UNIT-IV: Mobile Ad hoc Network (MANET)

1 Types of Wireless Network

- In wireless network data is transmitted through wireless links from one point to another point that is i.e. there is no need of wired link between the two nodes for transmission. They just need to be in the transmission range of each other. Wireless networks or divided into two categories. Infrastructure wireless network and infrastructure less or Ad-hoc wireless network.

1.1 Infrastructure Network

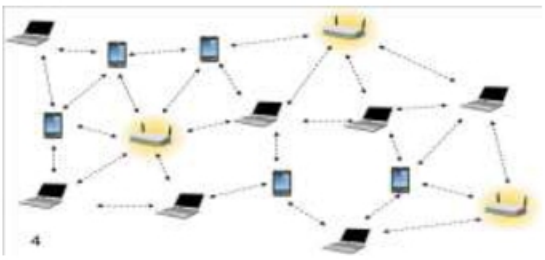
- Infrastructure network is type of wireless network which have fixed network topology. In this network wireless nodes connect through the fixed point which is known as access point or base station. In most cases the access point or base station are connected to the main network through wired link.
- The access point or base station is one of the important elements in such types of networks. All of the wireless links must pass from the access point or base station. Whenever a node is in the coverage of several base stations then it connect to any one of them on the basis of some defined criteria in the system.

1.2 Ad-hoc Network

- A type of wireless network which is decentralized that is there is no central administration is called Ad-hoc network. Also the Ad-hoc network is infrastructure less and nodes are independent. Each node works as a hosts as well as router to forwards the data on behalf of other nodes.
- The nodes are free to join or left the network without any restriction. In Ad-hoc networks the nodes can be stationary or mobile. Therefore one can say that Ad-hoc networks basically have two different forms, one in which the nodes are stationary is called static Ad-hoc networks (SANET) and the other in which nodes are mobile is called mobile Ad-hoc networks (MANET).
- From the introduction of new technologies such as IEEE 802.11 the commercial implementation of Ad-hoc network becomes possible. One of the good features of such networks is the flexibility and can be deployed very easily. Thus it is suitable for the emergency situation. But on the other side handling the operation of Ad-hoc network is also very difficult.
- Each node is answerable to handle its operation independently. Topology changes are very recurrent and thus there will be require of a proficient routing protocol, whose construction is a complex task.

2 Introduction to Mobile Ad hoc Network (MANETs):

- **Mobile Ad hoc Networks (MANETs)** are wireless networks which are characterized by dynamic topologies and no fixed infrastructure. Each node in a MANET is a computer that may be required to act as both a host and a router and, as much, may be required to forward packets between nodes which cannot directly communicate with one another. Each MANET node has much smaller frequency spectrum requirements that for a node in a fixed infrastructure network.
- A MANET is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized, where all network activity including discovering the topology and delivering messages must be executed by the nodes themselves, i.e., routing functionality will be incorporated into mobile nodes.



- A mobile ad hoc network is a collection of wireless nodes that can dynamically be set up anywhere and anytime without using any pre-existing fixed network infrastructure.

❖ **MANET- Characteristics**
- Dynamic network topology
- Bandwidth constraints and variable link capacity
- Energy constrained nodes
- Multi-hop communications
- Limited security
- Autonomous terminal
- Distributed operation
- Light-weight terminals
- Self-creation self-organization and self-administration

❖ **Need for Ad Hoc Networks**
- Setting up of fixed access points and backbone infrastructure is not always viable
  - Infrastructure may not be present in a disaster area or war zone
  - Infrastructure may not be practical for short-range radios; Bluetooth (range ~ 10m)
  - Do not need backbone infrastructure support
  - Are easy to deploy
  - Useful when infrastructure is absent, destroyed or impractical

3 Properties of MANETs

- MANET enables fast establishment of networks. When a new network is to be established, the only requirement is to provide a new set of nodes with limited wireless communication range. A node has limited capability, that is, it can connect only to the nodes which are nearby. Hence it consumes limited power.
- A MANET node has the ability to discover a neighboring node and service. Using a service discovery protocol, a node discovers the service of a nearby node and communicates to a remote node in the MANET.
- MANET nodes have peer-to-peer connectivity among themselves.
- MANET nodes have independent computational, switching (or routing), and communication capabilities.
- The wireless connectivity range in MANETs includes only nearest node connectivity.
- The failure of an intermediate node results in greater latency in communicating with the remote server.
- Limited bandwidth available between two intermediate nodes becomes a constraint for the MANET. The node may have limited power and thus computations need to be energy efficient.
- There is no access-point requirement in MANET. Only selected access points are provided for connection to other networks or other MANETs.
- MANET nodes can be the iPods, Palm handheld computers, Smartphones, PCs, smart labels, smart sensors, and automobile-embedded systems.
- MANET nodes can use different protocols, for example, IrDA, Bluetooth, ZigBee, 802.11, GSM, and TCP/IP.MANET node performs data caching, saving, and aggregation.
- MANET mobile device nodes interact seamlessly when they move with the nearby wireless nodes, sensor nodes, and embedded devices in automobiles so that the seamless connectivity is maintained between the devices.

4 Advantages and Disadvantages of Ad hoc Network

	Advantages	Disadvantages
Infrastructure	The Ad hoc networks do not rely on any infrastructure. They work independently, are more robust, and it is cheaper to form an ad hoc network. There is no installation, maintenance cost.	Without any help from infrastructure, the nodes have to work harder. They have to hop the messages, secure their own resource from attackers, and perform a routing table.
Mobility	Unlike the infrastructure network, in which node's moving is restricted by cell's border, in the ad hoc networks, a node can theoretically move freely. As long as this node can hop to a node inside the network, it can also communicate with other node in this network.	In the practical, it is hard to form a network, in which a node can move freely
Scalability		Depend on routing algorithm, how the Ad hoc networks can perform well. In a network with a large number of nodes and high mobility, a table driven algorithm would not perform well, because there will be big overhead. Generally, the infrastructure networks perform better in this situation. The infrastructure networks have only specific tasks, so they can handle more nodes
Routing	In the infrastructure networks, if the access point is defected, there will be no more communication in the affected cell	Mobility and increased or decreased number of nodes can force some routing algorithms to alter their routing table.
Security	Some attacks can cause malfunction. If one of participant is attacked and it doesn't work anymore. The network can relay the messages through other route (if alternative route is available).	Internal attacks may be possible via Ad hoc transmissions. It means, the attacker can disguise itself as an Ad hoc participant. It can spy, modify, or delete the hopped messages.

5 Challenges and Issues in MANET

The absence of fixed infrastructure in MANETs creates a number of different challenges and issues like mobility, security, bandwidth constraints, hidden and exposed node problems and routing mechanisms. To design a good wireless ad hoc network, various challenges have to be taken into account:

- *Dynamic Topology*: Nodes are free to move in an arbitrary fashion resulting in the topology changing arbitrarily. This characteristic demands dynamic configuration of the network.
- *Limited security*: Wireless networks are vulnerable to attack. Mobile ad hoc networks are more vulnerable as by design any node should be able to join or leave the network at any time. This requires flexibility and higher openness.
- *Limited Bandwidth*: Wireless networks in general are bandwidth limited. In an ad hoc network, it is all the more so because there is no backbone to handle or multiplex higher bandwidth
- *Routing*: Routing in a mobile ad hoc network is complex. This depends on many factors, including finding the routing path, selection of routers, topology, protocol etc.

The most challenging issue is the design of MAC protocols that define how the wireless medium can be shared by all nodes. Because of the nature of the network, a distributed random access MAC is preferred over centralized MAC. CSMA (Carrier Sense Multiple Access) is one of the earliest mechanisms that may be adopted for Ad hoc networks. In CSMA, a transmitter first senses the wireless channel in vicinity and restrain itself from transmission if the channel is already in use.

Distributed random access protocols such as CSMA suffer from hidden and exposed nodes problem. It is also assumed that each node can communicate with another node only if there is a link between them. In a typical exposed node problem, a node within the range of the transmitter may be unnecessarily prohibited to access the medium and thus degrade the network throughput.

5.1 Hidden-Node Problem

- Let us consider a network scenario consisting of three mobile nodes S1, R1 and S2, as shown in figure. It is assumed that all the nodes transmit at the same power. The circles around these nodes show the radio-coverage area. As illustrated in figure, the node R is located in the middle of nodes S1 and S2. In other words the node R1 lies in the radio-coverage areas of the nodes S1 as well as the node S2.
- Further, node R1 receives transmission from both nodes S1 and S2 but S1 and S2 cannot receive transmission from each other since S1 is out of range of S2 and in a similar manner, S2 is out of range of S1. When the node S1 transmits to the node R1, the node S2 cannot detect this transmission using the carrier sense mechanisms. Is S2 also transmits to R1 then this transmission collides with that of from S1 for R1. Hence, the node R1 can be said to be a hidden nodes for S1 and S2. This increase data packet collisions and hence reduce throughput.

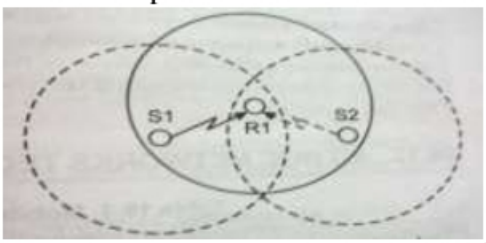


Figure: Hidden-node problem in MANET

5.2 Exposed-node Problem

- CSMA/CA with RTS/CTS mechanism resolves the hidden-node problem only if the nodes are synchronized. When a node receives an RTS data packet from a neighbouring node, but not the corresponding CTS data packet, that node deduces that it is an exposed node and is permitted to transmit to other neighbouring nodes.
- If the nodes are not synchronized, the problem may occur that the sender does not receive the CTS data packet or the ACK packet during the course of transmission of data of second node.
- In wireless network, the exposed-node problem occurs when a node is checked from sending packets to other nodes because of a neighbouring transmitter.
- Let us consider an example of 4 nodes labeled R1, S1, S2 and R2 as shown in figure. Where the two nodes are out of range of each other (node R1 is out of range of node R2 and vice versa), yet the two nodes in the middle (node S1 and node S2) are in range of each other.
- If a transmission between nodes S1 and R1 takes place, the node S2 is checked from transmitting to the node R2 since it concludes, based on carrier sense that it interferes with the transmission by its neighbouring node S1. But, the node R2 could still receive the transmission of the node S2 without interference since it is out of range from the node S1.
- When the node S1 transmits to the node R1, the node S2 detects this transmission using carrier sense mechanism. Node S2 avoids transmitting to the node R2, hence the node S2 is exposed to S1a transmission. This particular situation reduces bandwidth utilization and therefore reduces throughput.
- The possible solution is the use of directional antennas, and separate channels for control and data. Modified CSMA/CA with RTS/CTS mechanism also helps to resolve the exposed-node problem.
- The duration of data transfer is included in RTS and CTS control packet itself which instructs other nodes not to transmit for this duration. If a RTS/CTS packet collides, the nodes wait for a random time which is calculated using binary exponential back-off algorithm. This scheme is also called Multiple Access Collision Avoidance (MACA). The only drawback is that it cannot avoid RTS/CTS control packet collisions.

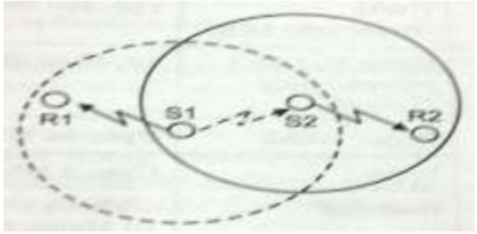
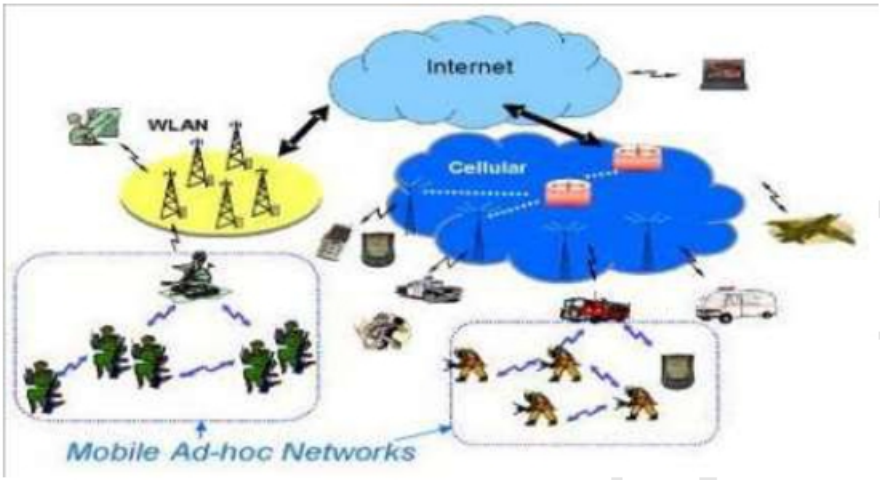


Figure: Exposed node problem in MANET

6 Applications of MANETS

The set of applications for MANETs is diverse, ranging from small, static networks that are constrained by power sources, to large-scale, mobile, highly dynamic networks. The design of network protocols for these networks is a complex issue. Regardless of the application, MANETs need efficient distributed algorithms to determine network **organization, link scheduling, and routing. Some of the main application areas of MANET's are:**



- ***Military battlefield***– soldiers, tanks, planes. Ad- hoc networking would allow the military to take advantage of commonplace network technology to maintain an information network between the soldiers, vehicles, and military information headquarters.
- ***Sensor networks*** – to monitor environmental conditions over a large area
- ***Local level*** – Ad hoc networks can autonomously link an instant and temporary multimedia network using notebook computers or palmtop computers to spread and share information among participants at e.g. conference or classroom. Another appropriate local level application might be in home networks where devices can communicate directly to exchange information.
- ***Personal Area Network (PAN)*** – pervasive computing i.e. to provide flexible connectivity between personal electronic devices or home appliances. Short-range MANET can simplify the intercommunication between various mobile devices (such as a PDA, a laptop, and a cellular phone). Tedious wired cables are replaced with wireless connections. Such an ad hoc network can also extend the access to the Internet or other networks by mechanisms e.g. Wireless LAN (WLAN), GPRS, and UMTS.
- ***Vehicular Ad hoc Networks*** – intelligent transportation i.e. to enable real time vehicle monitoring and adaptive traffic control
- ***Civilian environments*** – taxi cab network, meeting rooms, sports stadiums, boats, small aircraft
- ***Emergency operations*** – search and rescue, policing and firefighting and to provide connectivity between distant devices where the network infrastructure is unavailable. Ad hoc can be used in emergency/rescue operations for disaster relief efforts, e.g. in fire, flood, or earthquake. Emergency rescue operations must take place where nonexisting or damaged communications infrastructure and rapid deployment of a communication network is needed. Information is relayed from one rescue team member to another over a small hand held.

Table: Mobile Ad hoc network applications

Application	Possible Scenarios/Services
Tactical networks	• Military communication and operations • Automated battlefields
Emergency Services	• Search and rescue operations • Disaster recovery • Replacement of fixed infrastructure in case of environmental disasters • Policing and fire fighting • Supporting doctors and nurses in hospitals
Commercial and civilian environments	• **E-commerce:** Electronic payments anytime and anywhere • Business dynamic database access, mobile offices • **Vehicular Services:** Road or accident guidance, transmission or road and weather conditions, taxi cab network, inter-vehicle networks • Sports stadiums, trade fairs, shopping malls • Networks of visitors at airports
Home and enterprise networking	• Home/office wireless networking • Conferences, meeting rooms • Personal area networks (PAN), personal networks (PN) • Networks at construction sites
Education	• Universities and campus setting • Virtual classrooms • Ad hoc communications during meeting or lectures
Entertainment	• Multi-user games • Wireless P2P networking • Outdoor Internet access • Robotic pets • Theme parks
Sensor networks	• Home applications; smart sensors and actuators embedded in consumer electronics • Data tracking of environmental conditions, animal movements, chemical/biological detections
Context aware services	• **Follow-on Services:** Call-forwarding, mobile workspace • **Information Services:** Location Specific services, time dependent services • **Infotainment:** Touristic information
Coverage extension	• Extending cellular network access • Linking up with the Internet, intranets, etc

7 Routing

IP addressing is based on the concept of hosts and networks. A host is essentially anything on the network that is capable of receiving and transmitting IP packets on the network, such as a workstation or a router. Routing is a process of moving data from one host computer to another. The difference between routing and bridging is that bridging occurs at Layer 2 (the link layer) of the OSI reference model, whereas routing occurs at Layer 3 (the network layer). Routing determines the optimal routing paths through a network.

❖ Routing Algorithms

The routing algorithm is stored in the router's memory. The routing algorithm is a major factor in the performance of your routing environment. The purpose of the routing algorithm is to make decisions for the router concerning the best paths for data. The router uses the routing algorithm to compute the path that would best serve to transport the data from the source to the destination. Note that you do not directly choose the algorithm that your router uses. Rather, the routing protocol you choose for your network determines which algorithm you will use. For example, whereas the routing protocol Routing Information Protocol (RIP) may use one type of routing algorithm to help the router move data, the routing protocol Open Shortest Path First (OSPF) uses another. The routing algorithm cannot be changed. The only way to change it is to change routing protocols. The overall performance of your network depends mainly on the routing algorithm, so you should research the algorithms each protocol uses before deciding which to implement on your network. There are two major categories of routing algorithms - distance vector or link-state. Every routing protocol named "distance vector" uses the distance vector algorithm, and every link-state protocol uses the link-state algorithm.

❖ Routing Algorithms within Routing Protocols

One of the jobs of the routing protocol is to provide the information needed by the routing algorithm to compute its decisions. This is the point where many protocols differ. The information provided to the algorithm can be different from protocol to protocol.

The routing protocol gathers information about networks and routers from the surrounding environment and stores the information within a routing table in the router's memory. The routing algorithm is run using the information within this table to calculate the best path from one network to another. Calculating the new values within the formula then generates a sum. The result of this calculation is used then to determine where to send information. For example, the table below illustrates a sample routing table for a fictitious routing environment. The information that is passed to the routing algorithm within the routing table is gathered by the routing protocol through a process known as a routing update. Through a series of updates, each router will tell the other what information it has. Eventually, an entire routing table will be built.

Router Link	Metric
Router **A** to Router **B**	2
Router **B** to Router **C**	3
Router **A** to Router **C**	6
Router **C** to Router **D**	5

The sample routing algorithm states that the best path to any destination is the one that has the lowest metric value. A metric is a number that is used as a standard of measurement for the links of a network. Each link is assigned a metric to represent anything from monetary cost to use the line, to the amount of available bandwidth. When Router A is presented with a packet bound from Router C, the routing table shows two possible paths to choose from. The first choice is to send the packet from Router A directly over the link to Router C. The second option is to send the packet from Router A to Router B and then on to Router C. The routing algorithm is used to determine which option is best.

Some routing protocols might only provide one metric to the routing algorithm, whereas others might provide up to ten. On the other hand, whereas two protocols might both send only one metric to the algorithm, the origin of that metric might differ from protocol to protocol. One routing protocol might give an algorithm the single metric of cost, but that cost could represent something different than another protocol using the same metric.

The algorithm in our example states that the best path is the one with the lowest metric value. Therefore, by adding the metric numbers associated with each possible link, we see that the route from Router A to Router B to Router C has a metric value of 5, while the direct link to Router C has a value of 6. The algorithm selects the A-B-C path and sends the information along.

7.1 Distance Vector Algorithms

- A distance vector algorithm uses metrics known as costs in order to help determine the best path to a destination. The path with the lowest total cost is chosen as the best path.
- When a router utilizes a distance vector algorithm, different costs are gathered by each router. These costs can be completely arbitrary numbers. Costs can also be dynamically gathered values, such as the amount of delay experienced by routers when sending packets over one link as opposed to another. All the costs are compiled and placed within the router's routing table and then they are used by the algorithm to calculate a best path for any given network scenario.
- Although there are many resources that will offer complex mathematical representations of what distance vector algorithms are and how they compute their decisions, the core concept remains the same - by adding the metrics for every optional path on a network, you will come up with at least one best path. The formula for this is as follows:

$$M(i,k) = \min [M(i,t) + M(t,k)]$$

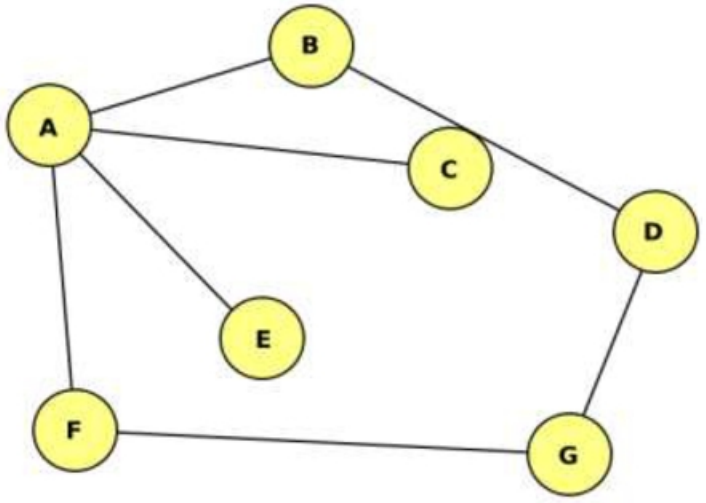
- This formula states that the best path between two networks (M(i,k)) can be found by finding the lowest (min) value of paths between all network points. Let's look again at the routing information in the table above. Plugging this information into the formula, we see that the route from A to B to C is still the best path:

$$5(A,C) = \min[2(A,B) + 3(B,C)]$$

- Whereas the formula for the direct route A to C looks like this:

$$6(A,C) = \min[6(A,C)]$$

- Distance vector algorithms are also known as Bellman-Ford routing algorithms and Ford-Fulkerson routing algorithms. In these algorithms, each router has a routing table which shows it the best route for any destination.
- In distance vector algorithms, each router has to follow the following steps:
    - It counts the weight of the links directly connected to it and saves the information to its table.
    - In a particular period of time, the router sends its table to its neighbor routers (not to all routers) and receives the routing table of each of its neighbors.
    - Based on the information the router receives from its neighbors' routing tables, it updates its own.
- Let's consider an example (the figure represented below).



- The cost of each link is set to 1. Thus, the least cost path is simply the path with the fewer hops. The table below represents each node knowledge about the distance to all other nodes:

Information stored at node	Distance to reach node						
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

- Initially, each node sets a cost of 1 to its directly connected neighbors and infinity to all the other nodes. Below is shown the initial routing table at node A:

Destination	Cost	Next Hop
B	1	B
C	1	C
D	∞	-
E	1	E
F	1	F
G	∞	-

- During the next step, every node sends a message to its directly connected neighbors. That message contains the node's personal list of distances. Node F, for example, tells node A that it can reach node G at cost of 1; node A also knows that it can reach F at a cost of 1, so it adds these costs to get the cost of reaching G by means of F. Because 2 is less than the current cost of infinity, node A records that it can reach G at a cost of 2 by going through F. Node A learns from C that node B can be reached from C at a cost of 1, so it concludes that the cost of reaching B via C is 2. Because this is worse than the current cost of reaching B, which is 1, the new information is ignored. The final routing table at node A is shown below:

Destination	Cost	Next Hop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

- The process of getting consistent routing information to all the nodes is called convergence. The final set of costs from each node to all other nodes is shown in the table below:

Information stored at node	Distance to reach node						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

- The cost of each link is set to 1. Thus, the least cost path is simply the path with the fewer hops.
- **One of the problems** with distance vector algorithms is called "**count to infinity**." Let's examine the following problem with an example:
- Consider a network with a graph as shown below. There is only one link between D and the other parts of the network.

with vectors

d [A][A] = 0 d [A][B] = 1 d [A][C] = 2 d [A][D] = 3

	A	B	C	D
A	0	1	2	3
B	1	0	1	2
C	2	1	0	1
D	3	2	1	0

- **Now the C to D link crashes So cost [C][D] = ∞ C used to forward any packets with address D directly on the CD** link, but now link is down, so C has to recompute its distance vector (and make a new choice of how to forward packets to D) - similarly D has to update its vector. After updating their vectors at C and D, we have

	A	B	C	D
A	0	1	2	3
B	1	0	1	2
C	2	1	0	3
D	∞	∞	∞	0

- C views B as the best route to D, with cost 1 + 2, so C sends new vector to B. B learns that its former choice for sending to D via C now has higher cost, so B should recompute its vector.

	A	B	C	D
A	0	1	2	3
B	1	0	1	4
C	2	1	0	3
D	∞	∞	∞	0

- View of B is that routing to D can either go via A or C with equal cost - B sends updated vector. Both A and C get updated vector from B and learn that their preferred route to D now has higher cost, so they recompute their own vectors.

	A	B	C	D
A	0	1	2	5
B	1	0	1	4
C	2	1	0	5
D	∞	∞	∞	0

- Then A and C send their vectors, B has to update its vector again, sending another round to A and C, obtaining.

	A	B	C	D
A	0	1	2	7
B	1	0	1	6
C	2	1	0	7
D	∞	∞	∞	0

- Notice that the routing table is very slowly converging to the fact that

  d [x][D] = ∞ for x = A or x = B or x = C

- This process loops until all nodes find out that the weight of link to D is infinity. In this way, experts say that distance vector algorithms have a slow convergence rate. In conclusion, distance vector algorithm is not robust. One way to solve this problem is for routers to send information only to the neighbors that are not exclusive links to the destination. For example, in this case, B should not send any information to C about D, because C is the only way to D.

❖ **Distance vector routing: Overview**

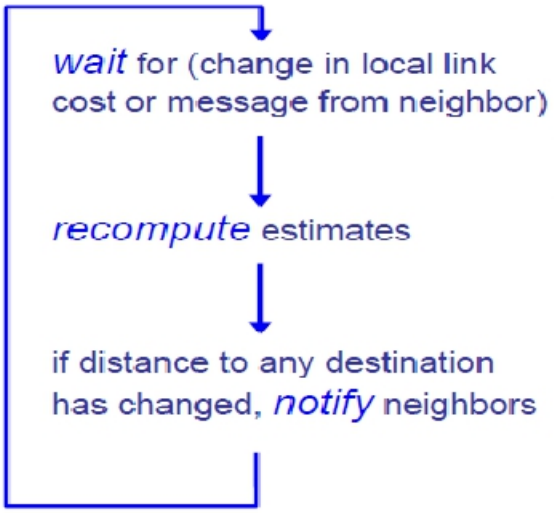
Iterative, asynchronous: each local iteration caused by:

- Local link cost change
- Distance vector update message from neighbor

Distributed:

- Each node notifies neighbors *only* when its DV changes
- Neighbors then notify their neighbors if necessary

Each node:

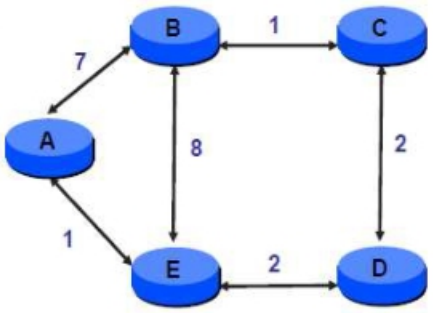


➢ **Step-by-Step:**

- $c(x,v)$ = cost for direct link from x to v
  - Node x maintains costs of direct links $c(x,v)$
- $D_x(y)$ = estimate of least cost from x to y
  - Node x maintains distance vector $\mathbf{D}_x = [D_x(y): y \in N]$
- Node x maintains its neighbors' distance vectors
  - For each neighbor v, x maintains $\mathbf{D}_v = [D_v(y): y \in N]$
- Each node v periodically sends D_v to its neighbors
  - And neighbors update their own distance vectors
  - $D_x(y) \leftarrow \min_v\{c(x,v) + D_v(y)\}$ for each node $y \in N$

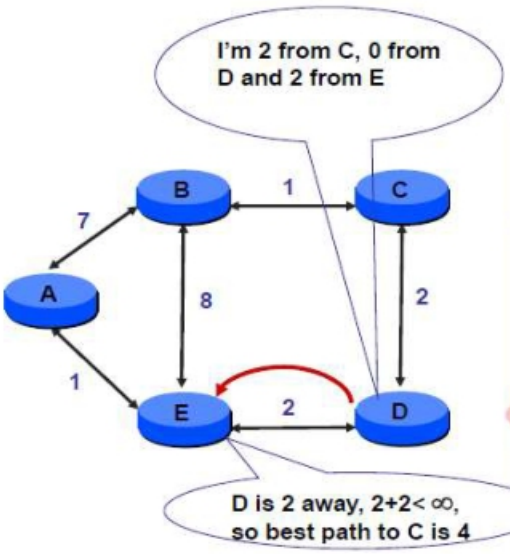
➢ **Example:**

- **Initial state**



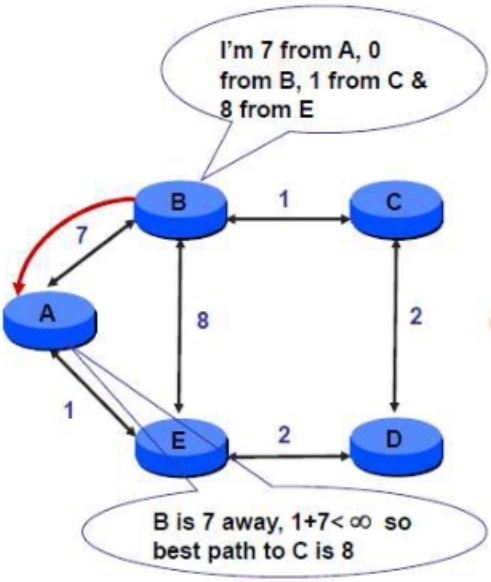
Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	∞	2	0

- **D sends vector to E**



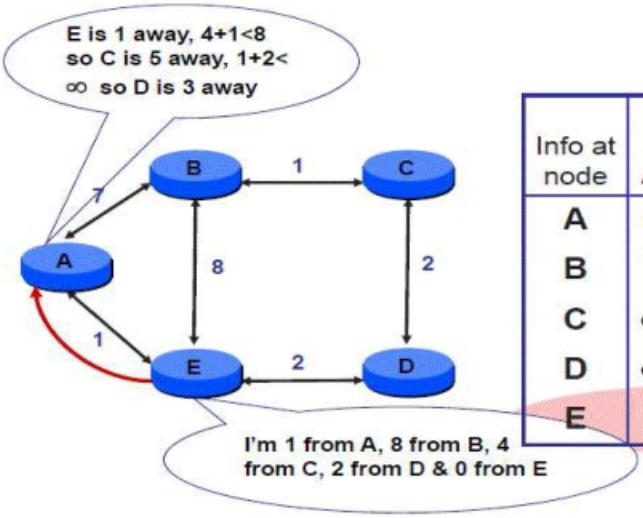
Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	**4**	2	0

- **B sends vector to E**



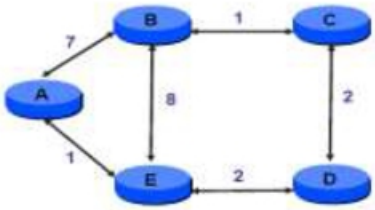
Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	**8**	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	4	2	0

- **E sends vector to A**



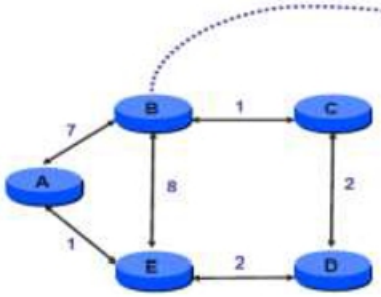
Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	**5**	**3**	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	4	2	0

- **….Until Convergence**



Info at node	Distance to Node				
	A	B	C	D	E
A	0	6	5	3	1
B	6	0	1	3	5
C	5	1	0	2	4
D	3	3	2	0	2
E	1	5	4	2	0

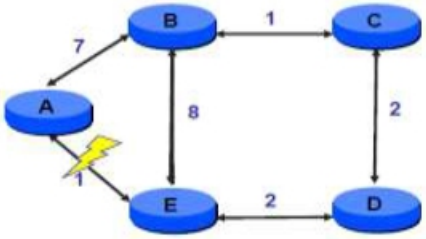
- **Node B's distance Vector**



Dest	Next hop		
	A	E	C
A	7	9	**6**
C	12	12	**1**
D	10	10	**3**
E	8	8	**5**

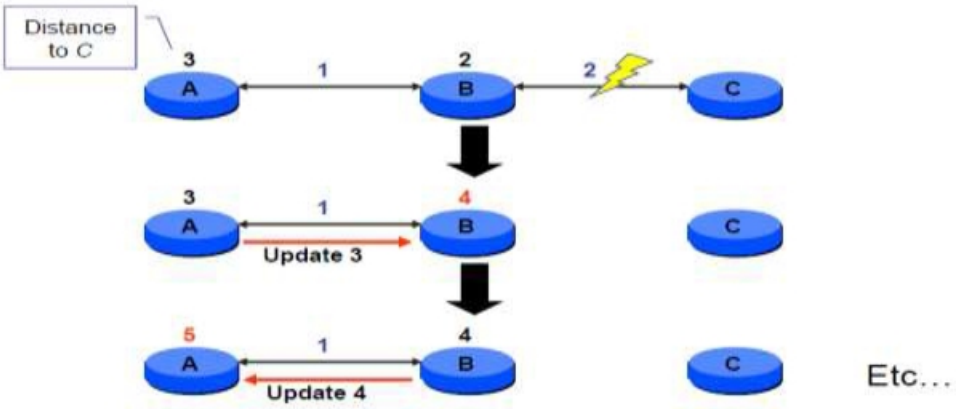
➢ **Handling Link Failure:**

- A marks distance to E as ∞ , and tells B
- E marks distance to A as ∞ , and tells B and D
- B and D recompute routes and tell C, E and E
- etc… until converge



Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	8	10	12
B	7	0	1	3	5
C	8	1	0	2	4
D	10	3	2	0	2
E	12	5	4	2	0

➢ **Counting to Infinity:**



7.2 Link-State Algorithms

Distance vector algorithms and link-state algorithms both favor the path with the lowest cost. However, link-state protocols work in more localized manner. Whereas a router running a distance vector algorithm will compute the end-to-end path for any given packet, a link-state protocol will compute that path as it relates to the most immediate link. That is, where a distance vector algorithm will compute the lowest metric between Network A and Network C, a link-state protocol will compute it as two distinct paths, A to B and B to C. This process is very efficient for larger environments. Link-state algorithms enable routers to focus on their own links and interfaces. Any one router on a network will only have direct knowledge of the routers and networks that are directly connected to it (or, the state of its own links). In larger environments, this means that the router will use less processing power to compute complicated paths. The router simply needs to know which one of its direct interfaces will get the information where it needs to go the quickest. The next router in line will repeat the process until the information reaches its destination. Another advantage to such localized routing processes is that protocols can maintain smaller routing tables. Because a link-state protocol only maintains routing information for its direct interfaces, the routing table contains much less information than that of a distance vector protocol that might have information for multiple routers. Like distance vector protocols, link-state protocols require updates to share information with each other. These routing updates, known as Link State Advertisements (LSAs), occur when the state of a router's links changes. When a particular link becomes unavailable (changes state), the router sends an update through the environment alerting all the routers with which it is directly linked.

❖ **In Link-State Algorithms, every router has to follow these steps:**

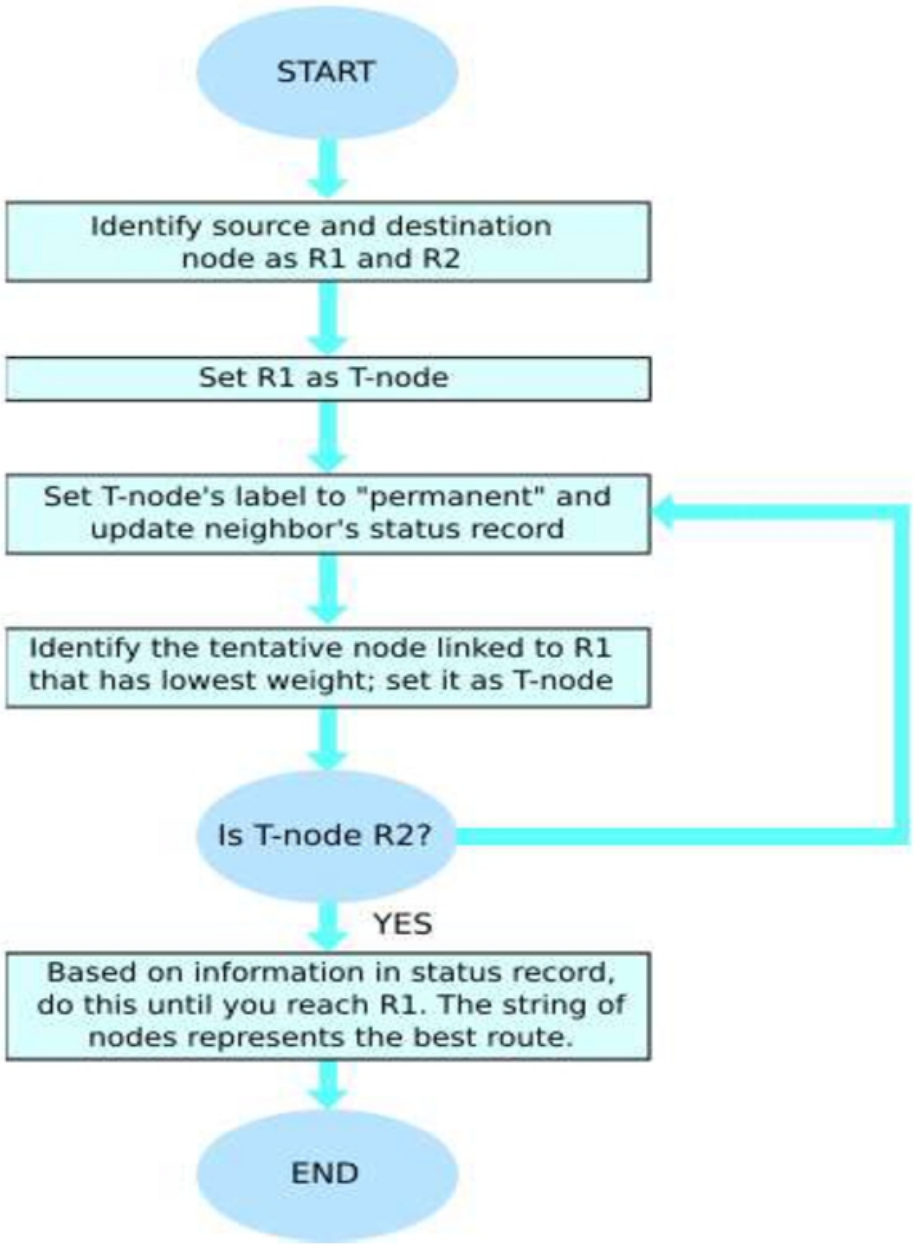
1. Identify the routers that are physically connected to them and get their IP addresses When a router starts working, it first sends a "HELLO" packet over network. Each router that receives this packet replies with a message that contains its IP address.
2. Routers measure the delay time (or any other important parameters of the network, such as average traffic) for neighbor routers. In order to do that, routers send echo packets over the network. Every router that receives these packets replies with an echo reply packet. By dividing round trip time by 2, routers can count the delay time. The delay time includes both transmission and processing times - the time it takes the packets to reach the destination and the time it takes the receiver to process it and reply.
3. Broadcast its information over the network for other routers and receive the other routers' information. In this step, all routers share their knowledge and broadcast their information to each other. In this way, every router can know the structure and status of the network.
4. Routers use an appropriate algorithm to identify the best route between two nodes of the network. In this step, routers choose the best route to every node. They do this using an algorithm, such as the Dijkstra shortest path algorithm. In this algorithm, a router, based on information that has been collected from other routers, builds a graph of the network. This graph shows the location of routers in the network and their links to each other. Every link is labeled with a number called the weight or cost. This number is a function of delay time, average traffic, and sometimes simply the number of hops between nodes. For example, if there are two links between a node and a destination, the router chooses the link with the lowest weight.

❖ **Dijkstra Algorithms**

➢ The Dijkstra Algorithm goes through the following steps:

1. The router builds a graph of the network. Then it identifies source and destination nodes, for example R1 and R2. The router builds then a matrix, called the "adjacency matrix." In the adjacent matrix, a coordinate indicates weight. [i, j], for example, is the weight of a link between nodes Ri and Rj. If there is no direct link between Ri and Rj, this weight is identified as "infinity."
2. The router then builds a status record for each node on the network. The record contains the following fields:
    - Predecessor field - shows the previous node.
    - Length field - shows the sum of the weights from the source to that node.
    - Label field - shows the status of node; each node have one status mode: "permanent" or "tentative."

3. In the next step, the router initializes the parameters of the status record (for all nodes) and sets their label to "tentative" and their length to "infinity".
4. During this step, the router sets a T-node. If R1 is to be the source T-node, for example, the router changes R1's label to "permanent." Once a label is changed to "permanent," it never changes again.
5. The router updates the status record for all tentative nodes that are directly linked to the source T-node.
6. The router goes over all of the tentative nodes and chooses the one whose weight to R1 is lowest. That node is then the destination T-node.
7. If the new T-node is not R2 (the intended destination), the router goes back to step 5.
8. If this node is R2, the router extracts its previous node from the status record and does this until it arrives at R1. This list of nodes shows the best route from R1 to R2.

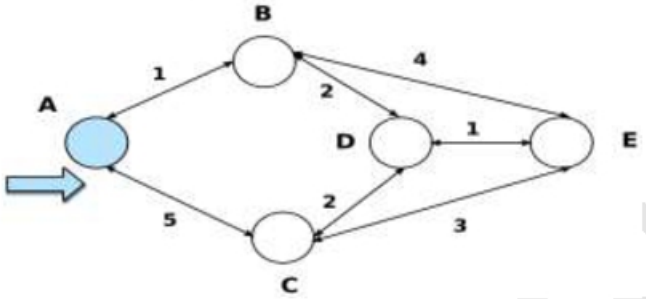


❖ **Dijkstra algorithm example:**

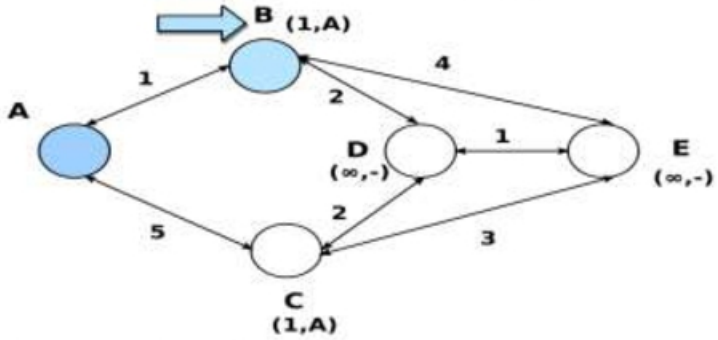
Example 1:

Let's find the best route between routers A and E. There are six possible routes between them (ABE, ACE, ABDE, ACDE, ABDCE, ACDBE), and it's obvious that ABDE is the best route because its weight is the lowest. But life is not always so easy, and there are some complicated cases in which we have to use algorithms to find the best route.

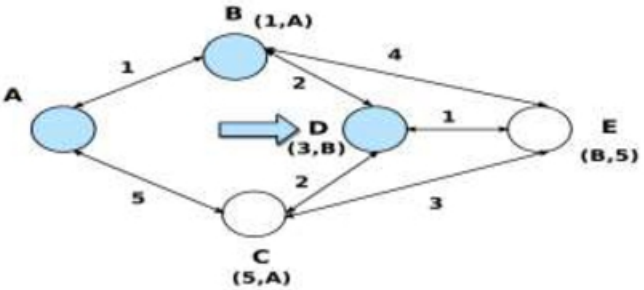
1. The source node (A) has been chosen as T-node, and so its label is permanent (permanent nodes are showed with filled circles and T-nodes with the -> symbol).



2. In this step, the status record of tentative nodes directly linked to T-node (B, C) has been changed. Also, because B has less weight, it has been chosen as T-node and its label has changed to permanent.



3. Like in step 2, the status records of tentative nodes that have a direct link to T-node (D, E), have been changed. Because router D has less weight, it has been chosen as T-node and its label has changed to permanent.

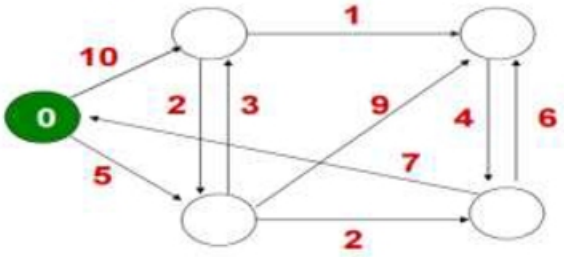


4. Because we do not have any tentative nodes, we just identify the next T-node. Because node E has the least weight, it has been chosen as T-node.

Now we have to identify the route. The previous node of E is node D, and the previous node of D is node B, and B's previous node is node A. So, we determine that the best route is ABDE. In this case, the total weigh is 4 (1+2+1). This algorithm works well, but it is so complicated that it may take a long time for routers to process it. That would cause the efficiency of the network to fail. Another note we should make here is that if a router gives the wrong information to other routers, all routing decisions will be ineffective.

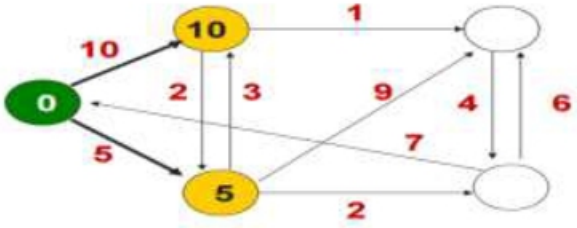
Example 2:

- **Step 1**



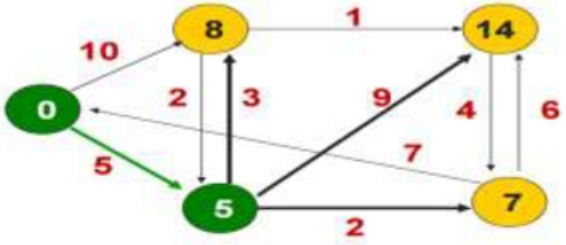
- Green nodes are "confirmed"
- Yellow nodes are "tentative"
- We can add ourselves to "confirmed"

- **Step 2:**



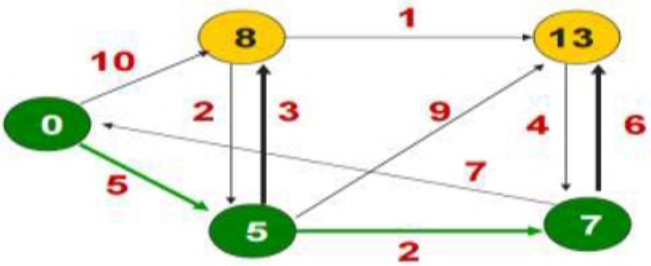
- Green nodes are "confirmed"
- Yellow nodes are "tentative"
- First look at neighbors
- "5" is cheaper than "10"
- We can confirm path with cost "5"

- **Step 3:**



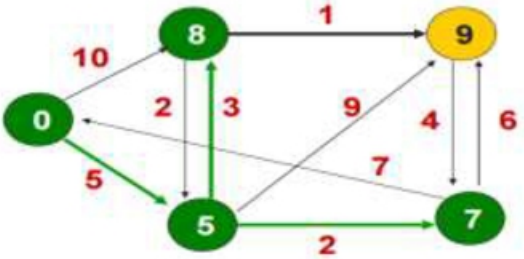
- Green nodes are "confirmed"
- Yellow nodes are "tentative"
- Update costs
- Look at 5's neighbors
- 7 is cheapest
- We can confirm path with cost 7

- **Step 4:**



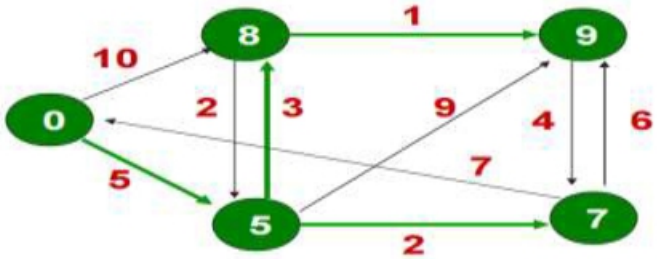
- Green nodes are "confirmed"
- Yellow nodes are "tentative"
- Update costs
- 7 has no new neighbors
- 8 is cheapest
- We can confirm 8

- **Step 5:**



- Green nodes are "confirmed"
- Yellow nodes are "tentative"
- Update costs
- No new neighbors
- 9 is cheapest
- We can path with cost 9

- **Step: Done:**



7.3 Link-State VS Distance-Vector

- **Message Complexity**
  - **LS:** with n nodes, E links, O(nE) message sent
  - **DV:** exchange between neighbors only
- **Speed of Convergence**
  - **LS:** relatively fast
  - **DV:** convergence time varies
    - May be routing loops
    - Count-to-infinity problem
- **Robustness:** what happens if router malfunctions?
  - **LS:**
    - Node can advertise incorrect link cost
    - Each node computes only its own table
  - **DV:**
    - Node can advertise incorrect path cost
    - **Each node's table used by others (error propagates)**

8 Routing in MANET's

Routing in Mobile Ad hoc networks is an important issue as these networks do not have fixed infrastructure and routing requires distributed and cooperative actions from all nodes in **the network. MANET's provide point to point** routing similar to Internet routing. The major difference between routing in MANET and regular internet is the route discovery mechanism. Internet routing protocols such as RIP or OSPF have relatively long converge times, which is acceptable for a wired network that has infrequent topology changes. However, a MANET has a rapid topology changes due to node mobility making the traditional internet routing protocols inappropriate. MANET-specific routing protocols have been proposed, that handle topology changes well, but they have large control overhead and are not scalable for large networks. Another major difference in the routing is the network address. In internet routing, the network address (IP address) is hier**archical containing a network ID and a computer ID on that network. In contrast, for most MANET's** the network address is simply an ID of the node in the network and is not hierarchical. The routing protocol must use the entire address to decide the next hop.

❖ **Some of the fundamental differences between wired networks & ad-hoc networks are:**

- Asymmetric links: - Routing information collected for one direction is of no use for the other direction. Many routing algorithms for wired networks rely on a symmetric scenario.
- Redundant links: - In wired networks, some redundancy is present to survive link failures and this redundancy is controlled by a network administrator. In ad-hoc networks, nobody controls redundancy resulting in many redundant links up to the extreme of a complete meshed topology.
- Interference: - In wired networks, links exist only where a wire exists, and connections are planned by network administrators. But, in ad-hoc networks links come and go depending on transmission characteristics, one transmission might interfere with another and nodes might overhear the transmission of other nodes.
- Dynamic topology: - The mobile nodes might move in an arbitrary manner or medium characteristics might change. This result in frequent changes in topology, so snapshots are valid only for a very short period of time. So, in ad-hoc networks, routing tables must somehow reflect these frequent changes in topology and routing algorithms have to be adopted.

❖ **Summary of the difficulties faced for routing in ad-hoc networks**

- ➢ Traditional routing algorithms known from wired networks will not work efficiently or fail completely. These algorithms have not been designed with a highly dynamic topology, asymmetric links, or interference in mind.
- ➢ Routing in wireless ad-hoc networks cannot rely on layer three knowledge alone. Information from lower layers concerning connectivity or interference can help routing algorithms to find a good path.
- ➢ Centralized approaches will not really work, because it takes too long to collect the current status and disseminate it again. Within this time the topology has already changed.
- ➢ Many nodes need routing capabilities. While there might be some without, at least one router has to be within the range of each node. Algorithms have to consider the limited battery power of these nodes.
- ➢ The notion of a connection with certain characteristics cannot work properly. Ad-hoc networks will be connectionless, because it is not possible to maintain a connection in a fast changing environment and to forward data following this connection. Nodes have to make local decisions for forwarding and send packets roughly toward the final destination.

- A last alternative to forward a packet across an unknown topology is flooding. This approach always works if the load is low, but it is very inefficient. A hop counter is needed in each packet to avoid looping, and the diameter of the ad-hoc network.

8.1 Types of MANET Routing Algorithms:

1. Based on the information used to build routing tables :
   - *Shortest distance algorithms*: algorithms that use distance information to build routing tables.
   - *Link state algorithms*: algorithms that use connectivity information to build a topology graph that is used to build routing tables.

2. Based on when routing tables are built:
   - ***Proactive algorithms*:** maintain routes to destinations even if they are not needed. Some of the examples are Destination Sequenced Distance Vector (DSDV), Wireless Routing Algorithm (WRP), Global State Routing (GSR), Source-tree Adaptive Routing (STAR), Cluster-Head Gateway Switch Routing (CGSR), Topology Broadcast Reverse Path Forwarding (TBRPF), Optimized Link State Routing (OLSR) etc.
     - Always maintain routes:- Little or no delay for route determination
     - Consume bandwidth to keep routes up-to-date
     - Maintain routes which may never be used
     - Advantages: low route latency, State information, QoS guarantee related to connection set-up or other real-time requirements
     - Disadvantages: high overhead (periodic updates) and route repair depends on update frequency
   - ***Reactive algorithms*:** maintain routes to destinations only when they are needed. Examples are Dynamic Source Routing (DSR), Ad hoc-On demand distance Vector (AODV), Temporally ordered Routing Algorithm (TORA), Associativity-Based Routing (ABR) etc
     - only obtain route information when needed
     - Advantages: no overhead from periodic update, scalability as long as there is only light traffic and low mobility.
     - Disadvantages: high route latency, route caching can reduce latency
   - ***Hybrid algorithms*:** maintain routes to nearby nodes even if they are not needed and maintain routes to far away nodes only when needed. Example is Zone Routing Protocol (ZRP).

Which approach achieves a better trade-off depends on the traffic and mobility patterns.

8.2 Dynamic Source Routing

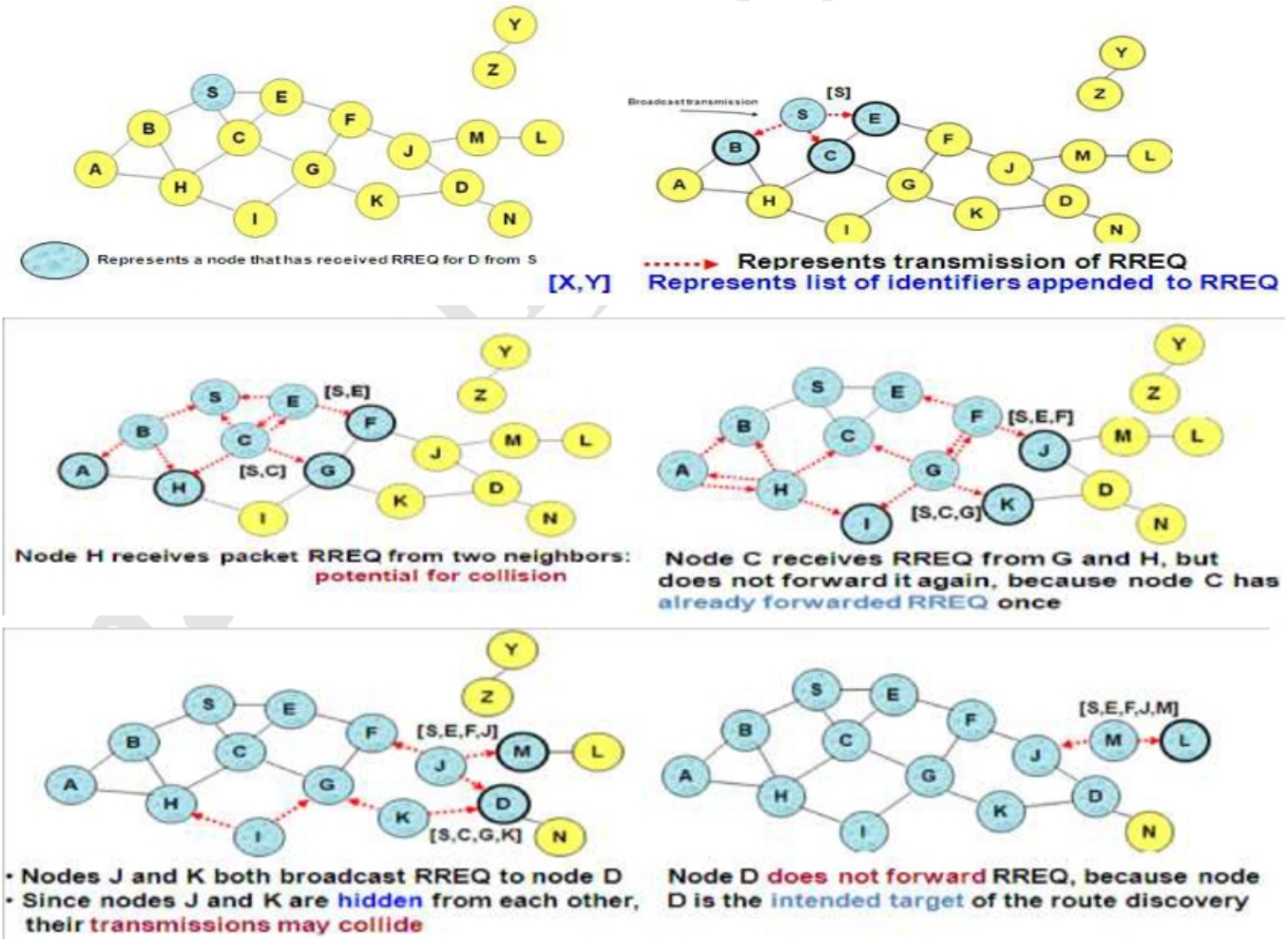
- The Dynamic Source Routing protocol (DSR) is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. The protocol is composed of the two main mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network. All aspects of the protocol operate entirely on-demand, allowing the routing packet overhead of DSR to scale automatically to only that needed to react to changes in the routes currently in use.

Route discovery. If the source does not have a route to the destination in its route cache, it broadcasts a route request (RREQ) message specifying the destination node for which the route is requested. The RREQ message includes a route record which specifies the sequence of nodes traversed by the message. When an intermediate node receives a RREQ, it checks to see if it is already in the route record. If it is, it drops the message. This is done to prevent routing loops. If the intermediate node had received the RREQ before, then it also drops the message. The intermediate node forwards the RREQ to the next hop according to the route specified in the header. When the destination receives the RREQ, it sends back a route reply message. If the destination has a route to the source in its route cache, then it can send a route response (RREP) message along this route. Otherwise, the RREP message can be sent along the reverse route back to the source. Intermediate nodes may also use their route cache to reply to RREQs. If an intermediate node has a route to the destination in its cache, then it can append the route to the route record in the RREQ, and send an RREP back to the source containing this route. This can help limit flooding of the RREQ. However, if the cached route is out-of-date, it can result in the source receiving stale routes.

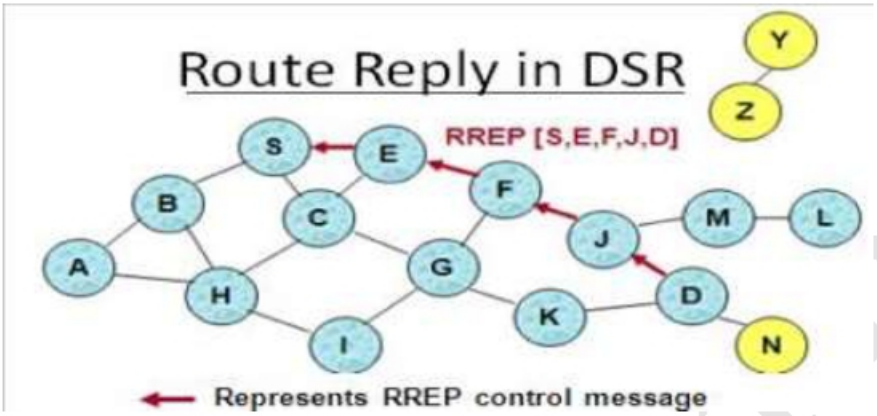
Route maintenance. When a node detects a broken link while trying to forward a packet to the next hop, it sends a route error (RERR) message back to the source containing the link in error. When an RERR message is received, all routes containing the link in error are deleted at that node.

As an example:

- Consider the following MANET, where a node S wants to send a packet to D, but does not know the route to D. So, it initiates a route discovery. Source node S floods Route Request (RREQ). Each node appends its own identifier when forwarding RREQ as shown below.

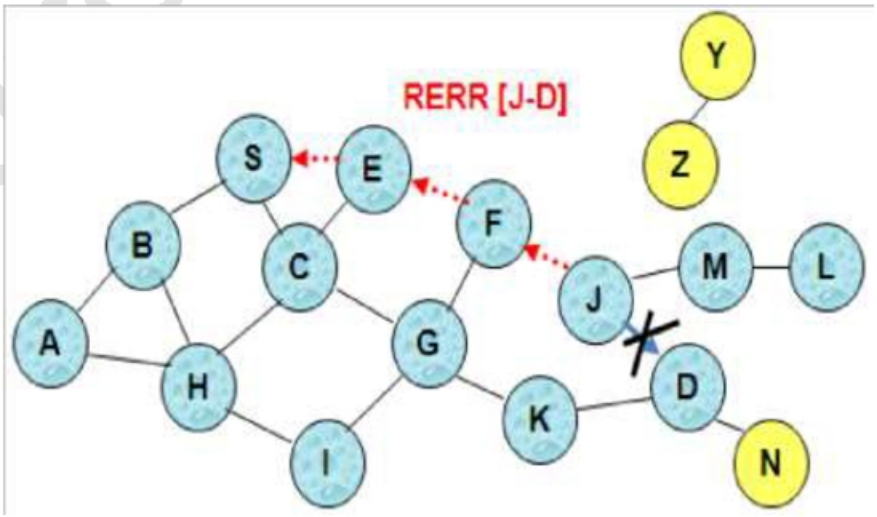


- Destination D on receiving the first RREQ, sends a Route Reply (RREP). RREP is sent on a route obtained by reversing the route appended to received RREQ. RREP includes the route from S to D on which RREQ was received by node D.



- Route Reply can be sent by reversing the route in Route Request (RREQ) only if links are guaranteed to be bi-directional. If Unidirectional (asymmetric) links are allowed, then RREP may need a route discovery from S to D. Node S on receiving RREP, caches the route included in the RREP. When node S sends a data packet to D, the entire route is included in the packet header {hence the name source routing}. Intermediate nodes use the source route included in a packet to determine to whom a packet should be forwarded.

- J sends a route error to S along route J-F-E-S when its attempt to forward the data packet S (with route SEFJD) on J-D fails. Nodes hearing RERR update their route cache to remove link J-D



Advantages of DSR:

- Routes maintained only between nodes who need to communicate-- reduces overhead of route maintenance
- Route caching can further reduce route discovery overhead
- A single route discovery may yield many routes to the destination, due to intermediate nodes replying from local caches

Disadvantages of DSR:

- Packet header size grows with route length due to source routing
- Flood of route requests may potentially reach all nodes in the network
- Care must be taken to avoid collisions between route requests propagated by neighboring nodes -- insertion of random delays before forwarding RREQ
- Increased contention if too many route replies come back due to nodes replying using their local cache-- Route Reply *Storm* problem. Reply storm may be eased by preventing a node from sending RREP if it hears another RREP with a shorter route
- An intermediate node may send Route Reply using a stale cached route, thus polluting other caches

An optimization for DSR can be done called as Route Caching. Each node caches a new route it learns by *any means*. In the above example, When node S finds route [S,E,F,J,D] to node D, node S also learns route [S,E,F] to node F. When node K receives Route Request [S,C,G] destined for node, node K learns route [K,G,C,S] to node S. When node F forwards Route Reply RREP [S,E,F,J,D], node F learns route [F,J,D] to node D. When node E forwards Data [S,E,F,J,D] it learns route [E,F,J,D] to node D. A node may also learn a route when it overhears Data packets. Usage of Route cache can speed up route discovery and can also reduce propagation of route requests. The disadvantages are, stale caches can adversely affect performance. With passage of time and host mobility, cached routes may become invalid.

8.3 Cluster based routing protocol (CBRP)

8.3.1 Concept of clustering

- Clustering is a process that divides the network into interconnected sub-structures called *clusters*.
- Each cluster has a *cluster-head* as a coordinator within the sub-structure, which acts as a medium for data transfer between the nodes.
- Cluster heads communicate with each other by using *gateway nodes*.
- The Gateway node has two or more cluster heads as its neighbors or— when the clusters are disjoint—at least one cluster head and another gateway node.

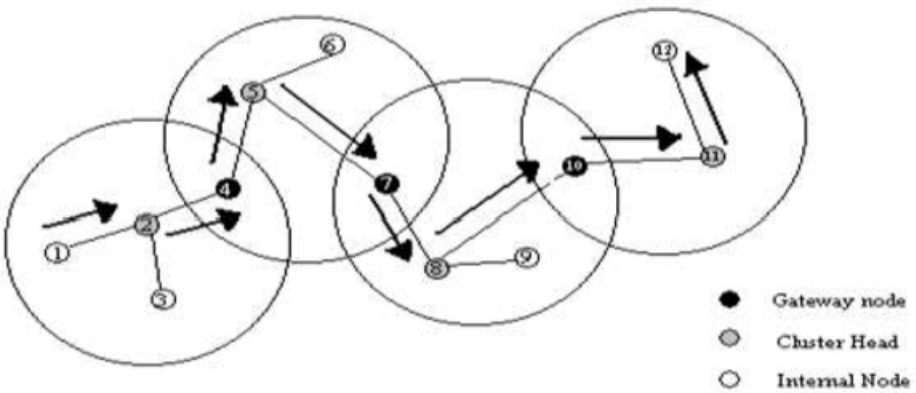


Figure: Different States in which a Node can Exist

8.3.2 Cluster formation

- Cluster formation takes place using two mechanisms :
  - **Identifier based Clustering:** A node elects itself as the cluster head if it has the lowest/highest ID in its neighborhood. (Lowest-Id Heuristic)
  - **Connectivity-based Clustering:** The node, which has the most neighbor nodes, is elected as the cluster head. (Highest Degree Heuristic)
- The CBRP uses a variation of the lowest-ID algorithm, which is an identifier-based algorithm.
- Each node uses a neighbor table. Information stored in a neighbor table are :
  - Unique Node IDs
  - Role in the cluster (i.e. Cluster head or Member node)
  - Status of the link to that node (Unidirectional/Bidirectional).
- The neighbor table is maintained by periodically broadcasting HELLO messages. A HELLO message contains information about a node's state, its neighbor table and its cluster adjacency table.

8.3.3 Mechanism in CBRP

❖ **Routing Process:**

- CBRP uses two data structures to support the routing process:
  1) **The Cluster Adjacency Table (CAT)** - The CAT stores information about neighboring clusters, i.e. whether the links are bi-directional or unidirectional.
  2) **The Two-Hop Topology Database** - The two-hop topology database contains all nodes that are at most two hops away.
- The routing process works in two steps:
  1) Discovery of a route from a source node 'S' to a destination node 'D'.
  2) Actual transmission of the data packets.

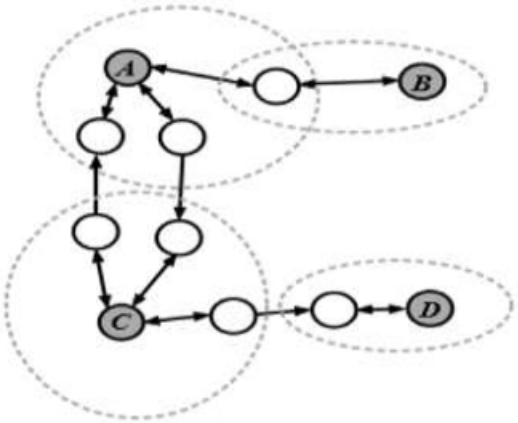
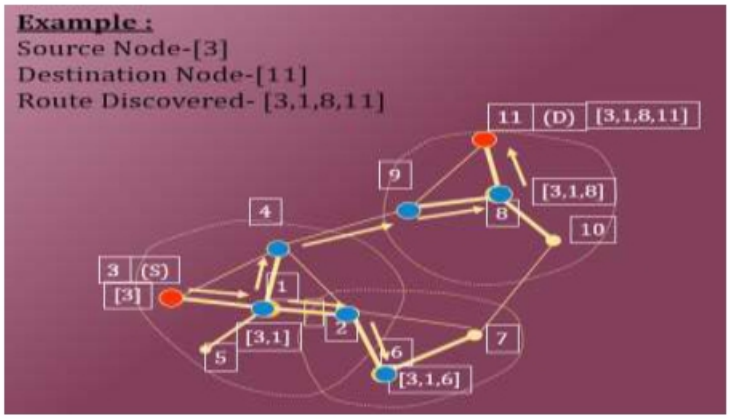


Figure: Presence of Bidirectional Links (A-B & A-C) & Unidirectional Links (C-D) between the Clusters.

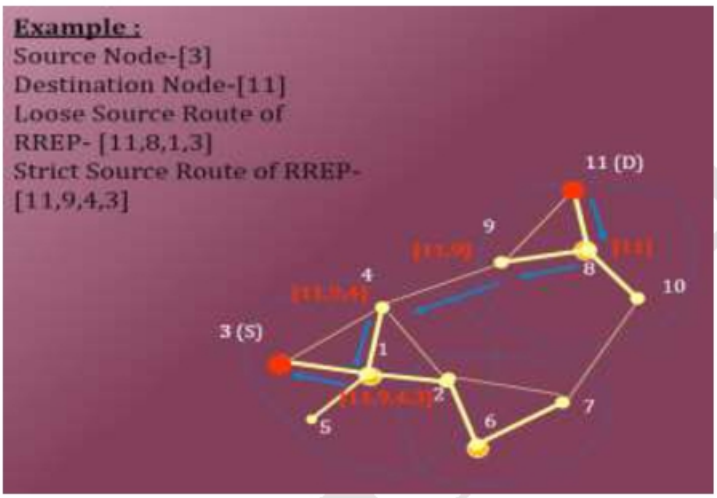
❖ **Route Discovery:**

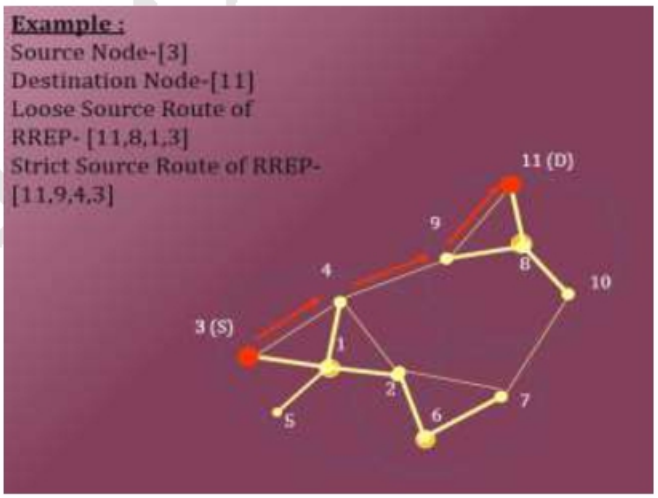
- In CBRP only cluster heads are flooded with route request package (RREQ). Gateway nodes receive the RREQs and forward them to the next cluster head.
- Initially, the source node 'S' broadcasts a RREQ with unique ID containing the address of the destination node 'D'.
- When a node 'N' receives a RREQ it does the following:
  - If 'N' is Gateway Node -> Forwards the RREQ to the next Cluster head 'C'.
  - If 'N' is Cluster Head -> Checks whether 'D' is a neighbor or is two-hops away from it. It then sends the RREQ to 'D'. Else broadcasts it to the neighboring Cluster head.



❖ **Route Reply:**

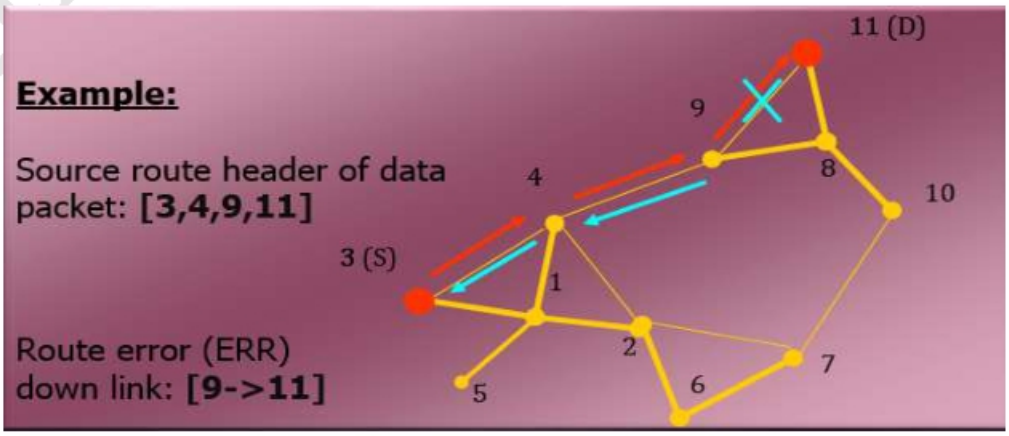
- If the RREQ reaches the destination node 'D',it contains the path called as "*loose source route*", [S,C1,C2,...,Ck,D].
- 'D' sends a Route Reply message (RREP) back to S using the reversed loose source route [D, Ck,...,C1,S], i.e. RREP is sent back to source along reversed loose source route of cluster heads.
- Every time a cluster head receives this RREP it computes a *strict source route*, which then consists only of nodes that form the shortest path within each cluster.





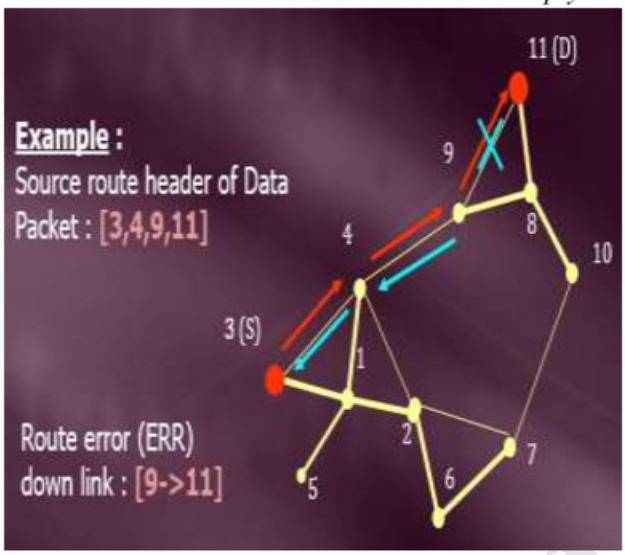
❖ **Route Error Detection:**

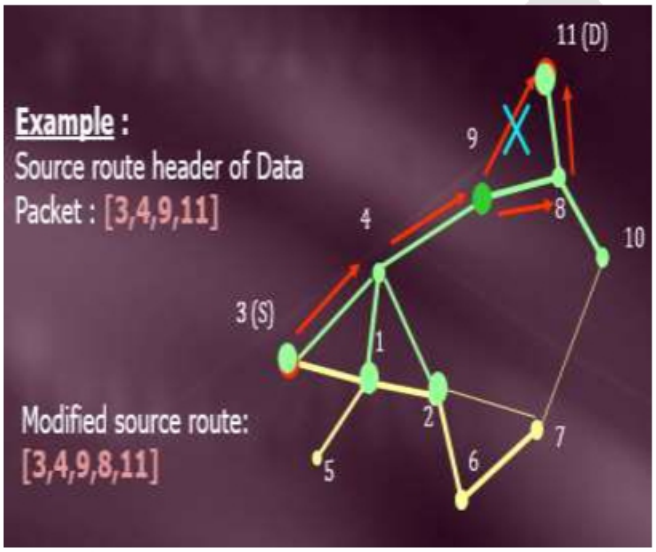
- After determining the route, source routing is used for actual packet transmission.
- A forwarding node sends a Route Error Message (ERR) to packet source if the next hop in source route is unreachable.

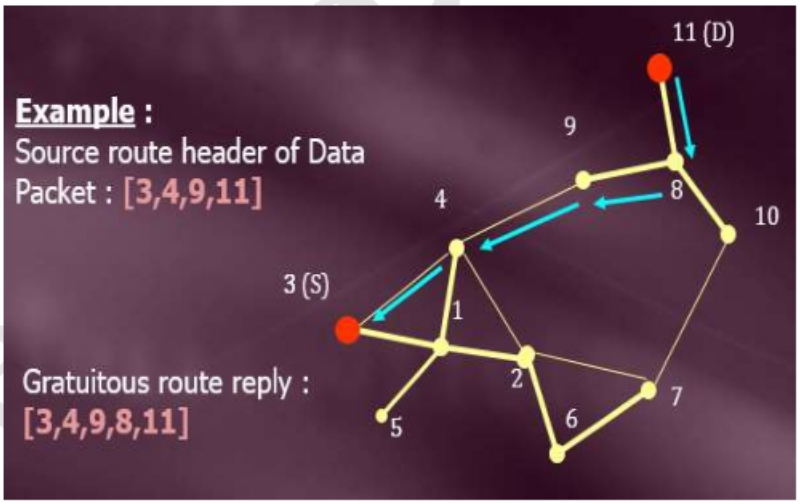


❖ **Local Route Repair:**

- Objective
  - Increase Packet Delivery Ratio.
  - Save Route Rediscovery flooding traffic.
  - Reduce overall route acquisition delay.
- A forwarding node repairs a broken route using its 2-hop-topology information and then modifies the source route header accordingly.
- Destination node sends a *Gratuitous Route Reply* to inform source of the modified route.







Advantages of CBRP

- Clustering approach minimizes on-demand route discovery traffic and routing overhead.
- **Uses "local repair" mechanism to reduce route acquisition delay and new route re**-discovery traffic.
- Increases the packet delivery ratio to a great extent.

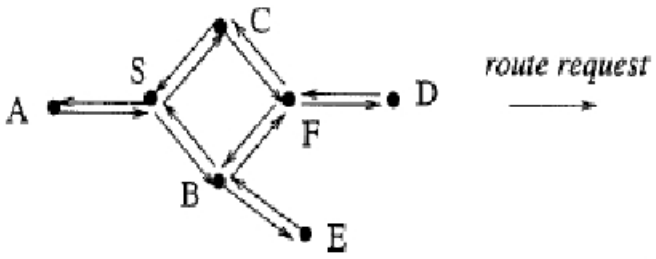
Disadvantages of CBRP

- With increase in cluster size, the overhead per packet increases due to source routing.
- Every node of the route has to be stored in the routed packet. So the packet size rises proportional to the path length of the route.
- The transmission time increases with increase in cluster size and path length of the route.

8.4 Location Routing applied to networks

- The use of location information has been used by Personal Communication Service (PCS), where paging on cell phones is limited to cells close to the last reported mobile host.
- **Metricom's Ricochet** uses packet radio to forward packets one hop closer to the destination
- DREAM maintains location information in the routing table, periodically broadcasting location information

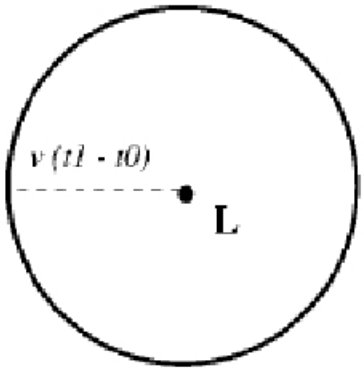
❖ Improving Flooding

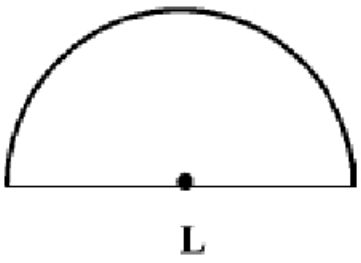


- S (sender) needs to send to D (destination), it broadcasts a route request to all neighbors (neighbors can communicate with each other directly over a wireless link)
- A node receiving the request compares D to itself, if it is a match, it is a route request to itself
- Otherwise the node broadcasts to its neighbors (requests are prevented from being repeatedly sent by sequence numbers)
- As the route progresses, the route so far is included with the request.
- If D receives the request, it sends a reply to S by reversing the route included with the request
- If S does not receive a reply within a timeout period, it sends another request with a new sequence number
- A route request is initiated when either the route is broken or the route is unknown
- S finds that the route is broken only when it attempts to use it and receives an error message from a node indicating that its next hop is not available
- This algorithm will reach every node that is reachable within an ad-hoc network
- This is where location information is useful, to reduce the number of nodes in the ad-hoc network receiving the route request information, i.e.: limited flooding

8.5 Location Aided Routing (LAR)

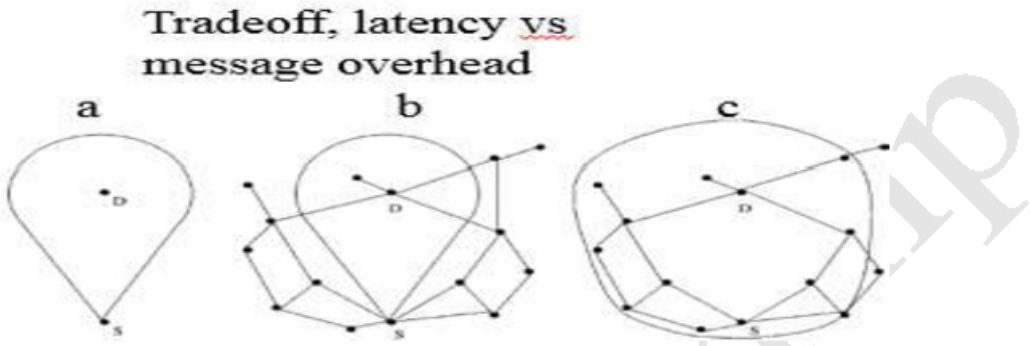
- Location of nodes may be found by GPS
- If the previous location at time t_0 is known and a velocity is known, the location at time t_1 can be predicted to be in a circle
- If it is known that the node is traveling in a particular direction, the location can be a semicircle.
- Route discovery then can proceed only within the expected location
- If it is not found, due to a higher than expected velocity, then the algorithm reverts to flooding with one modification





❖ **Request Zones**

- The LAR algorithm includes in the request zone all the nodes in the circle or semi-circle
- Other areas may be included
  - An expected zone does not include node S, both S and D must be in the request zone
- Fig a) may be inadequate (route not found within timeout period), the zone must be expanded



❖ **Determining Membership in a Zone**

- Nodes not in the request zone do not forward requests to its neighbors
- **LAR Scheme 1:** Rectangular area || X & Y axis
  - At time t_0, D is known to be at X_d, Y_d with velocity v
  - Figure A: S is outside the Expected zone
  - Figure B: S is inside the Expected zone
  - Nodes inside the request zone forward requests, others outside the request zone do not forward requests

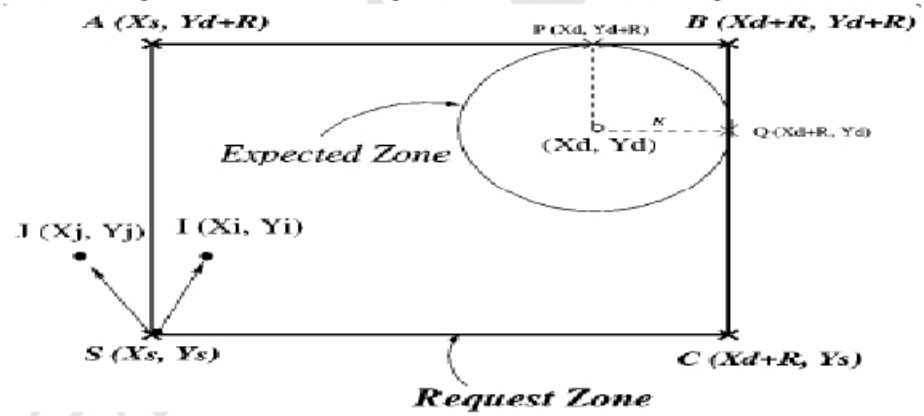


Figure A

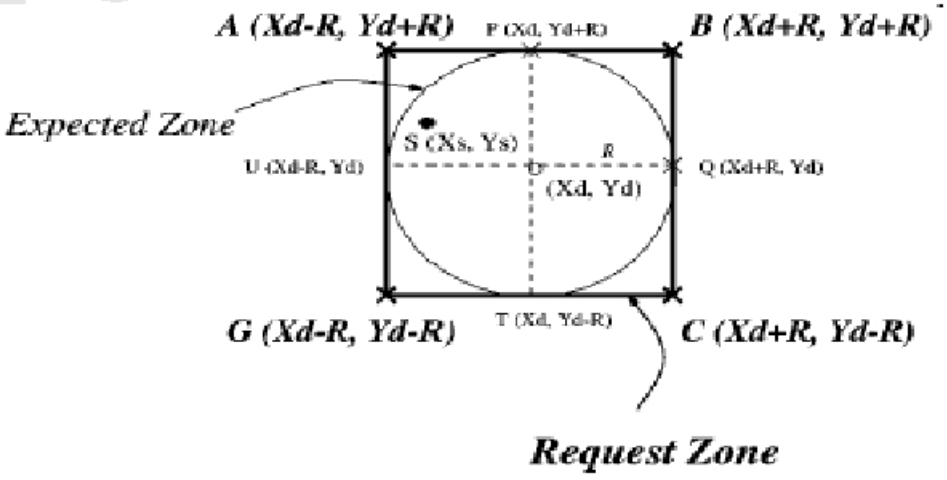
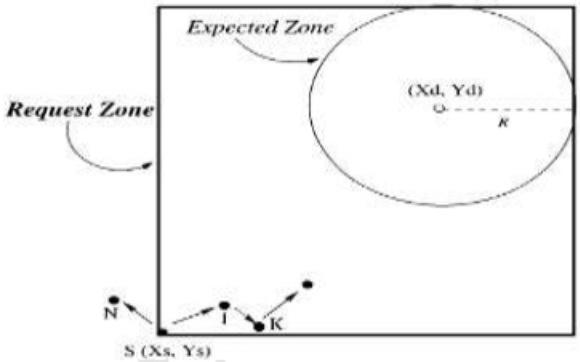


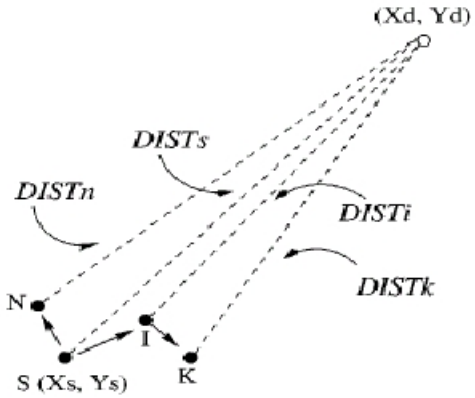
Figure B

- When D receives the route request, it replies with its current location, current time, and possibly current or average speed
- Node S records this information to predict Ds location in the future
- **LAR Scheme 2:** Instead of including the zone coordinates in the route request, Scheme 2 includes the known X_d and Y_d coordinates of node D at time t_0, also S includes $Dist_s$, the distance from S to D at time t_0 in the route request.
- When node I receives the request, it calculates its distance to D and records $Dist_i$
- For some parameters α and β,

  if ($\alpha(Dist_s) + \beta \geq Dist_i$) //they are closer to D
  Replace $Dist_s$ with $Dist_i$
  Forward the request
  Else
  Discard the request
- Initially α=1 and β=0; they can be used to trade off the probability of finding a route on the first attempt with the cost of finding a route, when the location error (for D) is non-zero, or when D will move a significant distance.
- Each node that receives the request and is closer to D forwards the request with its $Dist_j$ replacing the received $Dist_i$
- Nodes that receive a duplicate discards the duplicate
- **Scheme 1:** I and K forward the request, they are inside the expected zone, N discards the request, it is outside the expected zone



- **Scheme 2:** assume α=1 and β=0, N and I forward the request, they are closer to D than S, K discards the request, it is further from D than I



❖ **Location Estimates Error**

- **Scheme 1:** let e be the max error in the coordinates of D, then the expected zone is a circle of radius $e + v\ (t_1 - t_0)$, making the request zone correspondingly larger
- **Scheme 2:** no change is used, however, there is a higher chance that a path to D will not be included causing a timeout and another route discovery

8.6 Destination sequence distance vector routing (DSDV)

- Destination sequence distance vector (DSDV) routing is an example of proactive algorithms and an enhancement to distance vector routing for ad-hoc networks. Distance vector routing is used as routing information protocol (RIP) in wired networks. It performs extremely poorly with certain network changes due to the count-to-infinity problem. Each node exchanges its neighbor table periodically with its neighbors. Changes at one node in the network propagate slowly through the network. The strategies to avoid this problem which are used in fixed networks do not help in the case of wireless ad-hoc networks, due to the rapidly changing topology. This might create loops or unreachable regions within the network.
- DSDV adds the concept of sequence numbers to the distance vector algorithm. Each routing advertisement comes with a sequence number. Within ad-hoc networks, advertisements may propagate along many paths. Sequence numbers help to apply the advertisements in correct order. This avoids the loops that are likely with the unchanged distance vector algorithm.

Each node maintains a routing table which stores next hop, cost metric towards each destination and a sequence number that is created by the destination itself. Each node periodically forwards routing table to neighbors. Each node increments and appends its sequence number when sending its local routing table. Each route is tagged with a sequence number; routes with greater sequence numbers are preferred. Each node advertises a monotonically increasing even sequence number for itself. When a node decides that a route is broken, it increments the sequence number of the route and advertises it with infinite metric. Destination advertises new sequence number.

When X receives information from Y about a route to Z,

- Let destination sequence number for Z at X be S(X), S(Y) is sent from Y
- If S(X) > S(Y), then X ignores the routing information received from Y
- If S(X) = S(Y), and cost of going through Y is smaller than the route known to X, then X sets Y as the next hop to Z
- If S(X) < S(Y), then X sets Y as the next hop to Z, and S(X) is updated to equal S(Y)

Besides being loop-free at all times, DSDV has low memory requirements and a quick convergence via triggered updates. Disadvantages of DSDV are, large routing overhead, usage of only bidirectional links and suffers from count to infinity problem.

9 Security in MANET's

Securing wireless ad-hoc networks is a highly challenging issue. Understanding possible form of attacks is always the first step towards developing good security solutions. Security of communication in MANET is important for secure transmission of information. Absence of any central co-ordination mechanism and shared wireless medium makes MANET more vulnerable to digital/cyber attacks than wired network there are a number of attacks that affect MANET. These attacks can be classified into two types:

1) **External Attack**: External attacks are carried out by nodes that do not belong to the network. It causes congestion sends false routing information or causes unavailability of services.
2) **Internal Attack**: Internal attacks are from compromised nodes that are part of the network. In an internal attack the malicious node from the network gains unauthorized access and impersonates as a genuine node. It can analyze traffic between other nodes and may participate in other network activities.

- **Denial of Service attack**: This attack aims to attack the availability of a node or the entire network. If the attack is successful the services will not be available. The attacker generally uses radio signal jamming and the battery exhaustion method.
- **Impersonation**: If the authentication mechanism is not properly implemented a malicious node can act as a genuine node and monitor the network traffic. It can also send fake routing packets, and gain access to some confidential information.
- **Eavesdropping:** This is a passive attack. The node simply observes the confidential information. This information can be later used by the malicious node. The secret information like location, public key, private key, password etc. can be fetched by eavesdropper.
- **Routing Attacks:** The malicious node makes routing services a target because it's an important service in MANETs. There are two flavors to this routing attack. One is attack on routing protocol and another is attack on packet forwarding or delivery mechanism. The first is aimed at blocking the propagation of routing information to a node. The latter is aimed at disturbing the packet delivery against a predefined path.
- **Black hole Attack:** In this attack, an attacker advertises a zero metric for all destinations causing all nodes around it to route packets towards it. A malicious node sends fake routing information, claiming that it has an optimum route and causes other good nodes to route data packets through the malicious one. A malicious node drops all packets that it receives instead of normally forwarding those packets. An attacker listen the requests in a flooding based protocol.
- **Wormhole Attack:** In a wormhole attack, an attacker receives packets at one point in the network, —tunnelsll them to another point in the network, and then replays them into the network from that point. Routing can be disrupted when routing control message are tunnelled. This tunnel between two colluding attacks is known as a wormhole.
- **Replay Attack:** An attacker that performs a replay attack are retransmitted the valid data repeatedly to inject the network routing traffic that has been captured previously. This attack usually targets the freshness of routes, but can also be used to undermine poorly designed security solutions.
- **Jamming:** In jamming, attacker initially keep monitoring wireless medium in order to determine frequency at which destination node is receiving signal from sender. It then transmit signal on that frequency so that error free receptor is hindered.
- **Man- in- the- middle attack:** An attacker sites between the sender and receiver and sniffs any information being sent between two nodes. In some cases, attacker may impersonate the sender to communicate with receiver or impersonate the receiver to reply to the sender.
- **Gray-hole attack**: This attack is also known as routing misbehavior attack which leads to dropping of messages. Gray-hole attack has two phases. In the first phase the node advertise itself as having a valid route to destination while in second phase, nodes drops intercepted packets with a certain probability.