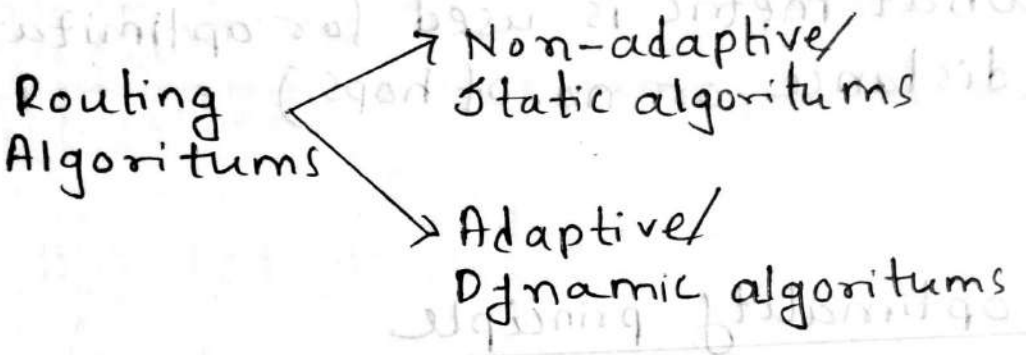
Farhan
160308

Routing Algorithms

Routing Algorithms



Non-adaptive algorithms

↳ Donot base their routing decisions on any measurements or estimates of the current topology and traffic. Instead the choice of the route to use to get from I to J (for all I and J) is computed in advanced offline and downloaded to routers when the network is booted.

Adaptive algorithms

↳ Change their routing decisions to reflect changes changes in the topology, and sometimes changes in the traffic as well. These algorithms differ in-

- where they get information (from adjacent or all routers)
- when they change the routes (when topology or load changes)

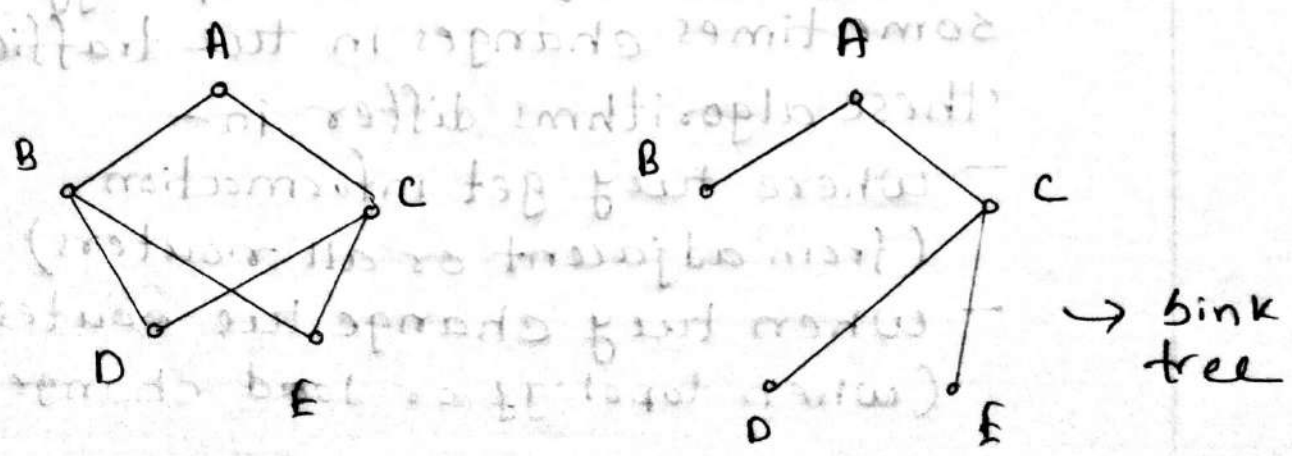
- what metric is used for optimization. (distance or no. of hops)

☐ The optimality principle

↳ If router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.

Sink tree

↳ As a direct consequence of the optimality principle, we can see that the set of optimal routes from all sources to a given destination form a tree rooted at the destination. Such a tree is called a sink tree.



↳ Sink tree

- A sink tree is not necessarily unique. other tree with the same path lengths may exist.

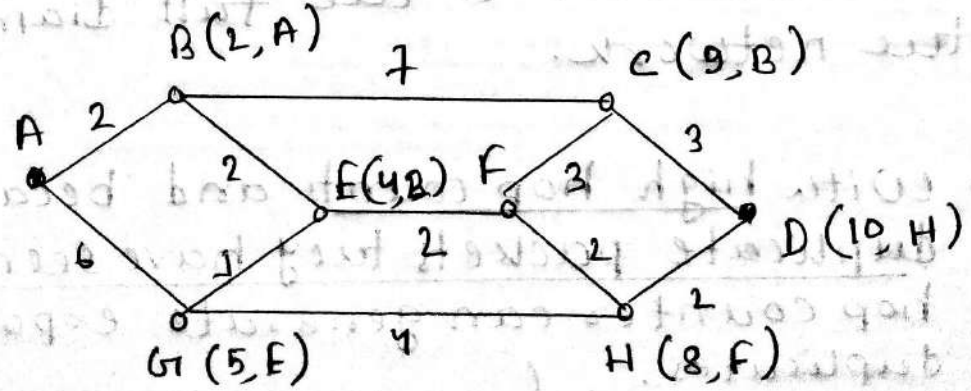
Directed Acyclic Graph

↳ If we allow all of the possible paths to be chosen, the tree becomes more general structure called a ✱ D A G.

☐ Shortest path Algorithm

↳ Dijkstra

- In general case, the labels of the edges could be computed as the function of the distance, bandwidth, average traffic, communication cost, measured delay and other factors.



Flooding

↳ Every incoming packet is sent out on every outgoing line except the one it arrived on.

– Local technique.

Hop counter

– Flooding generates vast numbers of duplicate packets. To damp the process, a hop counter is added in the header of each packet. It is decremented in each hop. When the counter reaches zero, the packet is discarded.

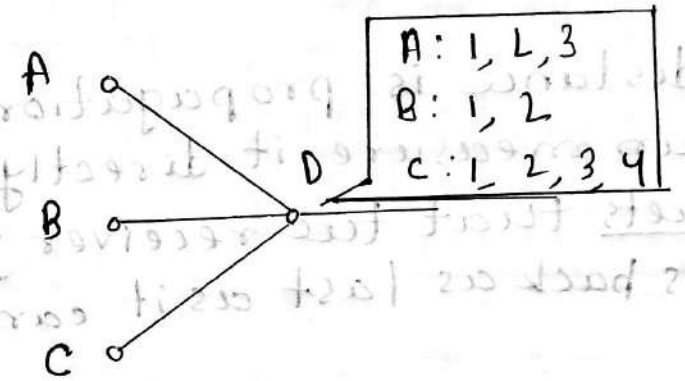
– Ideally, the counter is set to the length of the path from source to destination. If the source does not know the length, it is initialized to the full diameter of the network.

↳ With high hop counts and because routers duplicate packets they have seen before, hop counter can generate exponential duplicates.

↳ SOI^n is sequence number.

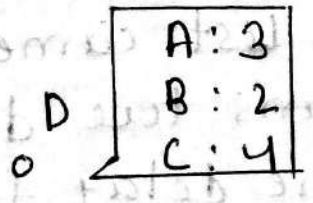
Sequence Numbers

— The source router put a sequence number in each packet it receives from its hosts. Each router then needs a list per source router telling which sequence number originated at the source have already been sent. If an incoming packet is on the list, it is not flooded.



↳ Problem is, the list can grow without Bound.

↳ Sol^n is to augment each list by a counter, k, meaning, all sequence no. through k have been seen.



Distance Vector Routing

- A Distance vector Routing algorithm operates by having each router maintain a table 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.
- If the distance is propagation delay, the router can measure it directly with special ECHO packets that the receiver just timestamps and sends back as fast as it can.

Distance Updating

- Assume, delay is used as metric and the router knows the delay to each of its neighbours. Once every T msec, each router sends to each neighbour a list of its estimated delays to each destination. It also receives a similar list from its neighbours.
- Imagine, such list came from neighbour X and X_i means the distance from X to router i. The delay from router to X is m. So the delay to i is $X_i + m$ msec.

- By performing this calculation for each neighbour, a router can find out best path to other.

- Example -

Router J has four neighbours - A, I, H, K

JA = 8, JI = 10, JH = 12, JK = 6

↗ from

To ↓	A	I	H	K
B	12	36	31	28
C	25	18	19	36
D	40	27	8	24
E	14	7	30	22

	JA	JI	JH	JK
B	20	46	43	34
C	33	28	31	42
D	48	37	20	30
E	22	17	42	28

	Line	Delay
A	A	8
B	A	20
C	I	28
D	H	20
E	I	17
H	H	12
I	I	10
K	K	6

↳ To

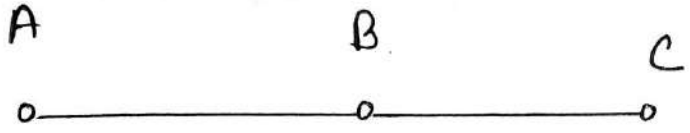
Full example on Book.

↳ Disadvantage of DVR is it causes the Count-to-Infinity problem.

- Long convergence problem.

↳ The setting of routers to best paths ~~according~~ accross the network.

Count-to-Infinity

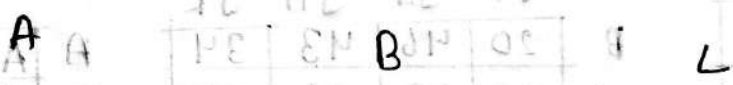


B	1
C	2

A	1
C	1

A	2
B	1

— Suppose, Link between A and B is Broken.



A	1
C	1

A	2
B	1

B can't reach A. But sees that C can reach A ←

A	③
C	1

A	2
B	1

A	3
C	1

A	⑤
B	1

→ C sees that B has a link to A and it changed. So, update.

A	⑧
C	1

A	5
B	1

A	8
C	1

A	⑬
B	1

↳ Sol^{n} is to use Link State Routing

Goes on...

Link-State Routing

- Five parts of link state routing -
  (1) Discover and learn network address of the neighbours.
  (2) Set the cost to each of its neighbours.
  (3) Construct packet to tell all it has learned
  (4) Send this packet and receive from other.
  (5) Compute shortest path to all other routers.

Learning about neighbours

- When a router is booted, its first task is to learn who its neighbours are. It accomplishes this goal by sending a special HELLO phacket on each point-to-point line. The router on the other end is expected to send back a reply giving its name.

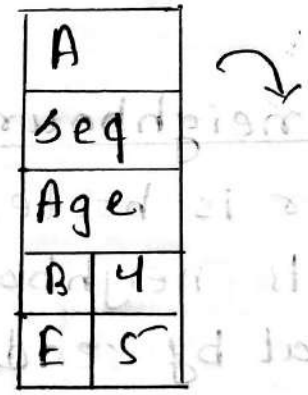
Setting Link Costs

- The router sends a special ECHO packet that the other side is required to send back immediately.

- By measuring the round trip time and dividing it by two, the sending router can get a reasonable estimate of the delay.

Building Link State Packets

- The packet included identity of the sender, sequence no, age and list of neighbours and their cost.



- When to build these packets? One possibility is to build them periodically, after T intervals. Other way is to build when some significant event occurs — a line or neighbour going down, or coming back, changing properties etc.

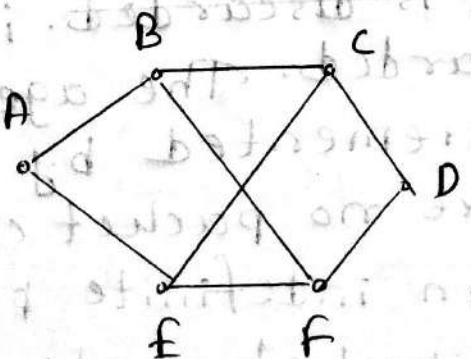
Distributing the link state packets

- Fundamental idea is to use flooding to distribute all the packets.
But flooding has few problems.
(1) If the sequence numbers wrap around, comfusion will arise.
(2) If the router crashes, count will start from 0 and will reject packets.
(3) If sequence no. is corrupted, it will reject other packets.

↳ Soln is Aging

- Age is included in the packet and decremented once per second. When the age is zero, the ~~packet is discarded~~. info. from that router is discarded. The age field is also ~~discarded~~ decremented by each router. to make sure no packet can get lost and live for an indefinite period. If age is zero, the packet is also discarded.

- Some refinements to this algorithm makes it more robust. When a link state package comes into a router for flooding, it is not queued for transmission immediately. It is put in a holding area to wait a short while. If another link state package from the same source comes in before the first packet is transmitted, sequence no. is compared. If same, new one is discarded. If different, older one is discarded.
- To guard against errors on the links, all link state packets are acknowledged.
- Example:-



			Sent			Ack			
Source	Seq.	Age	A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	Data
F	21	60	1	1	0	0	0	1	Data
E	21	59	0	1	0	1	0	1	Data
C	20	60	1	0	0	0	1	1	Data

↳ Data structure is router B.

- 1st row indicates that packet came from A to B. B forwarded to C, f and sent acknowledgement to A.
- 2nd row is for packet of f.
- 3rd row indicates, packet of E came from EAB and EfB. forwarded to C and sent ack. to A and f.
- Though C is a neighbour of B, packet of C came through A before it came from C.

Computing New Routes

- Now dijkstra's alg. can be run locally to construct true shortest path to all destinations.

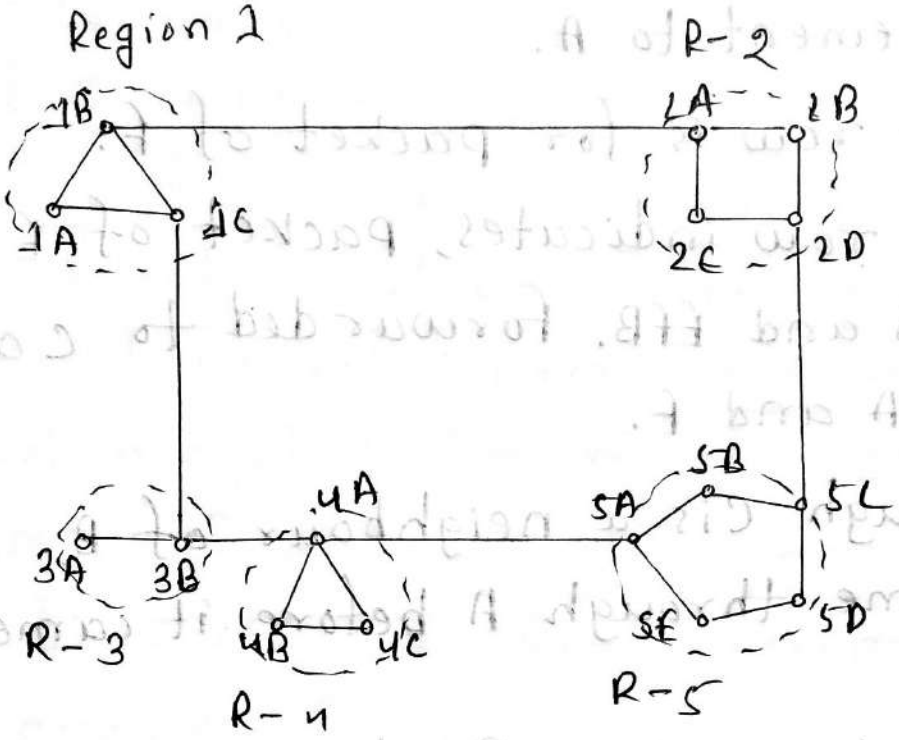
Advantage

- Does not suffer from slow convergence.

Disadvantage

- Requires more memory and computations.

Hierarchical Routing:-



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

↳ Table for 1A

- As networks grow in size, the router routing tables grow proportionally. At a certain point, the network may grow to the point where it is no longer ~~pos~~ feasible for every router to have an entry for every other router. So, the routing have to be done hierarchically, as in the telephone network.
- The routers are divided into regions. Each router only knows about the routers in the same region and the line to go to other regions.

- for huge networks, a two level hierarchy may not be sufficient. It may be necessary to group the regions into clusters, the clusters into ~~tones~~, ~~zones~~ into groups and so on.
- The optimal no of levels for an N router network is lnN. It requires elnN entries per router.

Disadvantage

- These gains in space are not free. There is a penalty to be paid — Increased path length. for example — the best route from 1A to 5C is via region 2. But here, all traffic go to region 5 via ~~1C~~ region 3, because that is better for ~~most~~ most destinations in region 5.

☐ Broadcast Routing

- In some applications, hosts need to send messages to many or all other hosts. Sending a packet to all destinations simultaneously is called broadcasting.
- Some broadcast algoritums are described below.

Multidestination Routing

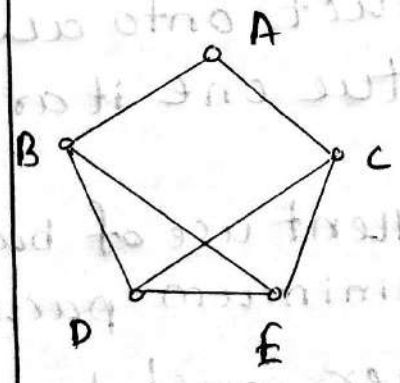
- Each packet contains either a list of destinations or a bitmap indicating the destinations.
- When a packet arrives at a router, the router checks all the destinations to determine the set of output lines that will be needed.
- The router generates a new copy of the ~~output lines~~ packet for each output lines to be used and includes in each packet only those destinations that are to use the line. In effect, the destination set is partitioned among the output lines.
- The network bandwidth is used efficiently.
- However, it still requires the source to know all the destinations and need to do much work to determine where the packet needs to be sent.

Flooding

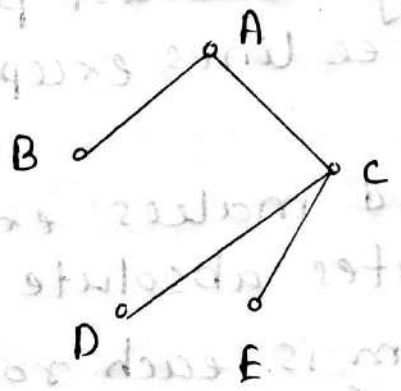
- When used ~~to~~ a sequence no., flooding uses links ~~effectively~~ efficiently.
- Although, flooding is ill suited for point-to-point lines, it is better for broadcasting.

Reverse Path forwarding

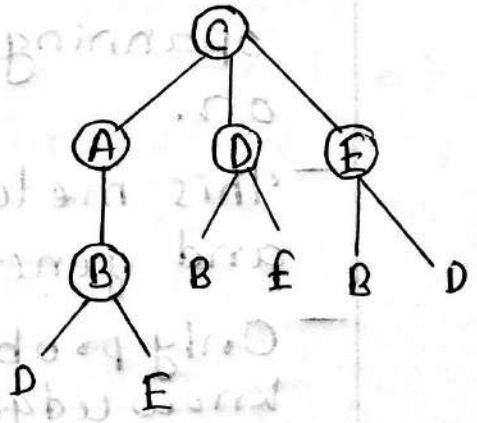
- When a broadcast packet arrives at a router, the router checks to see if the packet arrived on the link that is used for sending packets ~~too~~ towards the source of the broadcast. If so, there is good chance that the packet followed the best route from the router and the first copy to arrive. The router forwards the copies of it onto all links except the one it arrived on.
- If the broadcast packet arrived on a link other than the preferred one for reaching the source, the packet is discarded.



(a) Network



(b) Sink tree



(c) Tree built by algorithm for broadcasting from C. Circles are for preferred path.

- The principal advantage of RPF is that it is efficient while being easy to implement.
- It sends the broadcast packet over each link only once in each direction. It requires only that the routers know how to reach all destinations, without needing to remember sequence numbers or list all destinations in the packet.

Improvement on RPF

- Sink trees are spanning trees.
- If the routers knows which of its lines belong to the spanning tree, it can copy an incoming broadcast packet onto all the spanning tree lines except the one it arrived on.
- This method makes excellent use of bandwidth and generates absolute minimum packets.
- Only problem is each router must have knowledge of some spanning tree for the method to be applicable.
- Topology info is available on link state routing but not on distance vector routing.

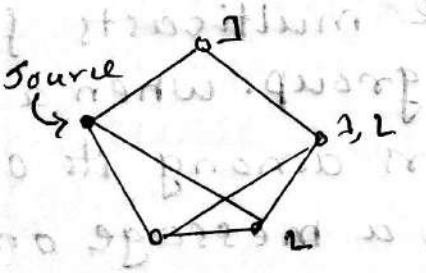
Multicast Routing

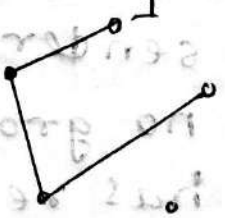
- We need a way to send messages to well defined groups that are numerically large in size but small compared to the network as a whole. Sending a message to such a group is called multicasting and the algorithm is called multicast routing.
- All multicasting schemes require some way to create and destroy the groups and to identify which routers are member of a group.

↳ Not concern of Routing algorithms

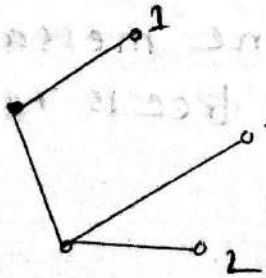
Pruning ↗ Better for dense groups

- The method is to prune the broadcast spanning tree by removing the links that do not lead to members. The result is an efficient multicast spanning tree.

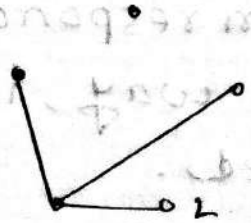




↗ Multicast tree for G-1



↗ Sink tree



↗ Multicast tree for G-2

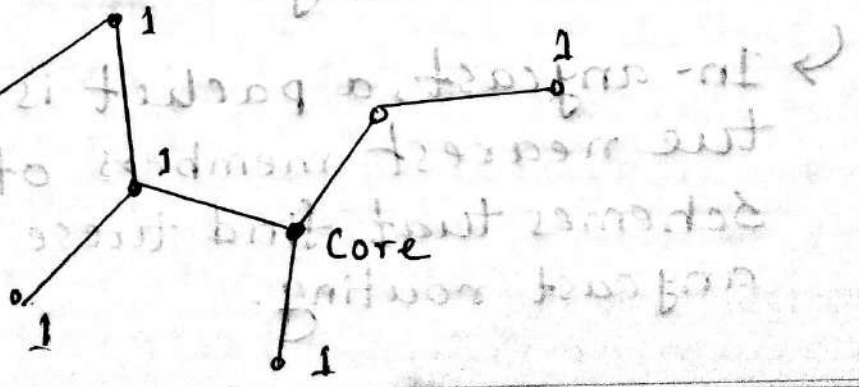
- Various ways of pruning the spanning tree are possible. One way is using link state routing. Each router is aware of the complete topology including which hosts belong to which groups. Each router can then construct its own pruned spanning tree for each sender to the group by constructing a sink tree for the sender as usual and then removing all links that do not connect group members to the sink node.
- With distance vector routing, a different pruning strategy can be followed. The basic algorithm is RPF. Whenever a router with no hosts interested in a particular group and no connections to other routers receives a multicast message for that group, it responds with a PRUNE message, telling the neighbour, that sent the message, not to send it anymore multicasts from the sender for that group. When a router with no group members among its own hosts has received such a message on all the lines to which it sends the multicast, it can respond with a prune message. In this way, the spanning tree is recursively pruned.

- Pruning results in efficient spanning trees that use only the links that are actually needed to reach members of the group.
- One potential disadvantage is that it is lots of work for routers, especially for large networks. Suppose that a network has n groups and each has average of m nodes. At each router, for each group, m pruned trees must be stored. for a total of mn trees.

Better for **sparse groups**

Core-based Trees

- An alternative design uses core based tree to compute a single spanning tree for the group.
- All of the routers agree on a ~~group~~ root and build the tree by sending a packet from each member to the root. The tree is the union of the paths traced by these packets.

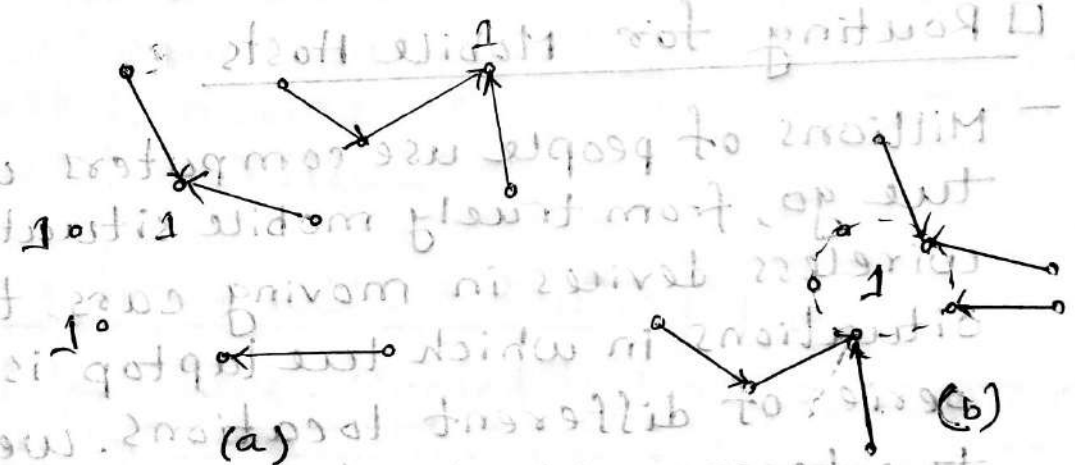


- To send to this group, a sender sends a packet to the core. When the packet reaches the core, it is forwarded down the tree.
- As a performance optimization, packets destined for the group do not need to reach the core before they are multicast. As soon as a packet reaches the tree, it can be forwarded up toward the root and down all the other branches.
- Often, it is reasonable when the core is in the middle of the senders. When there is only a single sender, as in a video stream, using the sender as root is optimal.
- Shared trees can be a major savings in storage costs, messages sent and computation. Each router has to keep only one tree per group.

☐ Anycast Routing

↳ In anycast, a packet is delivered to the nearest member of a group. Schemes that find these paths are called anycast routing.

- Why would we want anycast? Sometimes nodes provide a servile, such as time of the day or content distibution for which it is getting the right information all that matters, not the node that is contacted, any node will do.
- Regular distance vector and link state routing can produce anycast routers.



- We want to anycast to the members of group 1. They will all be given the address 1.
- Distance vector routing will distribute vectors as usual and nodes will choose the shortest path to destination 1. This will result in sending to the nearest instance of destination 1. This procedure works because the routing potocol does not realize that there are multiple instances of destination 1. It believes that all the instances of 1 are same.

- This procedure works for link state routing as well. But there is the added consideration that the routing protocol must not find seemingly short paths that pass through node 1. This would result in jumps through hyperspace, since the instance of node 1 are really nodes located in different parts of the network.

□ Routing for Mobile Hosts

- Millions of people use computers while on the go, from truely mobile situations with wireless devices in moving cars, to nomadic situations in which the laptop is used in a series of different locations. We will use the term mobile hosts to mean either category, as distinct from stationary hosts that never move.

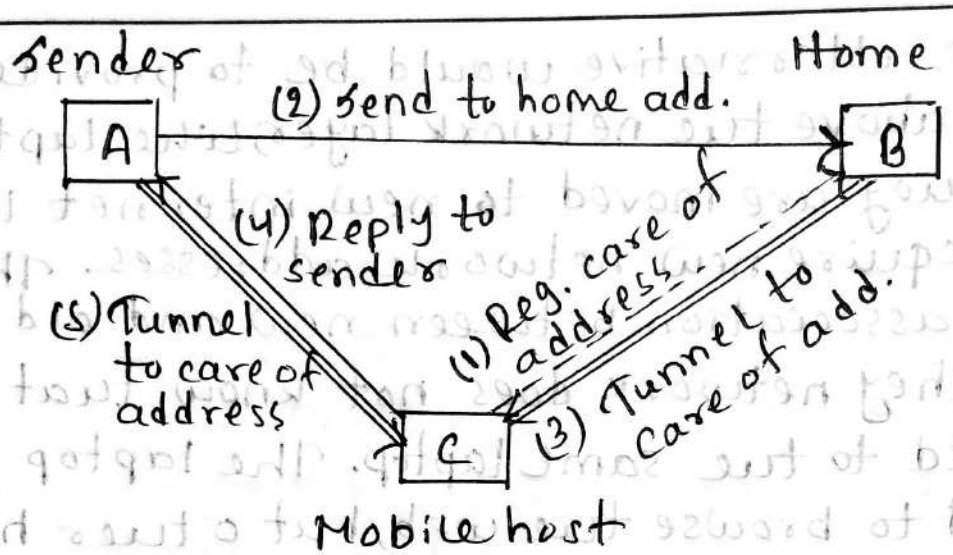
Why previous models are not used?

- A model would be to recompute ~~hosts~~ routes as the mobile host moves and the topology changes. We could then use routing schemes previously discussed. However, with a growing no. of hosts, this model would soon lead entire network to endlessly computing new routes.

- Another alternative would be to provide mobility above the network layer, like laptops. when they are moved to new internet locations they acquire new network addresses. There are no association between new and old addresses. The network does not know that they belonged to the same laptop. The laptop can be used to browse the web, but other hosts cannot send packets to it without building higher layer location servile. Moreover connections cannot be maintained while the host is moving, new connections must be started instedd.

Algorithm Used in Routing MH

- All hosts are assumed to have a permanent home location that never changes. Each hosts also has a permanent home address that can be used to determine its home location.
- The basic idea used for mobile routing in the internet and cellular networks is for the mobile host to tell a host at the home location where it is now. This host is called the home agent. Once it knows the current location of mobile host, it can forward packets to the host.



- The mobile host C needs to acquire a local network address before using the network. After acquiring the local address, also called the care of address, it sends a registration message to home agent with the care of address.
- Next, the sender sends a data packet to the permanent address.
- The home agent encapsulates the packet with new header and sends to the care of address. It is called tunneling.
- The mobile host then sends reply packet directly to the sender. The overall route is called triangular routing.
- The sender learns the current care of address and tunnels the next packets to the mobile host.

Routing in Ad-Hoc Networks

↳ Router themselves are mobile.
Ex- Military vehicles on a battlefield, a fleet of ships at sea.

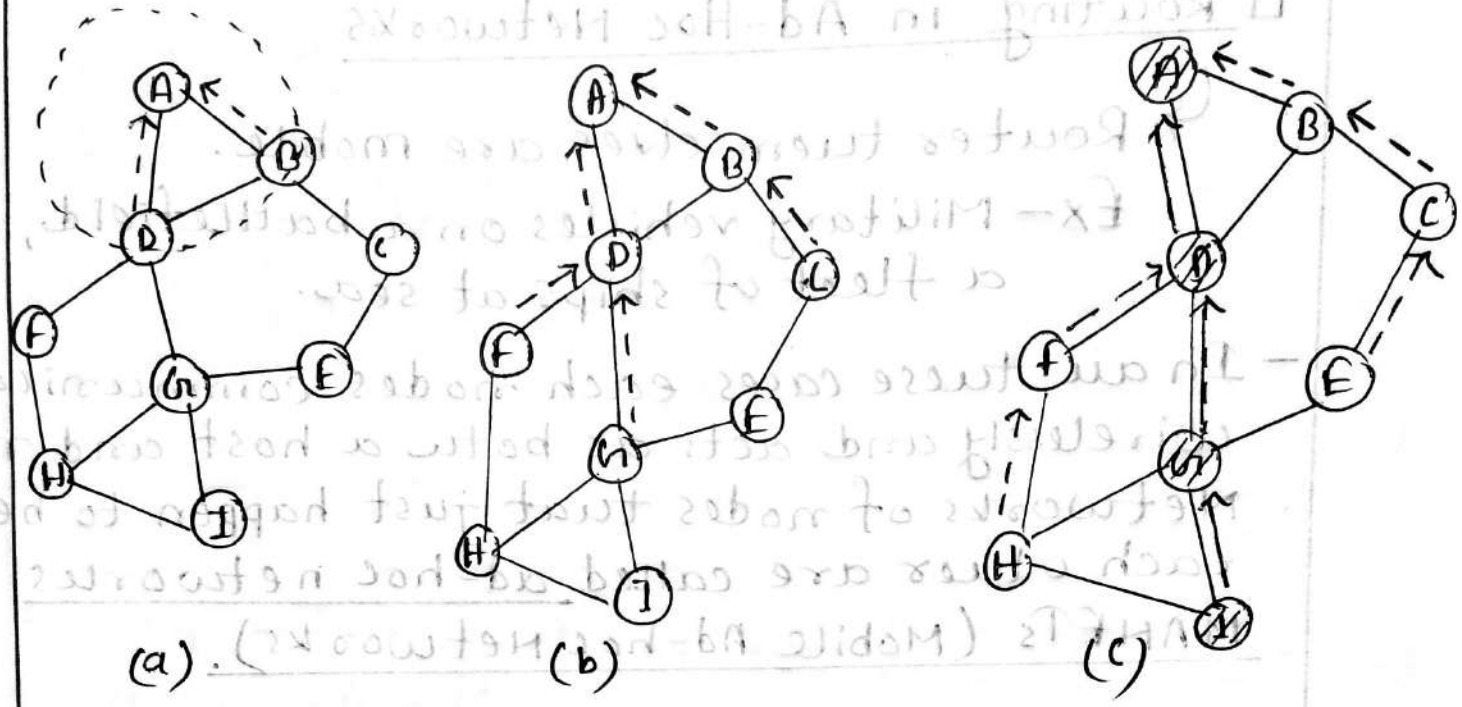
- In all these cases each nodes communicates wirelessly and acts as both a host and router. Networks of nodes that just happen to be near each other are called ad-hoc networks or MANETs (Mobile Ad-hoc Networks).

- With an ad-hoc network, the topology may be changing all the time. So, the desirability and even validity of paths can change without any warning.

Ad-hoc On-demand Distance Vector (AODV)

* Route Discovery:-

- In AODV, routes to a destination are discovered on demand, that is, only when somebody wants to send a packet to that destination.
- At any instance, topology of an adhoc network can be described by a graph of connected nodes. Two nodes are connected if they can communicate directly using their radios. Each node can communicate with all other nodes that lie within its coverage circle.



- To locate I, A constructs a ROUTE REQUEST packet and broadcast it using flooding. The transmission from A reaches B and D. Each nodes rebroadcasts the request which continues to reach node F, G, C. Packet from B to D is discarded because it is a duplicate of the packet from A. Then the request reaches H, I, E.
- Eventually the request reaches I. I constructs a ROUTE REPLY packet. This packet is unicast to the sender along the reverse of the path followed by the request. The arrows show reverse route information that are stored.
- Each intermediate node also increments a hopcount as it forwards the reply. It tells the node how far they are from the destination.

- The replies tell each intermediate node which neighbour to use to reach the destination - it is the node that sent them the reply.
- Intermediate nodes G and H put the best route ~~rou~~ they hear into their routing tables. When the reply reaches A, a new route ADGI is created.

Optimizing Route Discovery

- In a large network, many broadcasts are generated, even for close by destinations.
- To reduce overhead, the scope of the broadcasts is limited using the IP packets Time to Live field. This field is initialized by the sender and decremented on each hop. Discarded when zero.
- The route discovery process is then modified as follows. To locate a destination, the sender broadcasts a ROUTE REQUEST packet with Time to Live set to 1. If no response comes back within a reasonable time, another one is sent with Time to Live set to 2. Subsequent attempts use 3, 4, 5, etc.

* Route Maintenance:-

- Because nodes can move or be switched off, the topology can change spontaneously. In our example, if G is switched off, A will not realize that ADGI is no longer valid. The algorithm must be able to deal with this.
- Periodically each node sends a Hello message broadcast. Each of its neighbours is expected to respond. If no response comes, the broadcaster knows the neighbour is no longer connected.
- The broadcaster checks its routing table to see which destinations have routes using the now-gone neighbour. For each of these routes the active neighbours are informed that their route via broadcaster is now gone and must be purged from their routing tables. In our example, D purges its entries for G and I and notifies A. A purges its entries for I. Then the active neighbours inform their active ~~members~~ neighbours until all routers depending on the now gone router are purged from all routing tables.

- At this stage, the invalid routers have been purged from the network and senders can find new route by using the method we described earlier. However, the problem is slow convergence.
- For rapid convergence, routes include a sequence no. that is controlled by the destination. The destination sequence no. is like a logical clock. The destination increments it every time that it sends a fresh ROUTE REPLY. Senders ask for a fresh route by including the last sequence no. they used in the ROUTE REQUEST. The included no. may be the no. of the route that was just purged or 0 as initial value. The route will be broadcast until a route with higher sequence no. is found. Intermidiate nodes store the routes that have a higher sequence no.
- Intermidiate nodes only store the routes that are in use. It helps to save bandwidth and battery life.
- So far, we have considered only a single route. To further save resources, route discovery and maintenance are shared with routes overlap. For example, if B also wants to send to I, it will perform route discovery. In this case, request will first

reach D. D already has a route to I. Node D can then generate reply to tell B the route without any additional work.