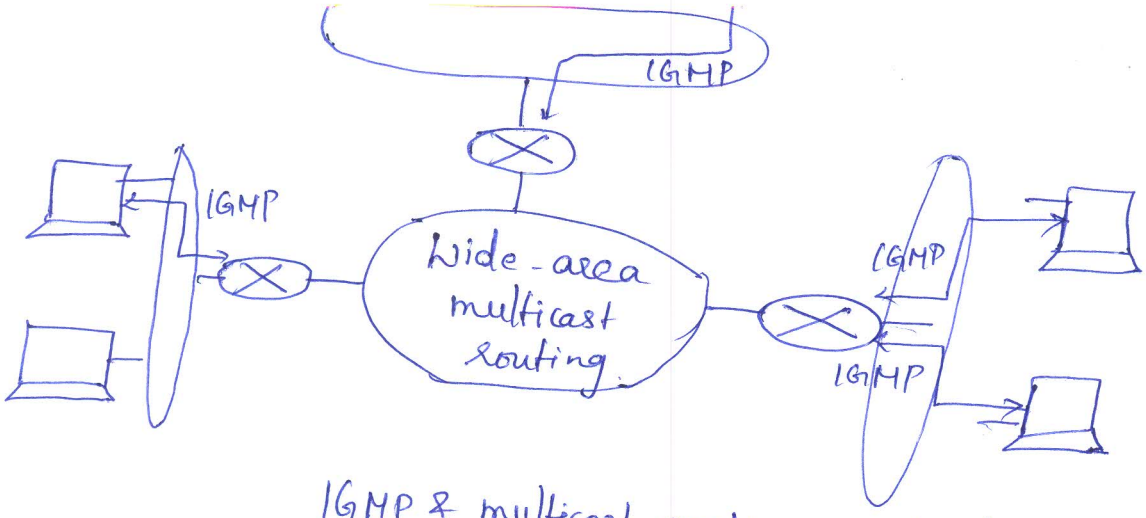
Multicast group.

Each host will also have unicast IP add' which doesn't depend on the multicast add'.

Internet Group Mgmt Protocol version 3.

Operates blw a host & its directly attached router.

IGMP provides the means for a host to inform its attached router than an app' running on the host wants to join a specific multicast group.



IGMP & multicast routing protocols.

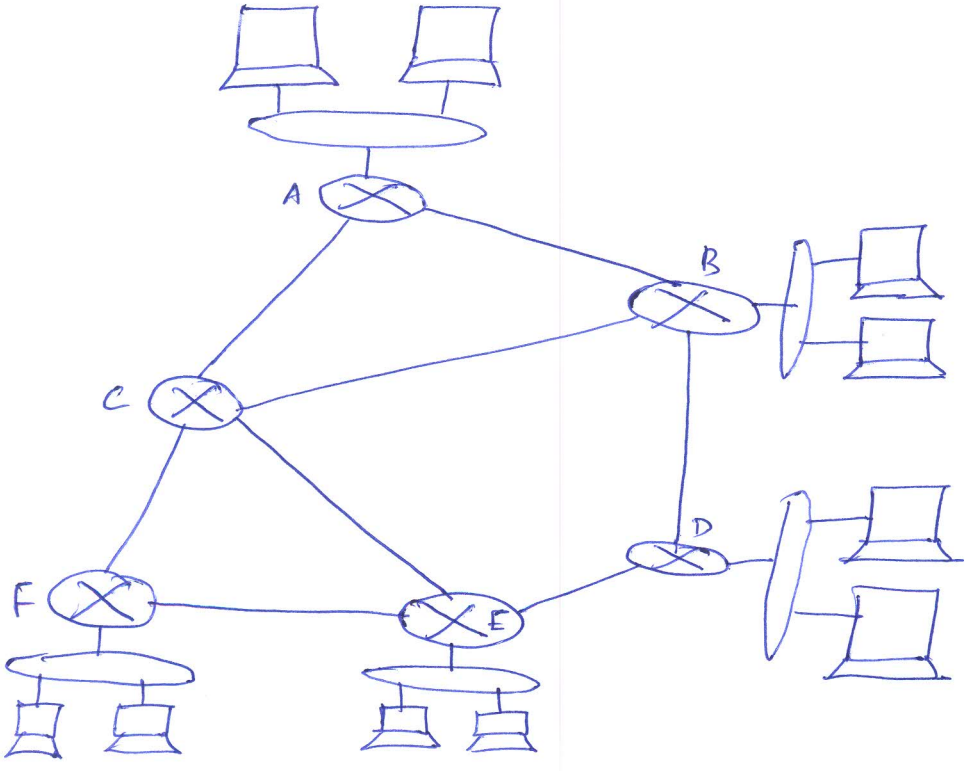
IGMP msgs are carried within an IP datagram, with an IP protocol no' of 2.

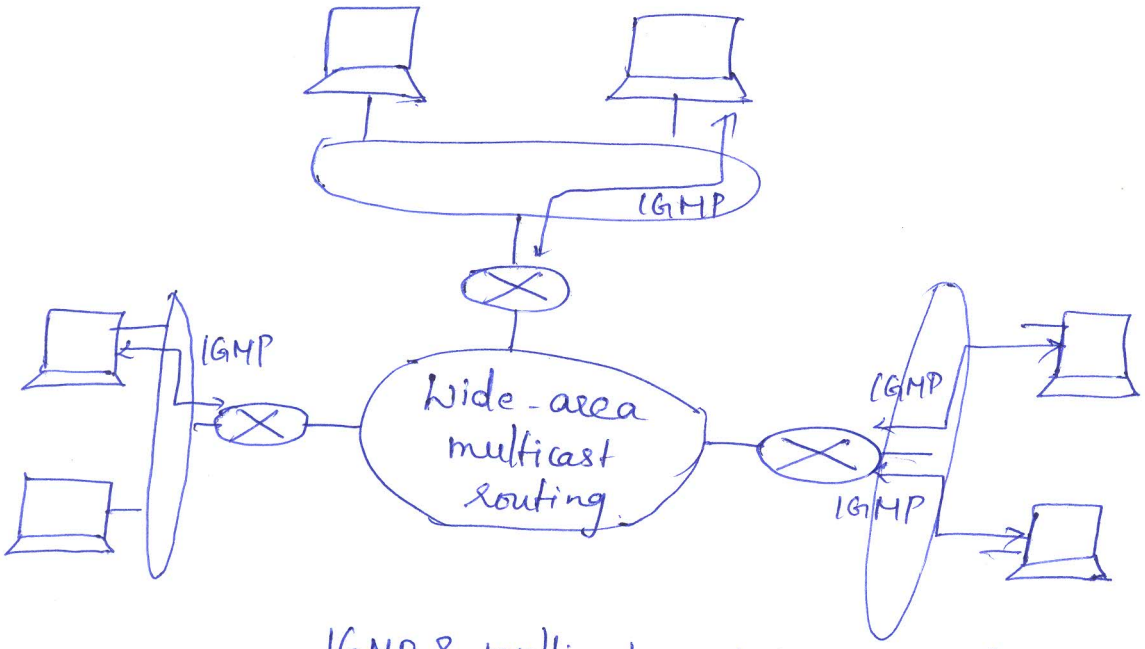
membership_query msg is sent by a router to all the hosts in the interface to identify the multicast group.

Hosts response to membership-query msg by membership_report msg.

leave_group msg (optional).

Multicast Routing Algorithms





IGMP & multicast routing protocols.

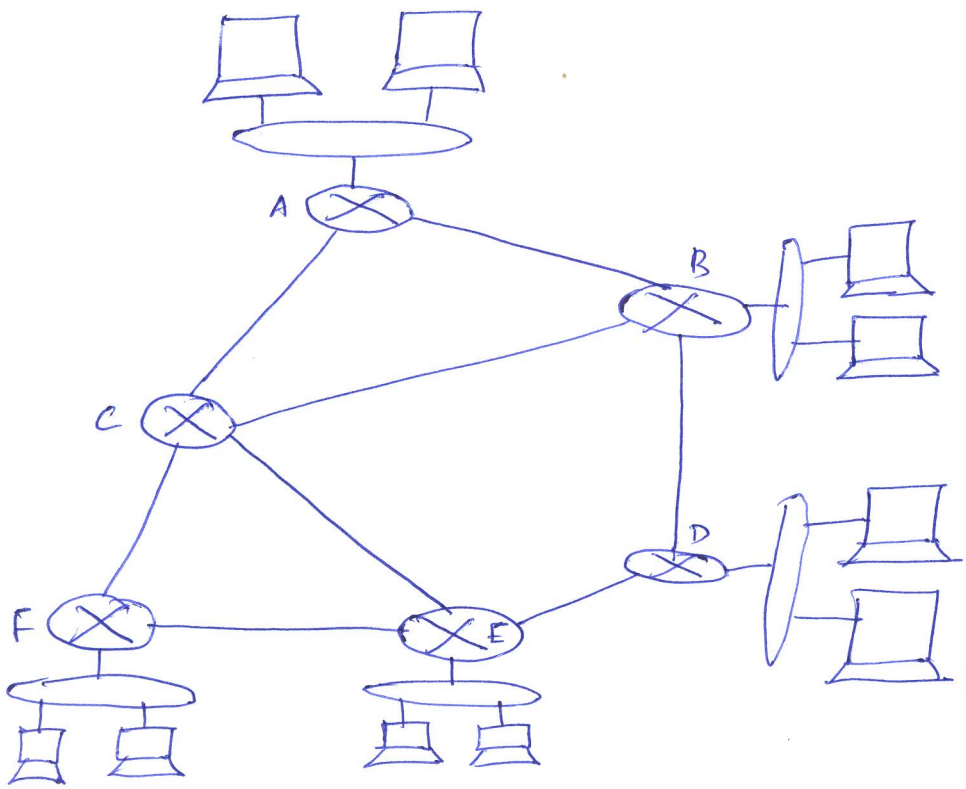
IGMP msgs are carried within an IP datagram, with an IP protocol no' of 2.

membership_query msg is sent by a router to all the hosts in the interface to identify the multicast group.

Hosts response to membership_query msg by membership_report msg.

leave_group msg (optional).

Multicast Routing Algorithms



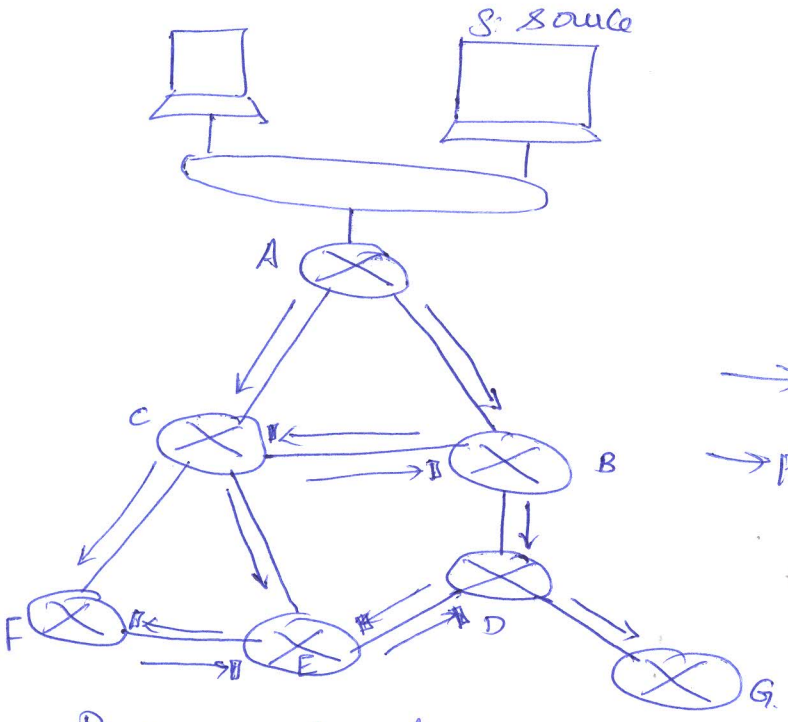
& approaches:

i) Multicast routing using a group-shared tree:

Center-based approach is used to construct the multicast routing tree, by nodes sending join msgs to the center node.

ii) Multicast routing using a source-based tree

A shared routing tree is constructed to route packets from all senders.



Reverse path forwarding (RPF)

Receiving unwanted multicast packets under RPF is called as pruning. A multicast router that receives multicast packets & has no attached hosts joined to that group will send a prune msg to its upstream router.

Multicast Routing in the Internet

Distance-Vector-multicast Routing Protocol (DVMRP) → source based trees with reverse path forwarding & pruning.

Protocol-Independent Multicast (PIM) routing protocol.

Dense mode - multicast group members are densely located. PIM dense mode is a flood & prune reverse path forwarding technique.

Sparse mode - the no' of routers with attached group members is small with respect to the total no' of routers.

PIM sparse mode uses rendezvous points to set up the multicast distribution tree.

In source-specific multicast (SSM), only a single sender is allowed to send traffic into the multicast tree, considerably simplifying tree construction & maintenance.