

Mobile Computing and Communications

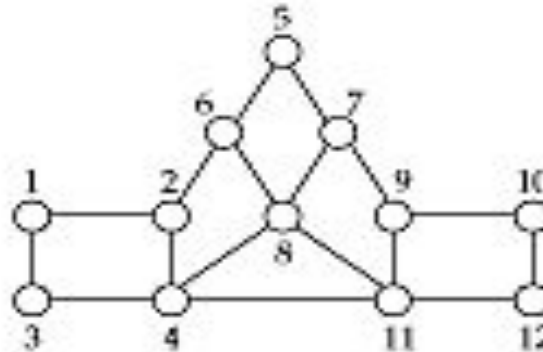
UNIT III Routing Protocols

Outline

- Routing Protocols
 - Reactive Protocols
 - Proactive Protocols
 - Hybrid Protocols

Traditional Routing

- A *routing protocol* sets up a *routing table* in *routers*



ROUTING TABLE AT 1

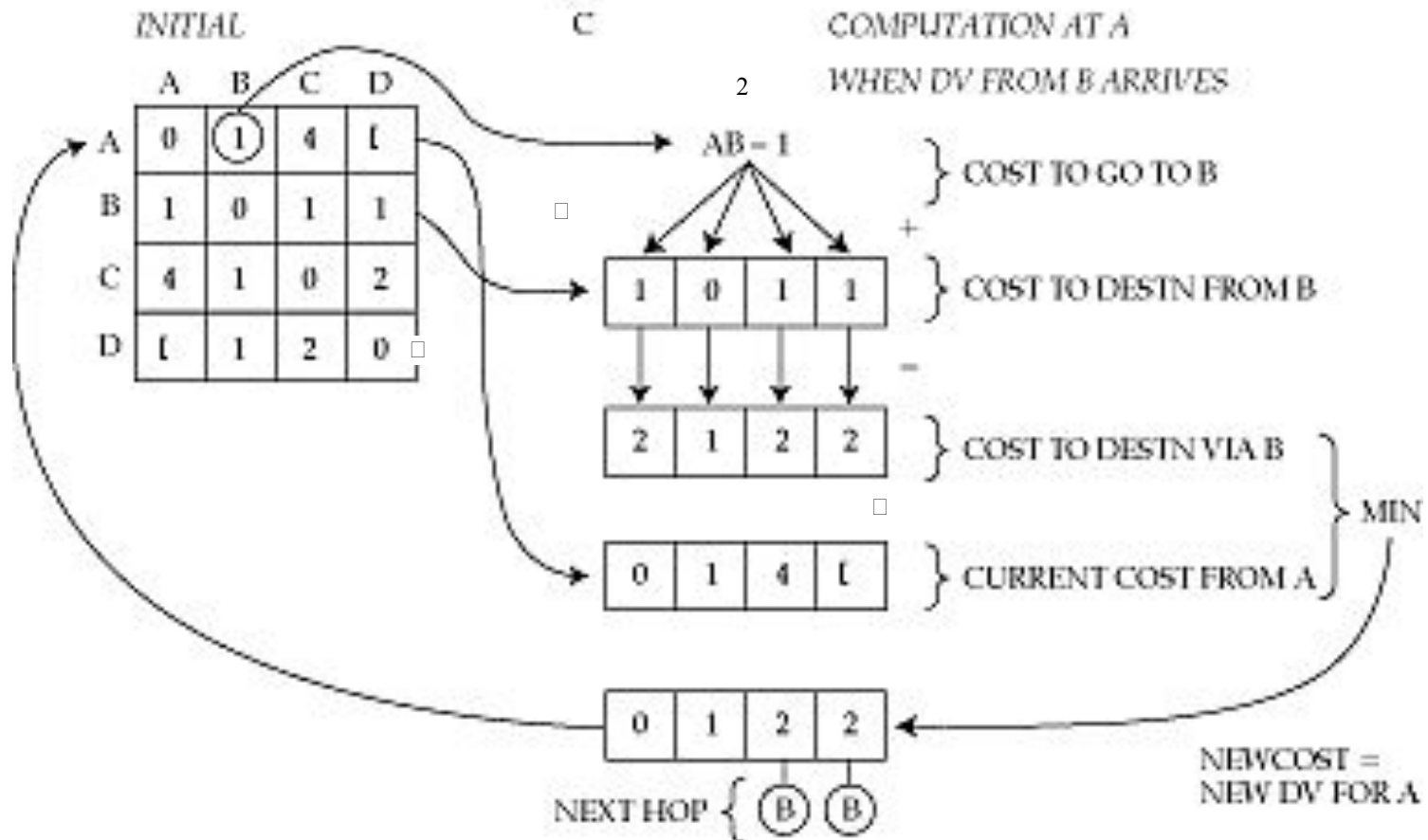
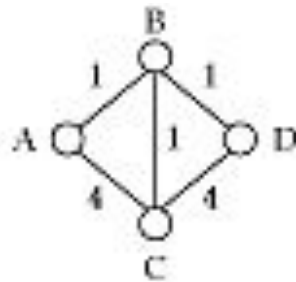
Destination	Next hop	Destination	Next hop
1	—	7	2
2	2□	8□	2□
3	3□	9□	2□
4	3□	10□	2□
5	2□	11□	3□
6	2	12	3

- A node makes a *local* choice depending on *global* topology

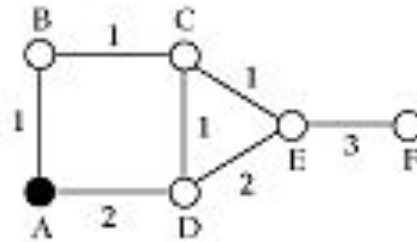
Distance-vector & Link-state Routing

- Both assume router knows
 - address of each neighbor
 - cost of reaching each neighbor
- Both allow a router to determine global routing information by talking to its neighbors
- **Distance vector** - router knows cost to each destination
- **Link state** - router knows entire network topology and computes shortest path

Distance Vector Routing: Example



Link State Routing: Example



B(A,1) means B was reached by A, cost 1

PERMANENT	TEMPORARY	COMMENTS
A	B(A,1), D(A,2)	ROOT AND ITS NEIGHBORS
A, B(A,1)	D(A,2), C(B,2)	ADD C(B,2)
A, B(A,1) D(A,2)	E(D,4), C(B,2)	C(D,3) DIDN'T MAKE IT
A, B(A,1) D(A,2), C(B,2)	E(C,3)	E(D,4) TOO LONG
A, B(A,1) D(A,2), C(B,2) E(C,3)	F(E,6)	
A, B(A,1) C(B,2), D(A,2) E(C,3), F(E,6)	NULL	STOP

A •

A • —¹• B

 D
 2
A • —¹• B

 D
 2
A • —¹• B —¹• C

 D
 2
A • —¹• B —¹• C —¹• E

 D
 2
A • —¹• B —¹• C —¹• E —³• F

Routing in Mobile Computing

Routing Protocols

■ Proactive protocols

- Traditional distributed shortest-path protocols
- Maintain routes between every host pair at all times
- Based on periodic updates; High routing overhead
- Example: DSDV (destination sequenced distance vector)

■ Reactive protocols

- Determine route if and when needed
- Source initiates route discovery
- Example: DSR (dynamic source routing)

■ Hybrid protocols

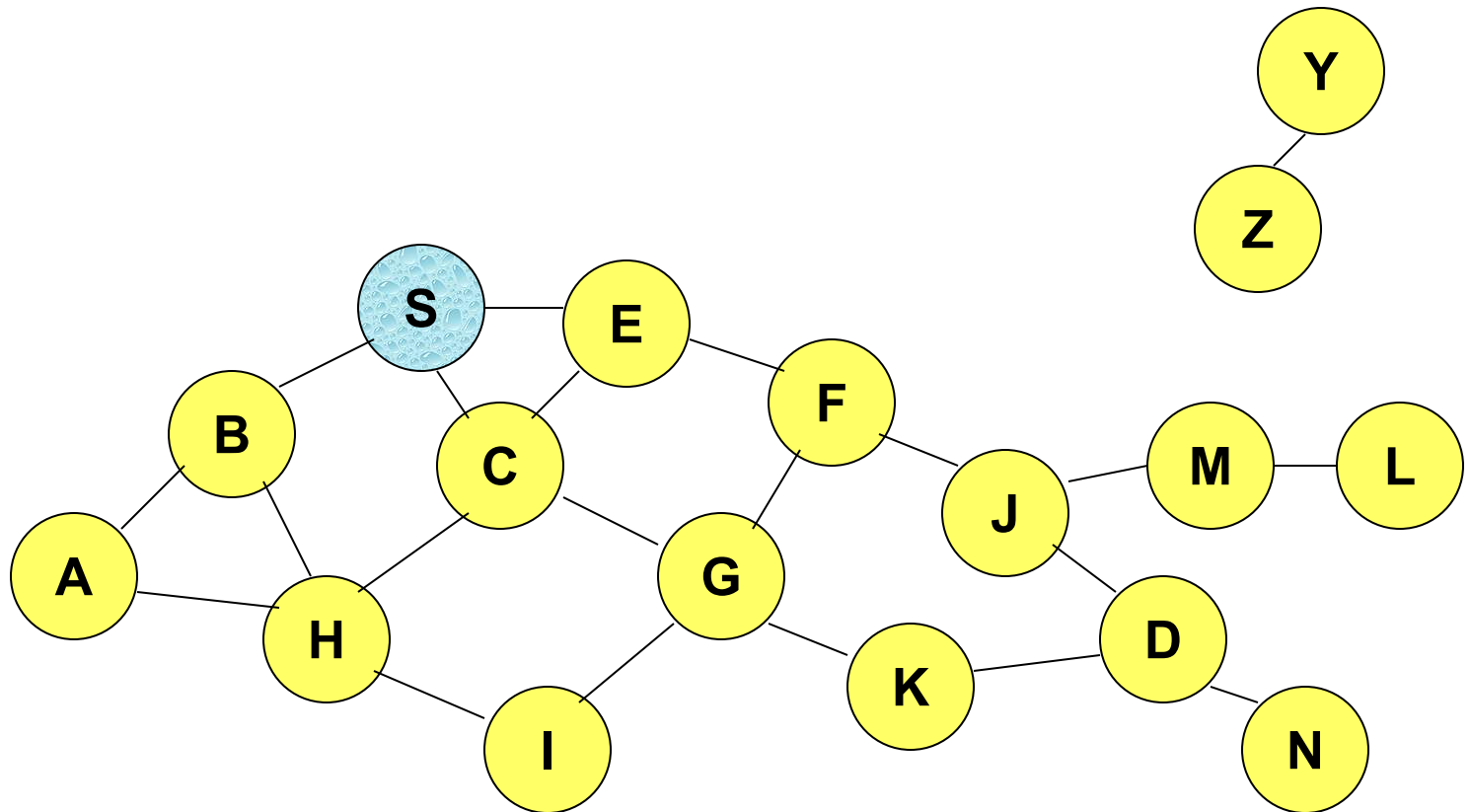
- Adaptive; Combination of proactive and reactive
- Example : ZRP (zone routing protocol)

Reactive Routing Protocols

Dynamic Source Routing (DSR) [Johnson96]

- When node S wants to send a packet to node D, but does not know a route to D, node S initiates a **route discovery**
- Source node S floods **Route Request (RREQ)**
- Each node *appends own identifier* when forwarding RREQ

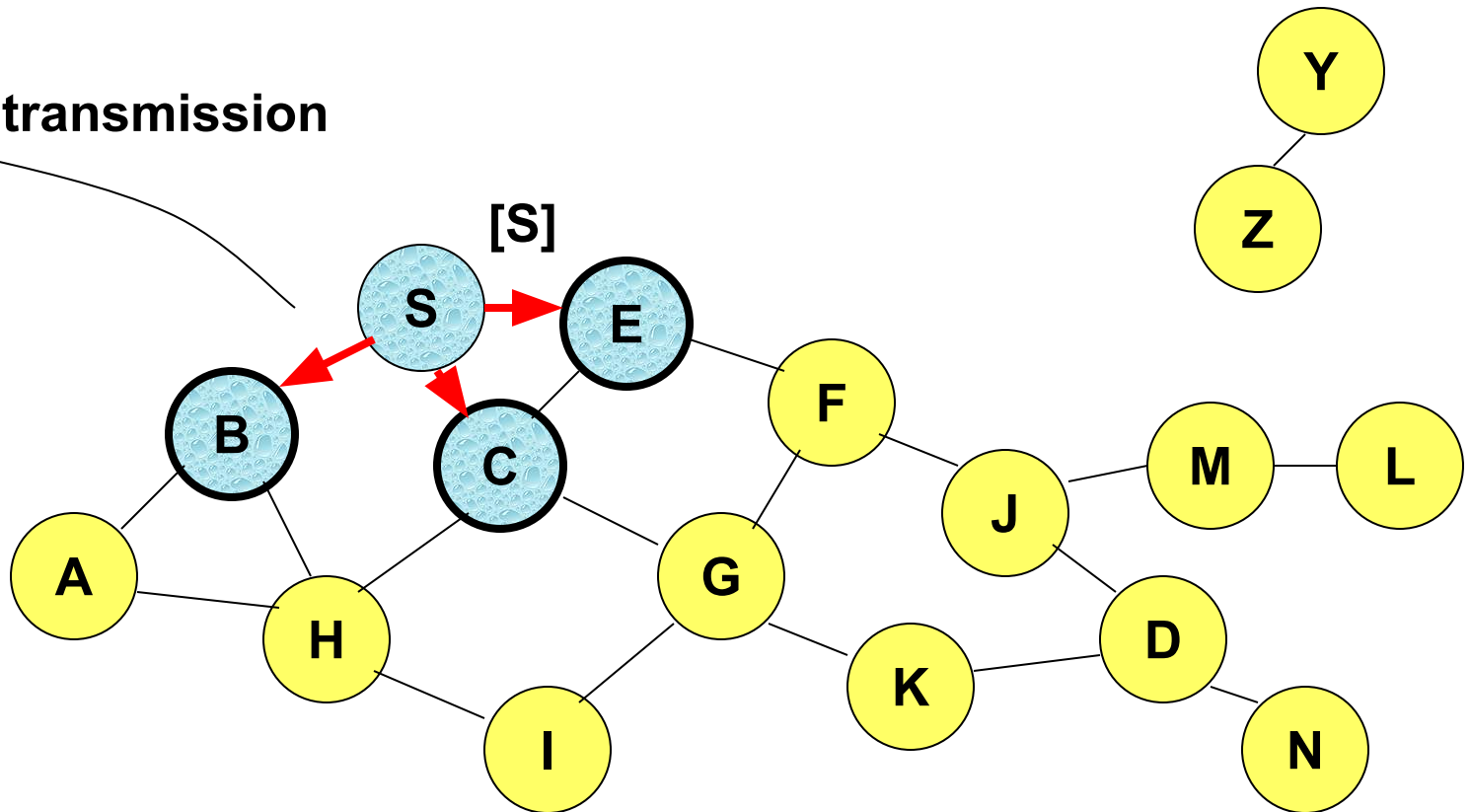
Route Discovery in DSR



Represents a node that has received RREQ for D from S

Route Discovery in DSR

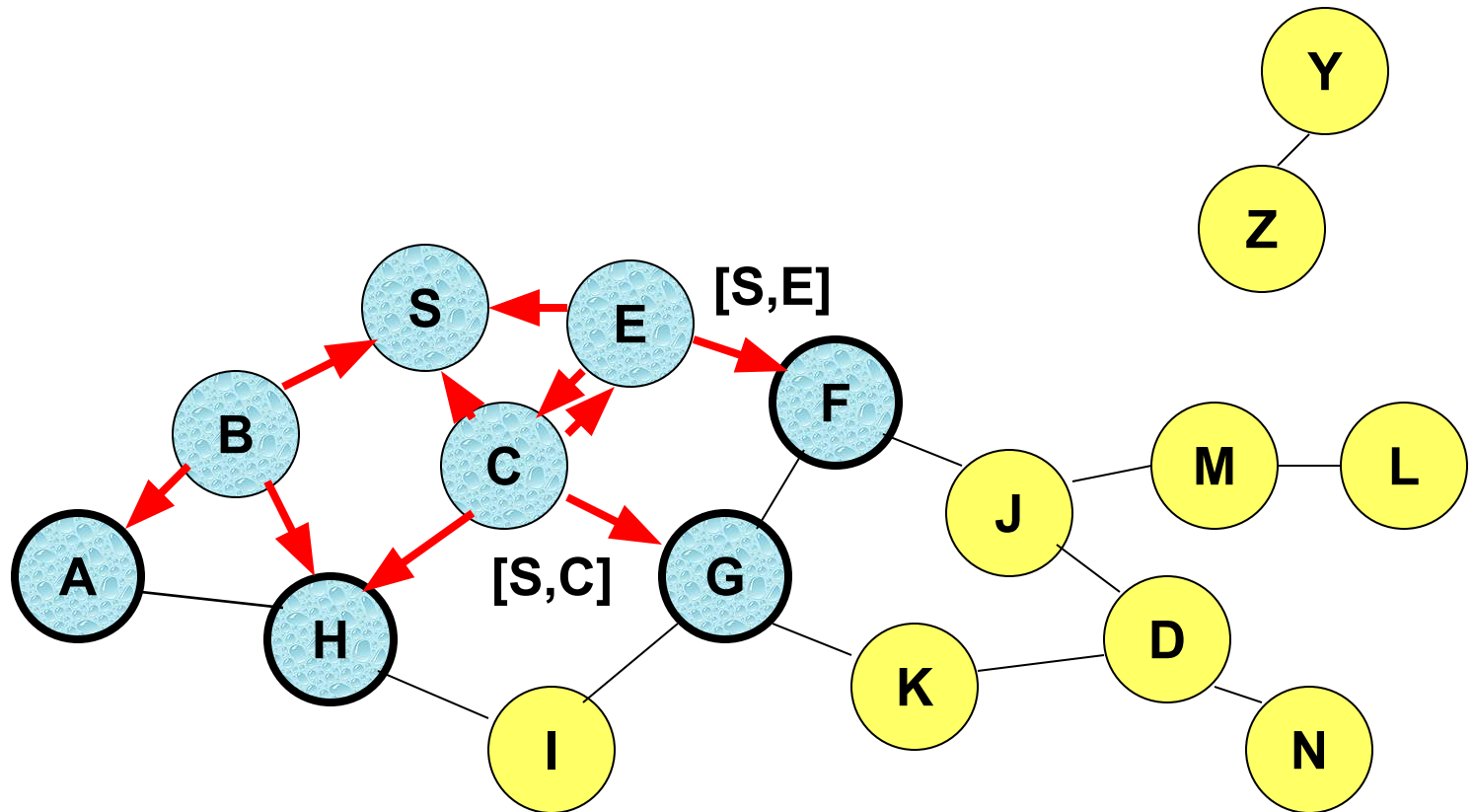
Broadcast transmission



→ Represents transmission of RREQ

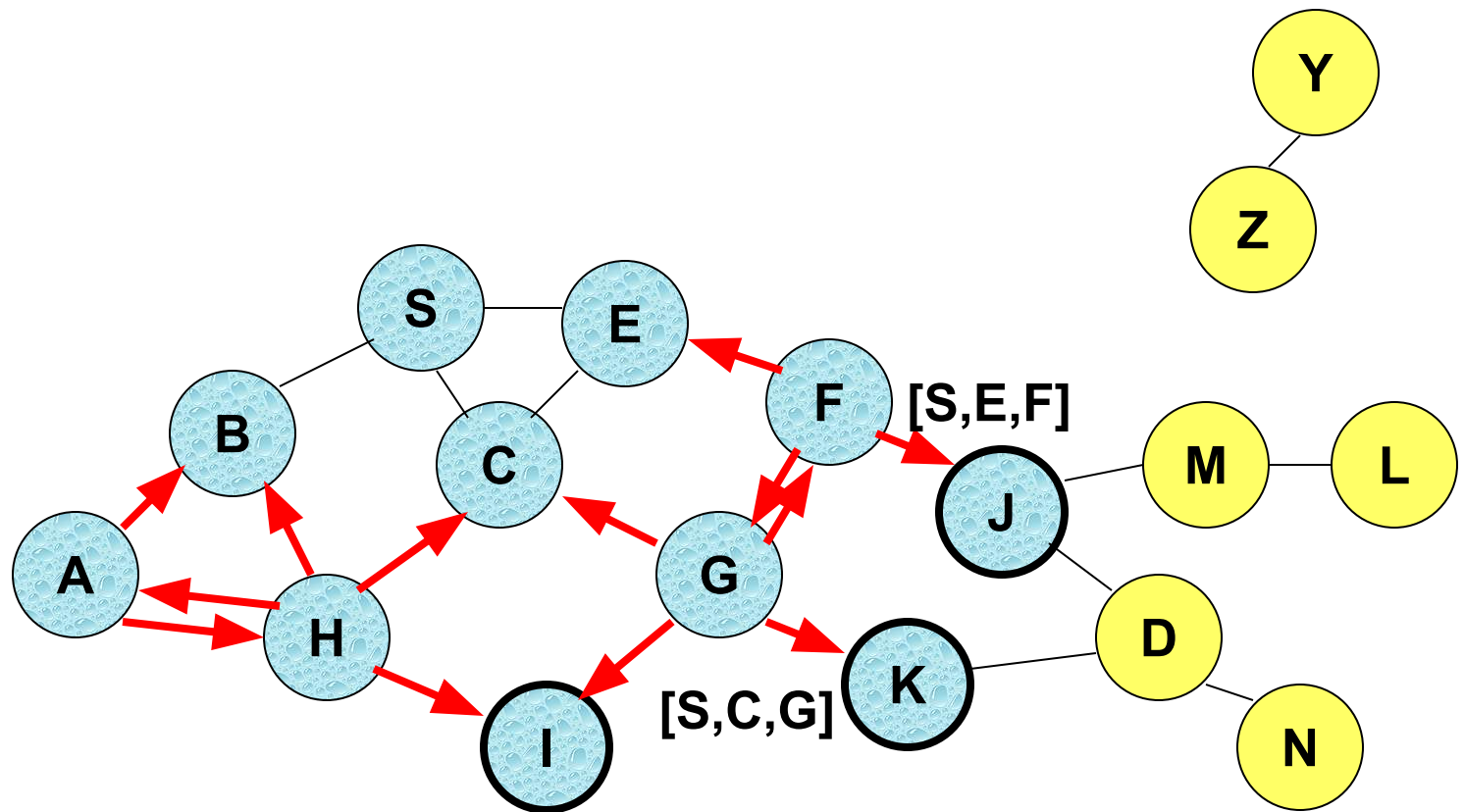
[X,Y] Represents list of identifiers appended to RREQ

Route Discovery in DSR



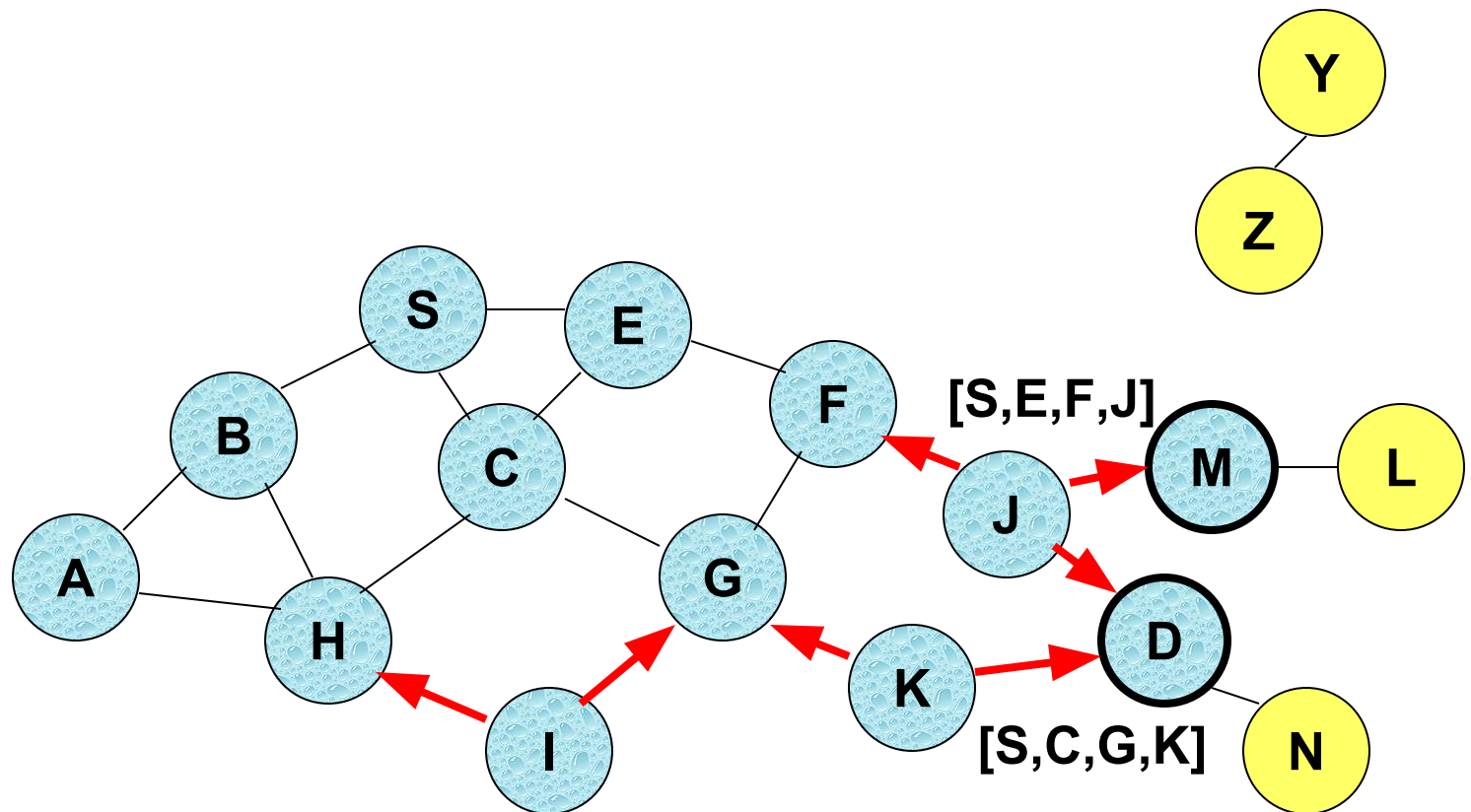
- Node H receives packet RREQ from two neighbors:
potential for collision

Route Discovery in DSR



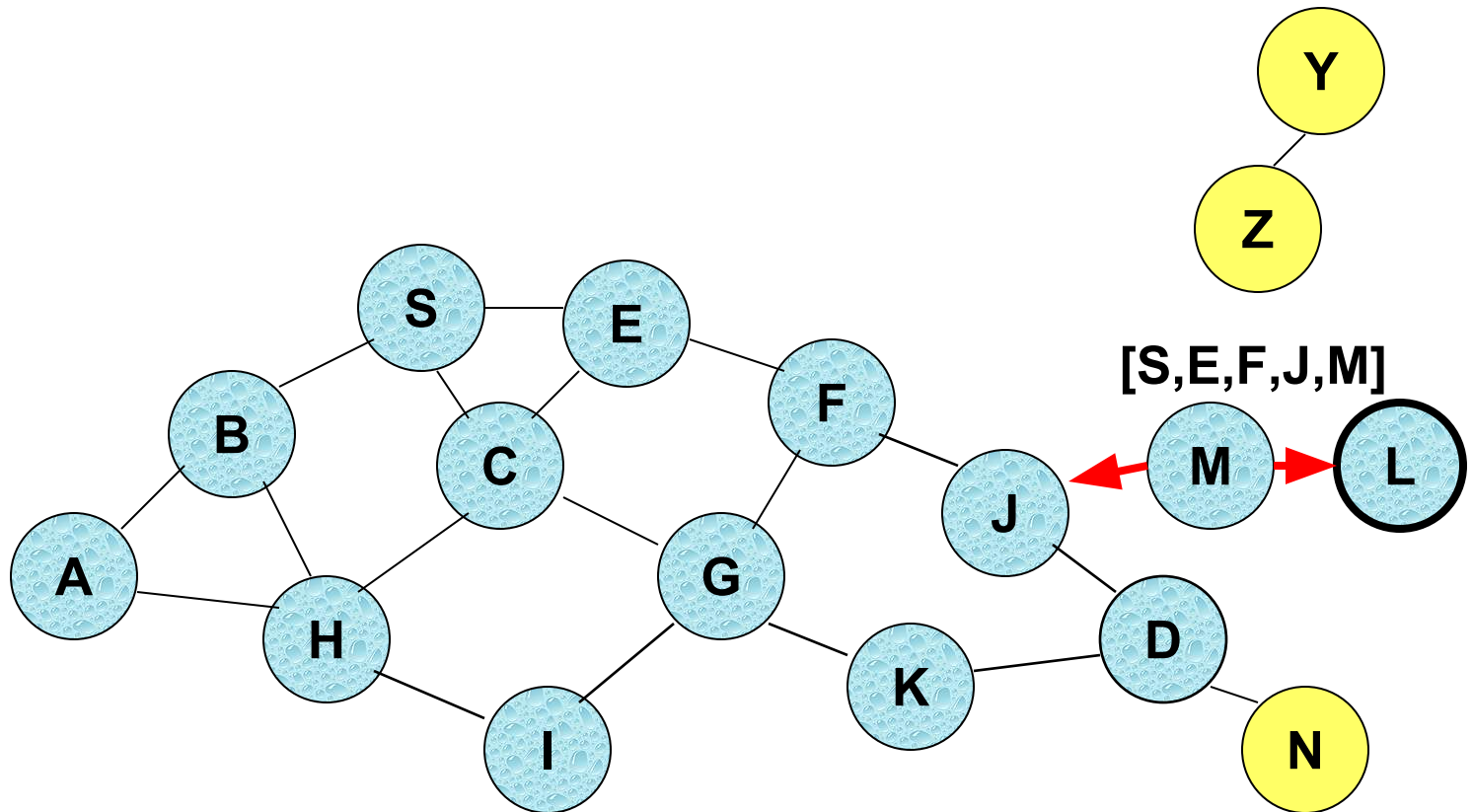
- Node C receives RREQ from G and H, but does not forward it again, because node C has **already forwarded RREQ** once

Route Discovery in DSR



- Nodes J and K both broadcast RREQ to node D
- Since nodes J and K are **hidden** from each other, their **transmissions may collide**

Route Discovery in DSR

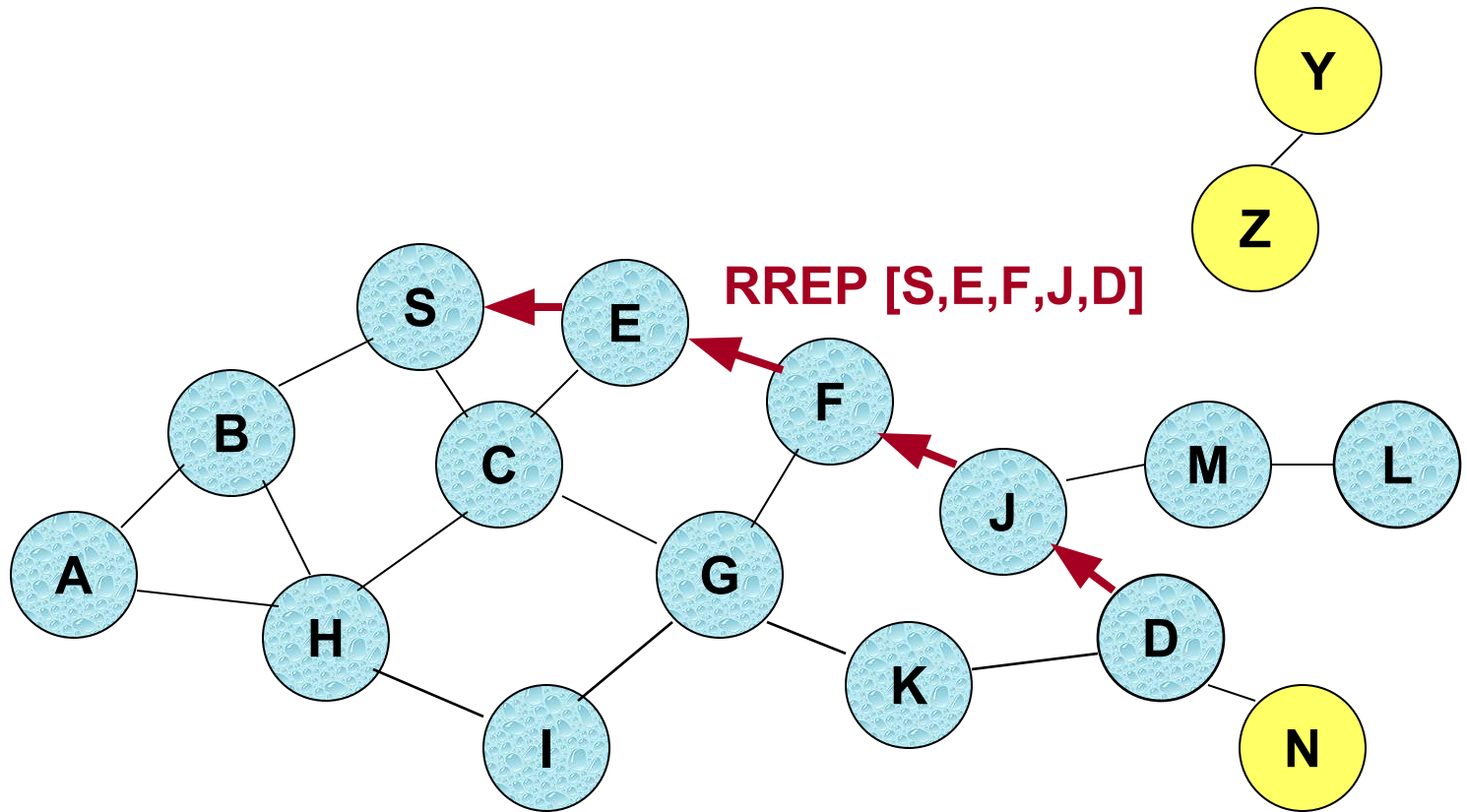


- Node D **does not forward** RREQ, because node D is the **intended target** of the route discovery

Route Discovery in DSR

- 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 in DSR

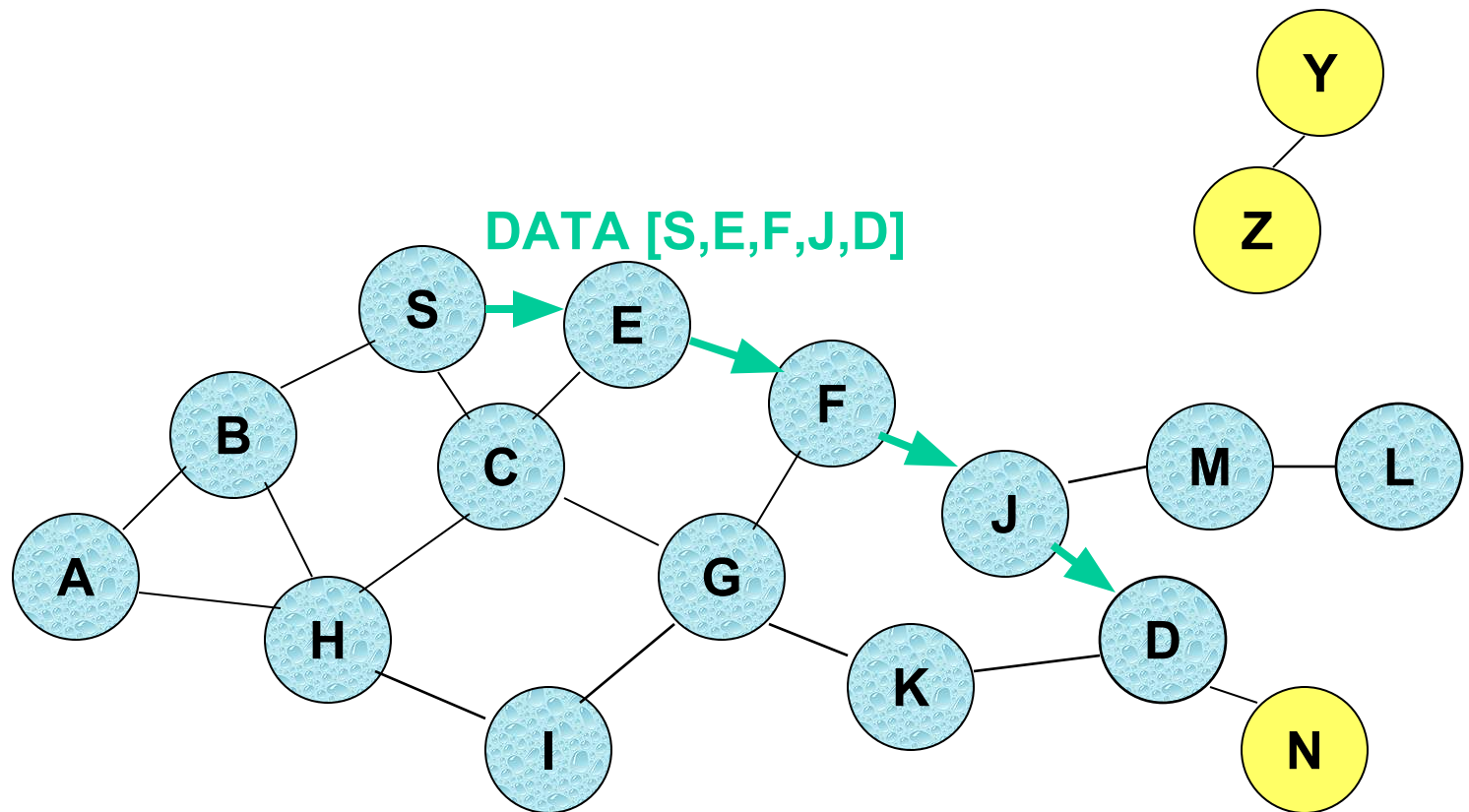


← Represents RREP control message

Dynamic Source Routing (DSR)

- 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

Data Delivery in DSR



Packet header size grows with route length

DSR Optimization: Route Caching

- Each node caches a new route it learns by *any means*
- 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
- **Problem:** Stale caches may increase overheads

Dynamic Source Routing: Advantages

- 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

Dynamic Source Routing: Disadvantages

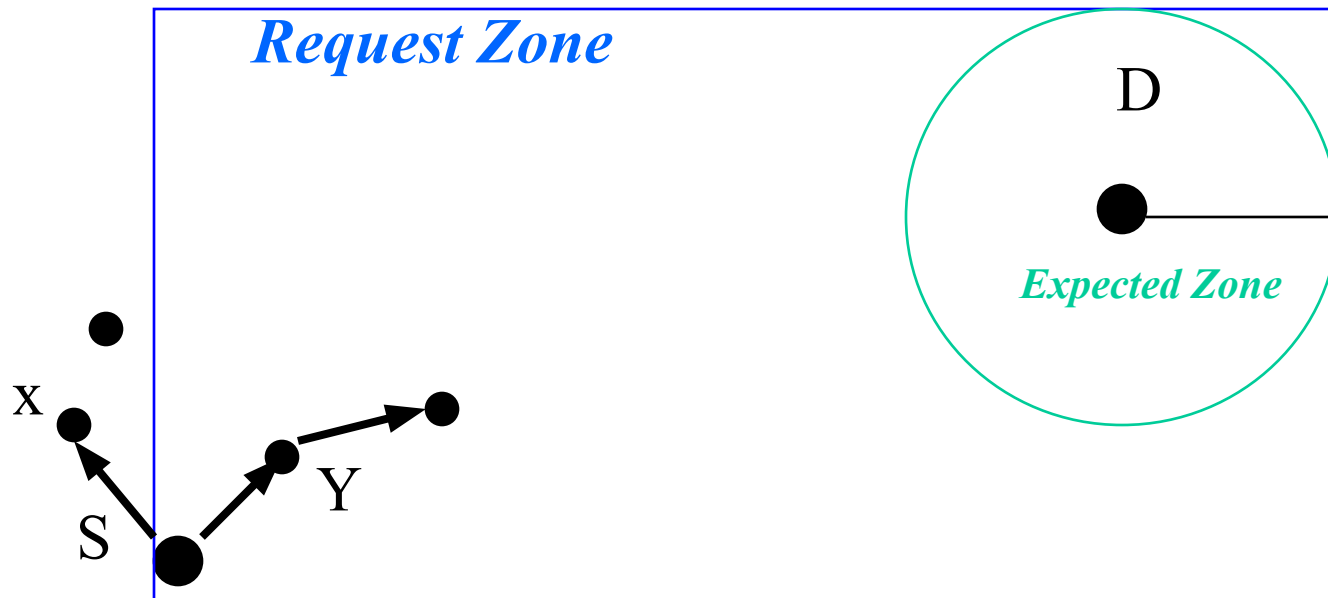
- Packet header size grows with route length due to source routing
- Flood of route requests may potentially reach all nodes in the network
- Potential collisions between route requests propagated by neighboring nodes
 - insertion of random delays before forwarding RREQ
- Increased contention if too many route replies come back due to nodes replying using their local cache
 - Route Reply *Storm* problem
- Stale caches will lead to increased overhead

Location-Aided Routing (LAR) [Ko98Mobicom]

- Exploits location information to limit scope of route request flood
 - Location information may be obtained using GPS
- *Expected Zone* is determined as a region that is expected to hold the current location of the destination
 - Expected region determined based on potentially old location information, and knowledge of the destination's speed
- Route requests limited to a *Request Zone* that contains the Expected Zone and location of the sender node

Request Zone

- Define a **Request Zone**
- LAR is same as flooding, except that only nodes in request zone forward route request
- Smallest rectangle including S and expected zone for D



Location Aided Routing (LAR)

- Advantages
 - reduces the scope of route request flood
 - reduces overhead of route discovery
- Disadvantages
 - Nodes need to know their physical locations
 - Does not take into account possible existence of obstructions for radio transmissions

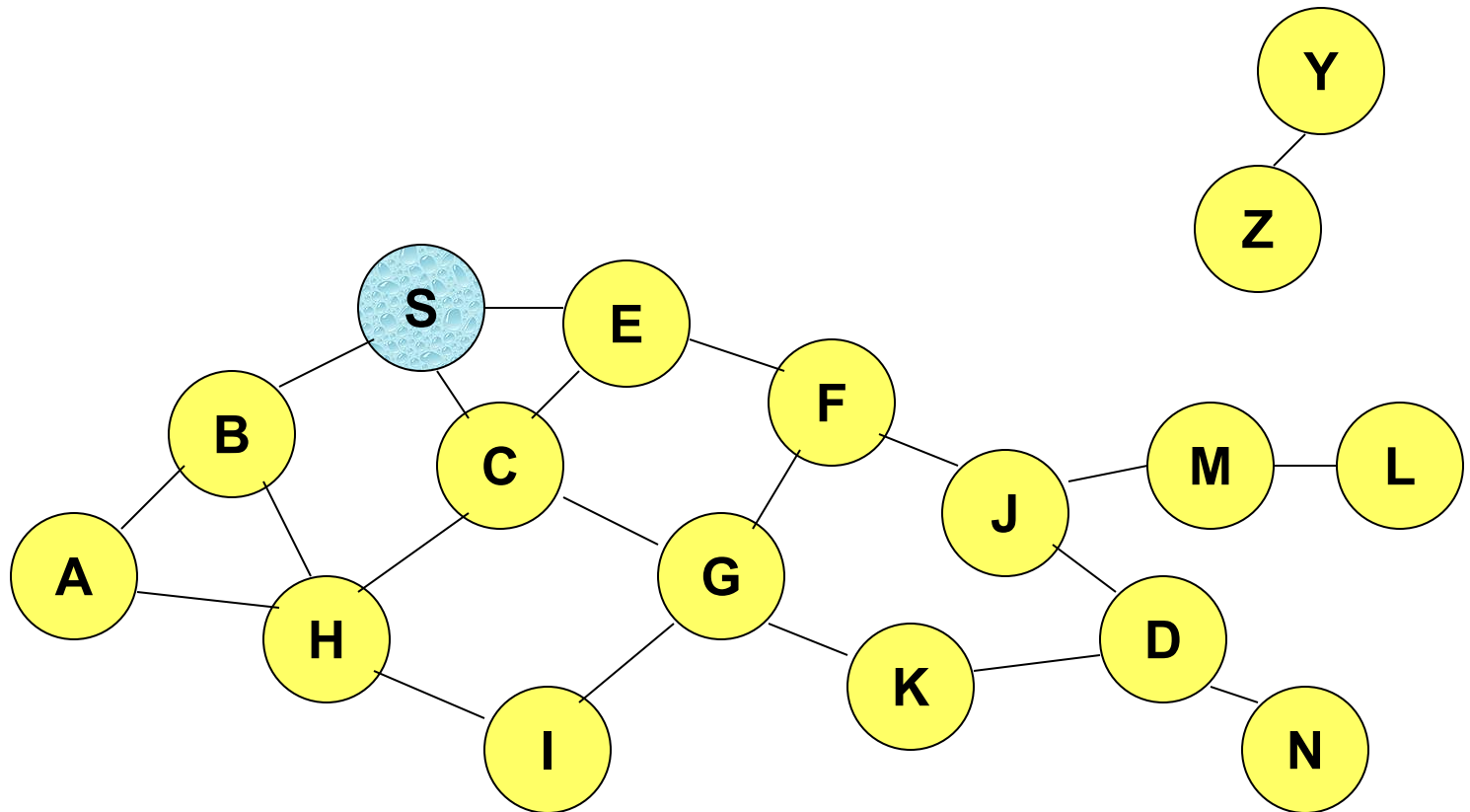
Ad Hoc On-Demand Distance Vector Routing (AODV) [Perkins99Wmcsa]

- DSR includes source routes in packet headers
- Resulting large headers can sometimes degrade performance
 - particularly when data contents of a packet are small
- AODV attempts to improve on DSR by maintaining routing tables at the nodes, so that data packets do not have to contain routes
- AODV retains the desirable feature of DSR that routes are maintained only between nodes which need to communicate

AODV

- **Route Requests (RREQ)** are forwarded in a manner similar to DSR
- 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

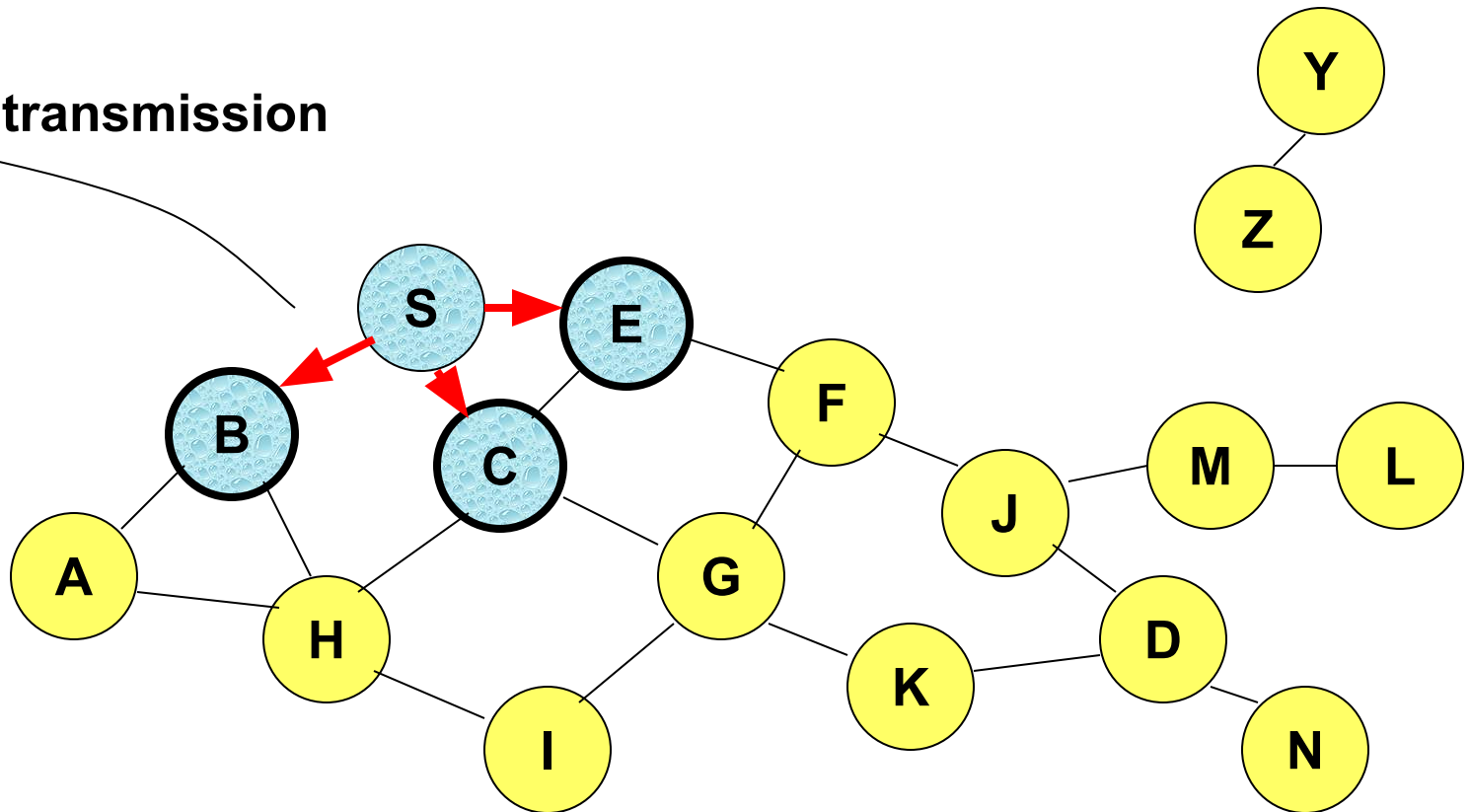
Route Requests in AODV



Represents a node that has received RREQ for D from S

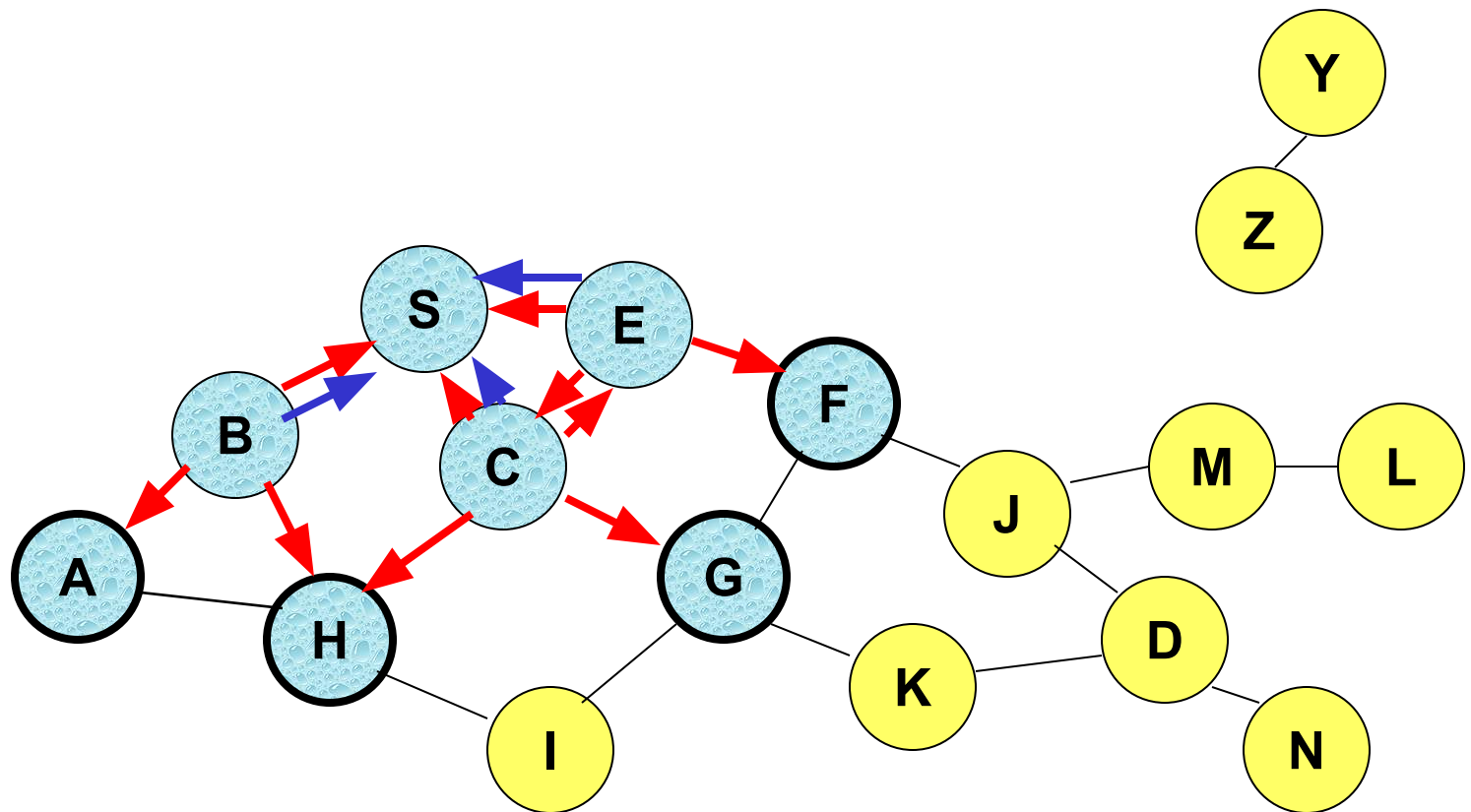
Route Requests in AODV

Broadcast transmission



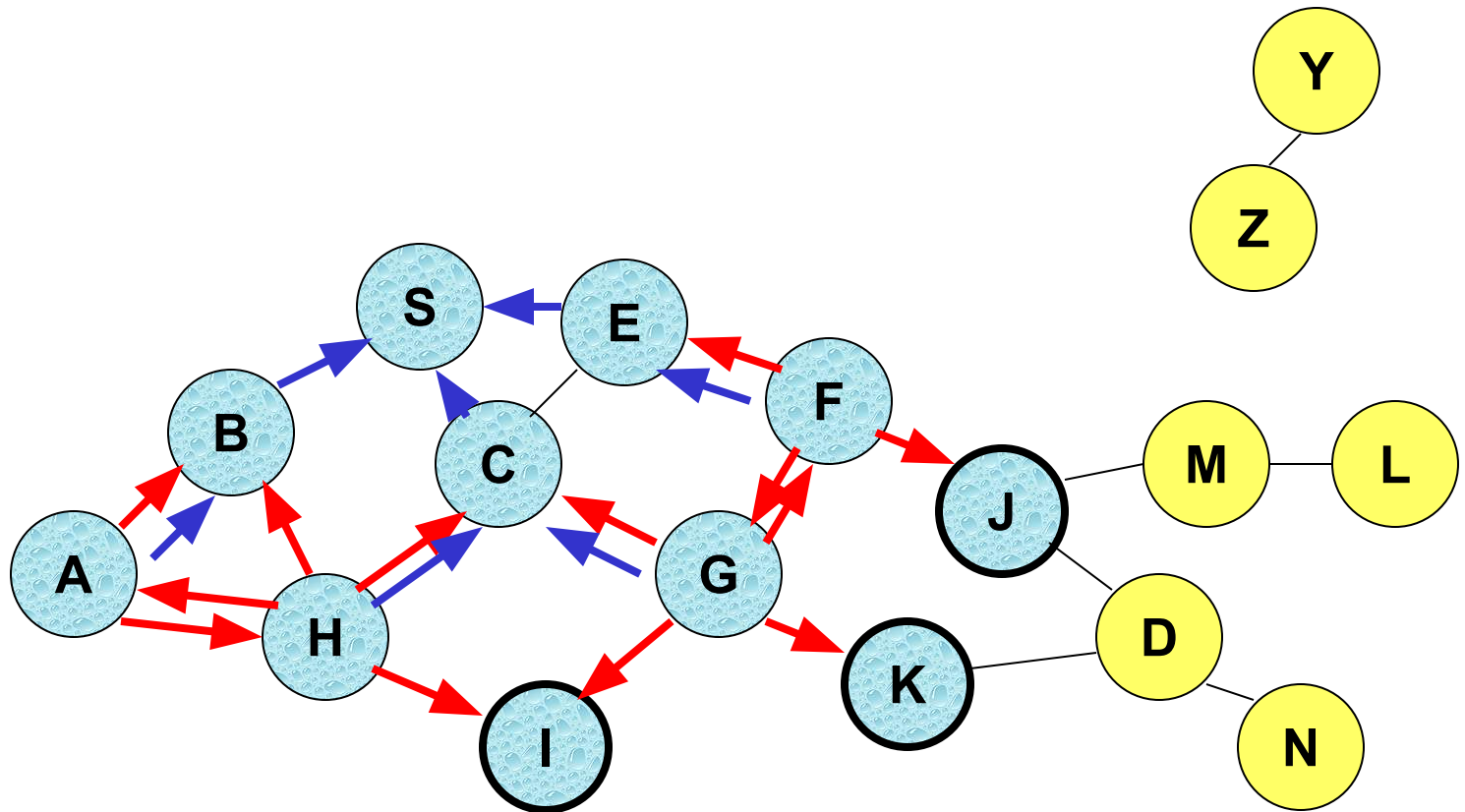
→ Represents transmission of RREQ

Route Requests in AODV



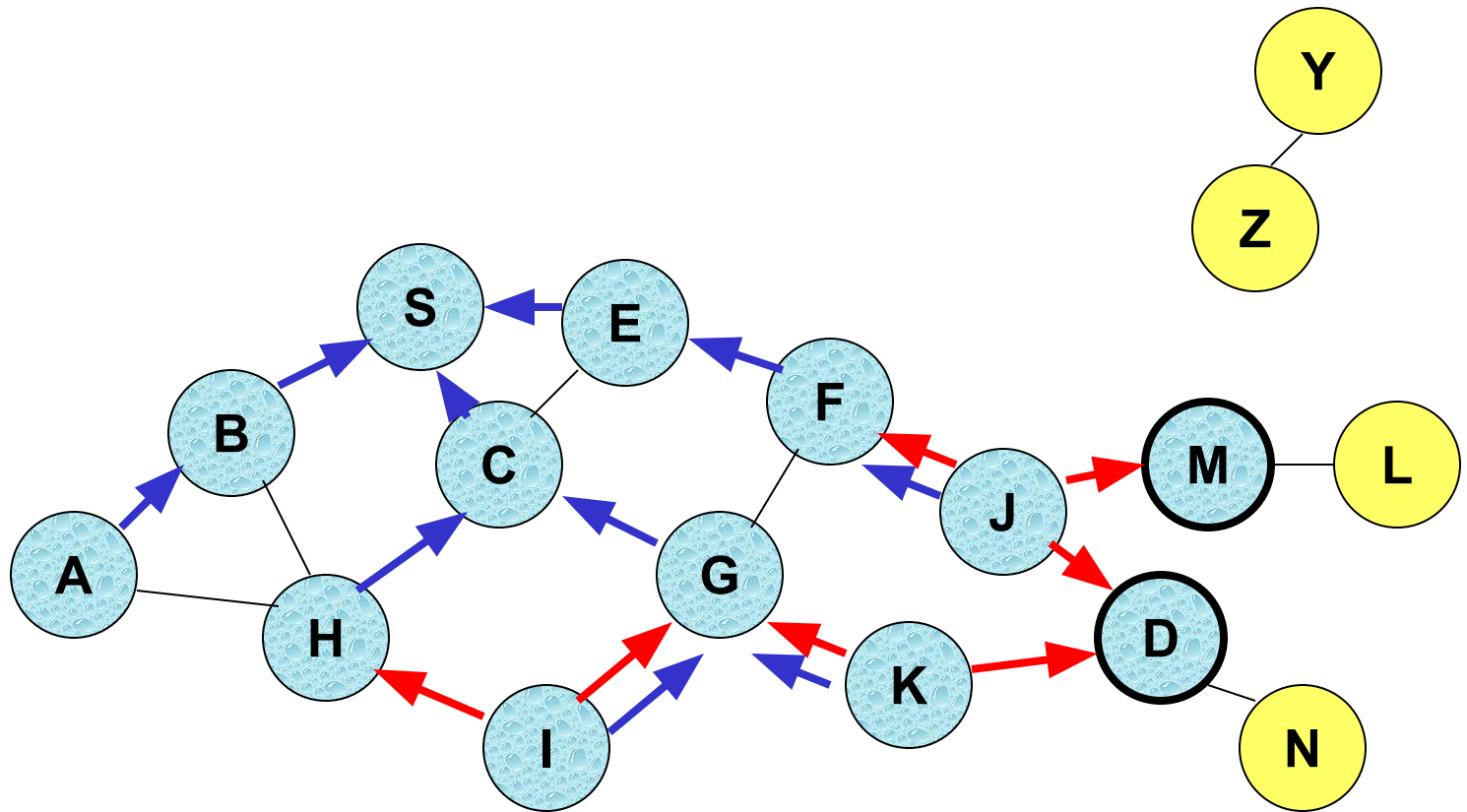
Represents links on Reverse Path

Reverse Path Setup in AODV

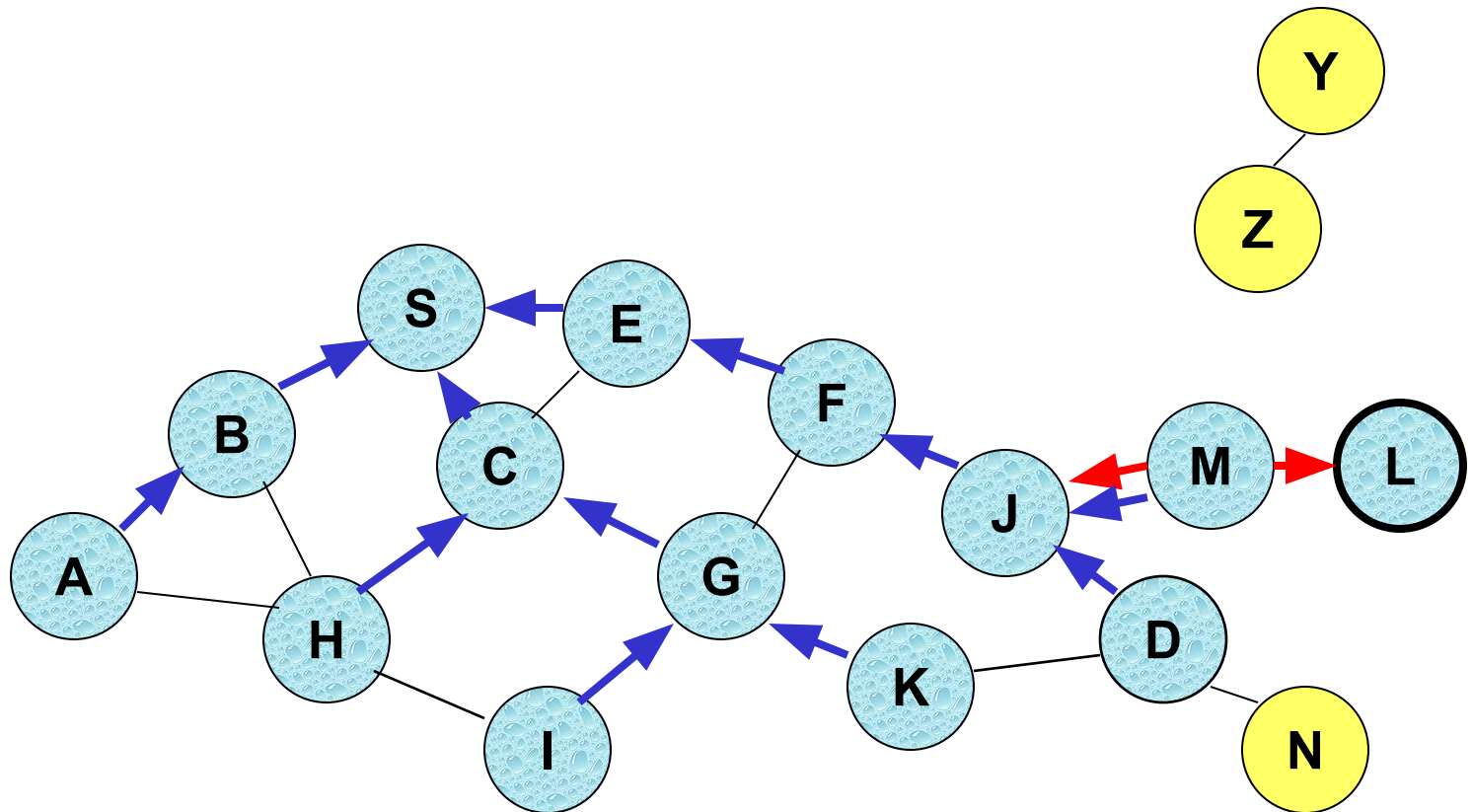


- Node C receives RREQ from G and H, but does not forward it again, because node C has **already forwarded RREQ** once

Reverse Path Setup in AODV

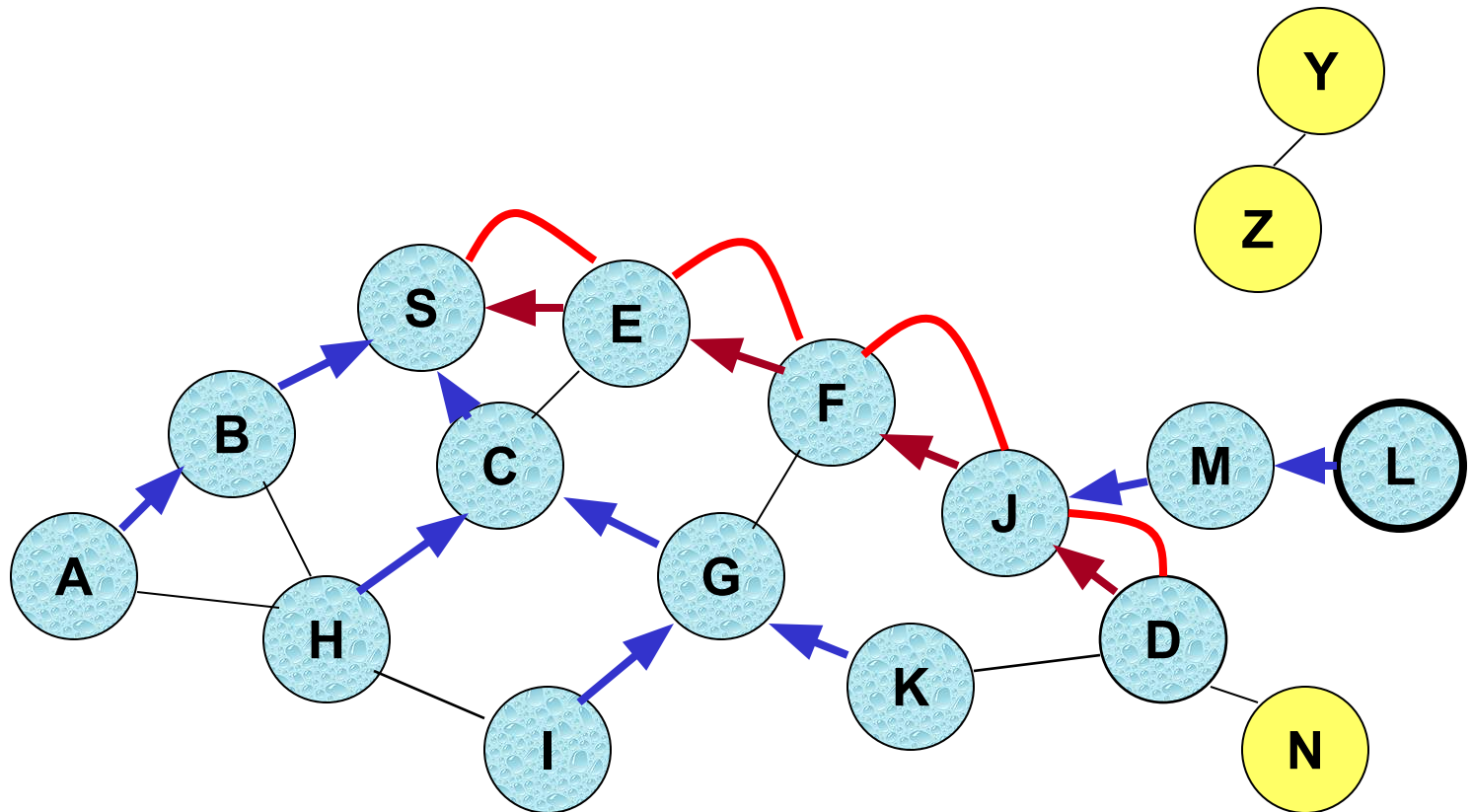


Reverse Path Setup in AODV



- 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

- 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
 - DSR may maintain several routes for a single destination
- 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

Proactive Routing Protocols

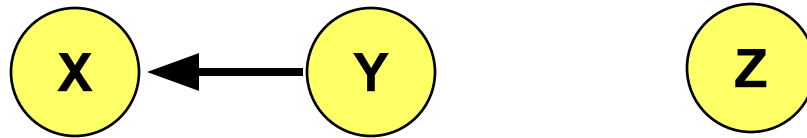
Destination-Sequenced Distance-Vector (DSDV)

[Perkins94Sigcomm]

- Each node maintains a routing table which stores
 - next hop, cost metric towards each destination
 - 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

Destination-Sequenced Distance-Vector (DSDV)

- 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

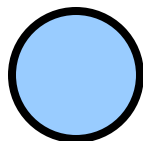
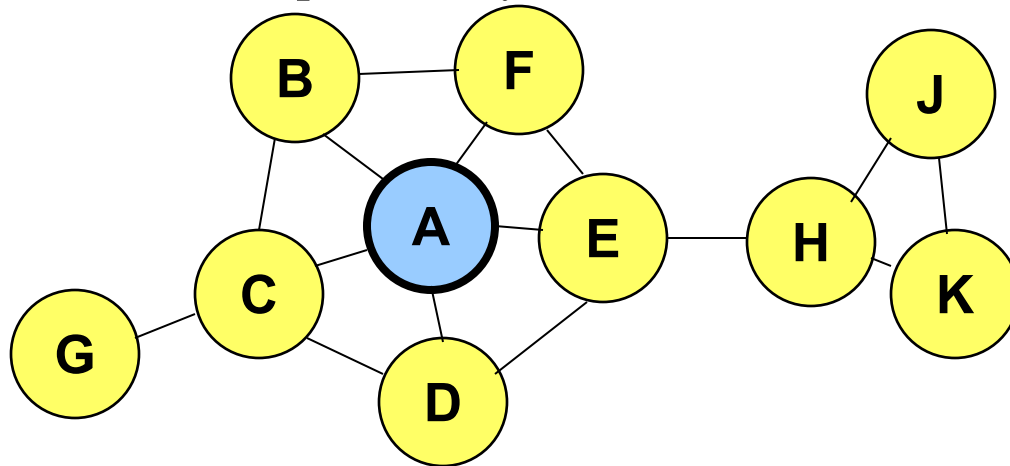


- If $S(X) > S(Y)$, then X ignores the routing information received from Y
- If $S(X) = S(Y)$, and cost of going through Y is smaller than the route known to X, then X sets Y as the next hop to Z
- If $S(X) < S(Y)$, then X sets Y as the next hop to Z, and $S(X)$ is updated to equal $S(Y)$

Optimized Link State Routing (OLSR)

[Jacquet00ietf]

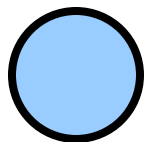
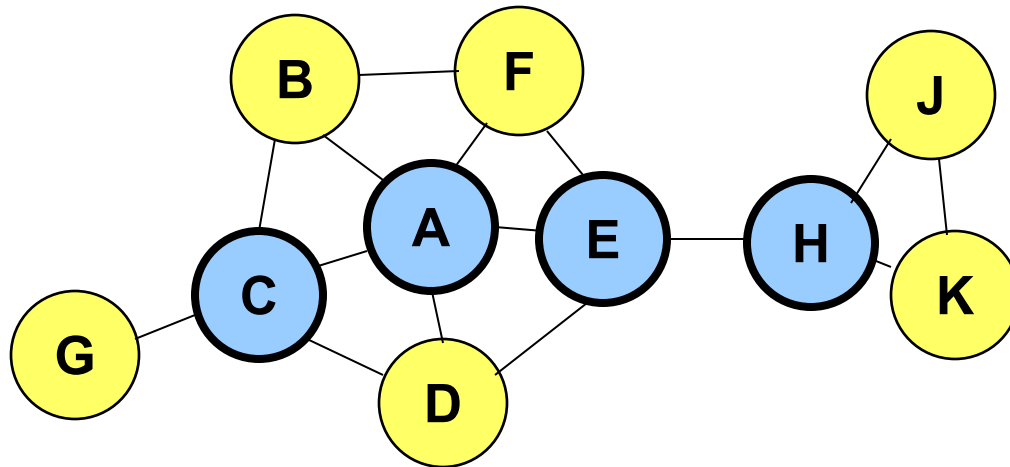
- Nodes C and E are multipoint relays of node A
 - Multipoint relays of A are its neighbors such that each two-hop neighbor of A is a one-hop neighbor of one multipoint relay of A
 - Nodes exchange neighbor lists to know their 2-hop neighbors and choose the multipoint relays



Node that has broadcast state information from A

Optimized Link State Routing (OLSR)

- Nodes C and E forward information received from A
- Nodes E and K are multipoint relays for node H
- Node K forwards information received from H



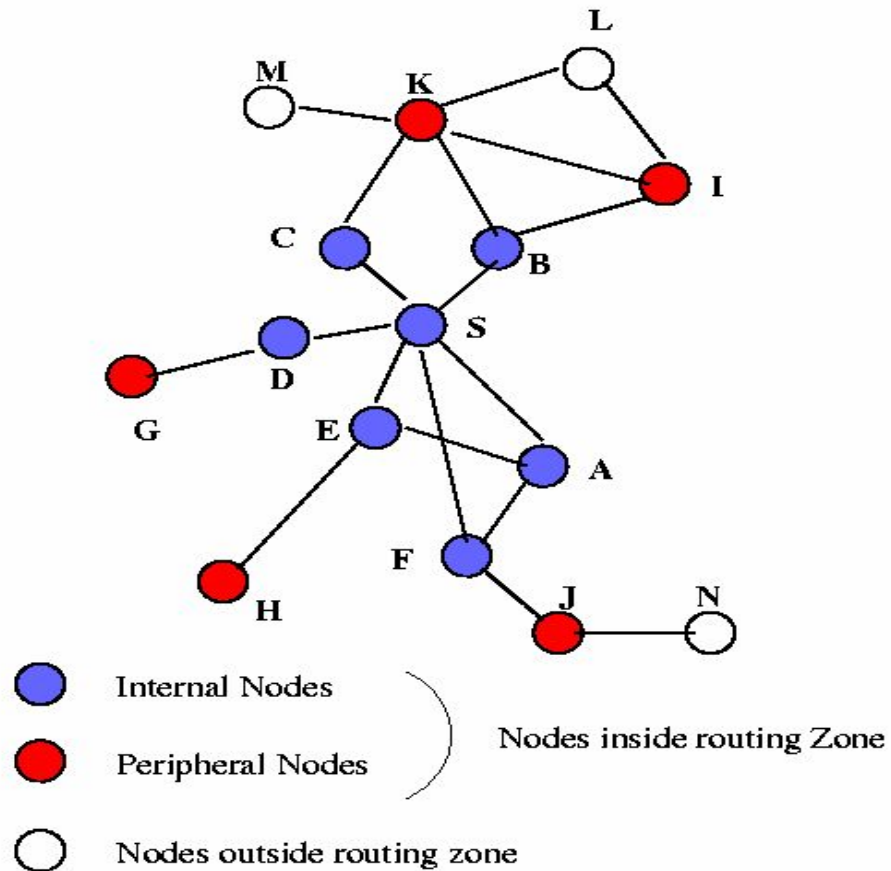
Node that has broadcast state information from A

Hybrid Routing Protocols

Zone Routing Protocol (ZRP) [Haas98]

- ZRP combines proactive and reactive approaches
- All nodes within hop distance at most d from a node X are said to be in the **routing zone** of node X
- All nodes at hop distance exactly d are said to be **peripheral** nodes of node X 's routing zone
- **Intra-zone routing**: Proactively maintain routes to all nodes within the source node's own zone.
- **Inter-zone routing**: Use an on-demand protocol (similar to DSR or AODV) to determine routes to outside zone.

Zone Routing Protocol (ZRP)



Radius of routing zone = 2