Mobile Adhoc Networks

Routing Protocols for Adhoc Wireless Network

- mobile nodes, network topology changing
- routing protocol of wired network can not be directly applied in adhoc network
- highly dynamic topology, absence of established infrastructure for centralized administration,
- bandwidth constrained wireless links and resource (energy) constrained nodes

Issues in designing of Routing protocols

- mobility of nodes, resource constraints, error-prone channel state and
- hidden, exposed terminal problems
- mobility
- highly dynamic due to node mobility, on-going suffers frequent path breaks
- disruption due to movement of intermediate nodes or end node movement
- wired network all nodes are stationary, find alternate routes during path breaks, convergence is very slow
- therefore wired network routing protocols can not be used in adhoc network, mobility of nodes results in frequently changing network topologies
- need efficient and effective mobility management

Issues in designing of Routing protocols

- bandwidth constraint
- abundant bandwidth in wired network, exploitation of wavelength division multiplexing (WDM)
- wireless network radio band is limited, data rate offered very less
- routing protocols use the bandwidth optimally
- limited bandwidth availability imposes a constraint on routing protocols in maintaining topological information,
- more control overhead more bandwidth wastage

Issues in designing of Routing protocols

- error-prone shared broadcast radio channel
- time varying characteristics in terms of link capacity and link error probability
- routing protocol interacts with MAC layer to find alternate routes through better quality links
- transmission - collision of data and control packets, hidden terminal problem
- routing protocol should find paths with less congestion

Issues in designing of Routing protocols

- Hidden and exposed terminal problems
- MACA requires that transmitting node first explicitly notifies all potentials hidden nodes about the forthcoming transmission by means of
- a two-way handshake control protocol called RTS-CTS protocol exchange - it reduces the probability of collisions
- to increase efficiency, improved version of MACA known as MACAW
- MACAW requires that the receiver acknowledges each successful reception of a data packet
- transmission is four-way exchange RTS-CTS-DATA-ACK

Issues in designing of Routing protocols

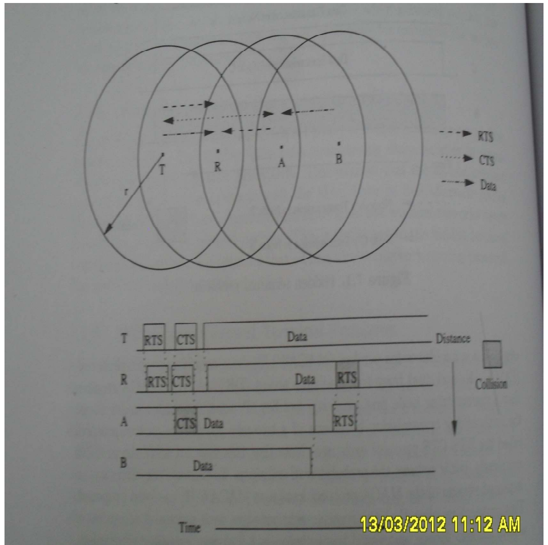


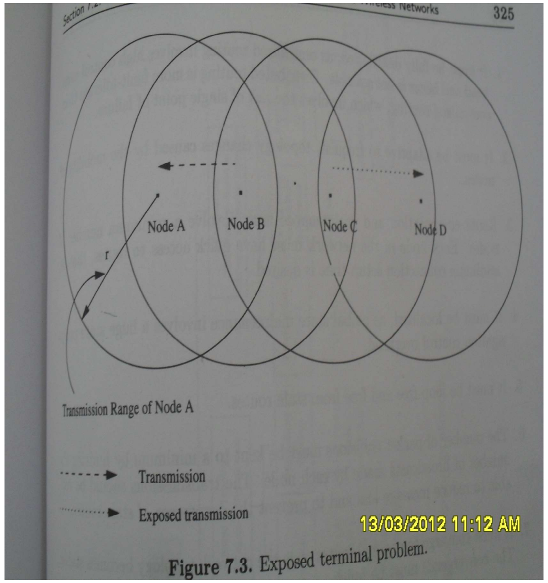
Figure 7.2. Hidden terminal problem with RTS-CTS-Data-ACK scheme.

src: textbook

Issues in designing of Routing protocols

- in absence of bit errors and mobility, RTS-CTS control packet exchange cannot ensure collision free data transmission that has no interference from hidden terminals
- important assumption that every node in the capture area of receiver (transmitter) receives the CTS (RTS) cleanly
- nodes that do not hear either of these clearly can disrupt the successful transmission of DATA or ACK
- one situation occurs when node A hidden from transmitter T and within the capture area of the receiver R,

Issues in designing of Routing protocols



src: textbook

Issues in designing of Routing protocols

- does not hear the CTS properly because it is within the capture area of node B that is transmitting and the hidden from both R and T,
- in this case node A did not successfully receive the CTS originated by node R and hence assumes that there is no on-going transmission in the neighborhood
- since node A is hidden from node T any attempt to originate its own RTS would result in collision of on-going transmission between nodes T and R
- exposed terminal problem (discussed earlier)
- resource constraints: battery life and processing power
- increasing both makes node bulky and less portable
- optimally manage resources

Characteristics of Ideal Routing protocol

- must be fully distributed; centralized routing involves high control overhead and hence not scalable
- distributed routing more fault tolerant
- centralized routing risk of single point of failure
- adaptive to frequent topology changes due to mobility of nodes
- route computation and maintenance must involve minimum number of nodes
- each node must have quick access to routes, i.e., minimum connection setup time is desired
- must be localized, as global state maintenance a huge state propagation control overhead
- must be loop free and free and from stale routes

Characteristics of Ideal Routing protocol

- number of packets collisions must be kept to minimum by limiting the number of broadcasts made by each node
- transmission should be reliable to reduce message loss and to prevent the occurrence of state routes
- must converge to optimal routes once the network topology becomes stable, convergence must be quick
- must optimally use scarce resources such as bandwidth, computing power, memory and batter power
- must provide QoS, support for time sensitive traffic
- every node should try to store information regarding the stable local topology only;
- changes in remote parts of network must not cause updates in the topology information maintained by the node

Classification of Routing protocols

- broadly classified into four categories based on
  - routing information update mechanism
  - use of temporal information for routing
  - routing topology
  - utilization of specific resources
- based on routing information update mechanism three major categories
- proactive or table-driven routing protocols
- network topology information maintained in routing tables, exchange periodically
- information flooded in the whole network
- whenever a node requires a path to a destination, it runs an appropriate path finding algorithm on the topology information it maintains

Classification of Routing protocols

- reactive or on-demand routing protocols
- do not maintain the network topology information
- they obtain the necessary path when it is required, by using a connection establishment process
- do not exchange routing information periodically
- hybrid routing protocols
- best features of the above two categories
- nodes within a certain distance from the node concerned, or within a particular geographical region,
- are said to be within the routing zone of the given node
- for routing within this zone, a table-driven approach is used
- nodes that are located beyond this zone, an on-demand approach is used

Classification of Routing protocols

- based on the use of temporal information for routing
- highly dynamic network, path breaks are frequent,
- the use of temporal information regarding the lifetime of the wireless links and lifetime of the paths selected assumes significance
- further classified into two types
- routing protocols using past temporal information
- use information about the past status of the links or the status of links at the time of routing to make routing decisions
- for example, the routing metric based on the availability of wireless links along with a shortest path-finding algorithm.
- topological changes may break the path, making the path undergo a resource wise expensive path reconfiguration process

Classification of Routing protocols

- routing protocols that use future temporal information
- use expected future status of wireless link to make approximate routing decisions
- future status like lifetime of the node, (remaining battery charge and discharge rate of non-replenishable resource),
- prediction of location, prediction of link availability
- based on routing topology
- Internet - hierarchical manner in order to reduce the state information maintained at the core routers
- adhoc network small number of nodes - either use flat topology or hierarchical topology for routing

Classification of Routing protocols

- flat topology based routing protocol
- use of flat addressing scheme similar to the one used in IEEE 802.3 LANs
- it assumes the presence of a globally unique addressing for nodes in an adhoc network
- hierarchical topology based routing protocol
- use of logical hierarchy in the network and associated addressing scheme
- hierarchy could be based on geographical information or on hop distance

Classification of Routing protocols

- Based on utilization of specific resources
- power aware routing
- aims at minimizing the consumption of important resource, battery power, may be locally or globally
- geographical information assisted routing
- improve the performance of routing and reduce the control overhead by utilizing the geographical information

Table driven Routing protocols

- extensions of the wired network routing protocols
- maintain global topology information in the form of tables at every node
- tables are updated frequently in order to maintain consistent and accurate network state information
- destination sequenced distance-vector routing protocol (DSDV)
- wireless routing protocol (WRP)
- source tree adaptive routing protocol (STAR)
- cluster head gateway switch routing protocol (CGSR)

Destination Sequenced Distance Vector Routing protocol (DSDV)

- one of the first protocol for adhoc network
- it is an enhanced version of distributed Bellman-Ford algorithm where
- each node maintains a table that contains the shortest distance and the first node on the shortest path to every other node in the network
- it incorporates table updates with increasing sequence number tags to prevent loops, to counter the count-to-infinity problem and for faster convergence
- it is table driven routing protocol, routes to all destinations are available at every node at all times

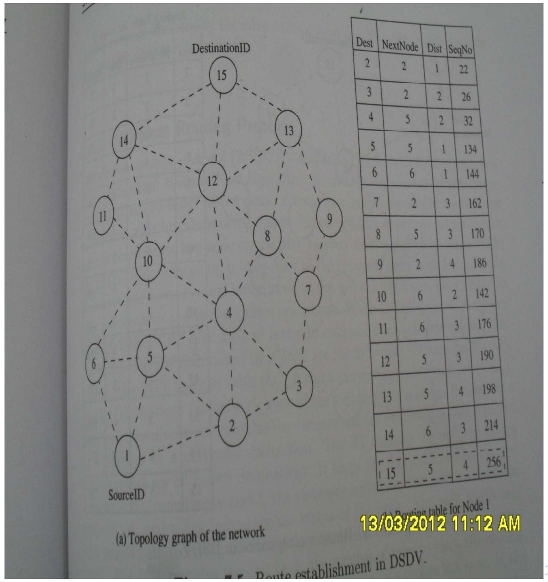
Destination Sequenced Distance Vector Routing protocol (DSDV)

- tables are exchanged between neighbors at regular intervals to keep an up-to-date view of the network topology
- tables are forwarded if a node observes a significant change in local topology
- table updates are of two types: incremental updates and full dumps
- incremental update takes a single network data packet unit (NDPU) while a full dump may take multiple NDPUs
- incremental updates are used when a node does not observe significant changes in the local topology
- full dump is done either when the local topology changes significantly or when an incremental update requires more than a single NDPU

Destination Sequenced Distance Vector Routing protocol (DSDV)

- tables updates are initiated by a destination with a new sequence number which is always greater than the previous one
- upon receiving an updated table, a node either updates its tables based on the received information or
- holds it for some time to select the best metric (lowest number of hops) received from multiple versions of the same update table from different neighboring nodes
- based on the sequence number of the table update, it may forward or reject the table
- node 1 is source node and node 15 is destination shortest route through node 5 and distance to it is 4

Destination Sequenced Distance Vector Routing protocol (DSDV)

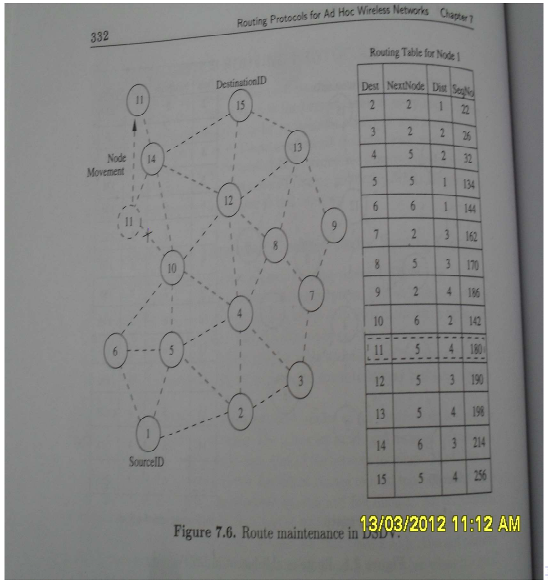


Dest	NextNode	Dist	SeqNo
2	2	1	22
3	2	2	26
4	5	2	32
5	5	1	134
6	6	1	144
7	2	3	162
8	5	3	170
9	2	4	186
10	6	2	142
11	6	3	176
12	5	3	190
13	5	4	198
14	6	3	214
15	5	4	256

Destination Sequenced Distance Vector Routing protocol (DSDV)

- reconfiguration of a path used by an on-going data transfer session is handled by the protocol in the following way:
- the end node of the broken link initiates a table update message with the broken link's weight assigned to infinity (∞) and
- with a sequence number greater than the stored sequence number for that destination
- each node upon receiving an update with weight ∞, quickly disseminates it to its neighbors in order to propagate the broken link information to the whole network
- a single link break leads to the propagation of table update information to the whole network
- a node always assigns an odd sequence number to the link break update to differentiate it from the even sequence number generated by the destination

Destination Sequenced Distance Vector Routing protocol (DSDV)



332 Routing Protocols for Ad Hoc Wireless Networks Chapter 7

Routing Table for Node 1

Dest	NextNode	Dist	SeqNo
2	2	1	22
3	2	2	26
4	5	2	32
5	5	1	134
6	6	1	144
7	2	3	162
8	5	3	170
9	2	4	186
10	6	2	142
11	5	4	180
12	5	3	190
13	5	4	198
14	6	3	214
15	5	4	256

Figure 7.6. Route maintenance in DSDV.

13/03/2012 11:12 AM

Destination Sequenced Distance Vector Routing protocol (DSDV)

- consider the case when node 11 moves from the current position,
- a neighbor node perceives the link break, it sets all the paths passing through the broken link with distance ∞
- for example, node 10 knows about the link break, it sets the path to node 11 as ∞ and broadcasts its routing table to its neighbors
- those neighbors detecting significant changes in their routing tables rebroadcast it to their neighbors
- in this way the broken link information propagates throughout the network

Destination Sequenced Distance Vector Routing protocol (DSDV)

- node 1 sets the distance to node 11 as ∞ when node 14 receives a table update message from node 11,
- it informs the neighbors about the shortest distance to node 11
- this information is also propagated throughout the network
- all nodes receiving the new update message with the higher sequence number set the new distance to node 11 in their corresponding tables
- the updated table at node 1 (shown in figure) the current distance from node 1 to node 11 has increased from three to four hops

Destination Sequenced Distance Vector Routing protocol (DSDV)

- advantages
- the availability of routes to all destinations at all times implies that much less delay is involved in the route setup process
- the mechanism of incremental updates with sequence number tags makes the existing wired network protocols adaptable to adhoc networks
- the updates are propagated throughout the network in order to maintain an up-to-date view of the network topology at all the nodes

Destination Sequenced Distance Vector Routing protocol (DSDV)

- disadvantages
- the updates due to broken links lead to a heavy control overhead during high mobility
- small network with high mobility or a large network with low mobility can choke the available bandwidth
- the protocol suffers from the excessive control overhead that is proportional to the number of nodes in the network and therefore
- is not scalable in adhoc network having limited bandwidth and whose topologies are highly dynamic
- another disadvantage is that in order to obtain information about a particular destination node, a node has to wait for a table update message initiated by the same destination node,
- this delay could result in stale routing information at nodes

On Demand Routing protocols

- unlike table-driven routing protocol, on-demand routing protocols execute the path finding process and
- exchange routing information only when a path is required by a node to communicate with a destination
- Dynamic Source Routing (DSR) protocol
- on demand protocol designed to restrict the bandwidth consumed by control packets in adhoc network by
- eliminating the periodic table update messages required in the table-driven approach
- major difference between DSR and other on demand routing protocols is that
- it is beacon less and hence does not require periodic hello packet transmissions, which are used by a node to inform its neighbors of its presence

Dynamic Source Routing (DSR) protocol

- the basic approach of on demand routing protocol during the route construction phase is to establish a route by flooding RouteRequest packets in the network
- the destination node, on receiving a RouteRequest packet responds by sending a RouteReply packet back to the source, which
- carries the route traversed by the Route Request packet received
- consider a source node that does not have a route to the destination
- when it has data packets to be sent to that destination, it initiates a RouteRequest packet
- this RouteRequest is flooded throughout the network
- each node, upon receiving a RouteRequest packet, rebroadcast the packet to its neighbors if it has not forwarded already or
- if the node is not the destination node, provided the packet's time to live (TTL) counter has not exceeded

Dynamic Source Routing (DSR) protocol

- each RouteRequest carries a sequence number generated by the source node and the path it has traversed
- a node upon receiving a RouteRequest packet, checks the sequence number on the packet before forwarding it
- the packet is forwarded only if it is not a duplicate RouteRequest
- the sequence number on the packet is used to prevent loop formations and to avoid multiple transmissions of the same RouteRequest by an intermediate node that receives it through multiple paths
- thus all node except the destination forward a Route Request during the route construction phase
- a destination node after receiving the first RouteRequest packet, replies to the source node through the reverse path the RouteRequest packet had traversed.

Dynamic Source Routing (DSR) protocol

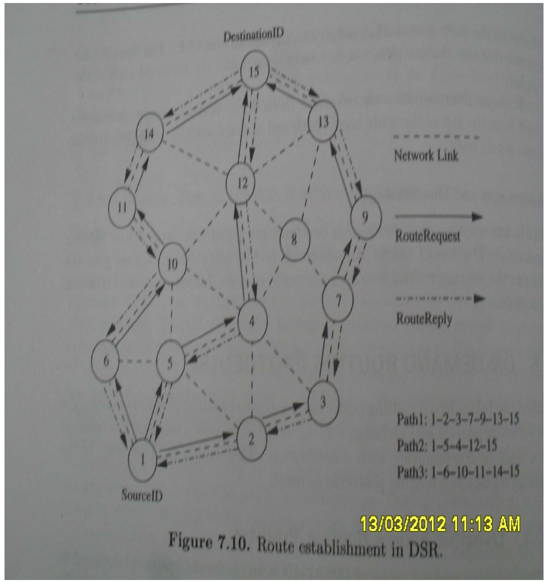


Figure 7.10. Route establishment in DSR.

src: textbook

Dynamic Source Routing (DSR) protocol

- node 1 initiates RouteRequest to obtain a path for destination node 15
- the protocol uses a route cache that stores all information extracted from the source route contained in a data packet
- nodes can learn about the neighboring routes traversed by data packets if operated in the promiscuous mode (the mode in which node can receive the packets that are neither broadcast nor addressed to itself)
- if an intermediate node receiving a RouteRequest has a route to the destination node in its route cache then
- it replies to the source node by sending a RouteReply with the entire route information from the source code to destination node

Dynamic Source Routing (DSR) protocol

- optimization techniques used with DSR to improve performance of the protocol
- DSR uses route cache at intermediate nodes, route cache is populated with routes
- cache also used to reply to the source by intermediate node
- using promiscuous mode, possible to update route cache and active routes maintained
- during network partitions, the affected nodes initiate RouteRequest packets
- exponential backoff algorithm is used to avoid frequent RouteRequest flooding in the network when the destination is in another disjoint set
- DSR allows piggy backing of a data packet on RouteRequest

Dynamic Source Routing (DSR) protocol

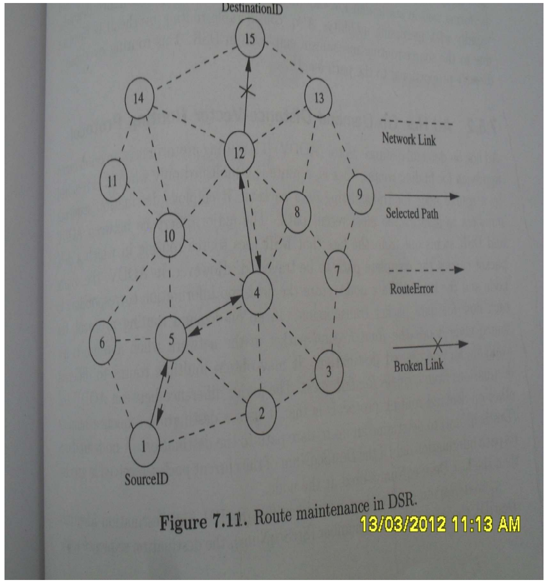


Figure 7.11. Route maintenance in DSR.

src: textbook (Route maintenance in DSR)

Dynamic Source Routing (DSR) protocol

- if optimization is not allowed in the DSR then route construction phase is simple
- all intermediate nodes flood the RouteRequest packet if it is not redundant
- RouteRequest packet received from node 1 by its neighbor nodes 2,5,6, they forward it
- node 4 receives the RouteRequest from both nodes 2 and 5 and discards the other redundant or duplicate RouteRequest packets
- RouteRequest is propagated till it reaches the destination which initiates RouteReply
- as part of optimization if the intermediate nodes are also allowed to originate RouteReply packets then a source node may receive multiple replies from intermediate nodes

Dynamic Source Routing (DSR) protocol

- intermediate node 10 has a route to the destination via node 14; it also sends the RouteReply to the source node
- source node selects the latest and best route and uses that for sending data packets
- when intermediate node in the path moves away, link breaks, say,
- link between nodes 12 and 15 fails,
- RouteError message is generated from the node adjacent to the broken link to inform the source node
- source node reinitiates the route establishment procedure
- the cached entries at the intermediate nodes and the source node are removed when a RouteError packet is received
- if a link breaks dues to the movement of edge nodes (node 1 and node 15) the source node again initiates the route discovery process

Dynamic Source Routing (DSR) protocol

- advantages
- protocol uses a reactive approach which eliminates the need to periodically flood the network with table update message which are required in table driven approach
- on-demand (reactive) approach route is established only when it is required and hence
- the need to find routes to all other nodes in the network as required by table driven approach is eliminated
- intermediate nodes utilizes the route cache information efficiently to reduce the control overhead

Dynamic Source Routing (DSR) protocol

- disadvantages
- the route maintenance mechanism does not locally repair a broken link
- stale route cache information could result in inconsistencies during the reconstruction phase
- connection setup delay is higher than in table driven protocols
- DSR protocols perform well in static and low mobility environments, performance degrades with increasing mobility
- routing overhead is involved due to source routing and proportional to path length

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- on demand, employs destination sequence numbers to identify the most recent path
- the major difference between AODV and DSR
- DSR uses source routing in which a data packet carries the complete path to be traversed
- AODV, source node and intermediate nodes store the next hop information corresponding to each flow for data packet transmission
- on-demand routing protocol, the source code floods the RouteRequest packet in the network when a route is not available for the desired destination
- it may obtain multiple routes to different destinations from a single RouteRequest

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- AODV uses a destination sequence number DestSeqNum to determine an uptodate path to the destination
- a node updates its path information only if the DeptSeqNum of the current packet received is greater than the last DeptSeqNum stored at the node
- A RouteRequest carries source identifier (SrcID), the destination identifier (DestID), the source sequence number (SrcSeqNum), destination sequence number (DestSeqNum), the broadcast identifier (BcastID), time to live (TTL) field
- DestSeqNum indicates the freshness of the route that is accepted by the source
- when an intermediate node receives a RouteRequest, it either forwards it or prepares RouteReply if it has a valid route to the destination

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- validity of a route at the intermediate node is determined by comparing the sequence number at the intermediate node with the destination sequence number in the RouteRequest packet
- if RouteRequest is received multiple times, indicated by BcastID-SrcID pair, the duplicate copies are discarded
- all intermediate nodes having valid routes to the destination, or the destination node itself, are allowed to send RouteReply packets to the source
- every intermediate node, while forwarding RouteRequest, enters the previous node address and its BcastID

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- a timer is used to delete this entry in case RouteReply is not received before the timer expires
- this helps in storing an active path at the intermediate node as AODV does not employ source routing of data packets
- when a node receives a RouteReply packet information about the previous node from which the packet was received is also stored in order to forward the data packet to this next node as the next hop toward the destination

Adhoc On Demand Distance Vector Routing Protocol (AODV)

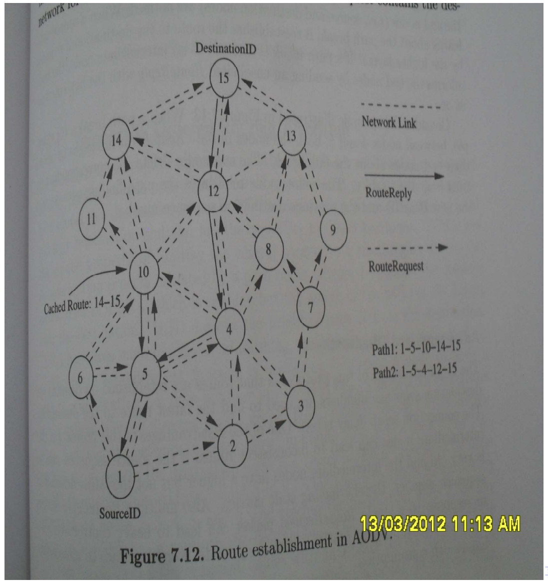


Figure 7.12. Route establishment in AODV.

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- node 1 initiates path finding process by originating RouteRequest to be flooded in the network for destination node 15
- assume that the RouteRequest contains destination sequence number as 3 and the source sequence number 1,
- when nodes 2,5,6 receives the RouteRequest packet, they check their routes to the destination,
- in case route to destination is not available, they further forward it to their neighbors i.e. nodes 3,4,10 neighbors of 2,5,6

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- assume that intermediate nodes 3, 10 have routes to the destination node, say 10-14-15 and 3-7-9-13-15 respectively,
- if the destination sequence number at intermediate node 10 is 4 and is 1 at intermediate node 3 then
- only node 10 is allowed to reply along the cached route to the source
- this is because node 3 has an older route to node 15 compared to the route available at the source node
- (the destination sequence number at node 3 is 1, but the destination sequence number is 3 at the source node) while
- node 10 has a more recent node (the destination sequence number is 4) to the destination

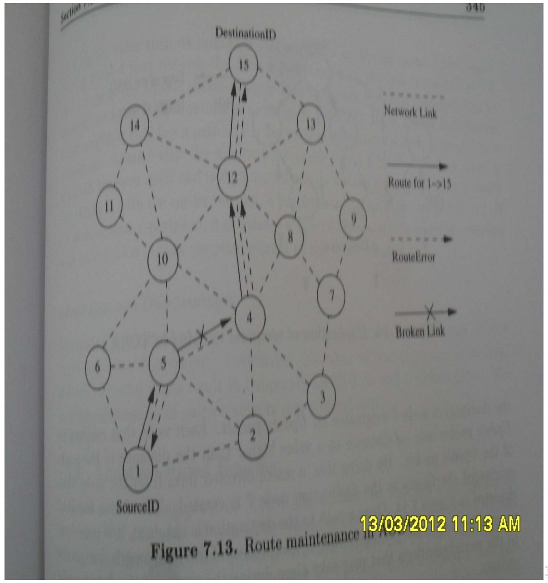
Adhoc On Demand Distance Vector Routing Protocol (AODV)

- if RouteRequest reaches the destination node 15 through path 4-12-15 or any other alternative route, the destination also sends RouteReply to the source
- in this case multiple RouteReply packets reach the source, all the intermediate nodes receiving a RouteReply update their route tables with the latest destination sequence number
- they also update the routing information if it leads to a shorter path between source and destination
- AODV does not repair a broken path locally, when a link breaks, determined by observing periodical beacons or through link-level acknowledgments, the end nodes are notified.

Adhoc On Demand Distance Vector Routing Protocol (AODV)

- when a source node learns about the path break, it reestablishes the route to the destination if required by the higher layers
- if a path break is detected at an intermediate node, the node informs the end nodes by sending an unsolicited RouteReply with the hop count set as ∞
- when a path breaks, say, between nodes 4 and 5 both nodes initiate RouteError message to inform their end nodes about the link break
- the end nodes delete the corresponding entries from their tables
- source node reinitiates the path finding process with the new BcastID and previous destination sequence number

Adhoc On Demand Distance Vector Routing Protocol (AODV)



Adhoc On Demand Distance Vector Routing Protocol (AODV)

- advantages
- routes are established on demand and destination sequence numbers are used to find the latest route to the destination
- connection setup delay is less
- disadvantages
- disadvantage is that intermediate nodes can lead to inconsistent routes if
- the source sequence number is very old and the intermediate nodes have a higher but not the latest destination sequence number, thereby having stale entries
- heavy control overhead due to multiple RouteReply in response to single RouteRequest
- periodic beaconing leads to unnecessary bandwidth consumption