Queen Mary
University of London

MANET Routing

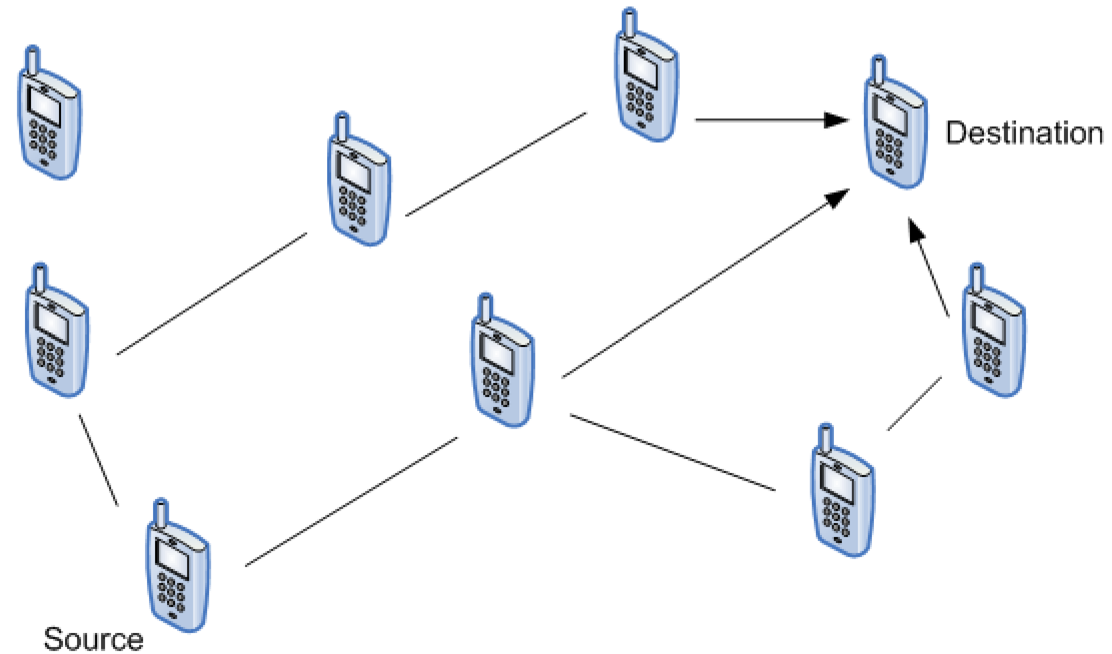
EBU5211: Ad Hoc Networks

Dr. Yan SUN (Cindy)

Outline

- Issues in designing a routing protocol for MANET
- Overview of current Ad Hoc routing protocols

MANET routing



- Routers: Any node in network can be considered as a router (infrastructure independent).
- Mobility: All nodes in network are moving randomly.
- Topology: Highly dynamic network topology .

Using wired IP routing for MANET

- Routing is one of main challenges in MANETs
  => high mobility of nodes leads to break links randomly
  => highly dynamic topology.
- So using wired IP routing ideas in MANETs causes:
  - High usage of (battery) power to search for “good” path of reliable links
  - Slow convergence
  - Little throughput
  - Signs of instability due to count-to-infinity problem.
- **Conclusion**:

  wired IP routing is not suitable for MANETs.

Design Challenges

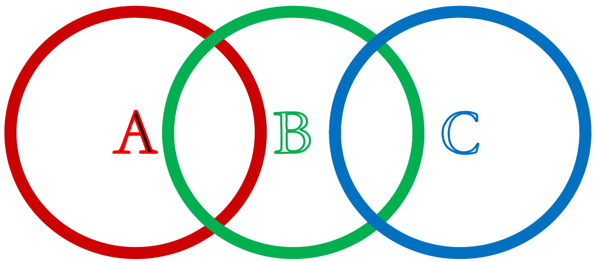
- Mobility :
  - Routing protocols for MANETS need efficient and effective mobility management
- Bandwidth constraint
  - Radio spectrum is limited and channel is shared by all users.
- Resource constraint:
  - Routing protocol must consider restrictions in battery, computing power and buffer storage for portable MANET terminals.

Design Challenges

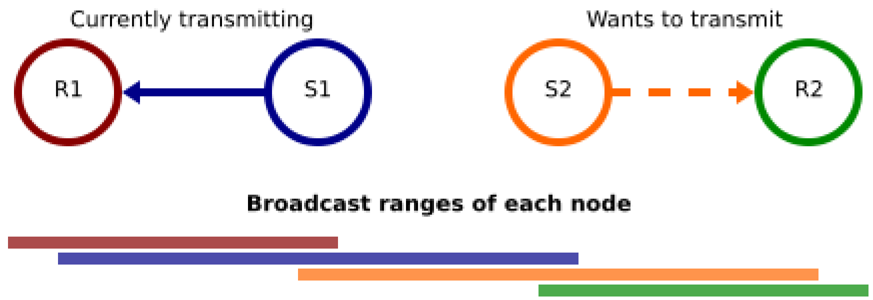
- "Hidden terminal" and "Exposed terminal" problem (interaction with MAC layer protocols)

Hidden terminal

A & C hidden
from each other
But
A ⇔ B or
C ⇔ B
interfere with other



Exposed terminal



Overview of MANET Routing Protocol

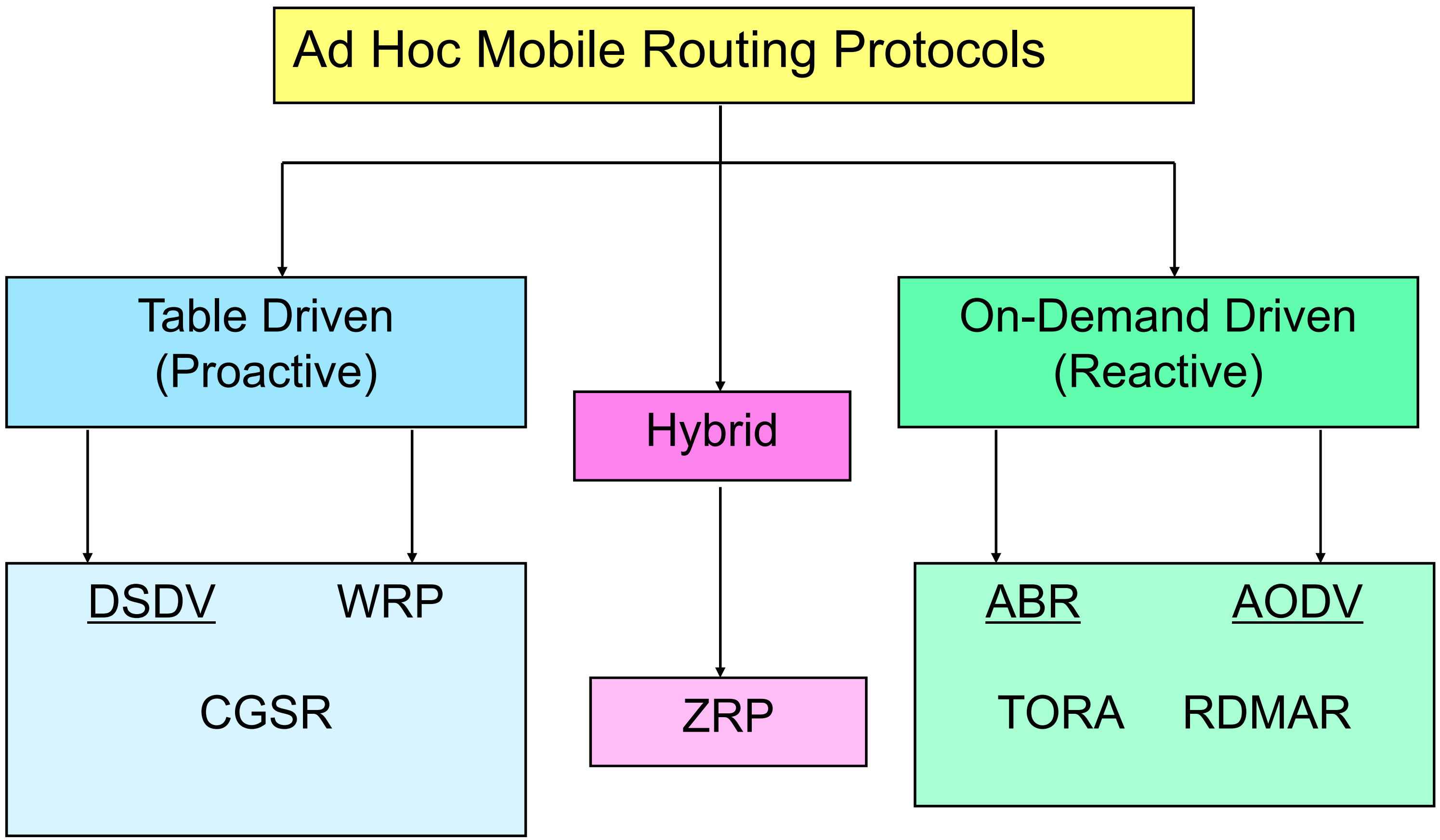


Table Driven

- Attempt to maintain consistent, up-to-date routing information from each node to every other node in the network.
- Involves constant propagation of routing information.
- A route to every other node in the ad hoc network is always available regardless of whether or not it is needed.
- Substantial signalling traffic and power consumption.
- Periodic Route updates.

On-demand Driven

Queen Mary University of London

- Creates routes only when desired by the source node
- When a node requires a route to a destination, it initiates a route discovery process within a network.
- NO periodic route updates
- Power efficient
- Bandwidth efficient

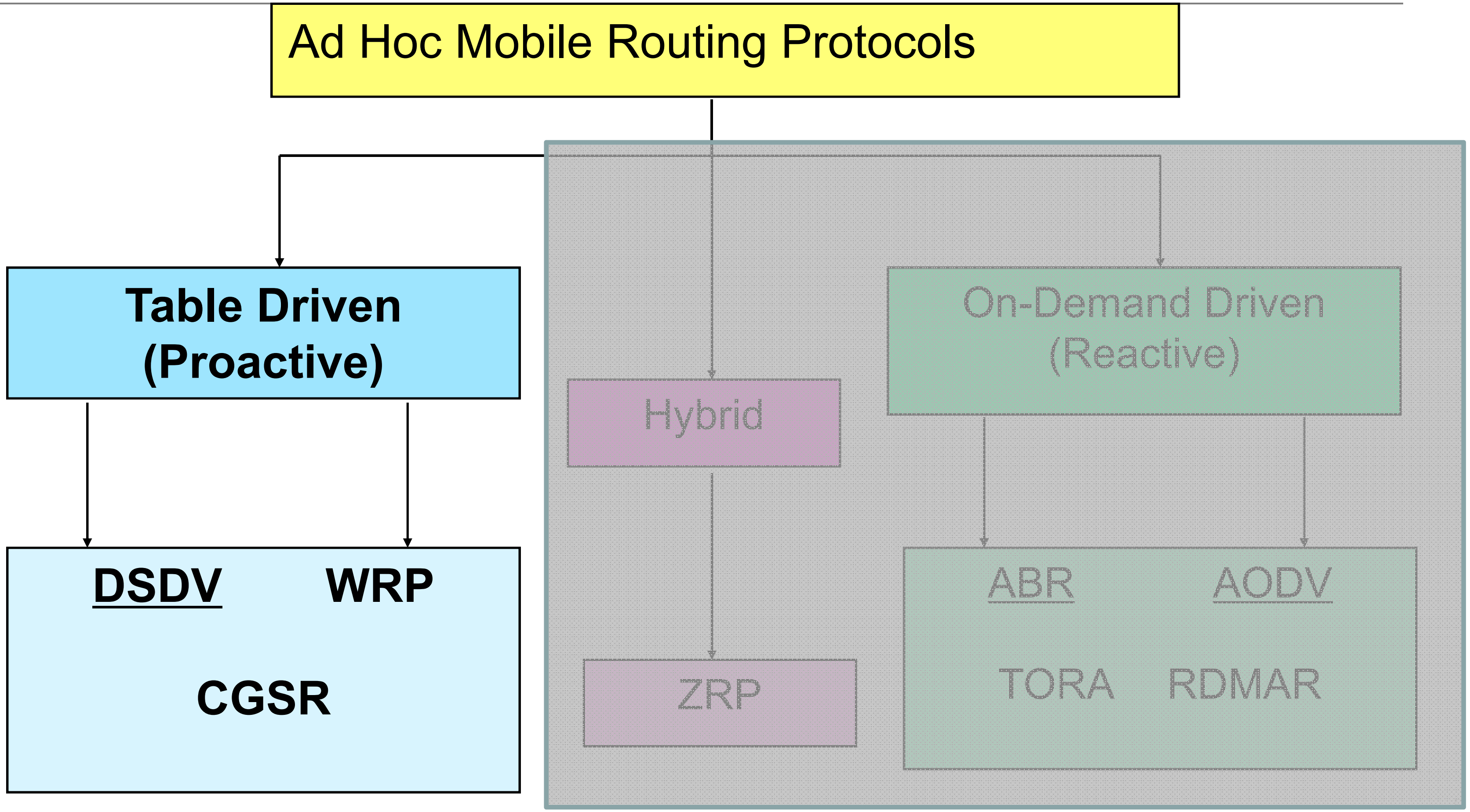
An Example of Ad Hoc Table Driven Routing Protocol

- **Table-Driven Routing Protocol Overview**
- **DSDV - Destination Sequenced Distance Vector Routing**

References: C. Siva Ram Murthy, B.S. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall, 2004.

C.K. Toh, *Ad Hoc Mobile Networks: Protocols and Systems,* Prentice Hall, 2002.

Table-Driven Routing Protocol



Destination Sequenced Distance Vector

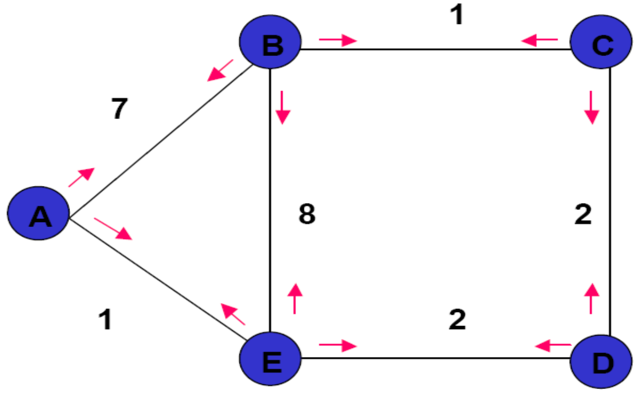
- It is based on classical Distributed Bellman-Ford (DBF) algorithm.
- In DBF, each node maintains the first node (hop) on the shortest path to every other node in the network.
- Each node maintains routing table for all possible DEST and the number of routing hops to reach that DEST.

Destination Sequenced Distance Vector

- Therefore, routing information is always available regardless whether the SRC needs the route or not. (Table Driven)
- A sequence numbering system (labelled the routes) is used to know the stale routes from the new routes.
- Route calculation: $D(i, j) = \min [d(i, k) + D(k, j)]$

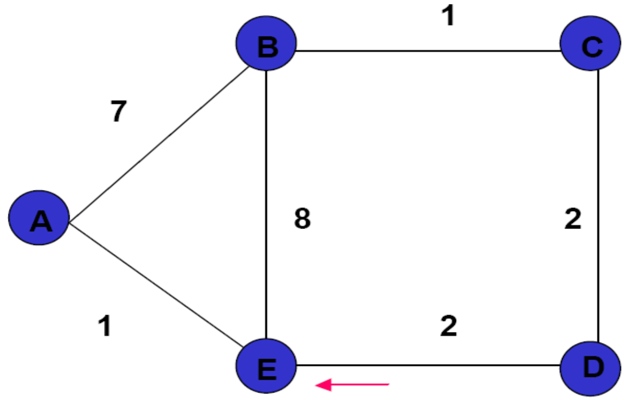
  where D(i, j) is the metric on the shortest path from node i to node j, d(i,k) is the cost of traversing directly from node i to node k, and k is one of the neighbours of node i.

Distributed Bellman-Ford algorithm



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

Initial Distances

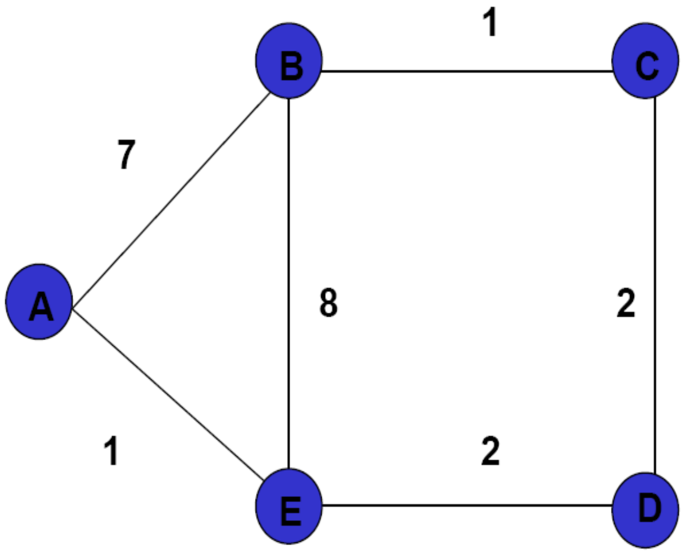


Info at Node	Distance to Node A	B	C	D	E
A	0	7	~	~	1
B	7	0	1	~	8
C	~	1	0	2	~
D	~	~	2	0	2
E	1	8	4	2	0

E receives D's Routes

Link cost or count (Metric): It is the hop count, bandwidth or even the really cost, if there are two different operators connected on one link.

DBF



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

Final Distances

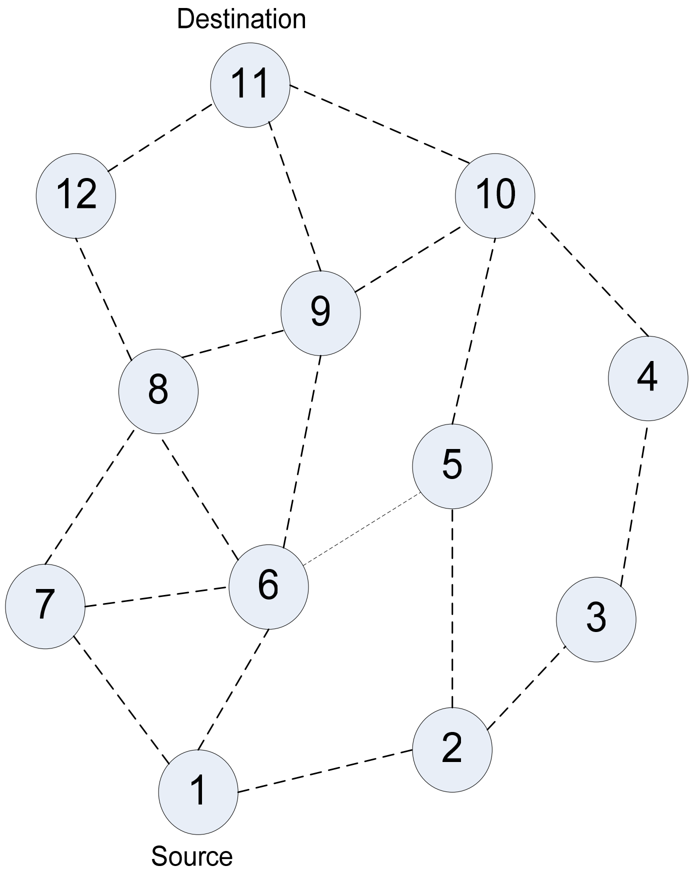
Routing Tables

- Routing tables are exchanged between neighbors at regular intervals to keep an up-to-date view of the network topology.
- Therefore, this can cause a lot of control traffic, rending an inefficient utilization of network resources. Solution!
- Table Updates Types:
  - Full Dump: It carries all available routing information. It happens when the local topology changes significantly (a lot of nodes are moving away)
  - Incremental Updates: It carries only the routing information that has change since the last full dump. It happens when a node does not observe significant changes in the local topology.

Basic Principle

- Mobile nodes exchange route information.
- Maintain routes to all possible DEST.
- Mobility – reflected as link changes
- Use of sequence number to ensure "freshness" of route information.
- Uses "hops" as a routing metric.
- Node keeps track of its own time (Time Triggered) and the sequence of events that happens (Event Triggered) for updating..
- Each node needs to assign sequence numbers to Distance Vector updates reflecting information about its neighbors
- Route with the most recent sequence number is used

An example

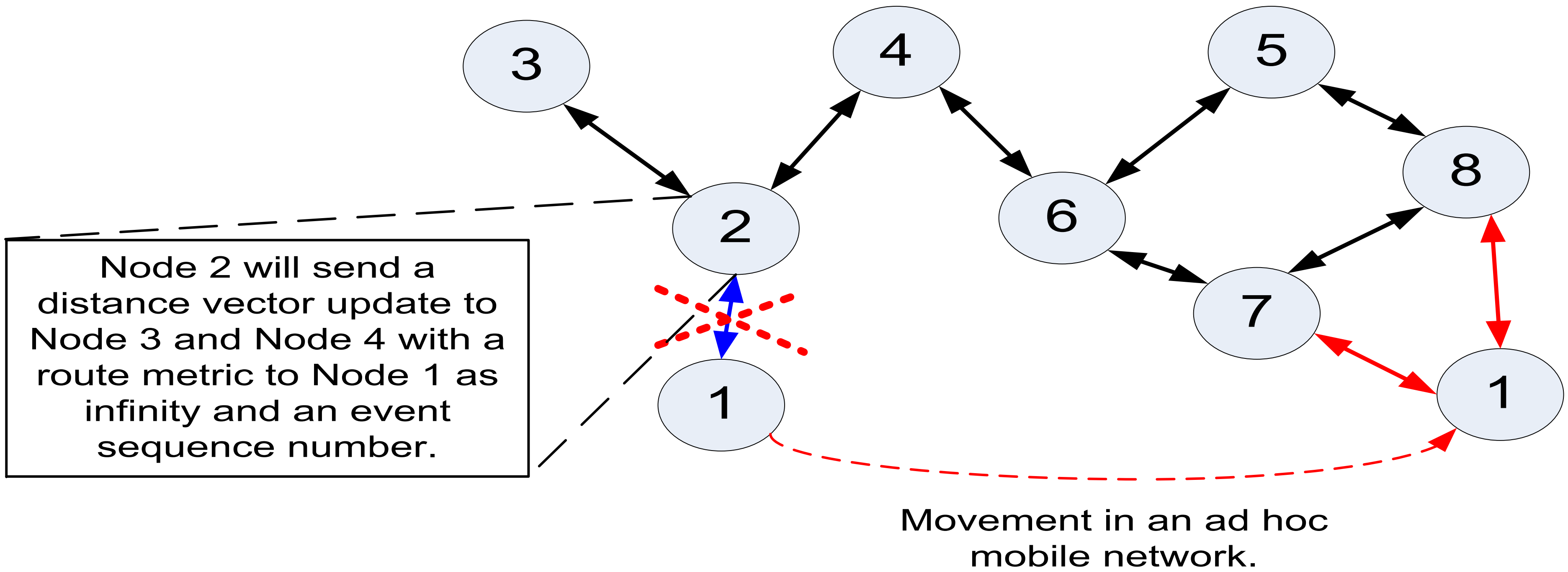


Dest	NextNode	Dist	SeqNo
2	2	1	22
3	2	2	26
4	2	3	30
5	6	2	34
6	6	1	38
7	7	1	144
8	7	2	148
9	6	2	160
10	6	3	168
11	6	3	180
12	7	3	188

Routing table for Node 1

- The shortest route to node 11 is available through node 6 with 3 hops away.

Case of Mobility



- The new neighbours of node 1 inform their neighbours about the shortest way to node 1. Then, this information propagates through the network.
- Eventually, every node receives a new route to node 1 with higher sequence number

Advantages and Disadvantages

- ✓ Availability of routes to all destination at all times implies much less delay is involved in the routes setup process.
- ✓ The incremental updates with sequence number makes the existing wired network protocols adaptable to ad hoc wireless networks

- ✘ Updates due to broken link lead to a heavy control overhead during high mobility
- ✘ 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 results in stale routing information at nodes.

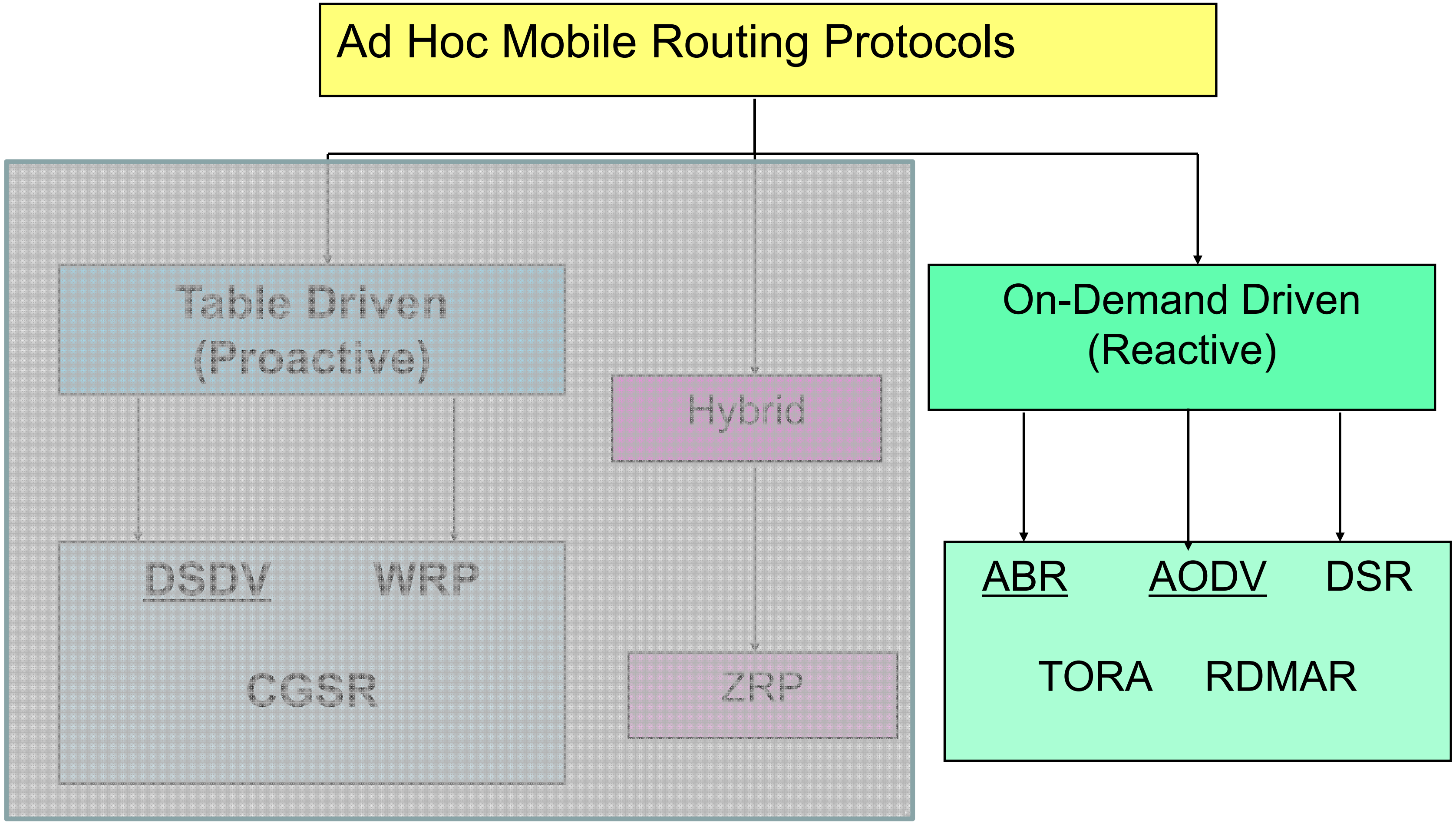
Examples of Ad Hoc On-Demand Driven Routing Protocol

- **On-Demand Driven Routing Protocol Overview**
- **ABR - Associativity Based Routing**
- **AODV-Ad Hoc On-Demand Distance Vector**
- **Table Driven vs on-Demand Driven**

References: C. Siva Ram Murthy, B.S. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall, 2004.

C.K. Toh, *Ad Hoc Mobile Networks: Protocols and Systems,* Prentice Hall, 2002.

On-Demand Driven Routing Protocol



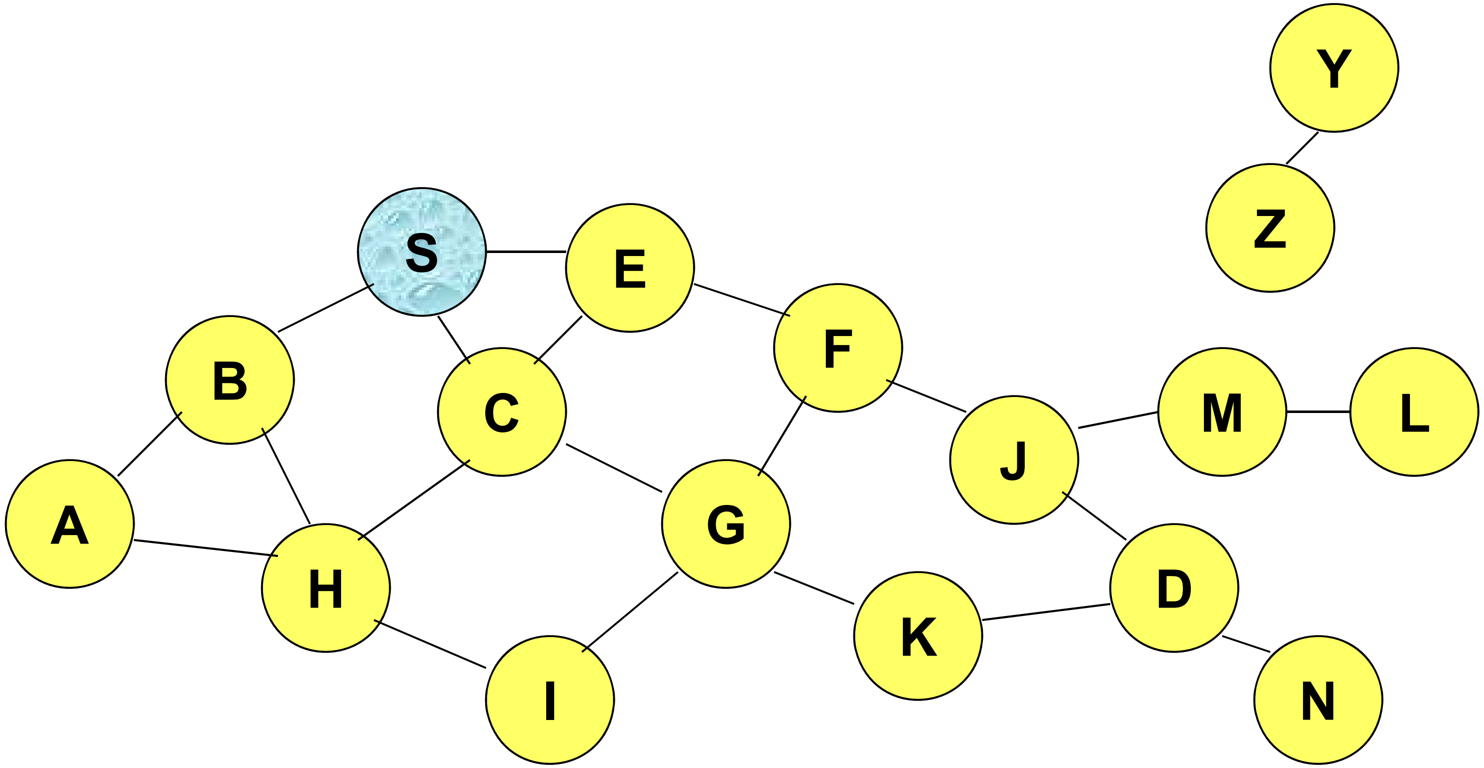
Ad hoc On-Demand Distance Vector Routing (AODV)

- Capable of unicast and multicast routing
- Reactive routing protocol
- Using sequence numbers on route updates to avoid the counting-to-infinity problem of other distance-vector protocols
- DestSeqNum: determine an up-to-date path to the destination
- Next-hop information stored in source node and intermediate nodes

AODV

- Route Requests (RREQ) are flooded in the ad hoc networkk
- When a node re-broadcasts a Route Request, it sets up a reverse path pointing towards the source
  - AODV assumes symmetric (bi-directional) links
- When the intended destination receives a Route Request, it replies by sending a Route Reply (RREP)
- Route Reply travels along the reverse path set-up when Route Request is forwarded

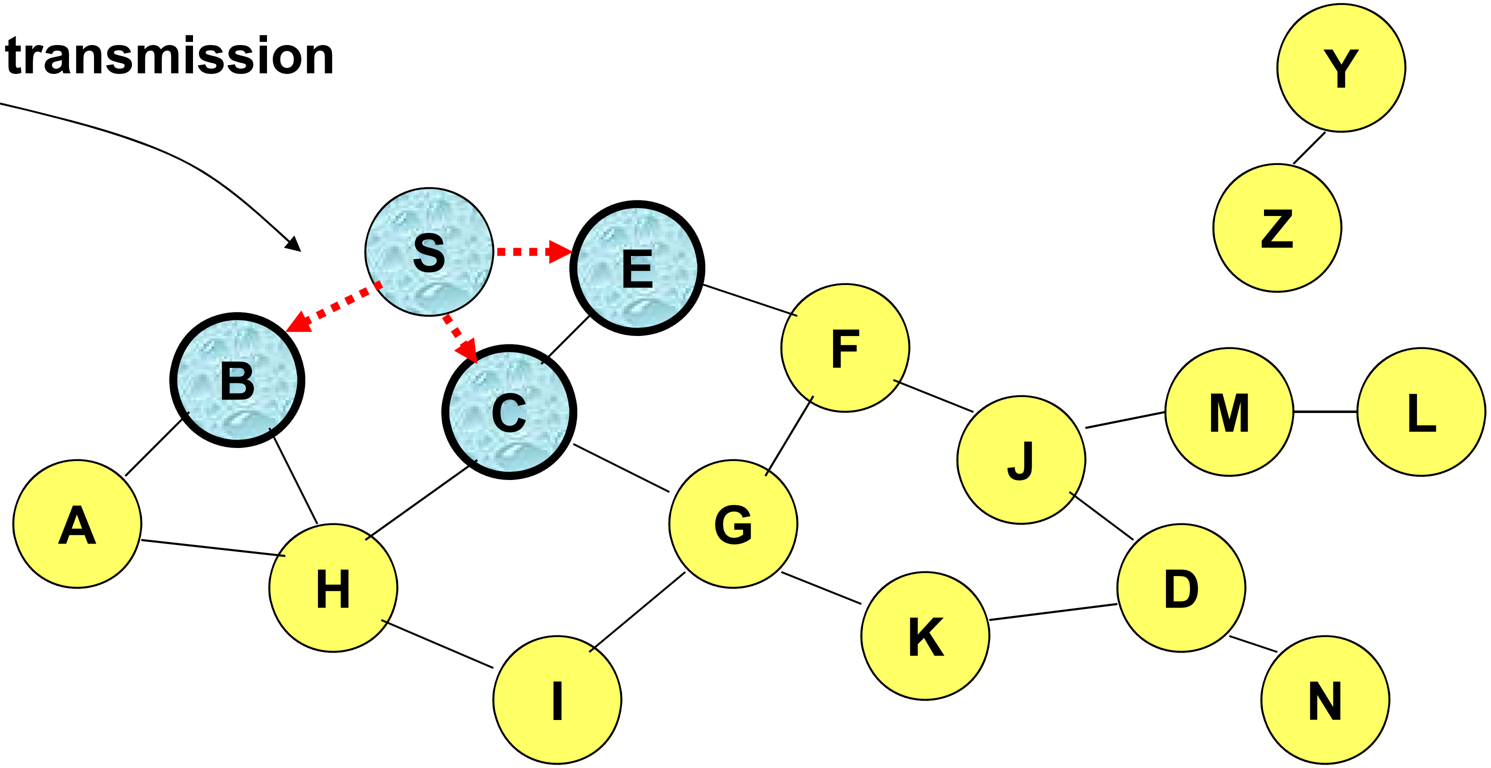
Route Requests in AODV



 Represents a node that has received RREQ for D from S

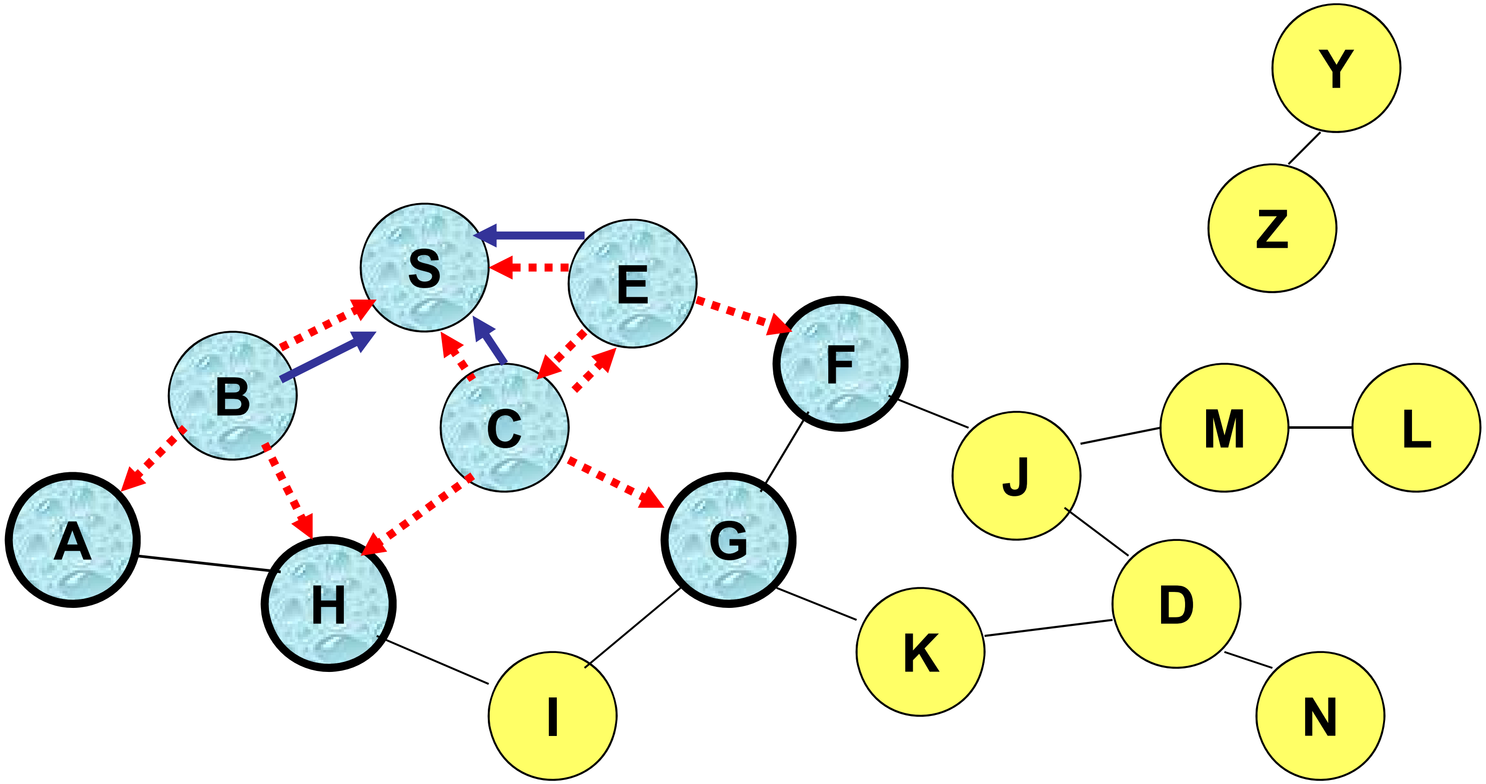
Route Requests in AODV

Broadcast transmission



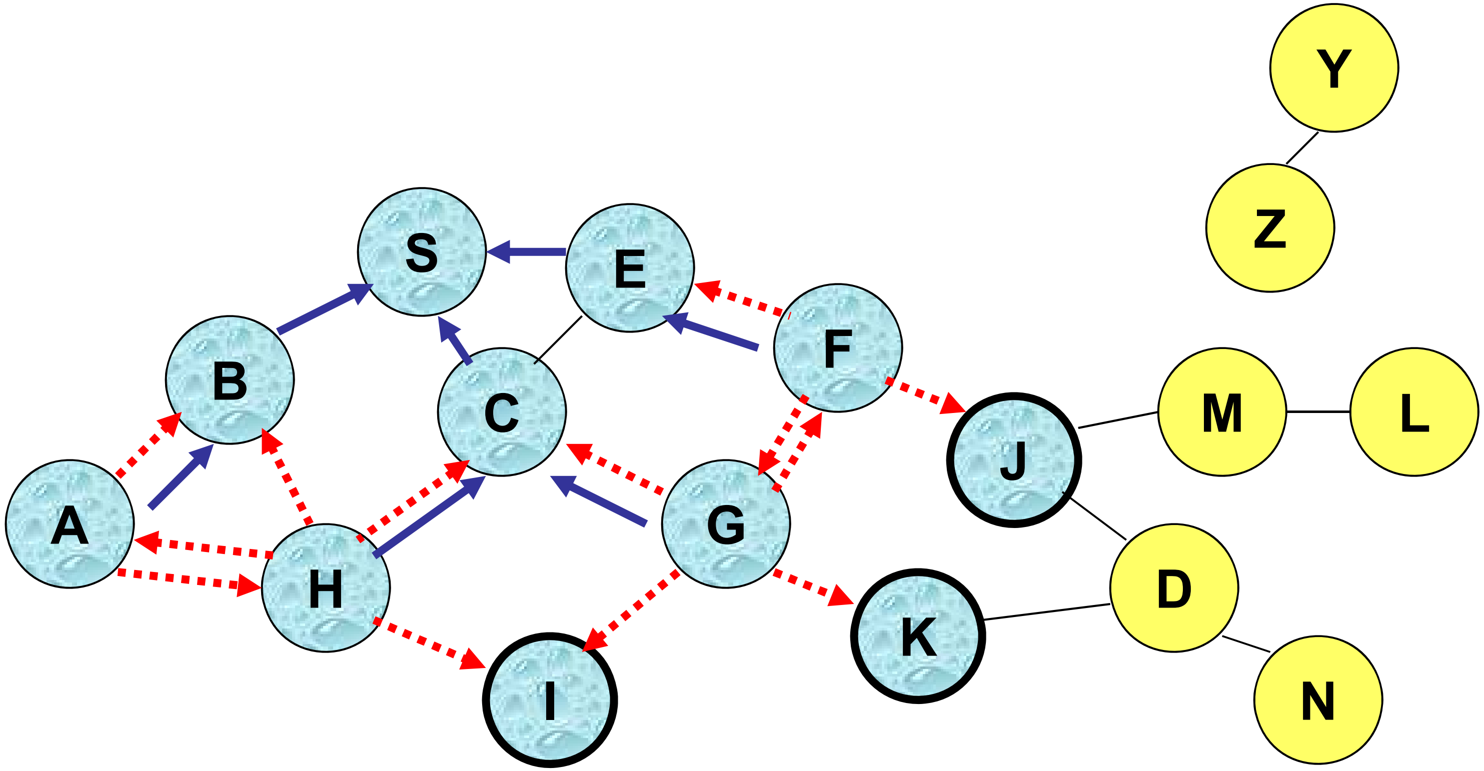
Represents transmission of RREQ

Route Requests in AODV



Represents links on Reverse Path

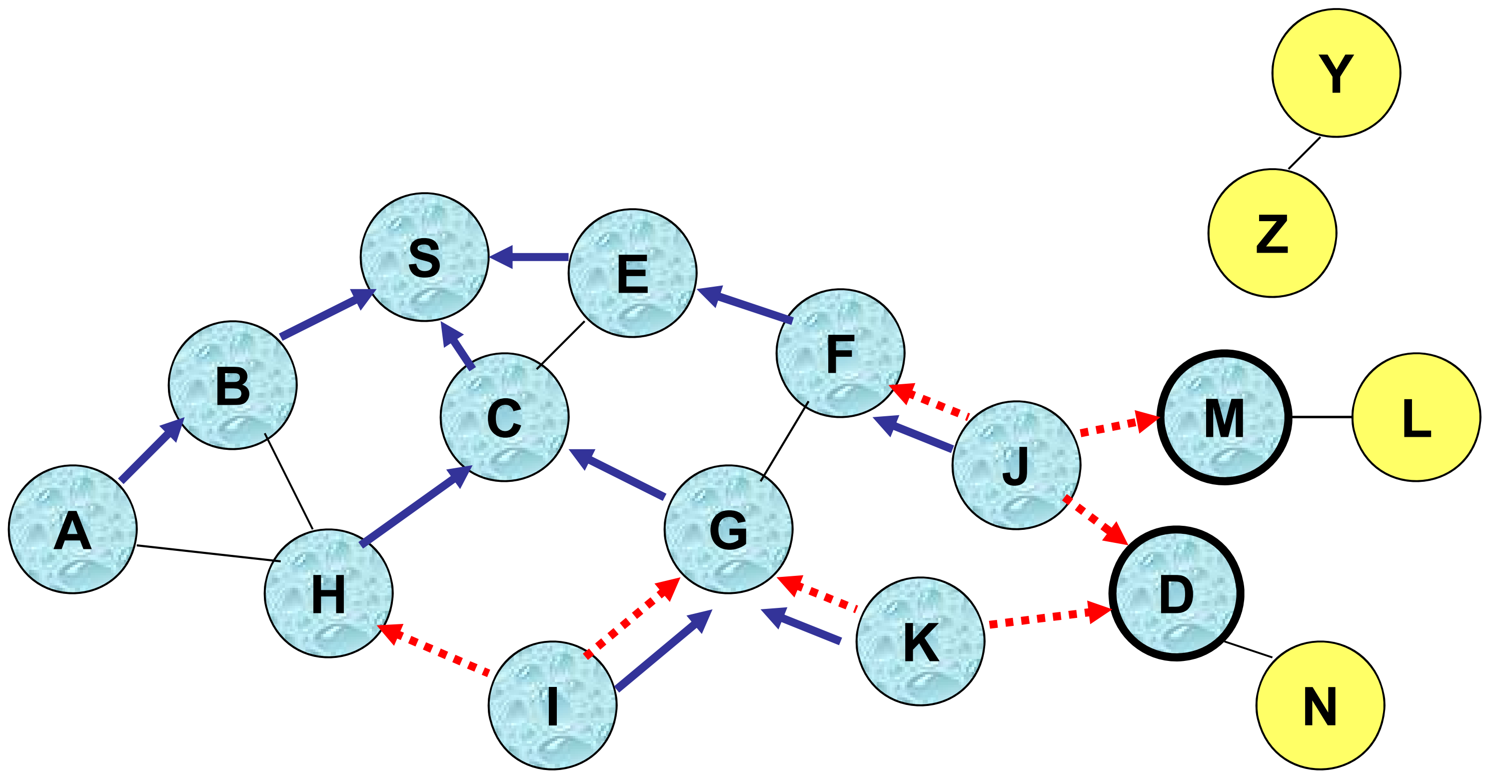
Reverse Path Setup in AODV



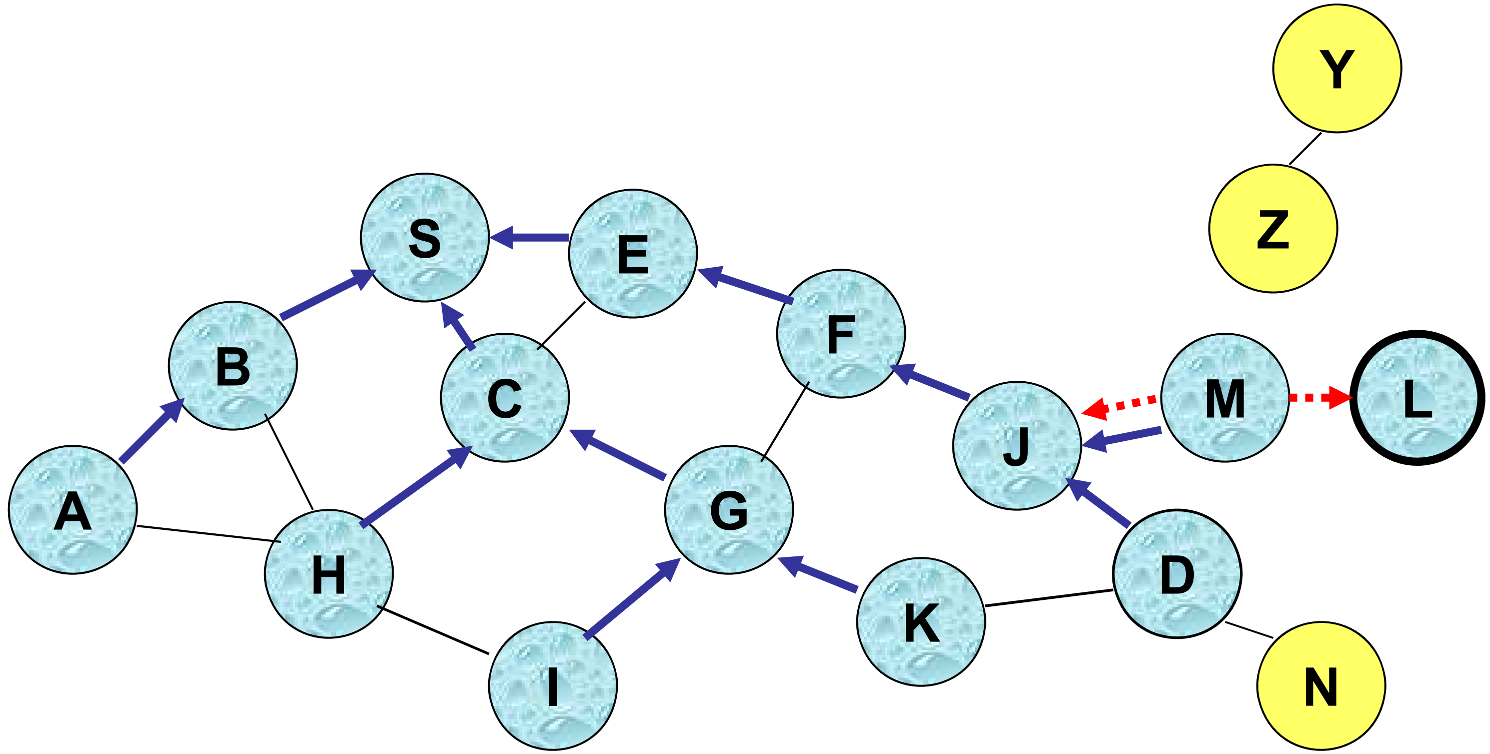
- Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once

Reverse Path Setup in AODV

Queen Mary University of London

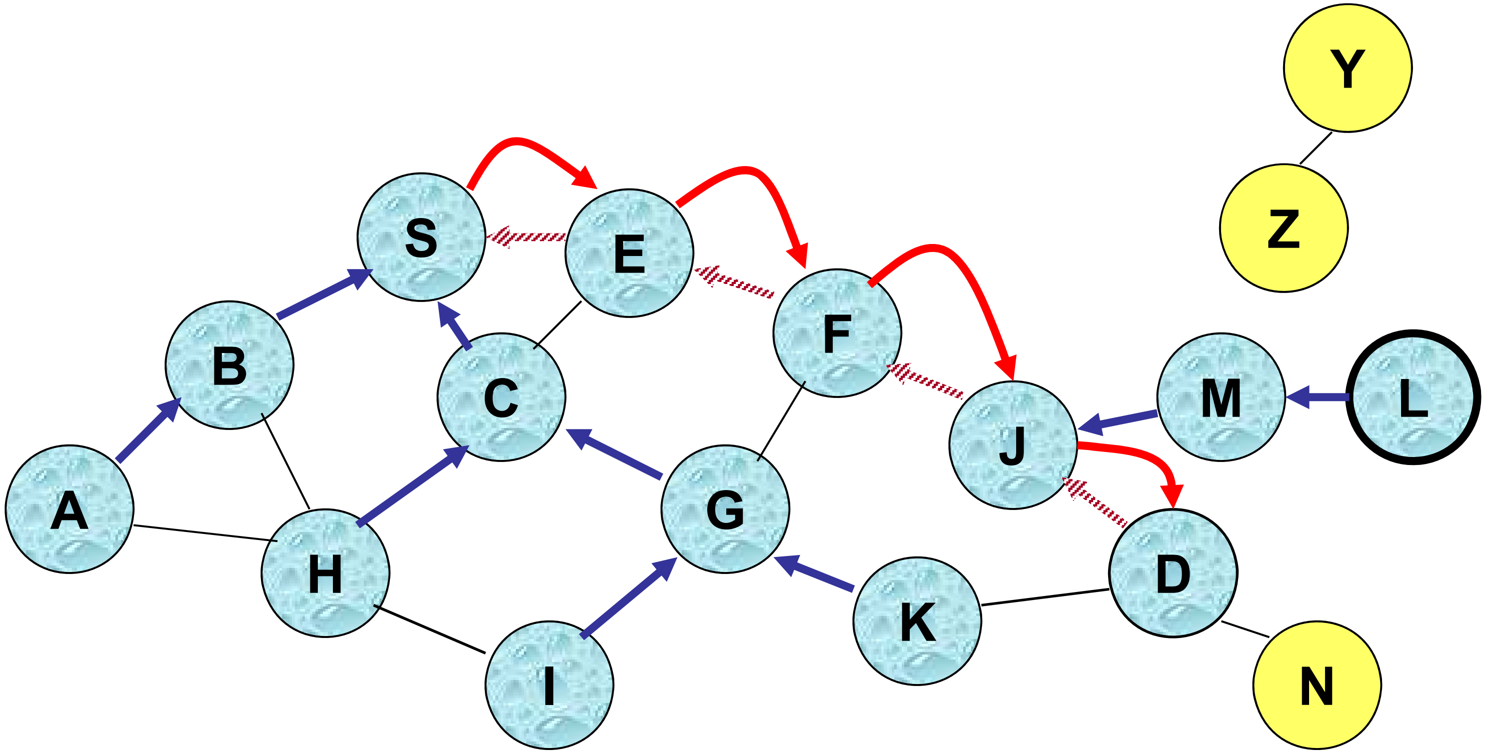


Reverse Path Setup in AODV

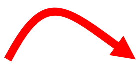


- Node D **does not forward** RREQ, because node D is the **intended target** of the RREQ

Forward Path Setup in AODV



Forward links are setup when RREP travels along the reverse path



Represents a link on the forward path

Route Request and Route Reply

- Route Request (RREQ) includes the last known sequence number for the destination
- An intermediate node may also send a Route Reply (RREP) provided that it knows a more recent path than the one previously known to sender
- Intermediate nodes that forward the RREP, also record the next hop to destination
- A routing table entry maintaining a reverse path is purged after a timeout interval
- A routing table entry maintaining a forward path is purged if *not used* for a *active_route_timeout* interval

Link Failure

- A neighbor of node X is considered active for a routing table entry if the neighbor sent a packet within *active_route_timeout* interval which was forwarded using that entry
- Neighboring nodes periodically exchange hello message
- When the next hop link in a routing table entry breaks, all active neighbors are informed
- Link failures are propagated by means of Route Error (RERR) messages, which also update destination sequence numbers

Route Error

- When node X is unable to forward packet P (from node S to node D) on link (X,Y), it generates a RERR message
- Node X increments the destination sequence number for D cached at node X
- The incremented sequence number *N* is included in the RERR
- When node S receives the RERR, it initiates a new route discovery for D using destination sequence number at least as large as *N*
- When node D receives the route request with destination sequence number N, node D will set its sequence number to N, unless it is already larger than N

AODV: Summary

- Route Request carries: SrcID, DestID, SrcSeqNum, DestSeqNum, BcastID, TTL
- Path Discovery
  - Reverse-Path Setup
  - Forward-Path Setup
- Path Maintenance
- No local path reconstruction

AODV: Summary

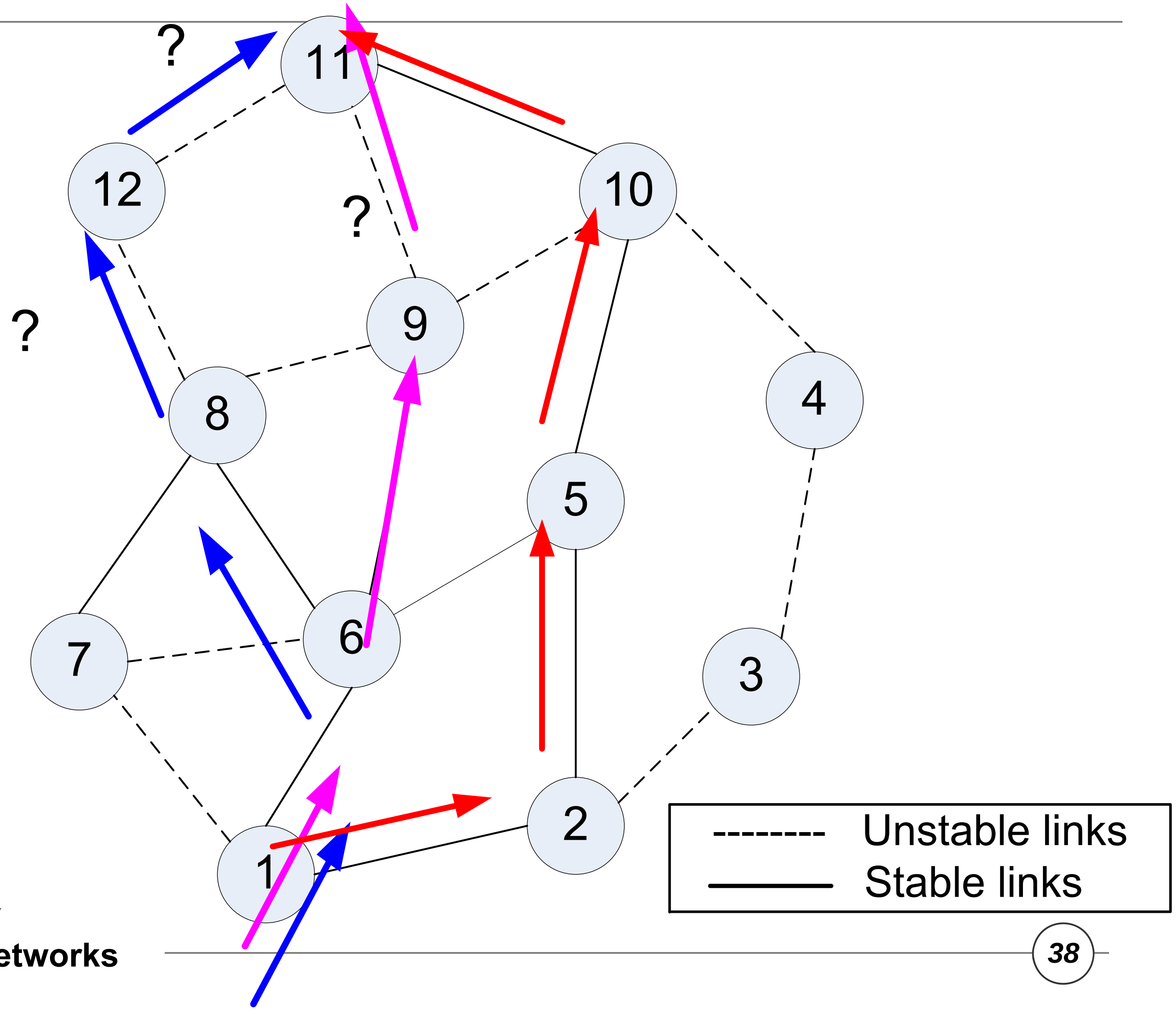
- Routes need not be included in packet headers
- Nodes maintain routing tables containing entries only for routes that are in active use
- At most one next-hop per destination maintained at each node
- Sequence numbers are used to avoid old/broken routes
- Sequence numbers prevent formation of routing loops
- Unused routes expire even if topology does not change

Associativity Based Routing (ABR)

Queen Mary
University of London

- Selects routes based on the stability of the wireless link.
- It is a beacon-based, on-demand routing protocol.
- Temporal stability is determined by counting the periodic beacons that a node received from its neighbours. Each link is classified as stable or unstable based on the beacon count corresponding to the neighbour node concerned.
- Stability in ABR refers to associativity ticks, signal strength and power life.

Example of Route Selection in ABR



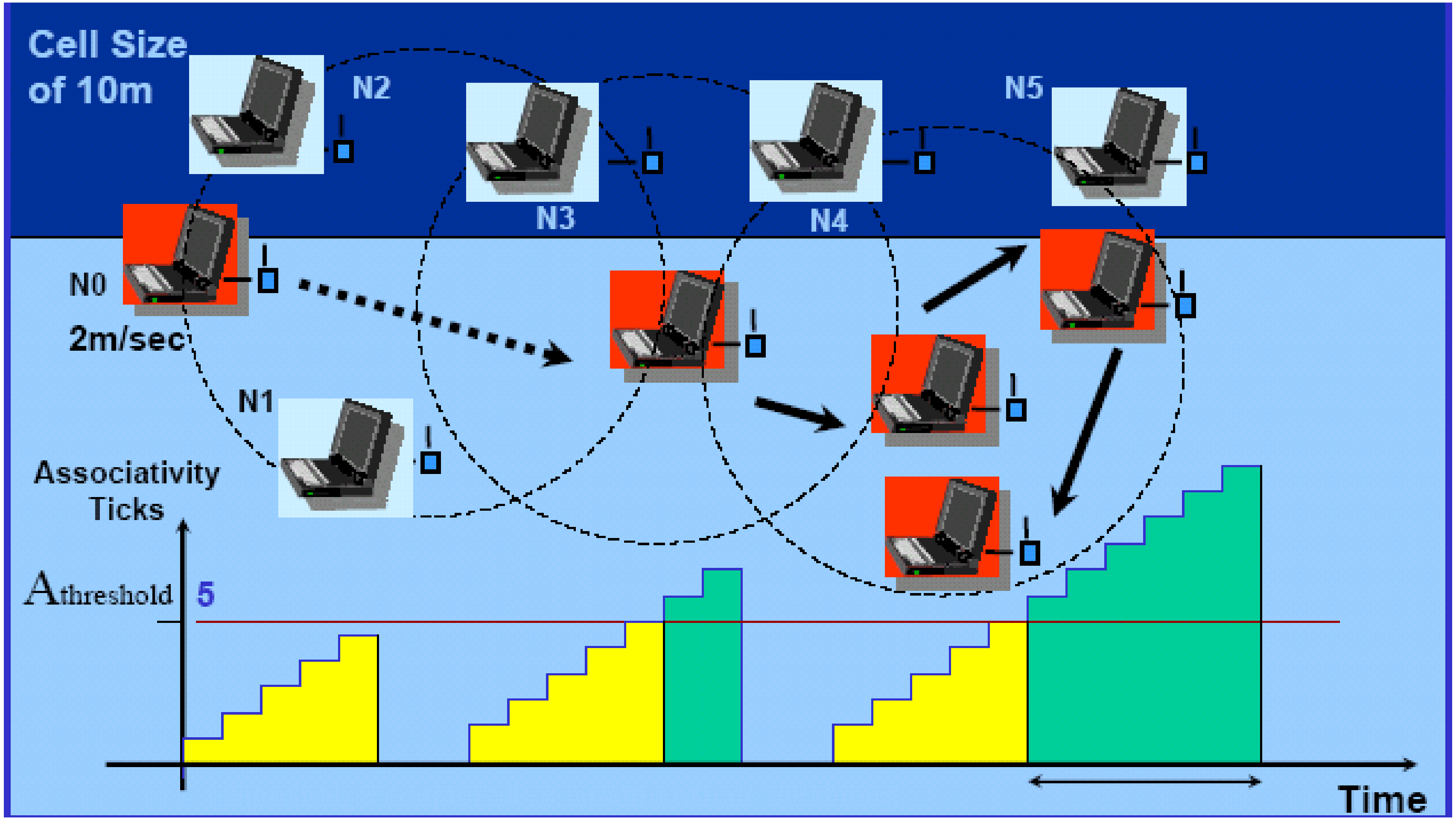
<u>Source:</u> node 1

<u>Destination:</u> node 11

Associativity Ticks

- Each node transmits beacons to identify itself and constantly updates its associativity ticks based on it neighbours.
- The threshold where associativity transitions take place is defined. If ticks>$A_{threshold}$, stable link, stable route (may not be the shortest parth). Associativity tick depends on transition range, speed and beaconing internal.
- Low associativity ticks = high state of mobility for node.

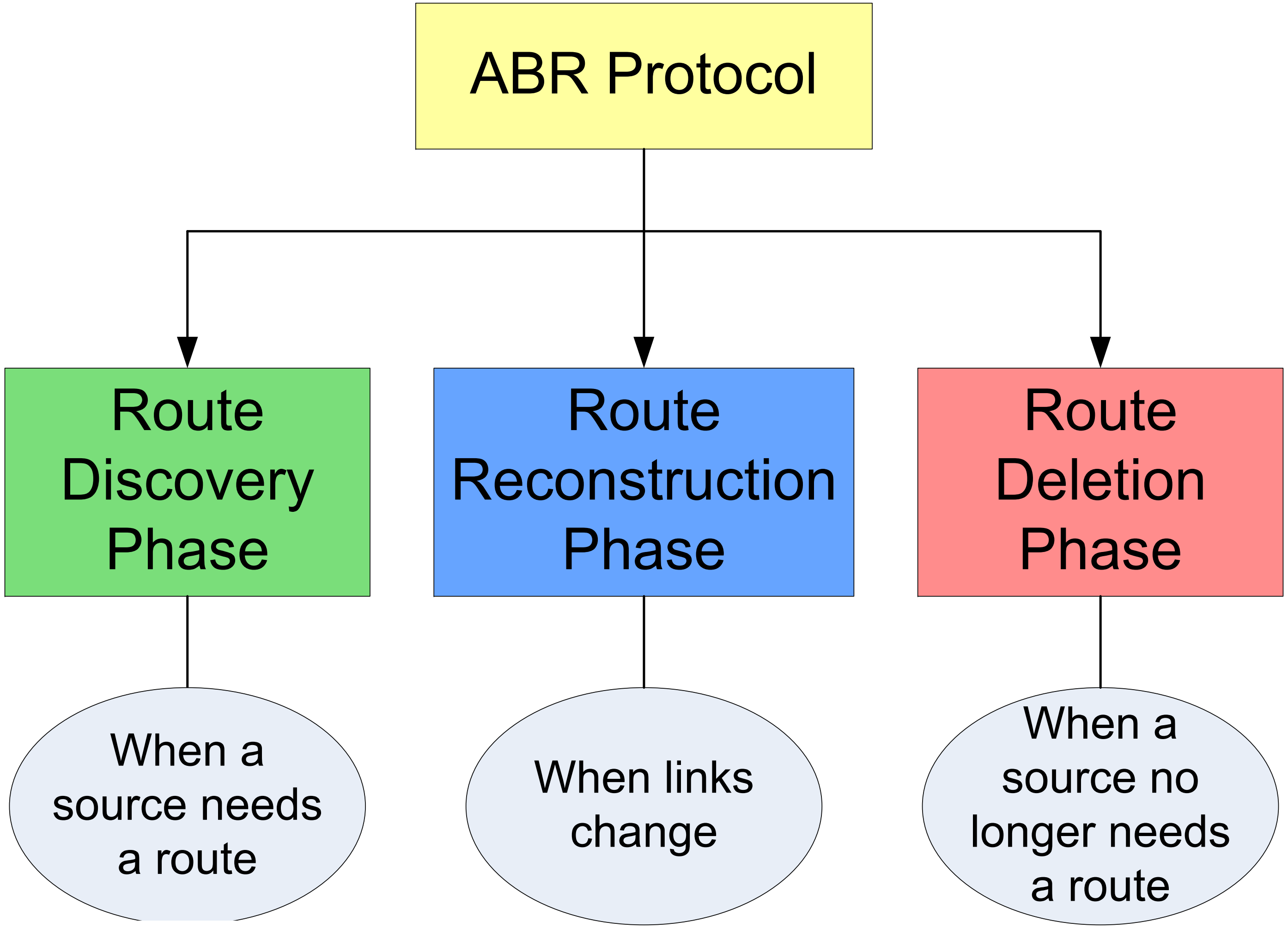
Associativity Ticks



Principles of ABR

- Principle 1 : A mobile host's association with neighbours changes under migration. Migration is such that after this unstable period, there EXIST a period of STABILITY, where the mobile host will spend more dormant time within a wireless cell before it moves again.
- Principle 2 : A mobile host is said to exhibit a HIGH state of mobility when it has LOW associativity ticks with its neighbours. If HIGH associativity ticks are observed, this implies host is in its stable state and it is ideal point to perform ad hoc routing.

ABR Protocol Description

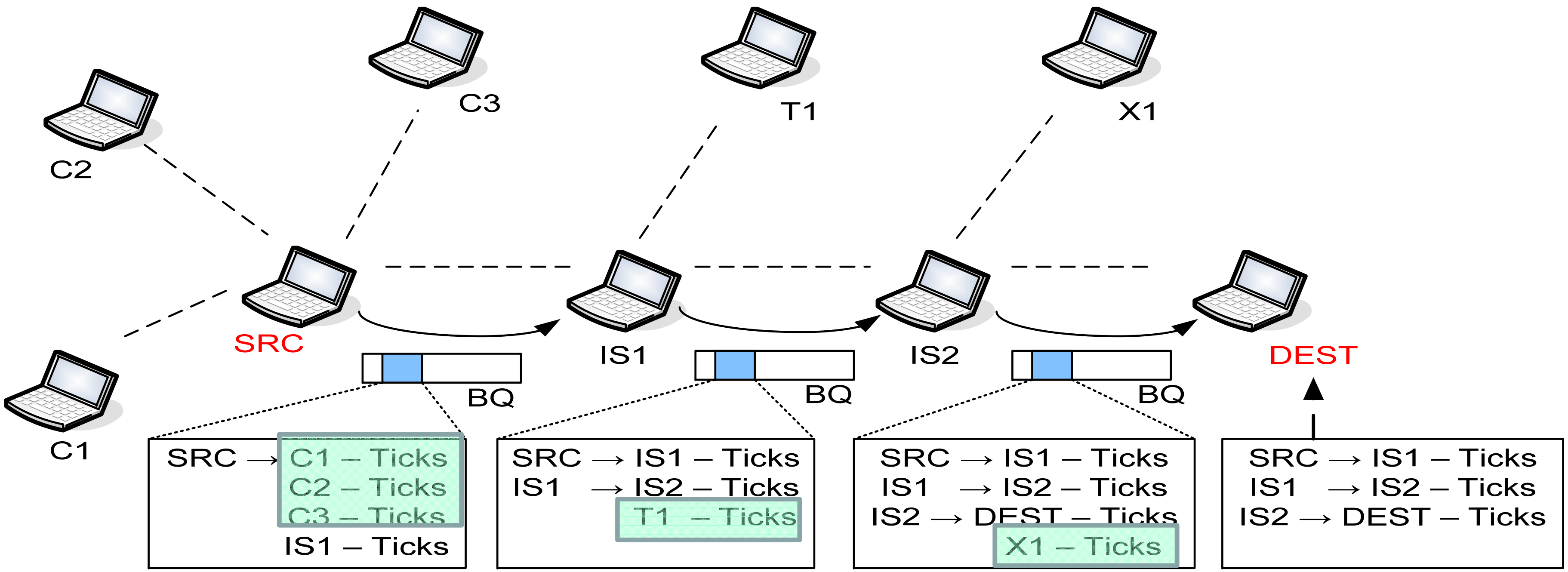


Route Selection Rules

On-demand routing

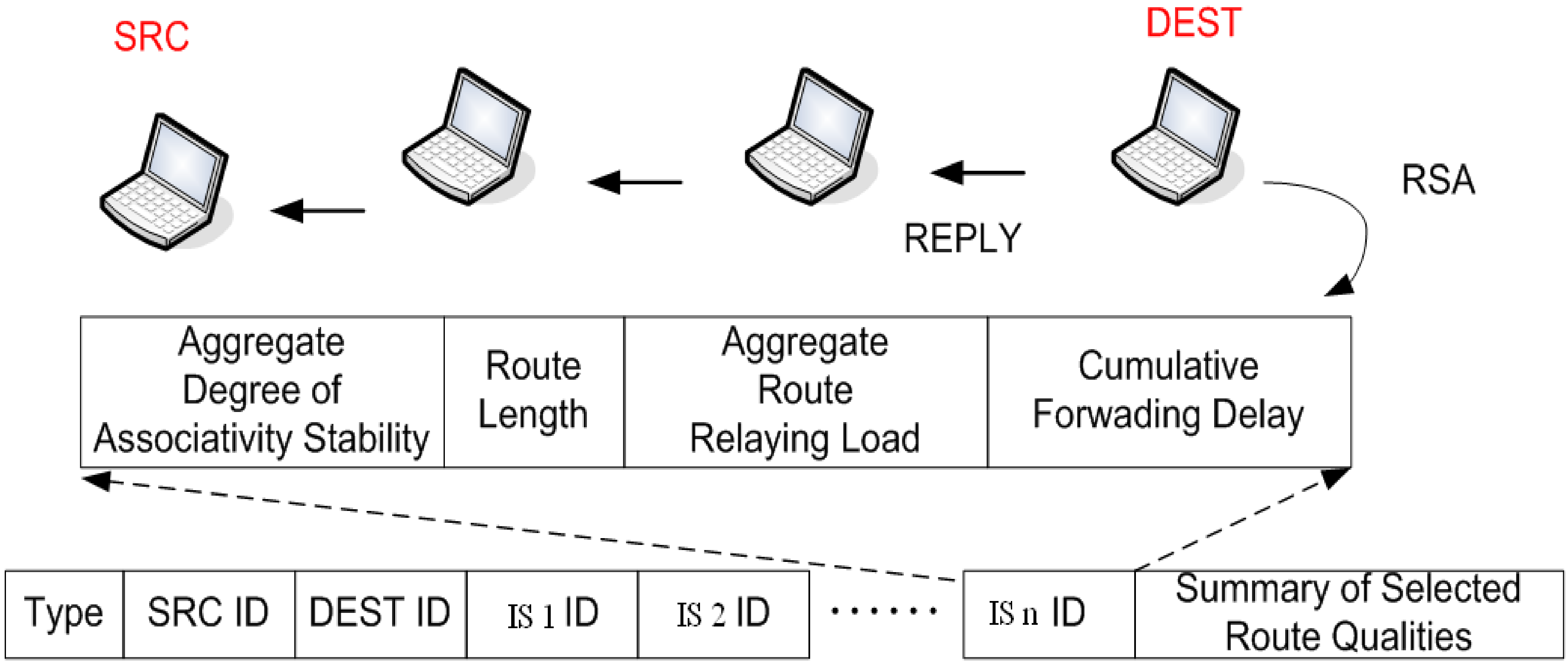
- Source (SRC) initiates search for route when needed
- Destination (DEST) selects most stable route:
  - DEST gets set of possible routes from SRC.
  - DEST chooses route with high associativity ticks (most stable route with low node mobility).
  - If two or more routes have same stability level, DEST chooses shortest path (least number of hops).
  - Else if same stability level and same number of hops, DEST selects one arbitrarily.

Route Discovery Phase: Broadcast Query, BQ



Do not send data that is not needed

Route Discovery Phase: REPLY



ABR Data Flow Acknowledgement

Passive Acknowledgement scheme:

- A node receives packet, and
- performs relaying via radio transmission to next neighbour

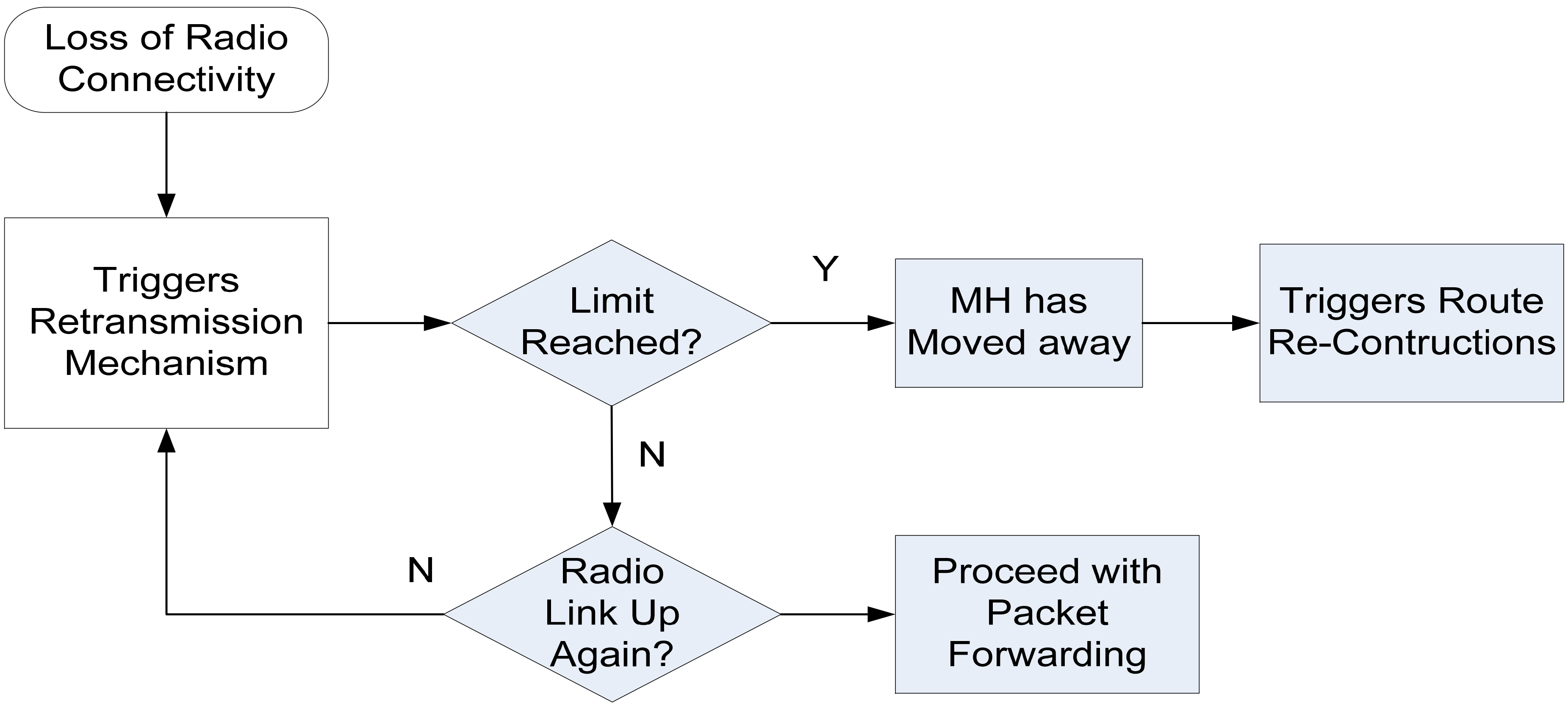
  => previous (upstream) neighbour hears transmission, and

  => indirectly used as an acknowledgement to packet sent.

Active Acknowledgement scheme:

- Active acknowledgement only sent by DEST because has no receiving neighbour

ABR Data Packet Retransmission

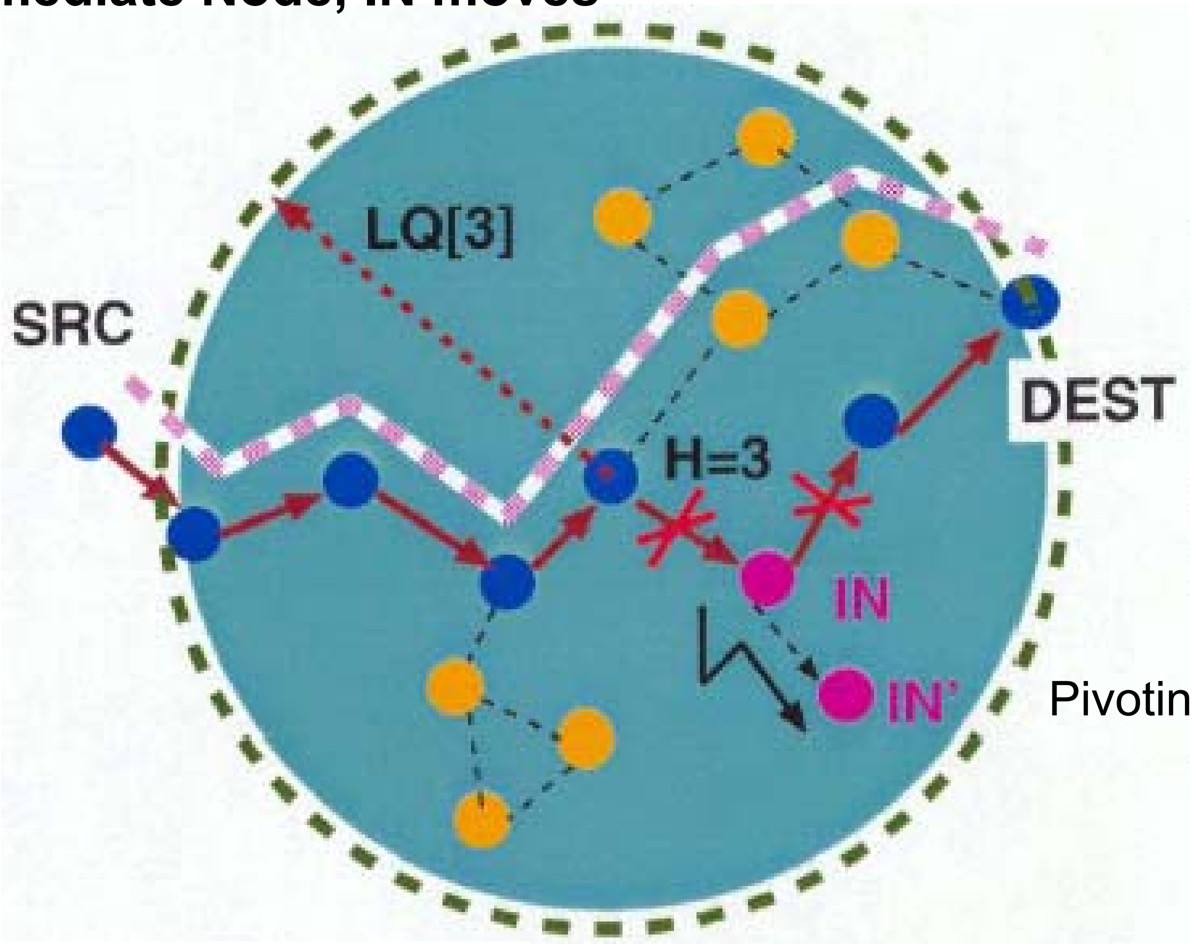


Route ReConstruction, RRC phase

- RRC is involved, when route changes due to source, destination, intermediate or subnet-bridging MH migration i.e. mobile device moves for unexpected engagement or switches off
  - associativity relationship among nodes is violated
- ABR route recovery is fast because it uses **partial route discovery.**
- RRC does not produce BQ again unless necessary.
  - avoids excessive control overhead and disturbing unconcerned nodes.

RRC Phase - scenario 1

Intermediate Node, IN moves

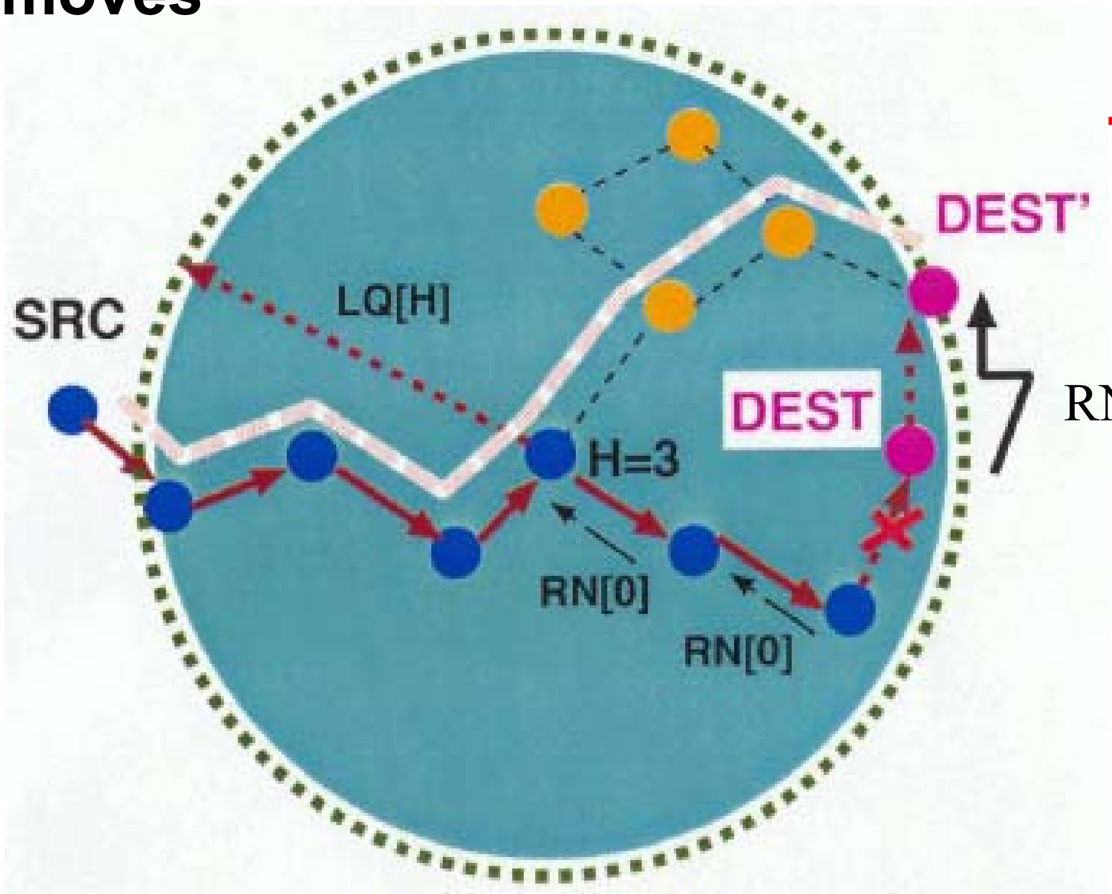


Original route

Pivoting node = start of LQ[H]

RRC Phase - scenario 2

DEST moves

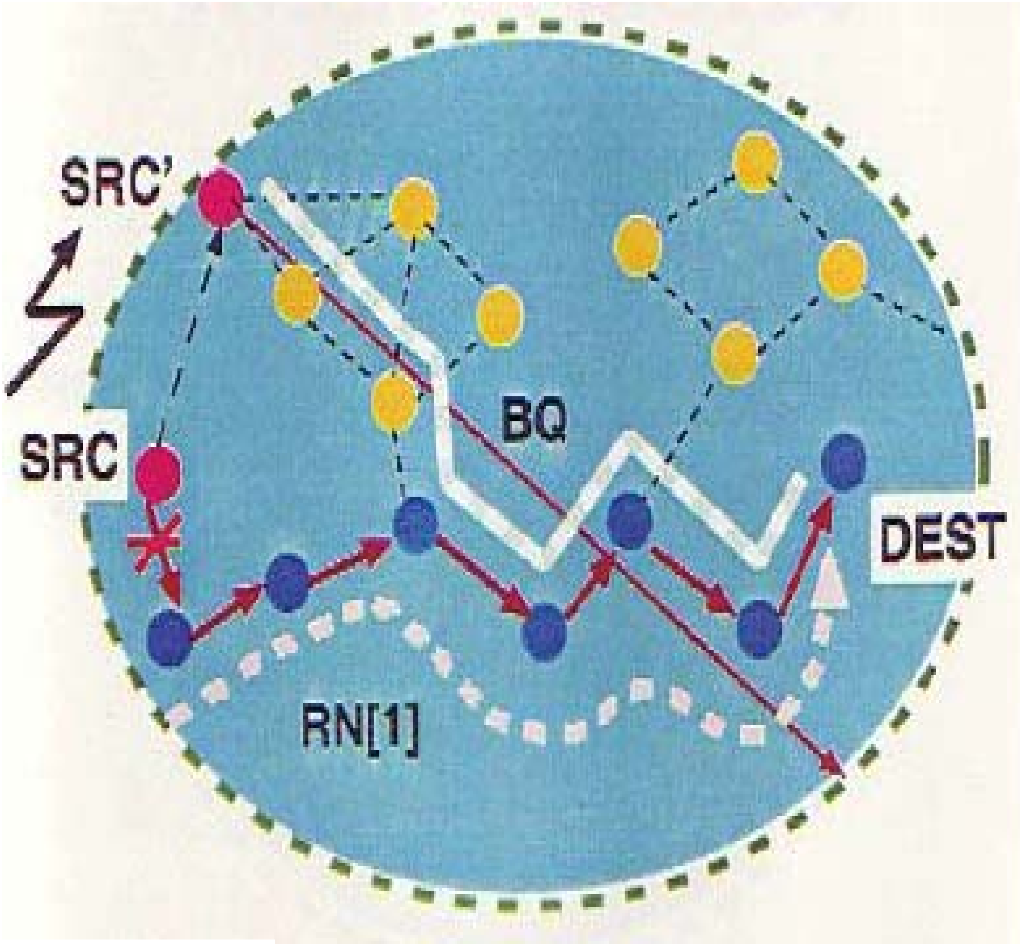


Original route

RN[]: Route Notification

RRC Phase - scenario 3

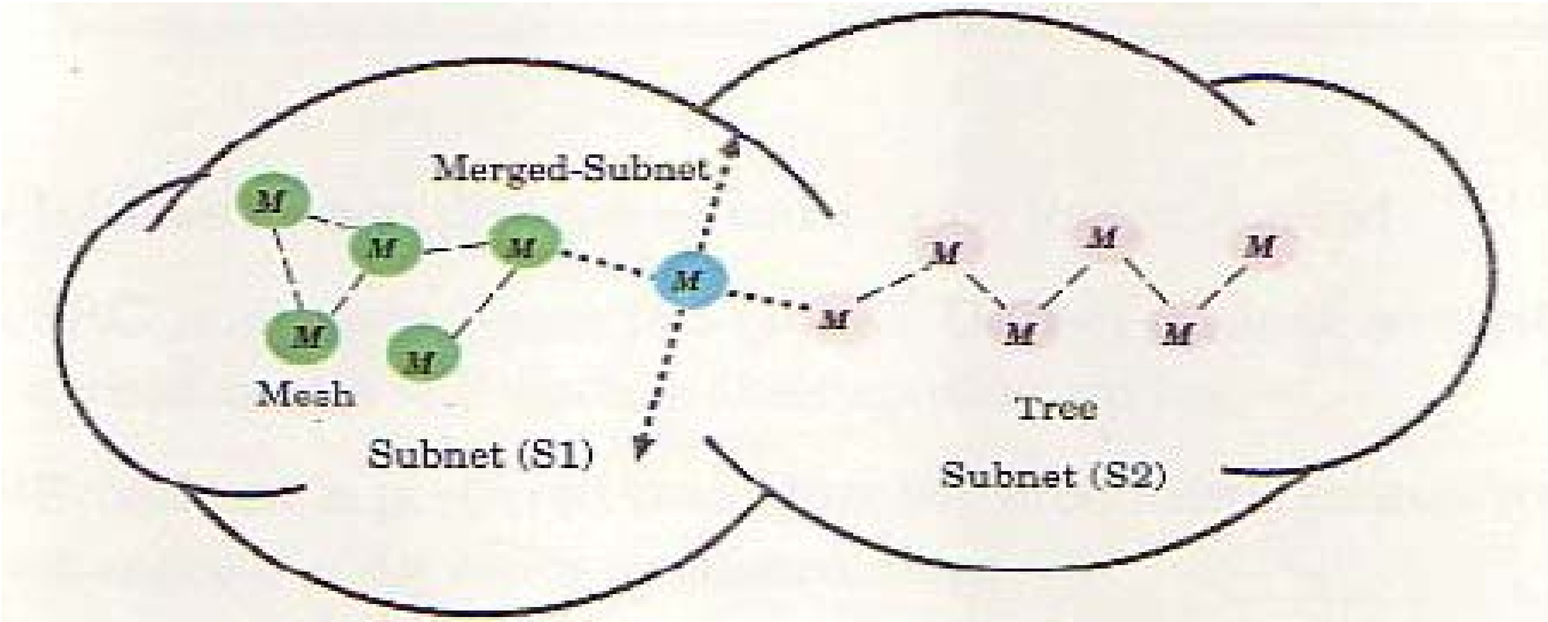
SRC Moves



Concurrent Nodes Movements

- Results in multiple invocations of Route ReConstructions, RRC
- SRC will send BQ to delete unfruitful Reconstructions (see Route deletion later)
- Ultimately only **ONE** Route Reconstruction succeeds.

Movement of Subnet Bridging Mobile Host, M



- Movement of M causes mobile network to be partitioned.
- If existing ad hoc routes are all within each subnet, then no RRC necessary.
- If routes span across subnets, movement of M causes network partition and BQ-REPLY cycle will be necessary.

Route Deletion Phase, RD

- When the SRC no longer desires route, Route Deletion (RD) broadcast is initiated
- **Hard State:** SRC broadcasts RD messages

  => all INs will update routing tables to:

  - Free up resources
  - Avoid keeping stale routes
- **Soft State:** - based on timer when there is no traffic activity related to route over a period of time. This approach is performed at each node in route.

Proactive Vs Reactive approach

Pro-active	Reactive
➢Will always react or do something	➢ React specifically to link changes
➢Reaction in addition to those for link changes	➢React to need by the source
➢Not efficient if little mobility	➢No periodic route update
➢Periodic route updates	➢Similar to on-demand protocols