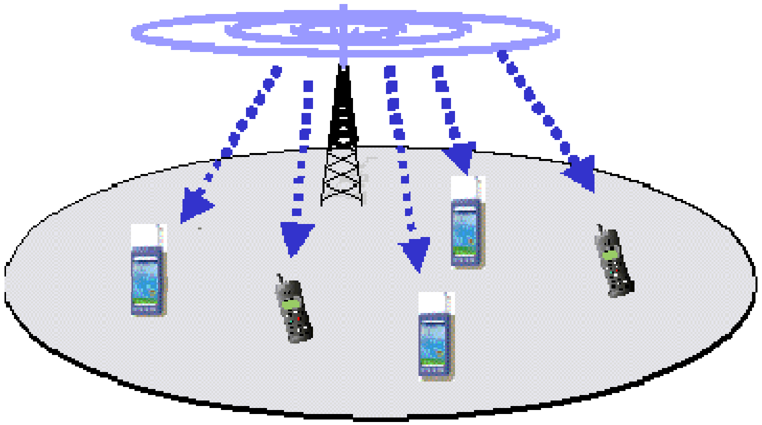
Lecture: Wireless Multihop Relay Networks

Hung-Yu Wei
National Taiwan University

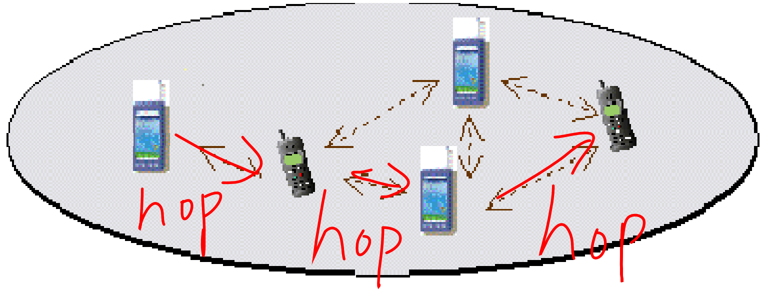
Two Wireless Network Paradigms

- Centralized network architecture
  - Single-hop connection
- Example:
  - cellular networks
  - 802.11 AP



- Distributed (peer-to-peer) network architecture
  - Multi-hop connection
- Example
  - Ad hoc network
  - Mesh network

relay

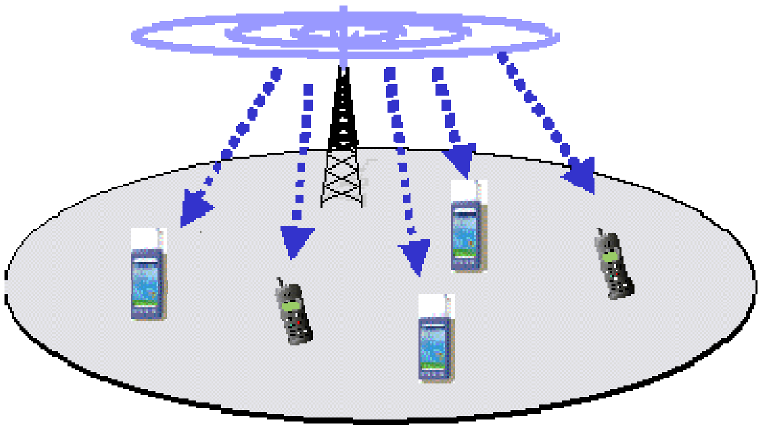


3 hop

Could become a hybrid one…

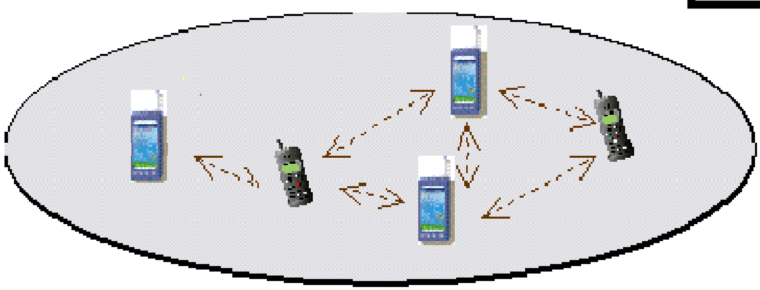
802.11s

802.11j Wimax relay

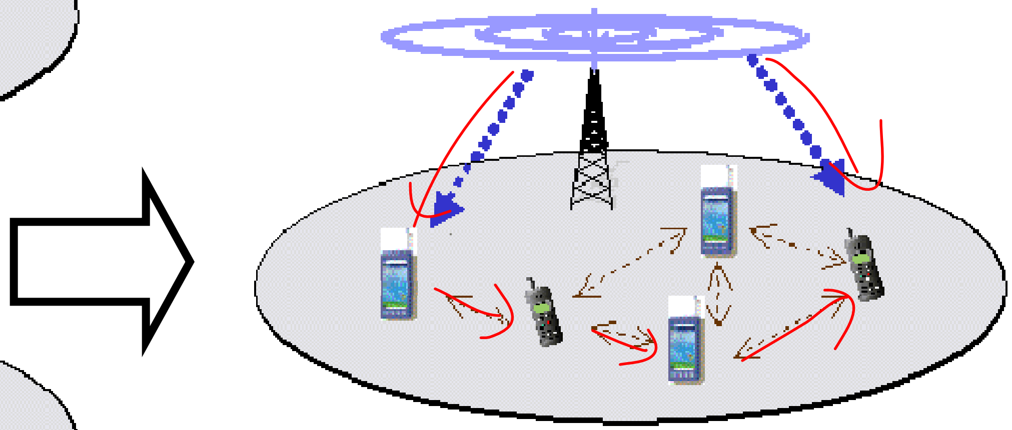


Cellular Network

+



Mobile *Ad Hoc* Network



Hybrid Wireless Relay Network

LTE relay

D2D

LTE device-to-device communication

Overview: Wireless Multihop Relay Networks

- Mobile Ad Hoc Networks (MANET)
  - Mobile
  - Ad hoc (peer-to-peer)
- Wireless Mesh Networks (WMN) 軍事用途
  - Infrastructure multihop relay
- Wireless Sensor Networks (WSN)
  - Sensor

  - Collecting data + data processing
  - Power-efficient
- Vehicular Ad Hoc Networks (VANET)
  - Vehicle-to-vehicle

Classification : multihop-relay networks

- Which nodes do relay for others?
- Mobility?
  - Static with wireless connections
  - Low mobility
  - High mobility (e.g. vehicular)
- Battery driven?
  - Does power efficient design a critical issue?
    - Example(i): sensor node with limited battery power
    - Example(ii): wireless mesh router (powered by electric wires)
- Architecture
  - Pure peer-to-peer
  - Limited infrastructure
  - Some centralized control

References

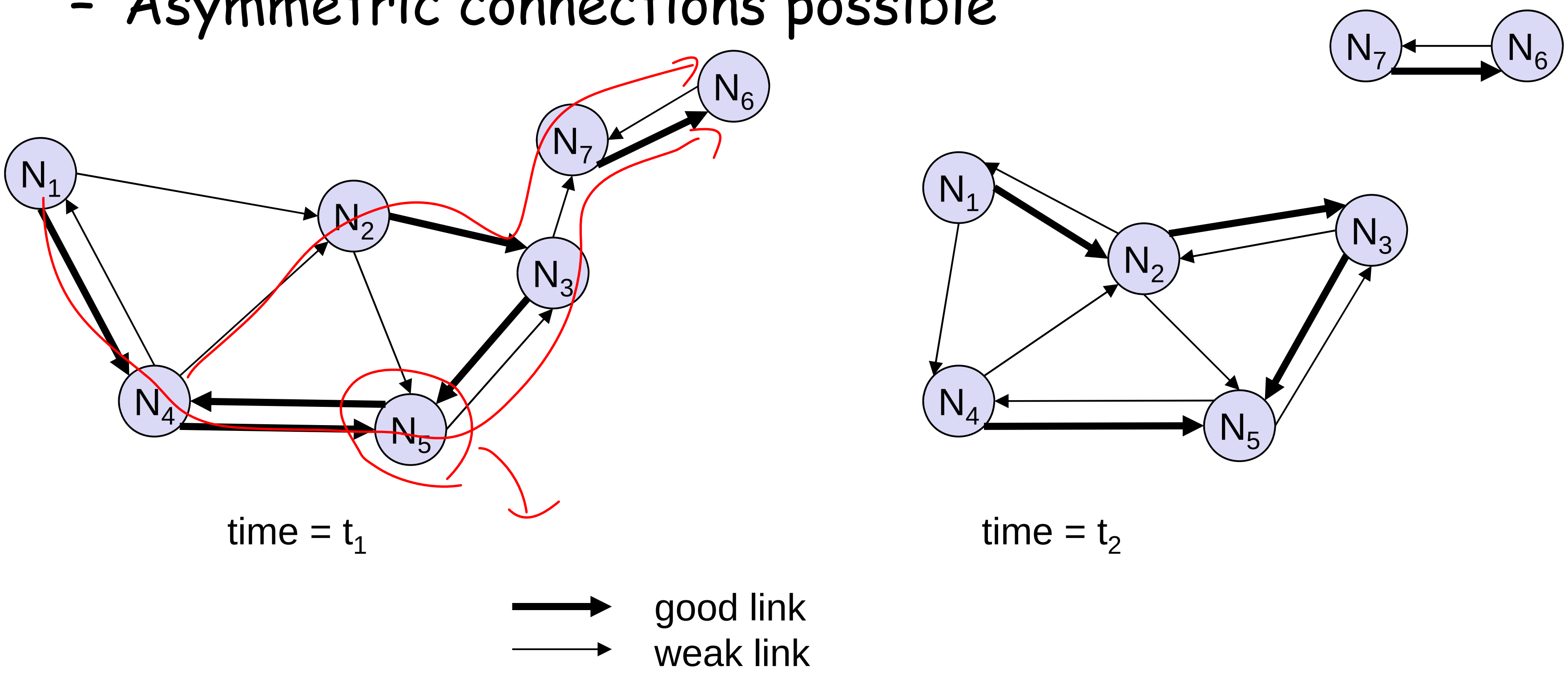
- http://www.crhc.uiuc.edu/wireless/tutorials.html
- S. M. Faccin, C. Wijting, J. Kenckt, and A. Damle, "Mesh WLAN networks: concept and system design," IEEE Wireless Communications, vol. 13, pp. 10-17, 2006.
- M. J. Lee, Z. Jianliang, K. Young-Bae, and D. M. Shrestha, "Emerging standards for wireless mesh technology," IEEE Wireless Communications, vol. 13, pp. 56-63, 2006.

Mobile Ad Hoc Networks: Routing

IP

Problem No. 1: Routing

- Highly dynamic network topology
  - Device mobility plus varying channel quality
  - Separation and merging of networks possible
  - Asymmetric connections possible

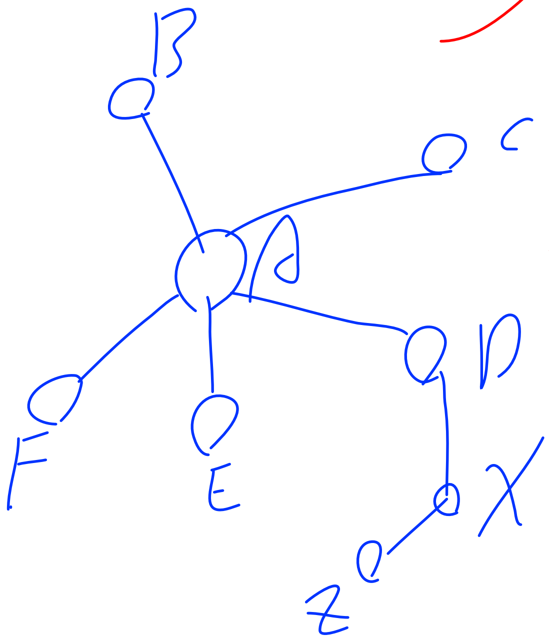


Traditional routing algorithms

- **Distance Vector** all info → neighbor
  - periodic exchange of messages with all physical neighbors that contain information about who can be reached at what distance
  - selection of the shortest path if several paths available
- **Link State** neighbor info → all
  - periodic notification of all routers about the current state of all physical links
  - router get a complete picture of the network

broadcast 給全部

destination	cost	next hop
X	5	Y
	4	Y



send to	link state	routing table
neighbor	X	link state routing
all	distance vector routing	inefficient

Problems of traditional routing algorithms

- Dynamic of the topology
  - frequent changes of connections, connection quality, participants
- Limited performance of mobile systems
  - periodic updates of routing tables need energy without contributing to the transmission of user data, sleep modes difficult to realize
  - limited bandwidth of the system is reduced even more due to the exchange of routing information
  - links can be asymmetric, i.e., they can have a direction dependent transmission quality

Mobile Ad Hoc Networks

- MANET
  - Packet radio networks (in 1970s)
  - Hot research issues in 1990-now
    - Significant interest from DARPA
    - Funding$$$
- Routing
  - A major issue to be solved in mobile ad hoc networks
  - Mobility
    - Old routes no longer valid
    - Discover new routes

Routing in MANET

- Holy Grail:
  - one "best" MANET routing protocol
- Reality:
  - No one-size fit all routing solution
  - Performance varies in different scenarios

Classifications of Routing Protocols

- On-demand routing protocol 被动
  - Also know as reactive routing protocol
  - Discover route "on-demand" (when needed)
  - Example
    - AODV (Ad hoc On-Demand Distance Vector Routing)
    - DSR (Dynamic Source Routing Protocol)
- 主动 Proactive routing protocol
  - Actively maintain valid routes
  - Example
    - OLSR (Optimized Link State Routing)
    - DSDV (Destination-Sequenced Distance-Vector)
- Hybrid routing protocol
  - Mixture of proactive and on-demand protocol
  - Example
    - ZRP (The Zone Routing Protocol)

proactive: distance vector / link state

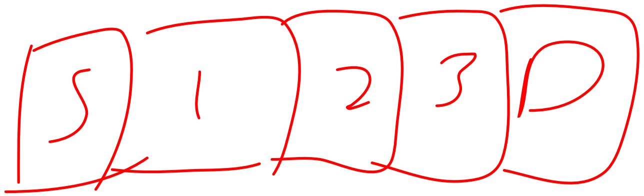
DSR

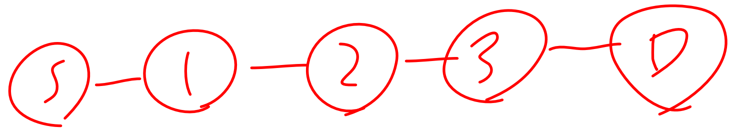
Flooding of Control Packets

- Many protocols perform (potentially *limited*) flooding of control packets, instead of data packets
- The control packets are used to discover routes
- Discovered routes are subsequently used to send data packet(s)
- Overhead of control packet flooding is amortized over data packets transmitted between consecutive control packet floods

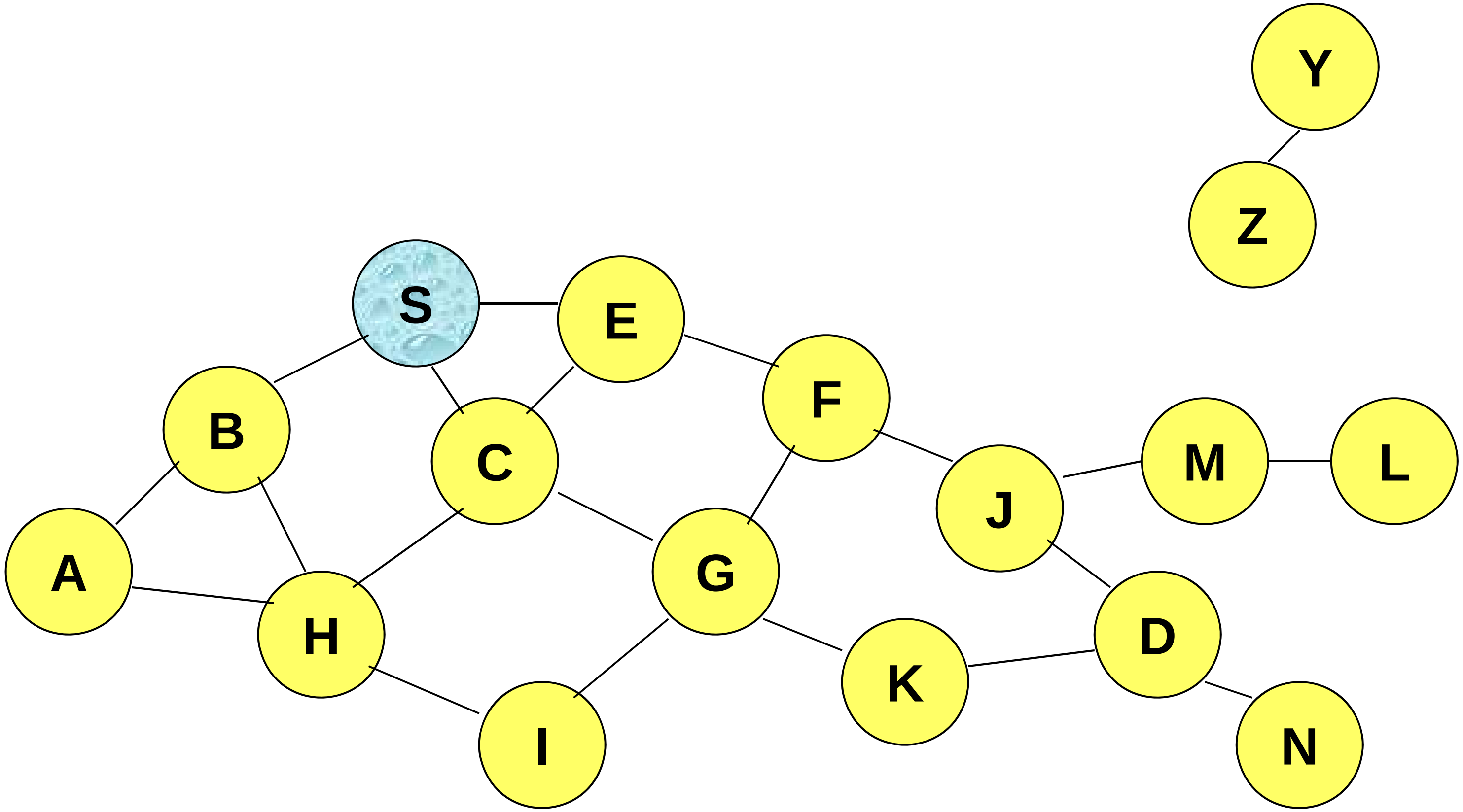
Dynamic Source Routing (DSR) [Johnson96]

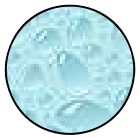
- When node S wants to send a packet to node D, but does not know a route to D, node S initiates a route discovery
- Source node S floods Route Request (RREQ)
- Each node appends own identifier when forwarding RREQ





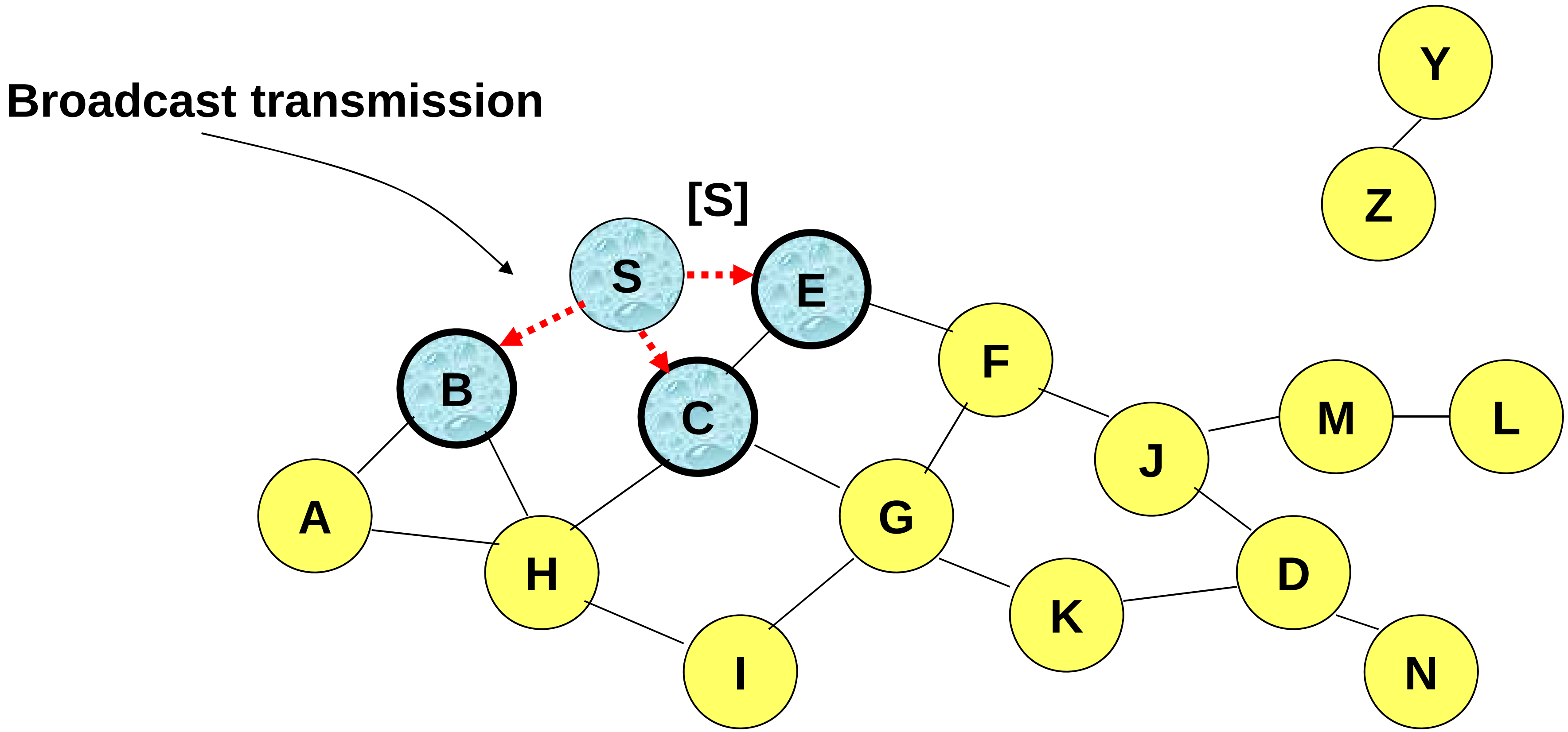
Route Discovery in DSR





Represents a node that has received RREQ for D from S

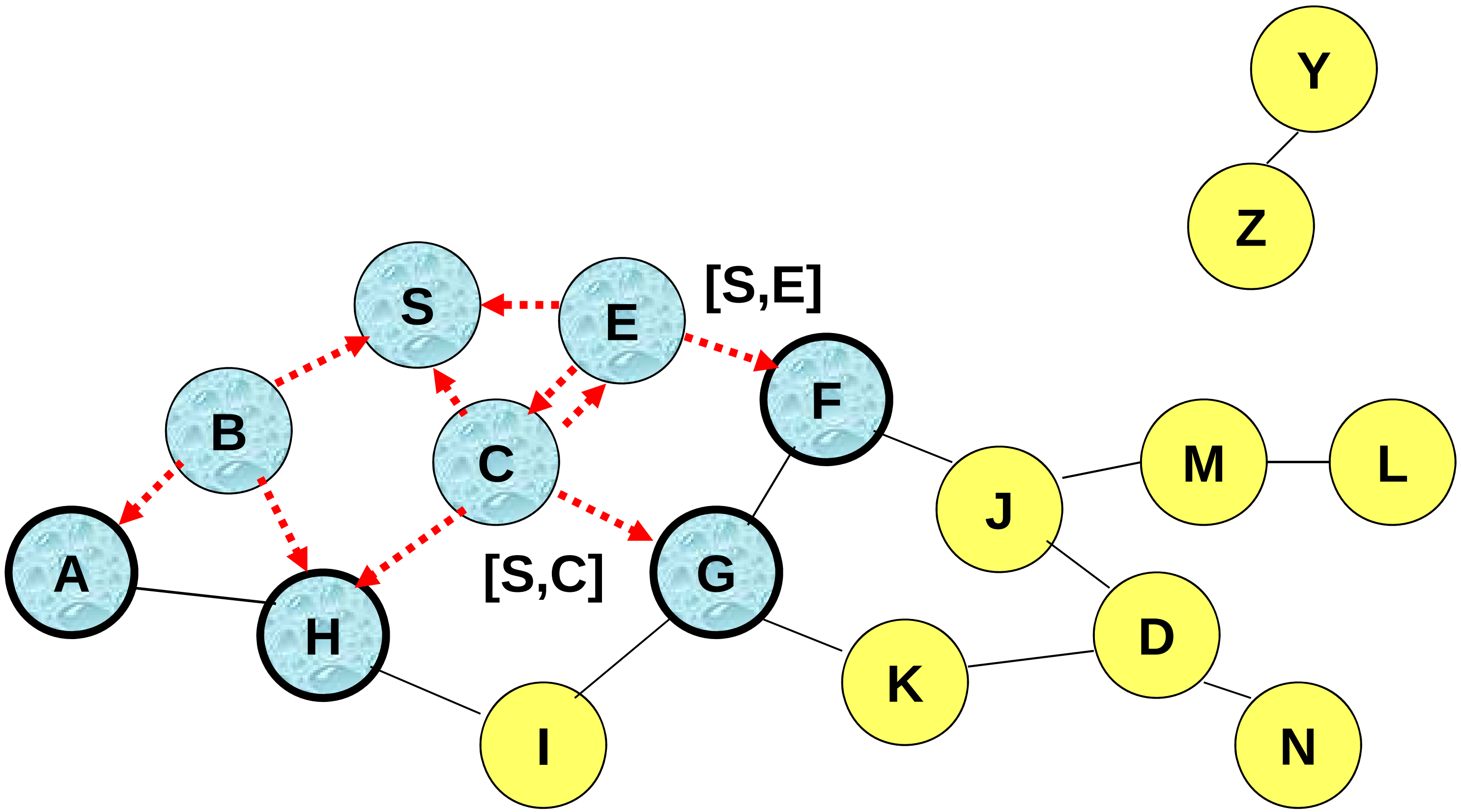
Route Discovery in DSR



Represents transmission of RREQ

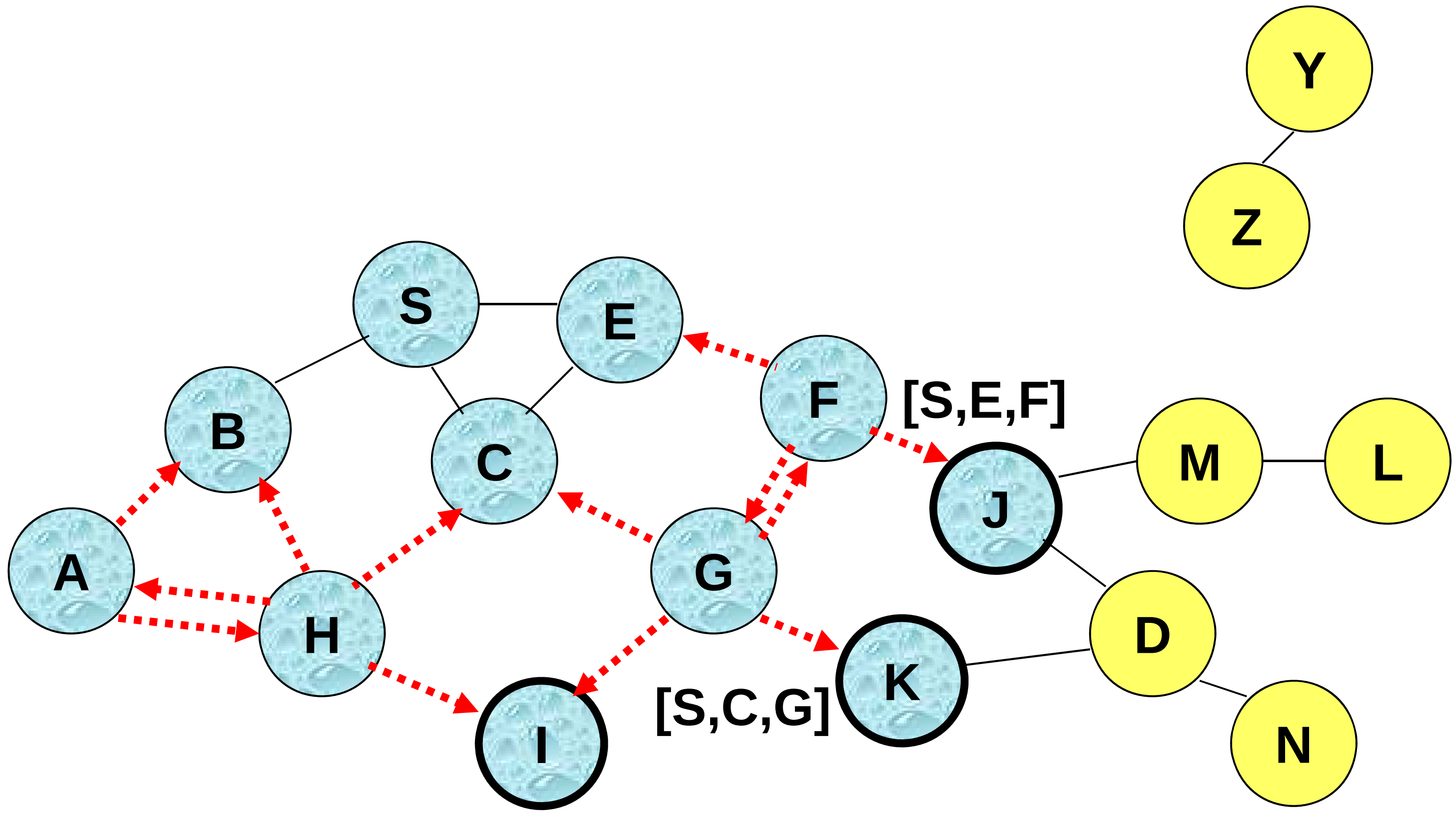
[X,Y] Represents list of identifiers appended to RREQ

Route Discovery in DSR



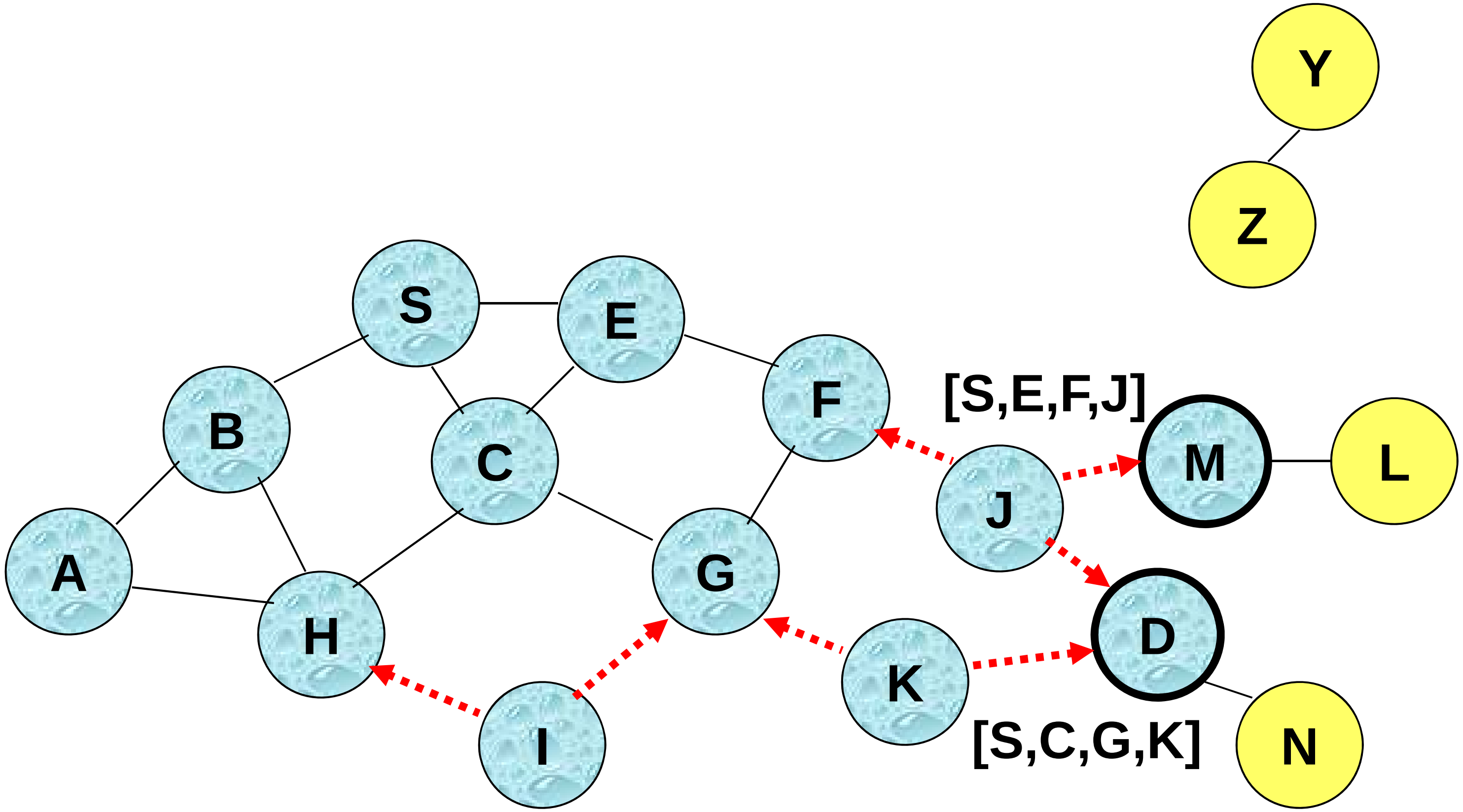
- **Node H receives packet RREQ from two neighbors: potential for collision**

Route Discovery in DSR



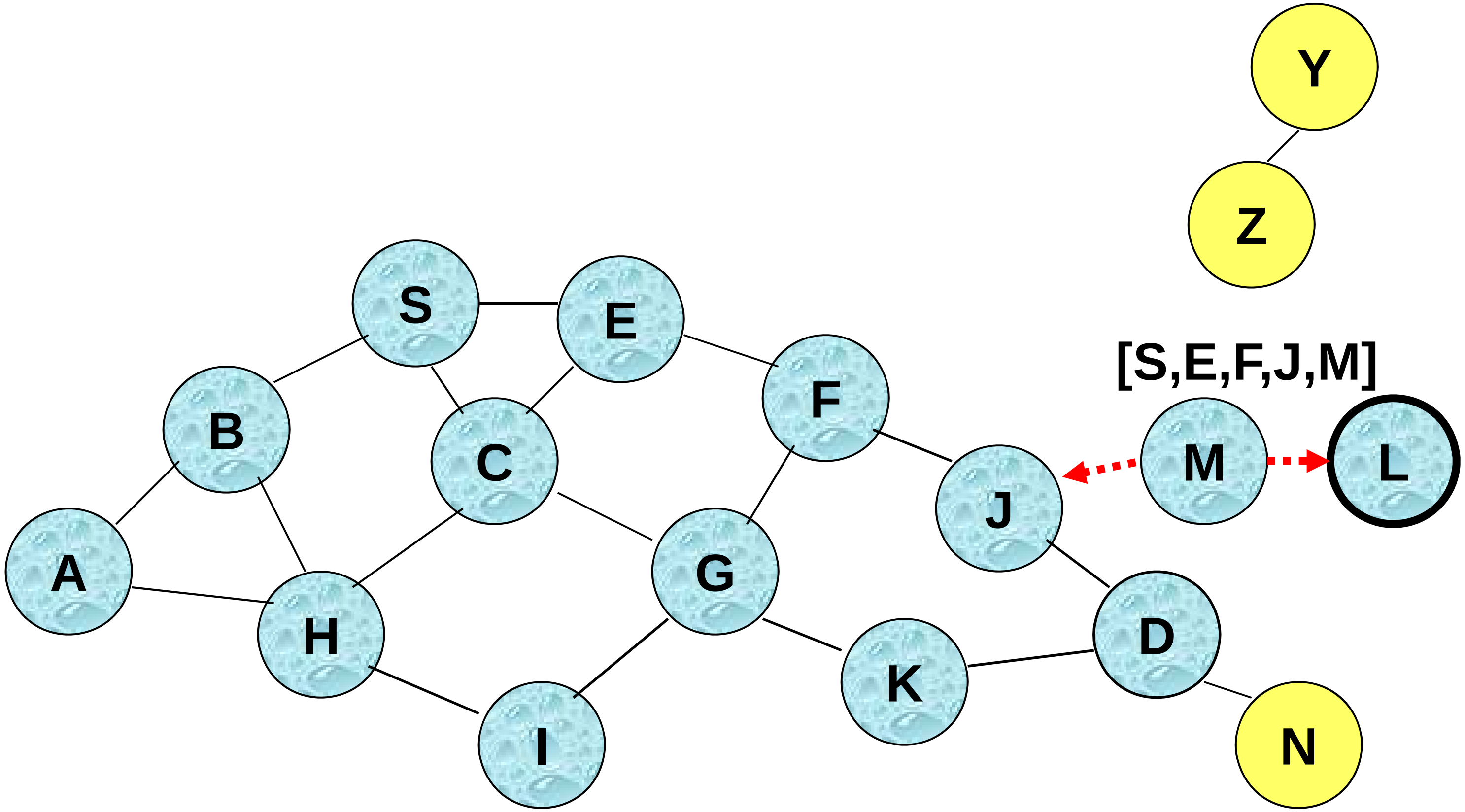
• **Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once**

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**

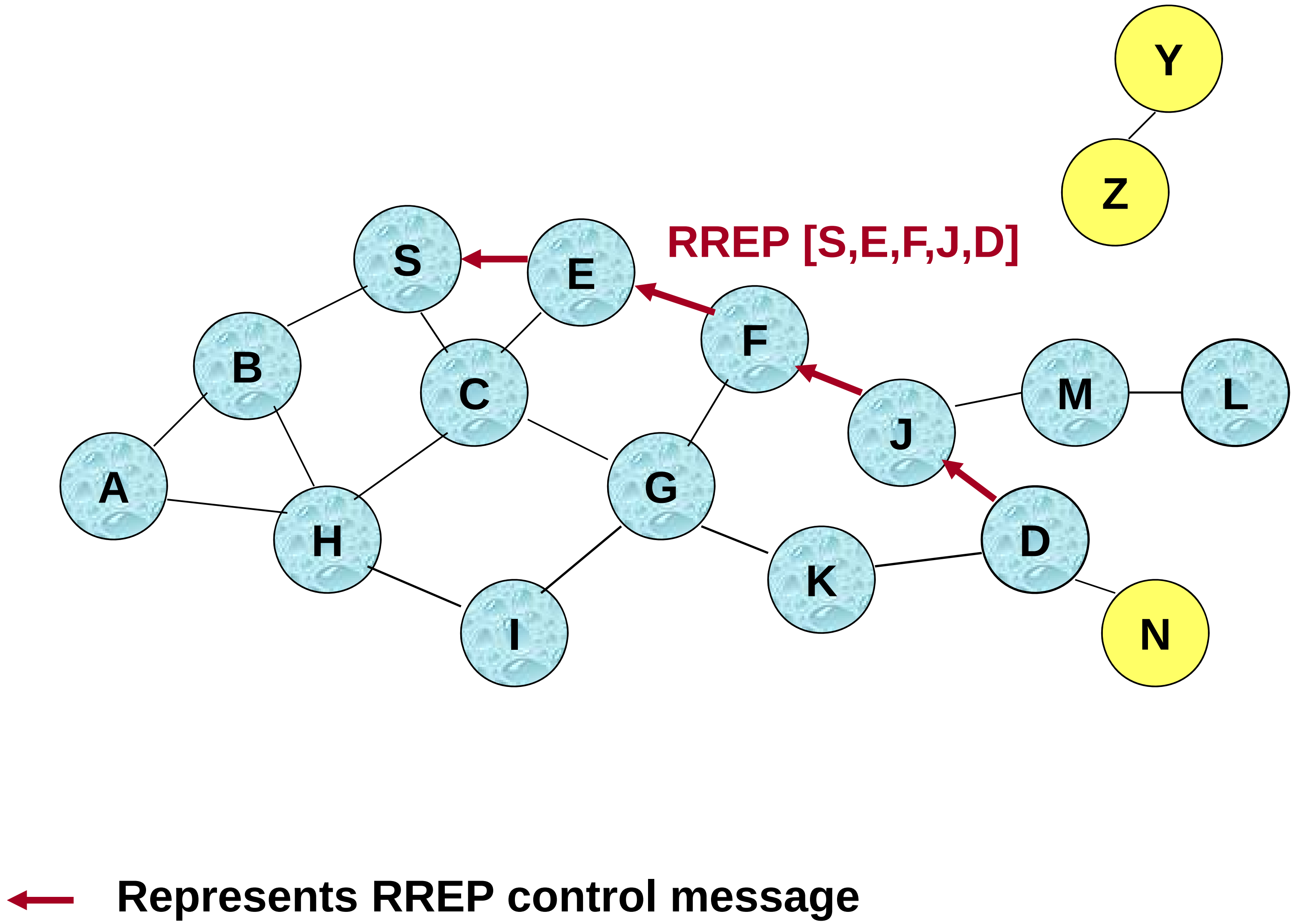
Route Discovery in DSR

- 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



Route Reply in DSR

- Route Reply can be sent by reversing the route in Route Request (RREQ) only if links are guaranteed to be bi-directional
  - To ensure this, RREQ should be forwarded only if it received on a link that is known to be bi-directional
- If unidirectional (asymmetric) links are allowed, then RREP may need a route discovery for S from node D
  - Unless node D already knows a route to node S
  - If a route discovery is initiated by D for a route to S, then the Route Reply is piggybacked on the Route Request from D.
- If IEEE 802.11 MAC is used to send data, then links have to be bi-directional (since Ack is used)

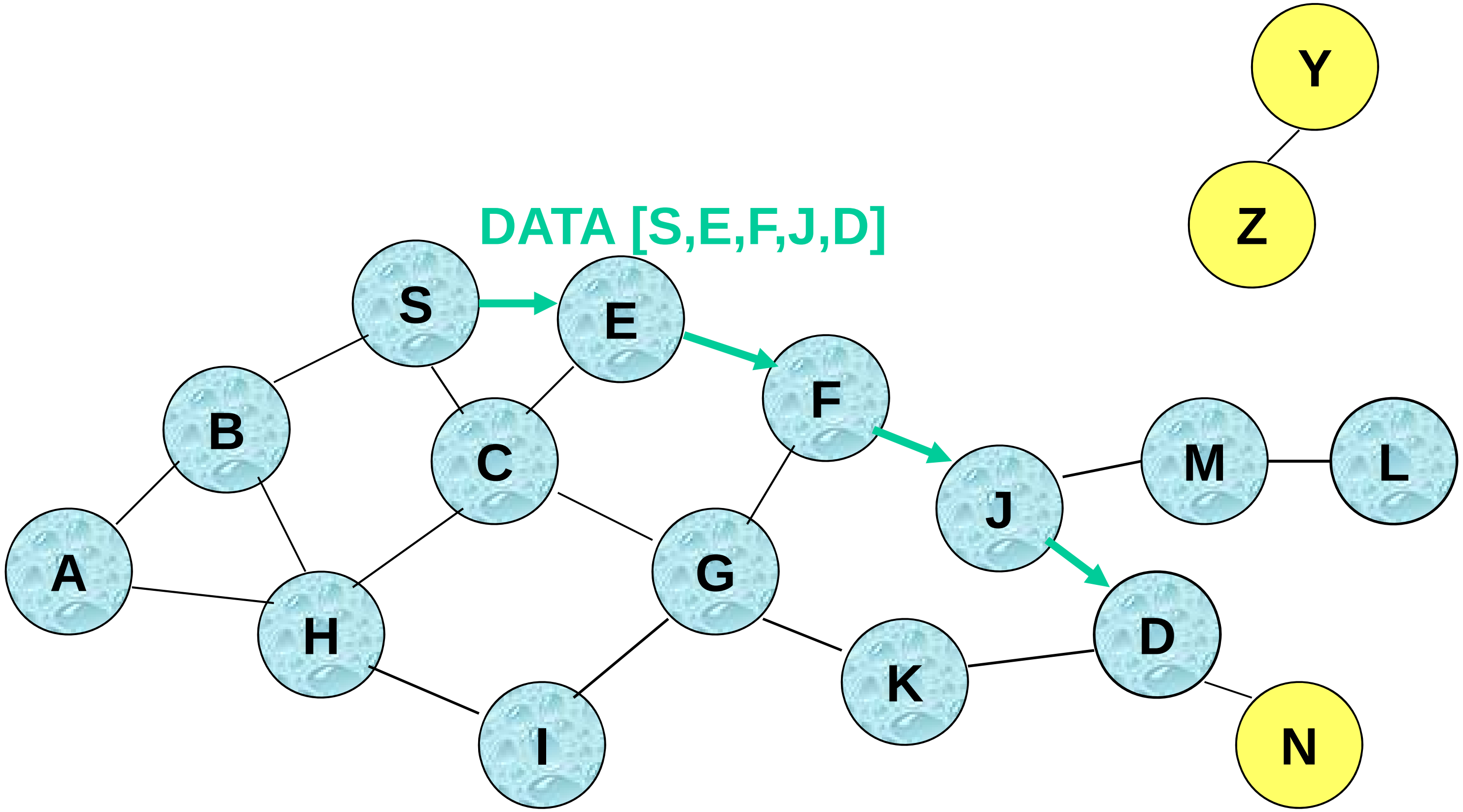
Dynamic Source Routing (DSR)

- 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

When to Perform a Route Discovery

- When node S wants to send data to node D, but does not know a valid route node D

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 packets

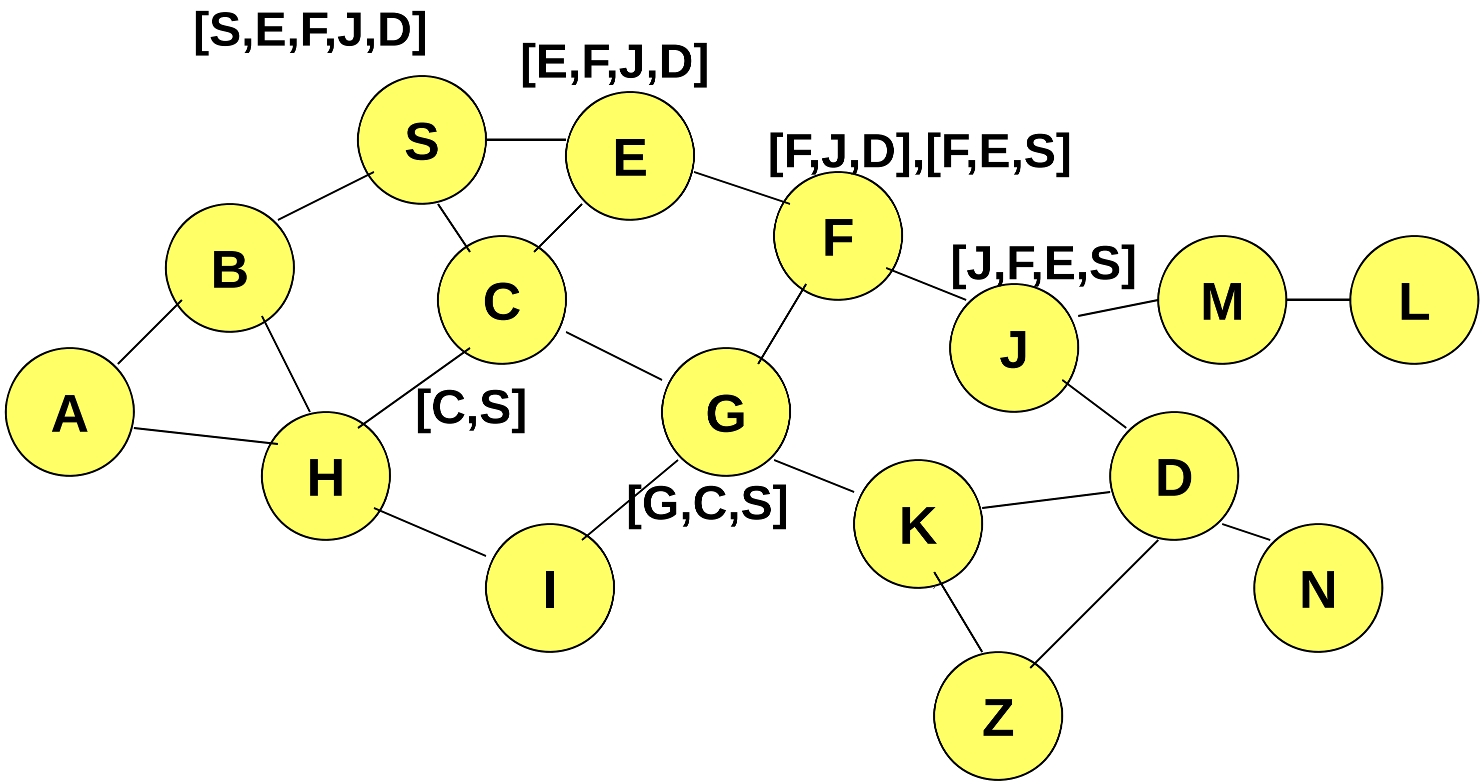
Use of Route Caching

- When node S learns that a route to node D is broken, it uses another route from its local cache, if such a route to D exists in its cache. Otherwise, node S initiates route discovery by sending a route request

- Node X on receiving a Route Request for some node D can send a Route Reply if node X knows a route to node D

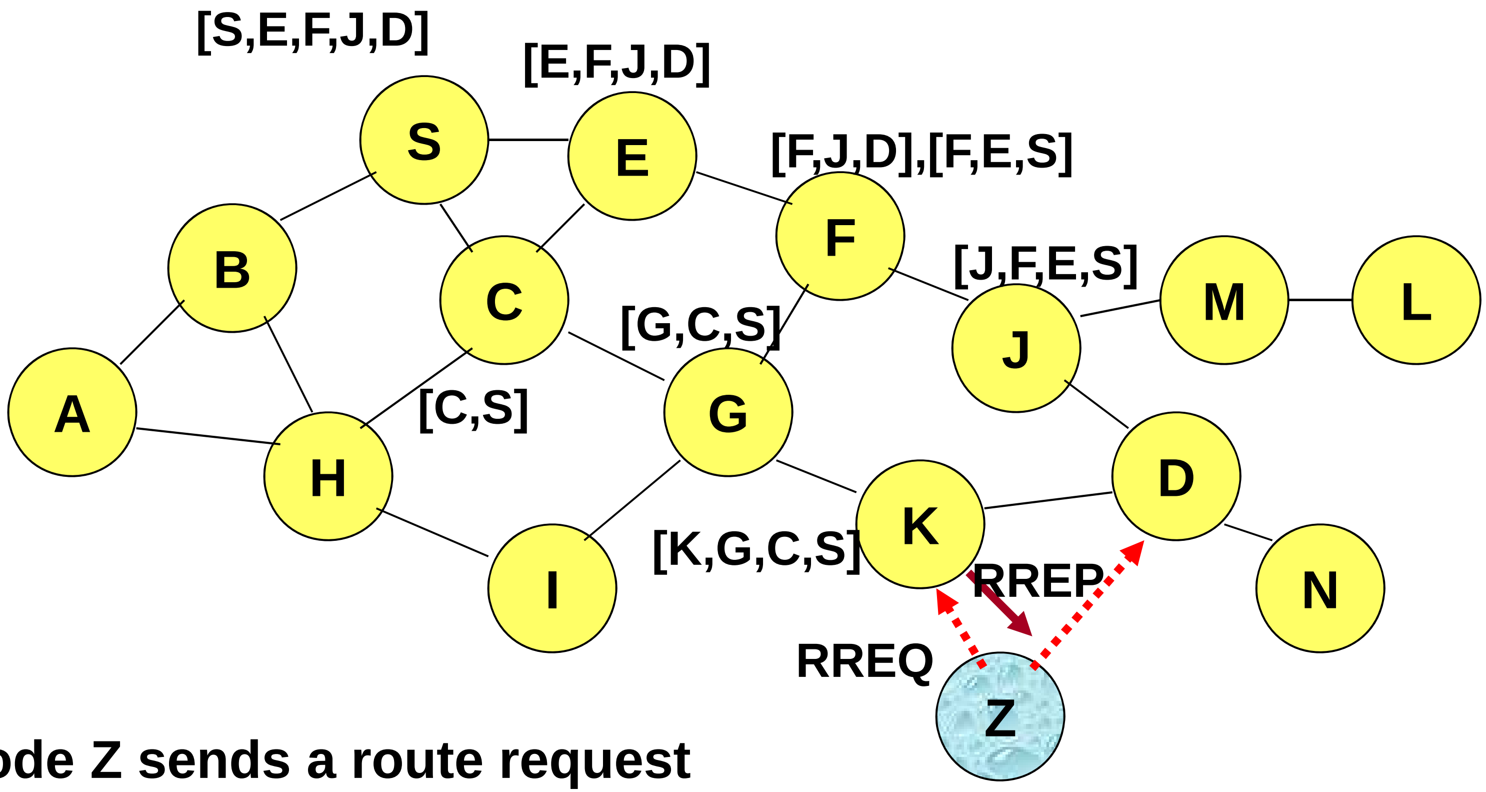
- Use of route cache
  - can speed up route discovery
  - can reduce propagation of route requests

Use of Route Caching



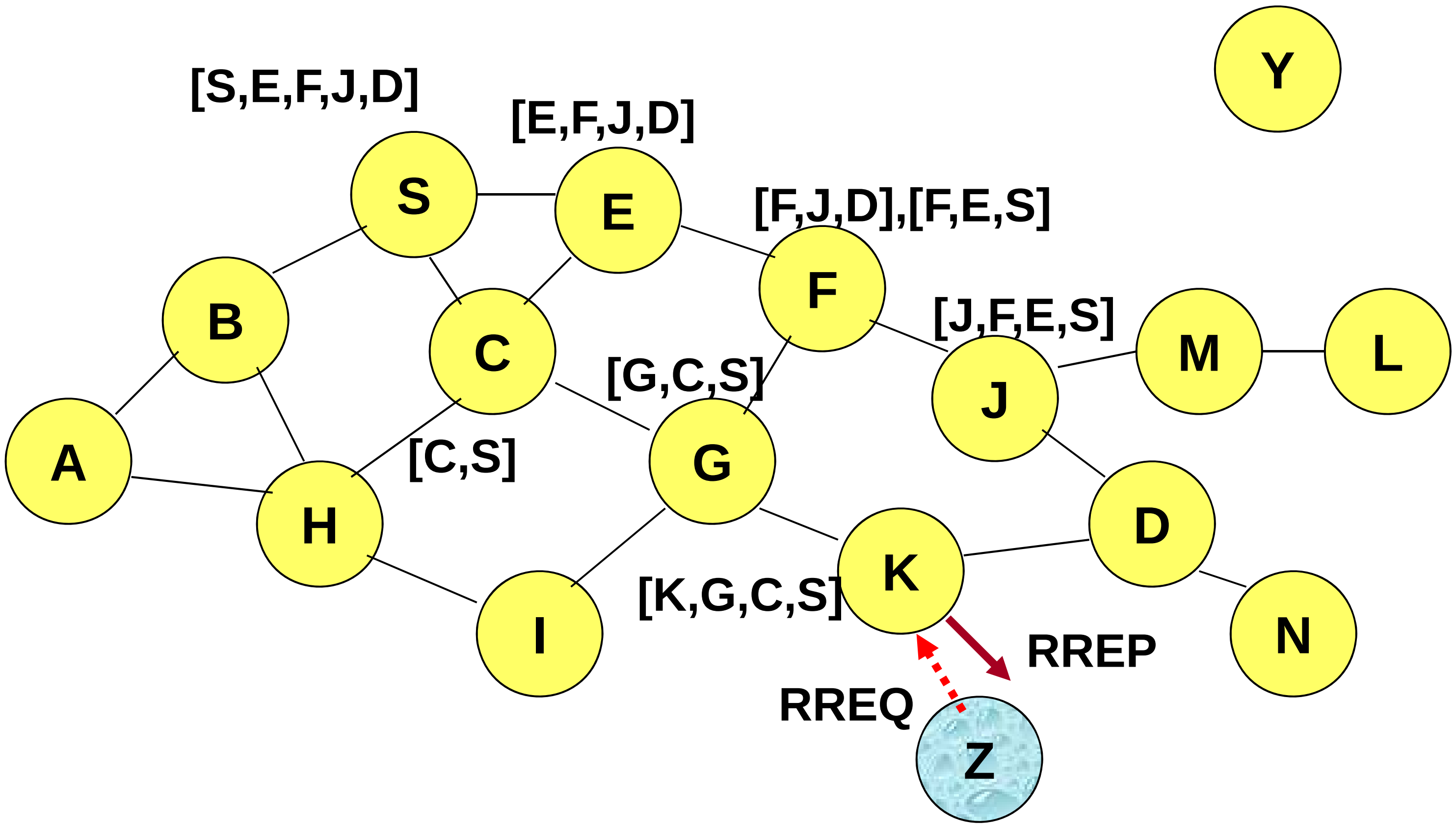
[P,Q,R]   Represents cached route at a node
(DSR maintains the cached routes in a tree format)

Use of Route Caching: Can Speed up Route Discovery



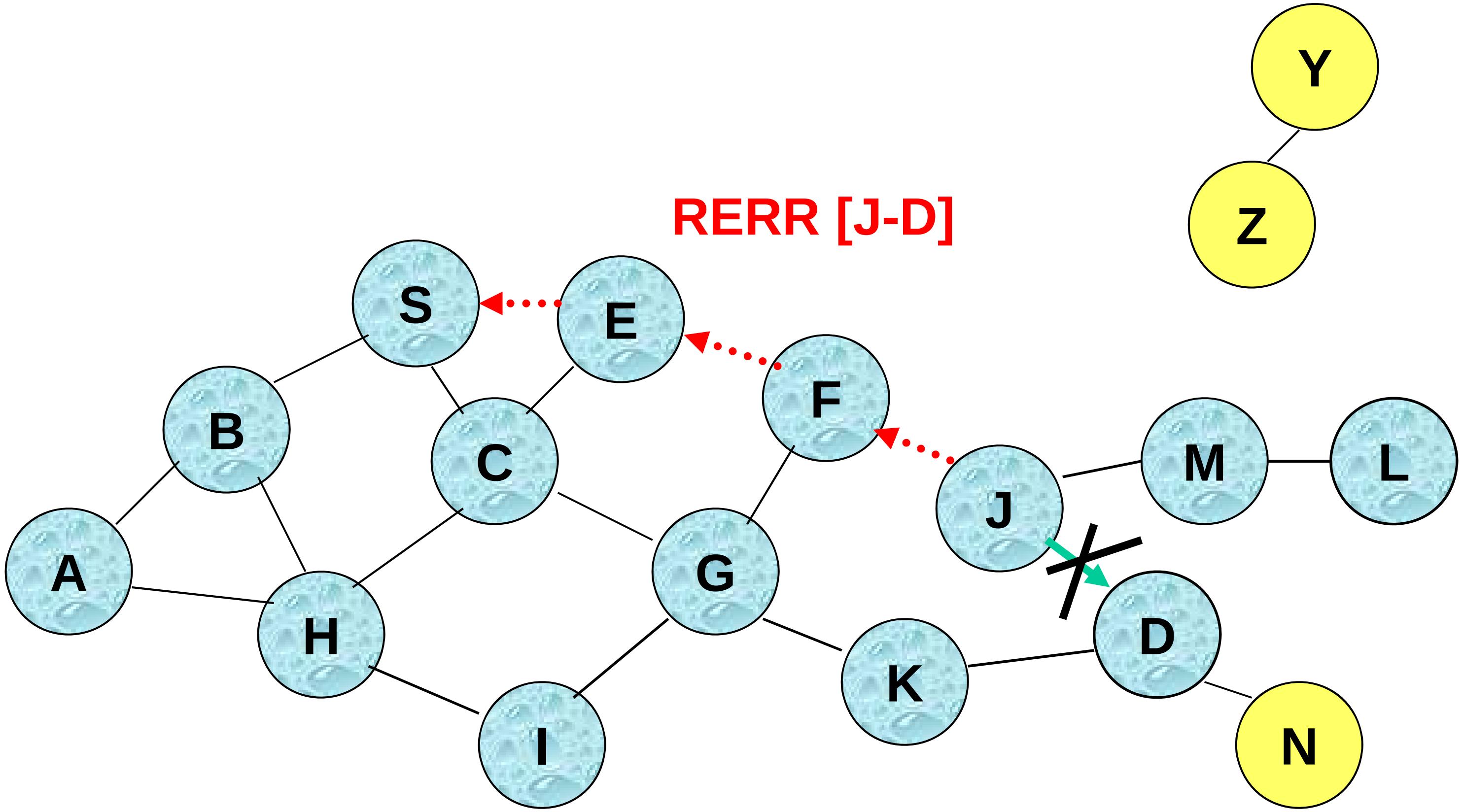
When node Z sends a route request for node C, node K sends back a route reply [Z,K,G,C] to node Z using a locally cached route

Use of Route Caching: Can Reduce Propagation of Route Requests



Assume that there is no link between D and Z.
Route Reply (RREP) from node K limits flooding of RREQ.
In general, the reduction may be less dramatic.

Route Error (RERR)



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

Route Caching: Beware!

- Stale caches can adversely affect performance

- With passage of time and host mobility, cached routes may become invalid

- A sender host may try several stale routes (obtained from local cache, or replied from cache by other nodes), before finding a good route

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
- 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

AODV

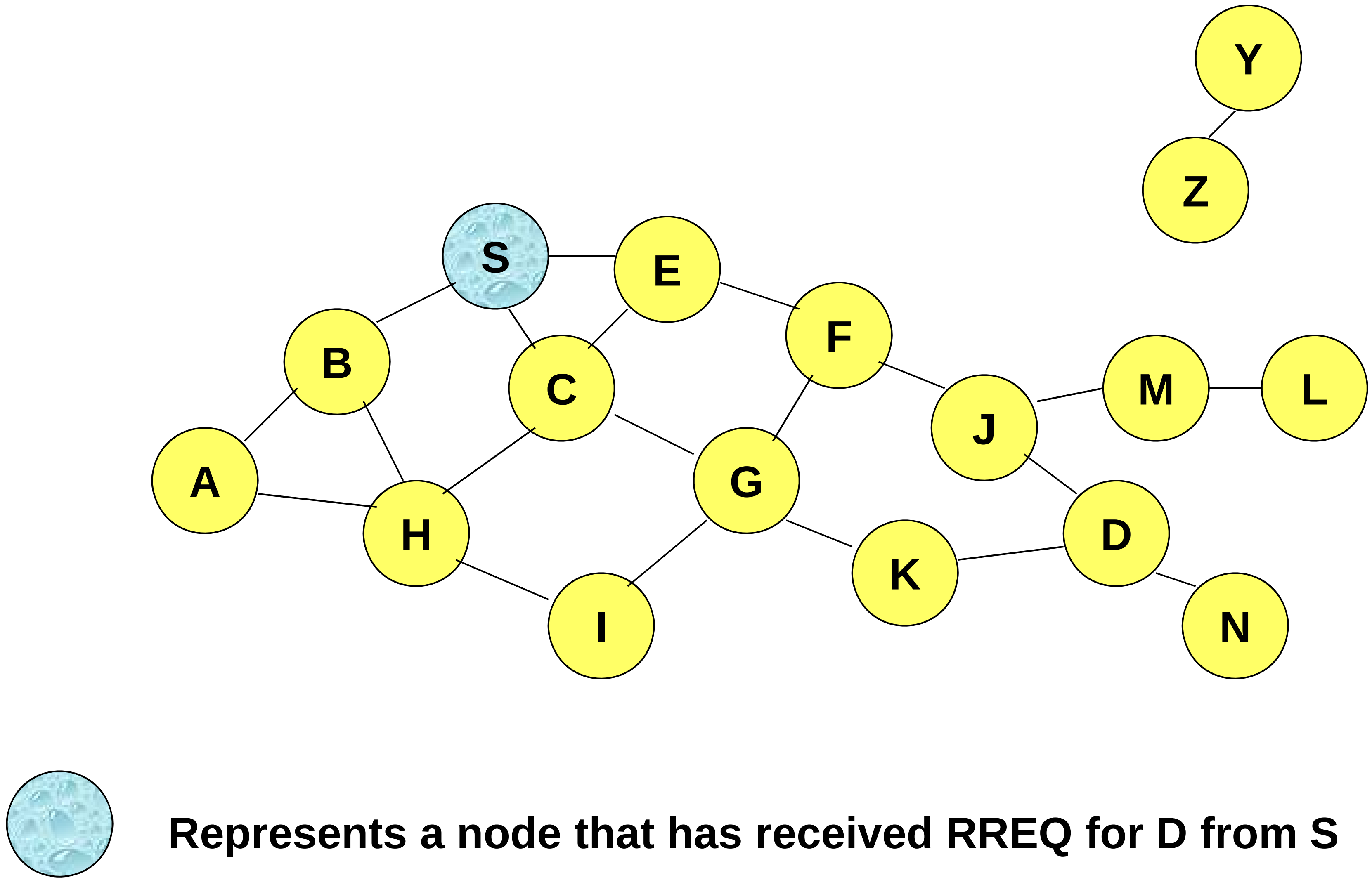
Ad Hoc On-Demand Distance Vector Routing (AODV) [Perkins99Wmcsa]

- 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

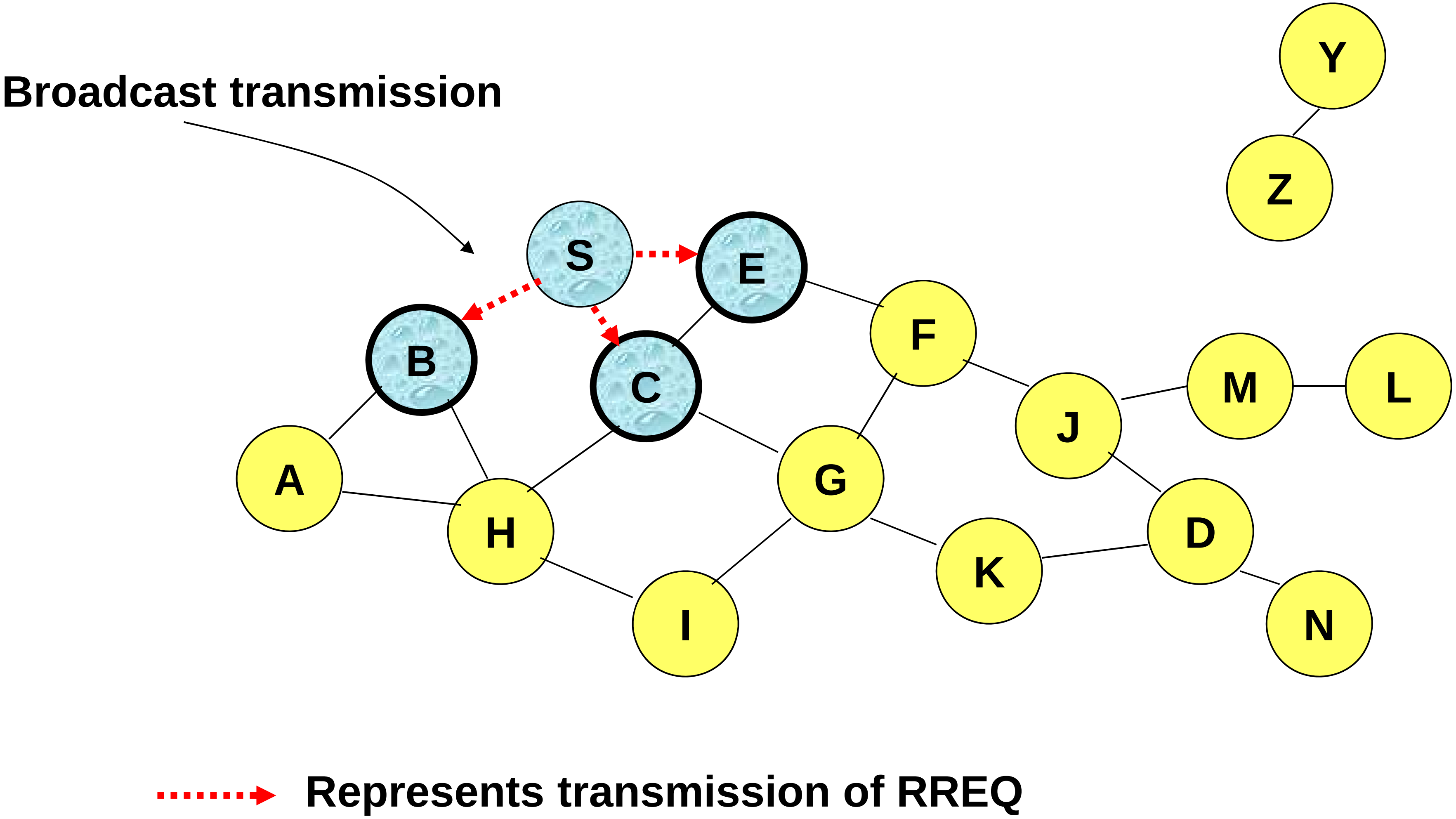
- Route Requests (RREQ) are forwarded in a manner similar to DSR
- When a node re-broadcasts a Route Request, it sets up a reverse path pointing towards the source
  - AODV assumes symmetric (bi-directional) links
- When the intended destination receives a Route Request, it replies by sending a Route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

Route Requests in AODV

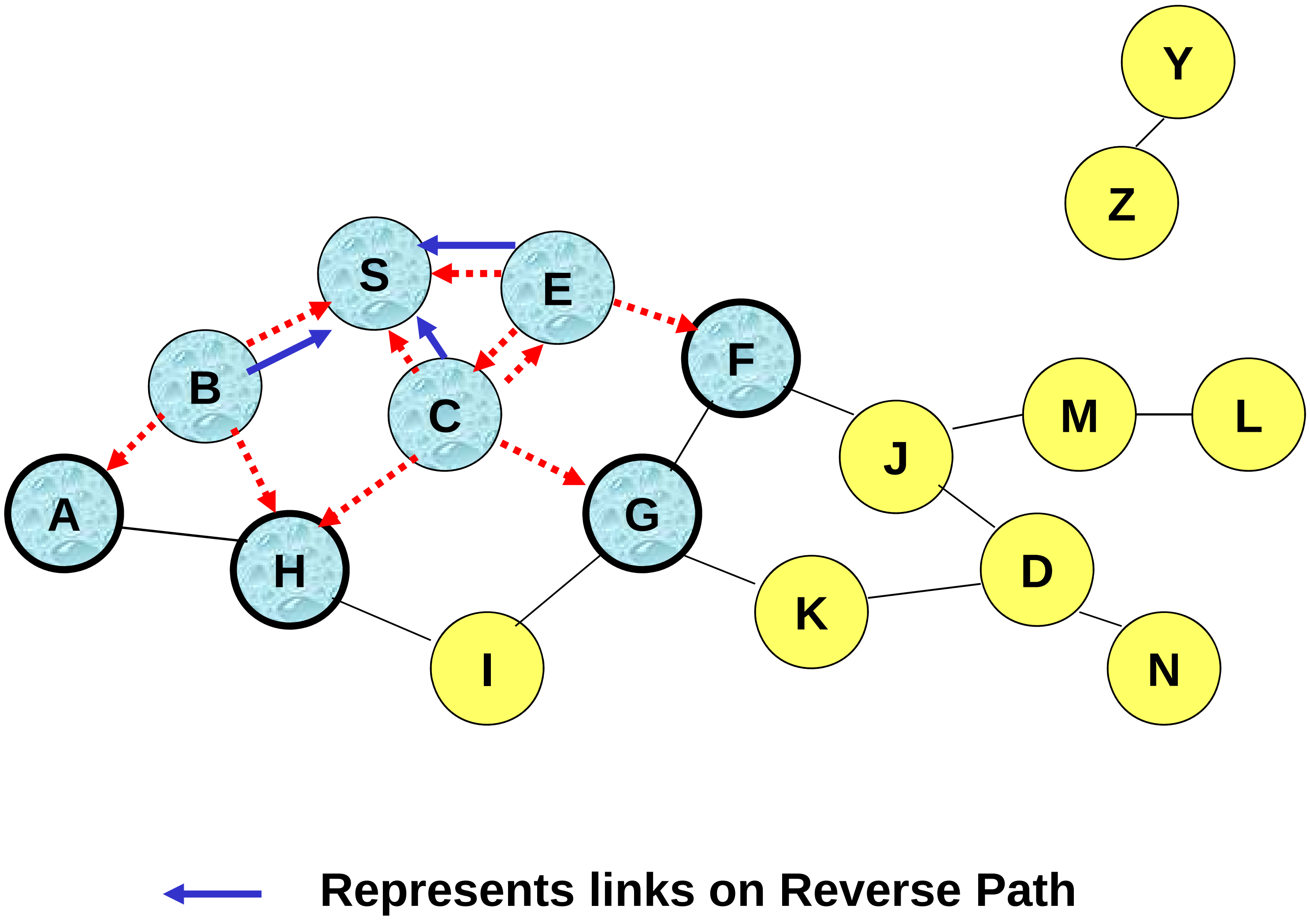


Represents a node that has received RREQ for D from S

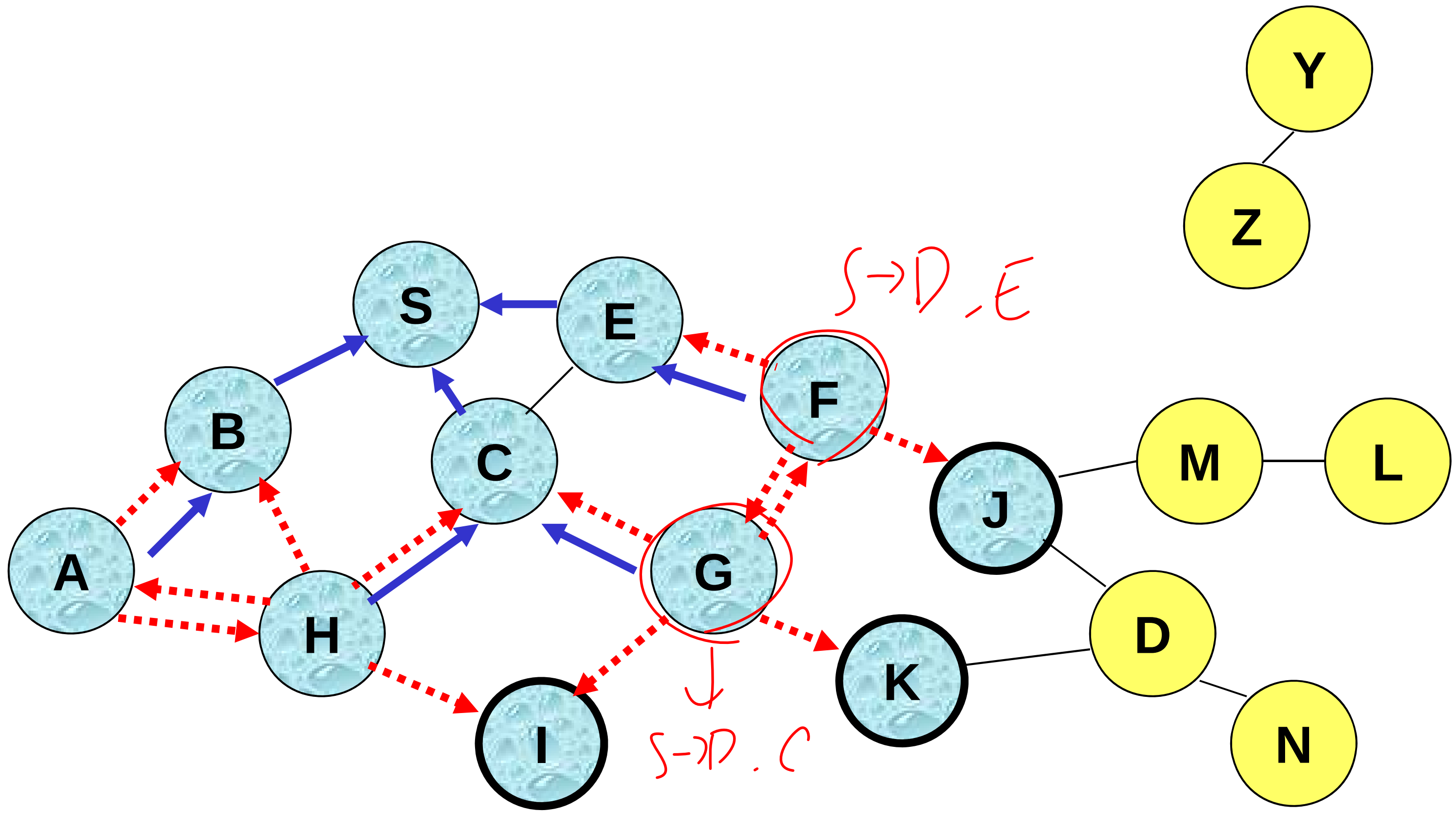
Route Requests in AODV



Route Requests in AODV

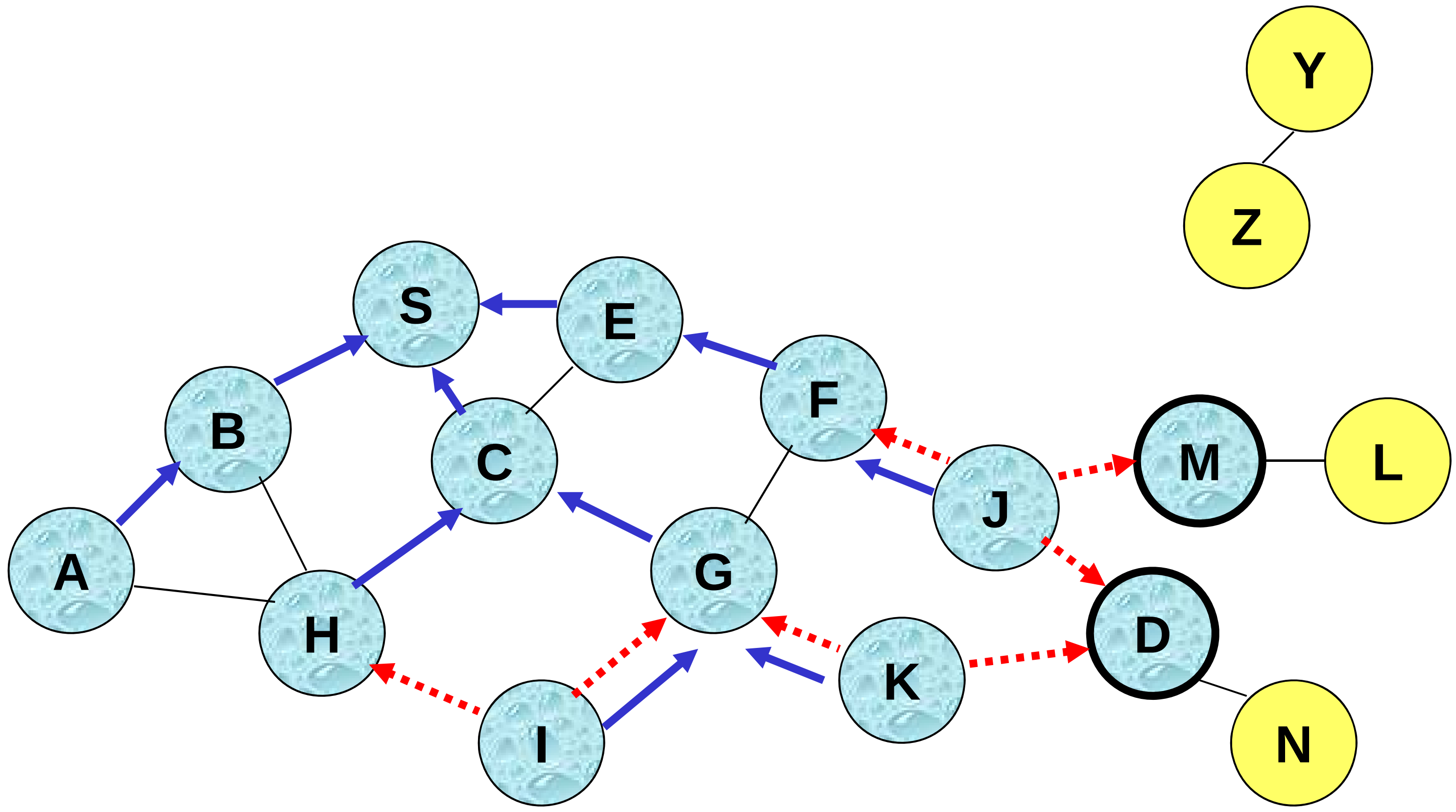


Reverse Path Setup in AODV

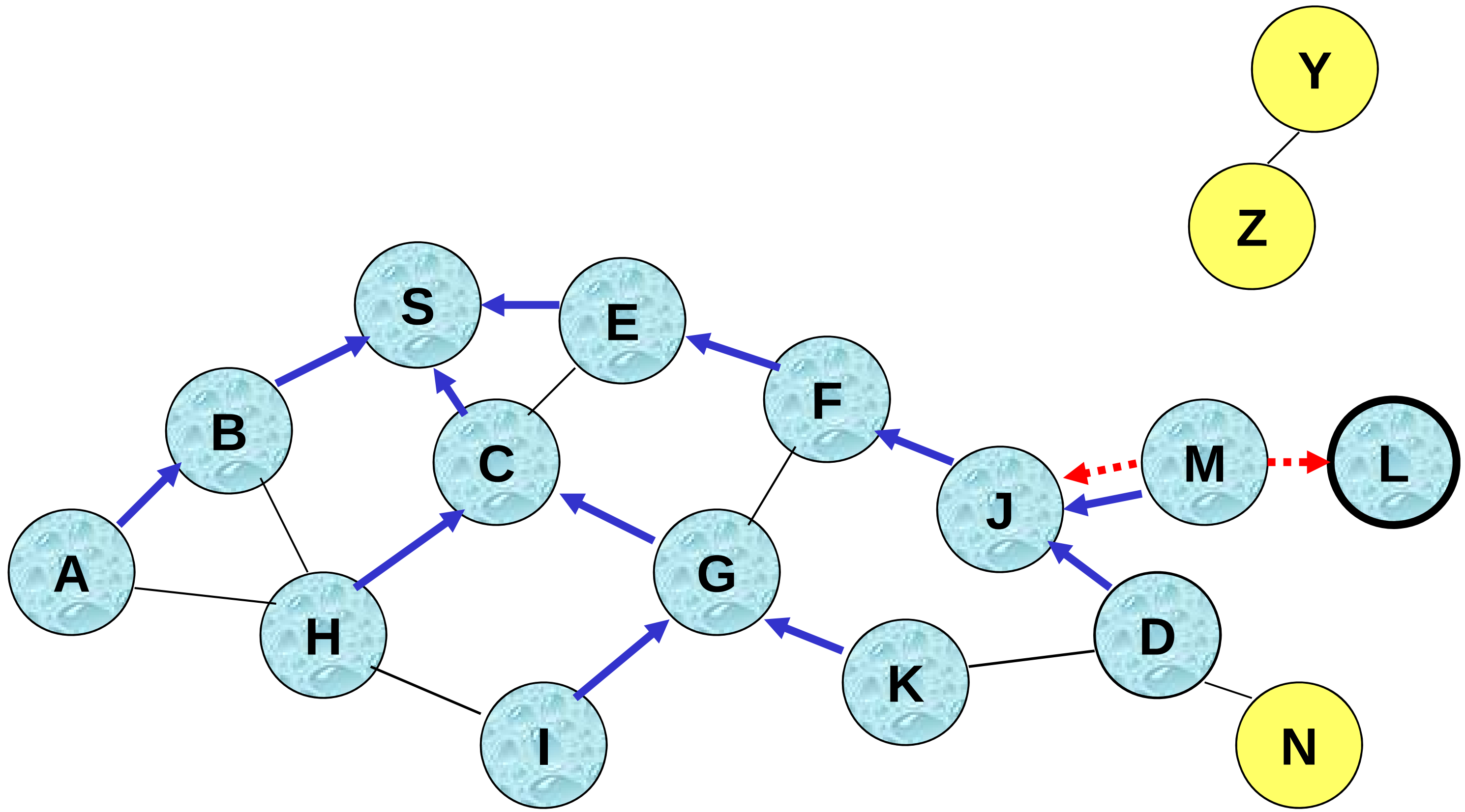


- Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once

Reverse Path Setup in AODV

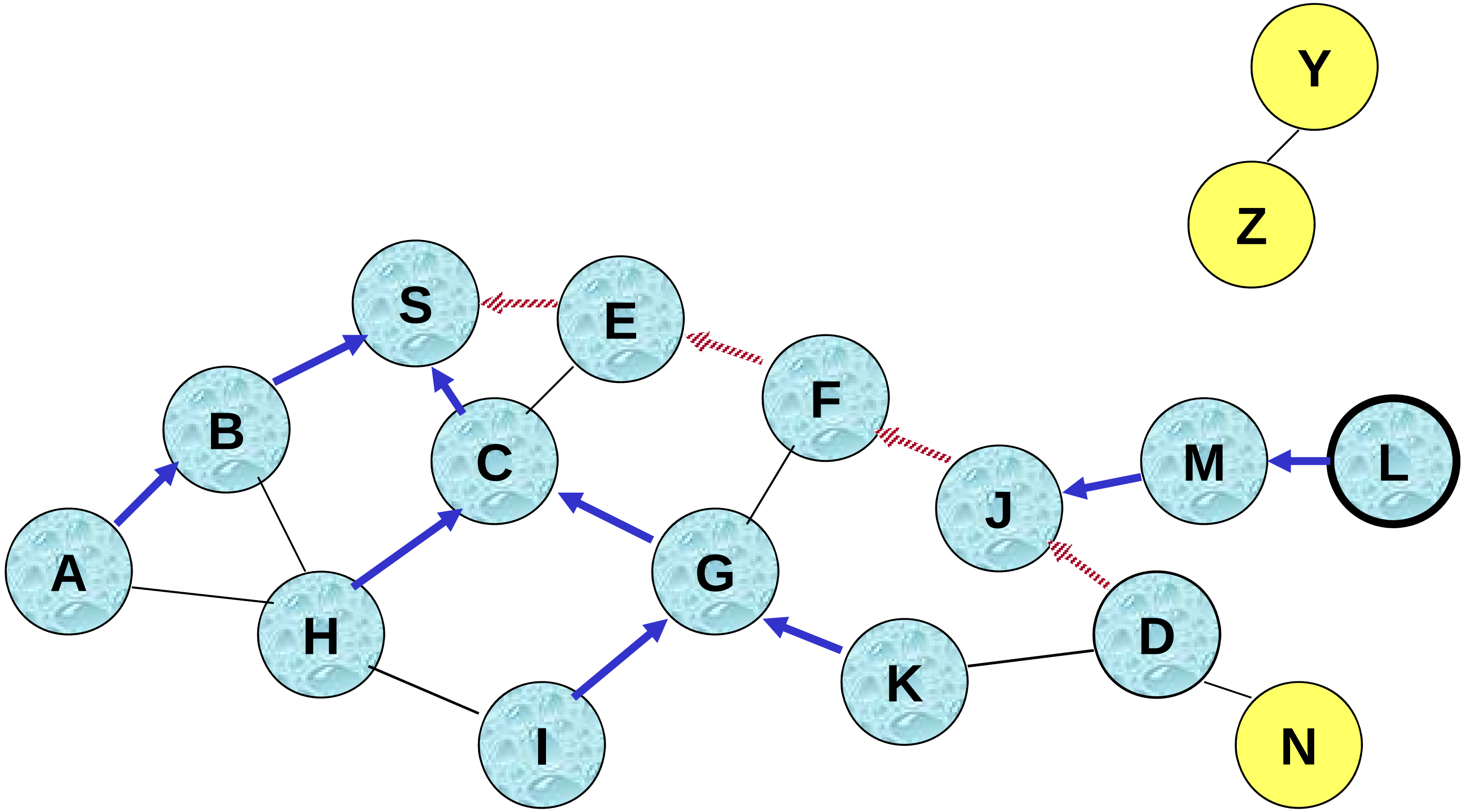


Reverse Path Setup in AODV



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

Route Reply in AODV



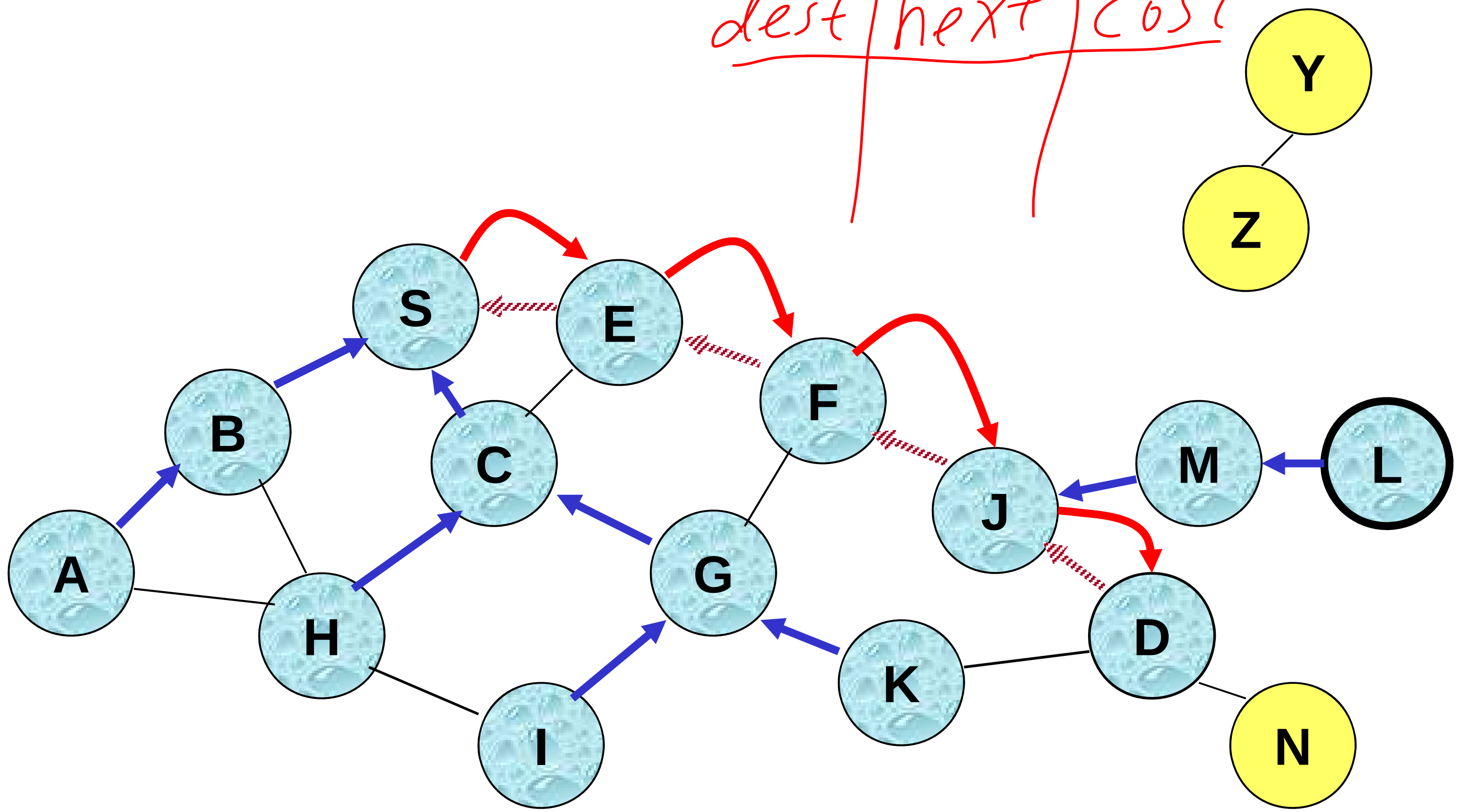
Represents links on path taken by RREP

Route Reply in AODV

- An intermediate node (not the destination) may also send a Route Reply (RREP) provided that it knows a more recent path than the one previously known to sender S
- To determine whether the path known to an intermediate node is more recent, *destination sequence numbers* are used
- The likelihood that an intermediate node will send a Route Reply when using AODV not as high as DSR
  - A new Route Request by node S for a destination is assigned a higher destination sequence number. An intermediate node which knows a route, but with a smaller sequence number, cannot send Route Reply

Forward Path Setup in AODV

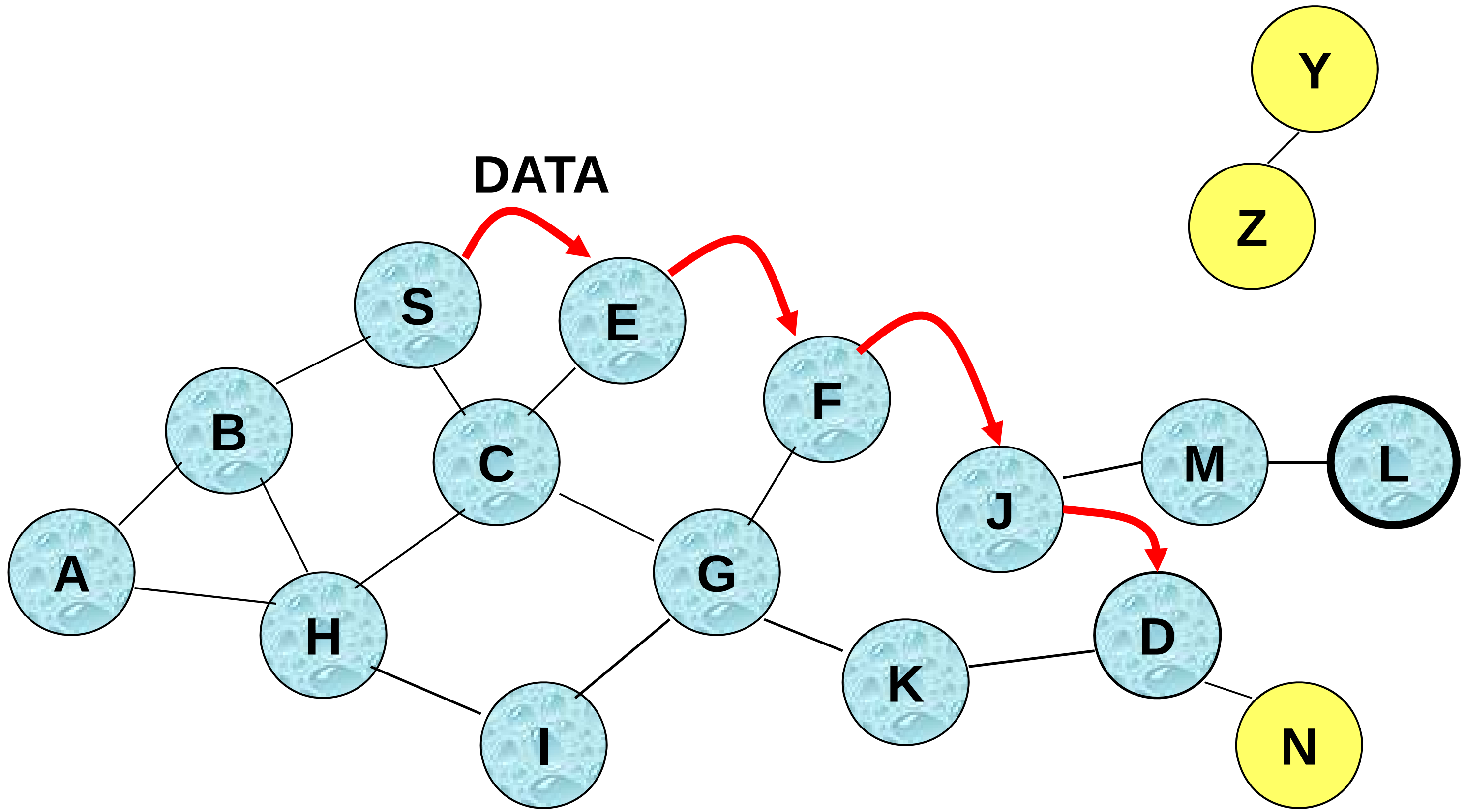
dest	next	cost



Forward links are setup when RREP travels along the reverse path

Represents a link on the forward path

Data Delivery in AODV



Routing table entries used to forward data packet.

Route is *not* included in packet header.

Timeouts

- A routing table entry maintaining a reverse path is purged after a timeout interval
  - timeout should be long enough to allow RREP to come back

- A routing table entry maintaining a forward path is purged if *not used* for a *active_route_timeout* interval
  - if no data is being sent using a particular routing table entry, that entry will be deleted from the routing table (even if the route may actually still be valid)

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 (route error) 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*

Link Failure Detection

- *Hello* messages: Neighboring nodes periodically exchange hello message

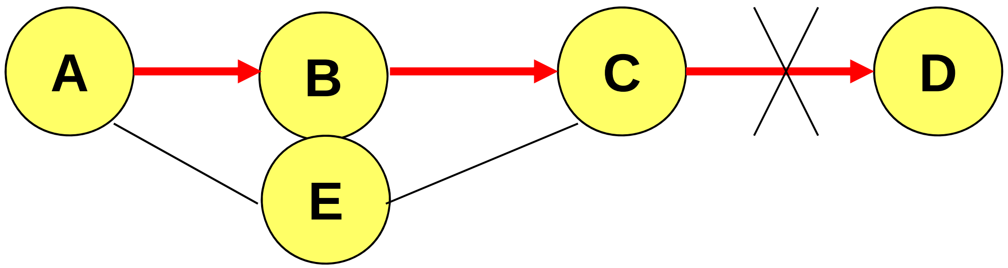
- Absence of hello message is used as an indication of link failure

- Alternatively, failure to receive several MAC-level acknowledgement may be used as an indication of link failure

Why Sequence Numbers in AODV

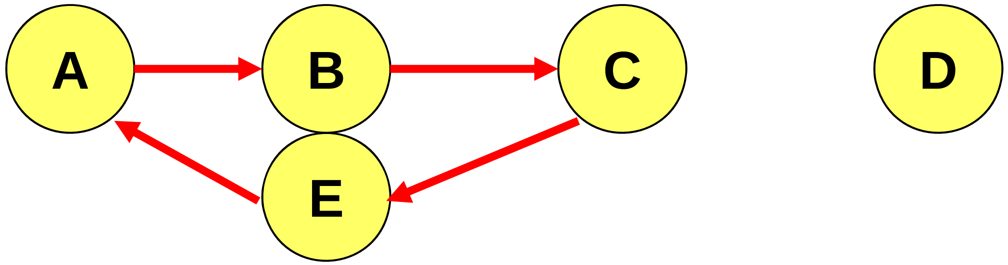
Loop

- To avoid using old/broken routes
  - To determine which route is newer
- To prevent formation of loops



  - Assume that A does not know about failure of link C-D because RERR sent by C is lost
  - Now C performs a route discovery for D. Node A receives the RREQ (say, via path C-E-A)
  - Node A will reply since A knows a route to D via node B
  - Results in a loop (for instance, C-E-A-B-C)

Why Sequence Numbers in AODV



- Loop C-E-A-B-C

Optimization: Expanding Ring Search

- Route Requests are initially sent with small Time-to-Live (TTL) field, to limit their propagation
  - DSR also includes a similar optimization

- If no Route Reply is received, then larger TTL tried

Summary: AODV

- 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
  - Multi-path extensions can be designed
  - DSR may maintain several routes for a single destination

- Unused routes expire even if topology does not change

OLSR

Proactive Protocols

- Most of the schemes discussed so far are reactive

- Proactive schemes based on distance-vector and link-state mechanisms have also been proposed

Link State Routing [Huitema95]

- Each node periodically floods status of its links
- Each node re-broadcasts link state information received from its neighbor
- Each node keeps track of link state information received from other nodes
- Each node uses above information to determine next hop to each destination

Optimized Link State Routing (OLSR) [Jacquet00ietf,Jacquet99Inria]

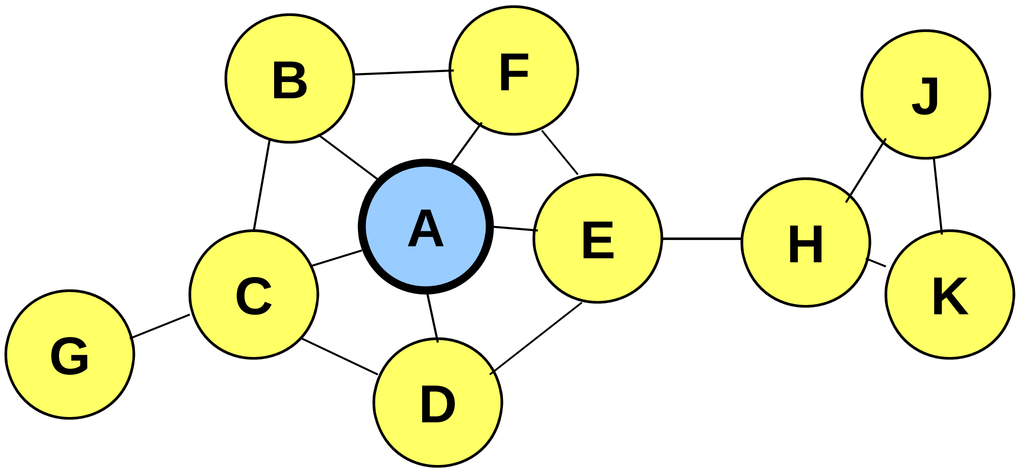
- The overhead of flooding link state information is reduced by requiring fewer nodes to forward the information

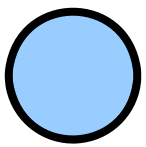
- A broadcast from node X is only forwarded by its *multipoint relays*
  - *Reduce the number of transmission*

- Multipoint relays of node X are its neighbors such that each two-hop neighbor of X is a one-hop neighbor of at least one multipoint relay of X
  - Each node transmits its neighbor list in periodic beacons, so that all nodes can know their 2-hop neighbors, in order to choose the multipoint relays

Optimized Link State Routing (OLSR)

- Nodes C and E are multipoint relays of node A

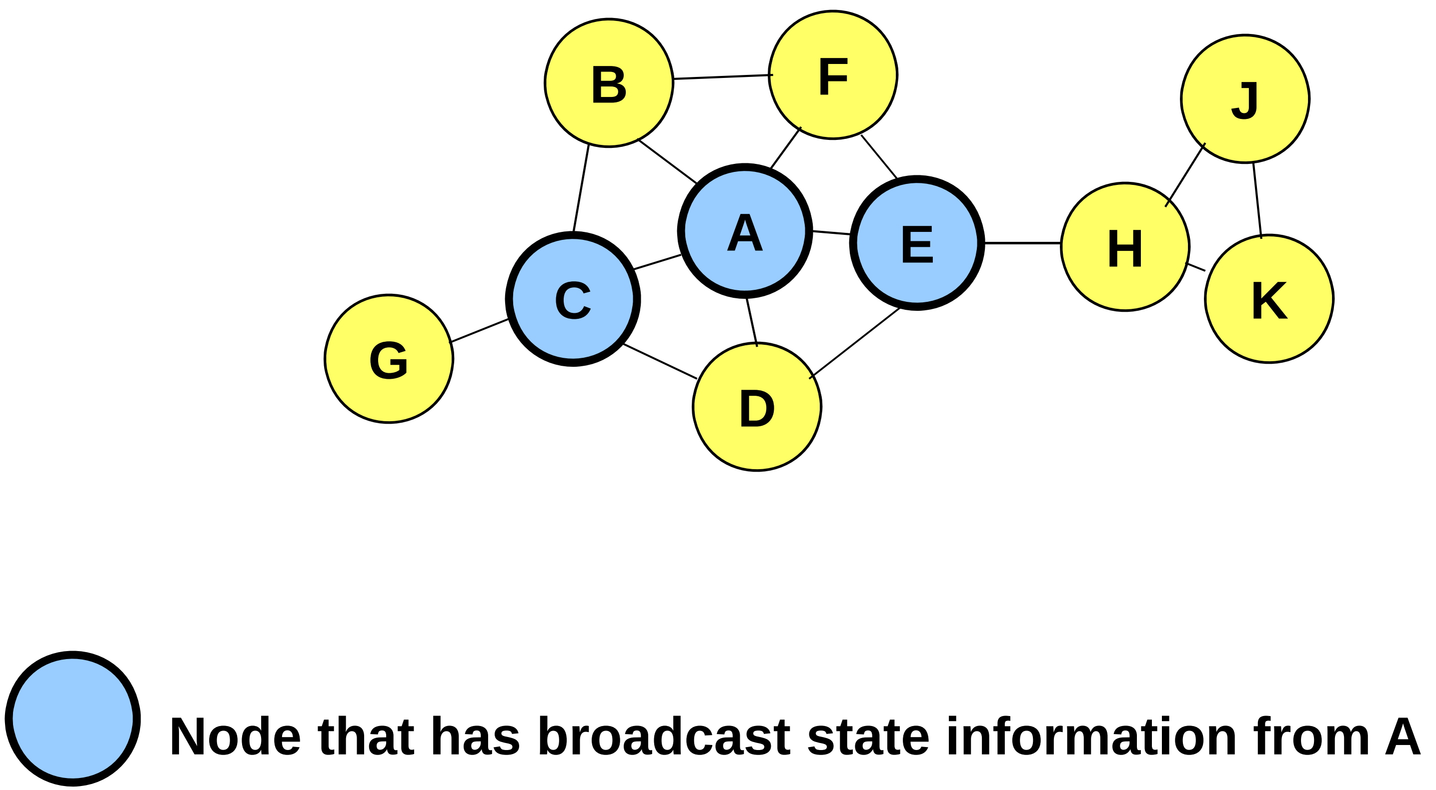




Node that has broadcast state information from A

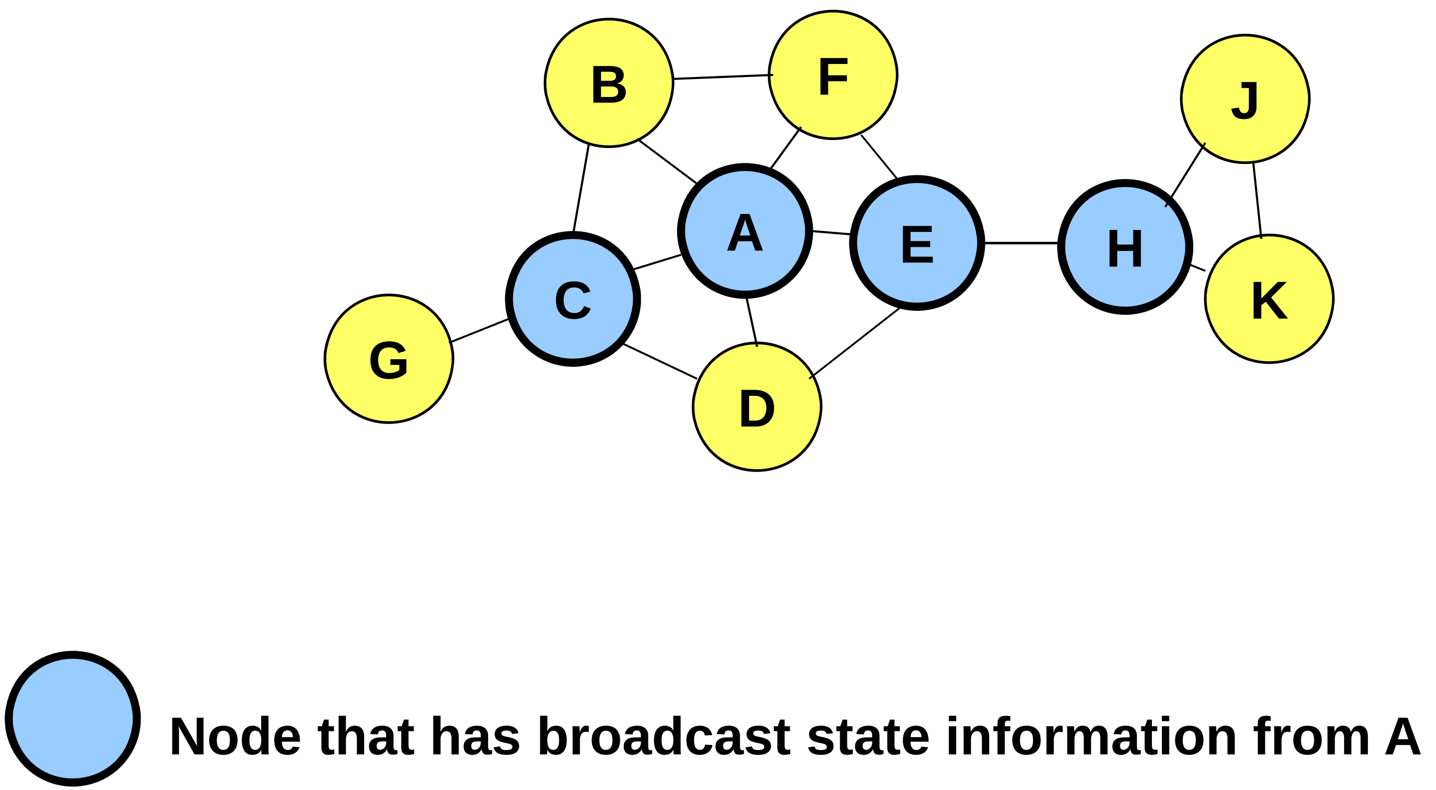
Optimized Link State Routing (OLSR)

- Nodes C and E forward information received from A



Optimized Link State Routing (OLSR)

- Nodes E and K are multipoint relays for node H
- Node K forwards information received from H
  - E has already forwarded the same information once



OLSR

- OLSR floods information through the multipoint relays

- The flooded information itself is for links connecting nodes to respective multipoint relays

- Routes used by OLSR only include multipoint relays as intermediate nodes

DSDV

Destination-Sequenced Distance-Vector (DSDV) [Perkins94Sigcomm]

- Each node maintains a routing table which stores
  - next hop towards each destination
  - a cost metric for the path to each destination
  - a destination sequence number that is created by the destination itself
  - Sequence numbers used to avoid formation of loops

- Each node periodically forwards the routing table to its neighbors
  - Each node increments and appends its sequence number when sending its local routing table
  - This sequence number will be attached to route entries created for this node

DSDV Protocol

- [Review]Routing Algorithm
  - Link-State algorithm:
    - Each node maintains a view of the network topology
  - Distance-Vector algorithm:
    - Every node maintains the distance of each destination
- DSDV is a variation of Distance Vector routing for MANET environment
  - DSDV is Destination Based
  - No global view of topology

DSDV Protocol

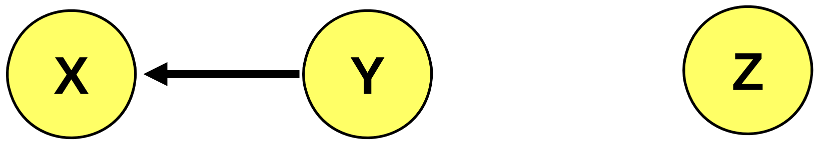
- DSDV is Proactive (Table Driven)
  - Each node maintains routing information for all known destinations
  - Routing information must be updated periodically
  - Traffic overhead even if there is no change in network topology
  - Maintains routes which are never used

DSDV Protocol

- Keep the simplicity of Distance Vector
- Guarantee Loop Freeness
  - New Table Entry for Destination Sequence Number
- Allow fast reaction to topology changes
  - Make immediate route advertisement on significant changes in routing table
  - but wait with advertising of unstable routes (damping fluctuations)

DSDV Basics

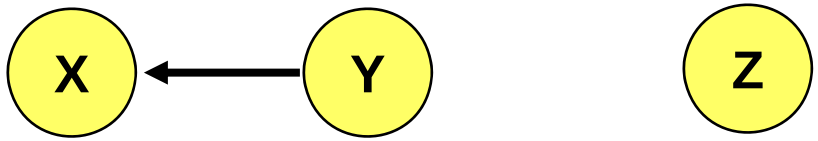
- Assume that node X receives routing information from Y about a route to node Z



- Define
  - S(X): the destination seq # for node Z stored at node X
  - S(Y): the destination seq # for node Z sent by node Y (sending with Y's routing table to node X)

DSDV Basics

- Node X takes the following steps:



  - 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 $S(Y)$

Route Selection and Seq

- Just like distance vector routing protocols
- Nodes learn paths that have a metric *and* a sequence number
  - Prefer route with highest sequence number
  - Among routes with equal sequence numbers, prefer route with lowest metric

DSDV (Table Entries)

Destination	Next	Metric	Seq. Nr	Install Time	Stable Data
A	A	0	A-550	001000	Ptr_A
B	B	1	B-102	001200	Ptr_B
C	B	3	C-588	001200	Ptr_C
D	B	4	D-312	001200	Ptr_D

- **Sequence number** originated from destination. Ensures loop freeness.
- **Install Time** when entry was made (used to delete stale entries from table)
- **Stable Data** Pointer to a table holding information on how stable a route is. Used to damp fluctuations in network.

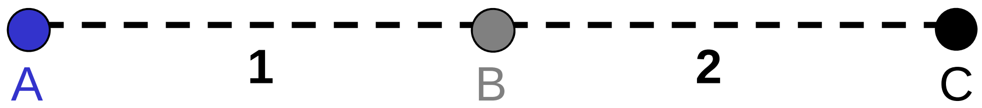
DSDV (Route Advertisements)

- Advertise to each neighbor own routing information
  - Destination Address
  - Metric = Number of Hops to Destination
  - Destination Sequence Number
- Rules to set sequence number information
  - On each advertisement increase own destination sequence number (use only even numbers)
  - If a node is no more reachable (timeout) increase sequence number of this node by 1 and set metric = ∞

DSDV (Route Selection)

- Update information is compared to own routing table
  - 1. Select route with higher destination sequence number (This ensure to use always newest information from destination)
  - 2. Select the route with better metric when sequence numbers are equal.

DSDV (Tables)



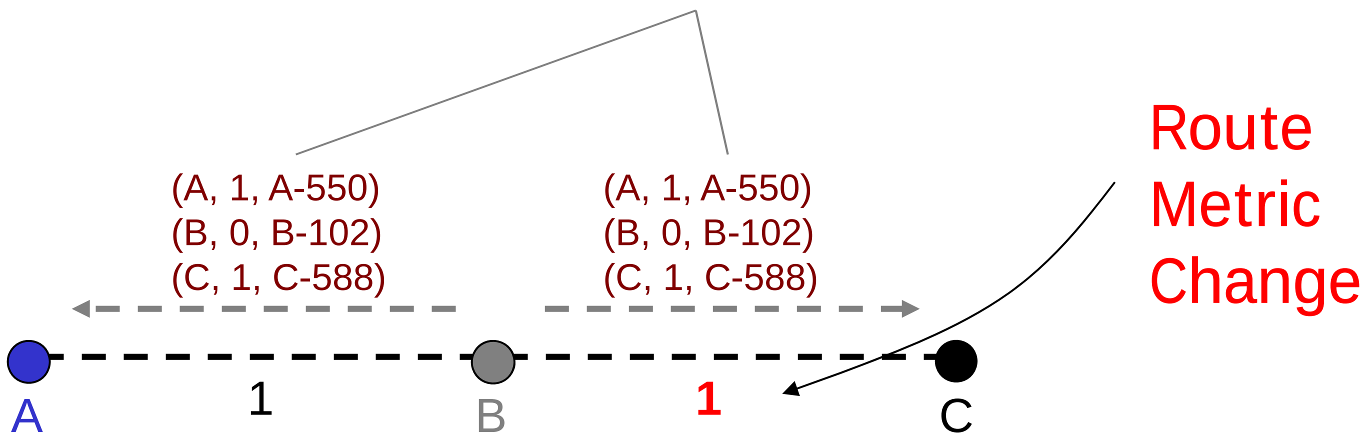
Dest.	Next	Metric	Seq
A	A	0	A-550
B	B	1	B-100
C	B	3	C-586

Dest.	Next	Metric	Seq
A	A	1	A-550
B	B	0	B-100
C	C	2	C-588

Dest.	Next	Metric	Seq.
A	B	3	A-550
B	B	2	B-100
C	C	0	C-588

DSDV (Route Advertisement)

B increases Seq.Nr from 100 -> 102
B broadcasts routing information
to Neighbors A, C including
destination sequence numbers



Dest.	Next	Metric	Seq
A	A	0	A-550
B	B	1	B-102
C	B	2	C-588

Dest.	Next	Metric	Seq
A	A	1	A-550
B	B	0	B-102
C	C	1	C-588

Dest.	Next	Metric	Seq.
A	B	2	A-550
B	B	1	B-102
C	C	0	C-588

DSDV (Respond to Topology Changes)

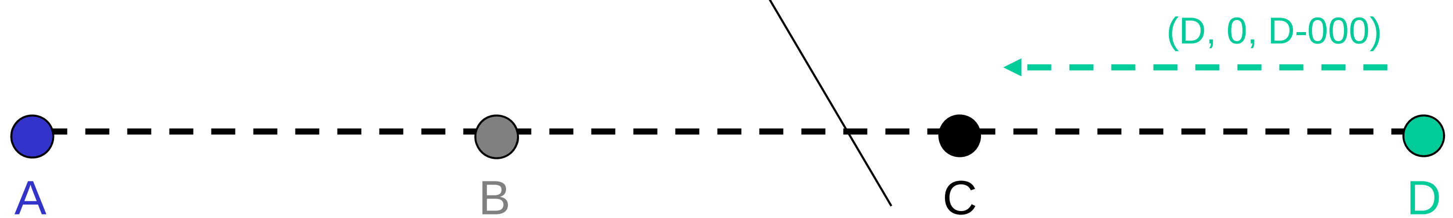
- Immediate advertisements
  - Information on new Routes, broken Links, metric change is immediately propagated to neighbors.

- Full/Incremental Update:
  - Full Update: Send all routing information from own table.
  - Incremental Update: Send only entries that has changed. (Make it fit into one single packet)

DSDV (New Node)

2. Insert entry for D with sequence number D-000
Then immediately broadcast own table

1. D broadcast for first time
Send Sequence number D-000



Dest.	Next	Metric	Seq.
A	A	0	A-550
B	B	1	B-104
C	B	2	C-590

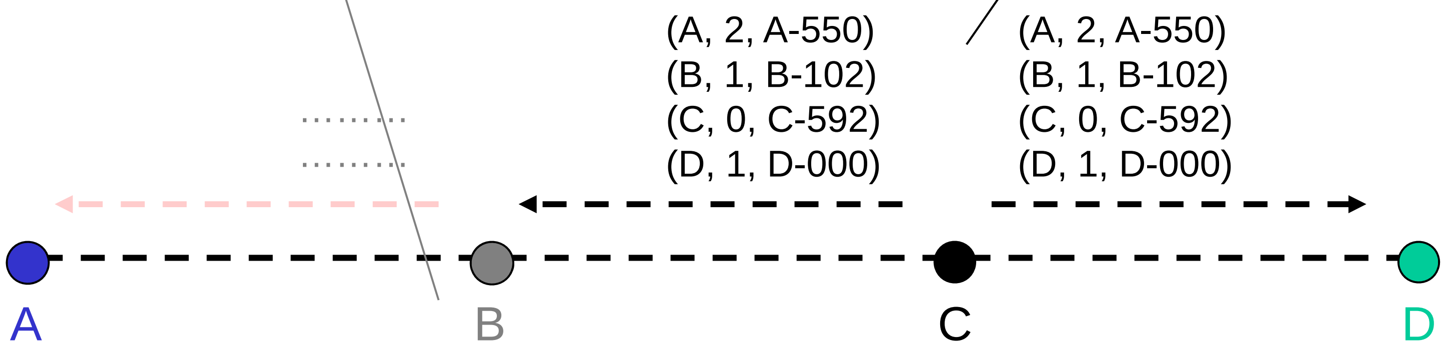
Dest.	Next	Metric	Seq.
A	A	1	A-550
B	B	0	B-104
C	C	1	C-590

Dest.	Next	Metric	Seq.
A	B	2	A-550
B	B	1	B-104
C	C	0	C-590
D	D	1	D-000

DSDV (New Node cont.)

3. C increases its sequence number to C-592 then broadcasts its new table.

4. B gets this new information and updates its table…….



Dest.	Next	Metric	Seq.
A	A	0	A-550
B	B	1	B-104
C	B	2	C-590

Dest.	Next	Metric	Seq.
A	A	1	A-550
B	B	0	B-102
C	C	1	C-592
D	C	2	D-000

Dest.	Next	Metric	Seq.
A	B	2	A-550
B	B	1	B-102
C	C	0	C-592
D	D	1	D-000

Summery: DSDV

- Advantages
  - Simple (almost like Distance Vector)
  - Loop free through destination seq. numbers
  - No latency caused by route discovery

- Disadvantages
  - No sleeping nodes
  - Overhead: most routing information never used

Hybrid Protocols

Zone Routing Protocol (ZRP) [Haas98]

Zone routing protocol combines

- Proactive protocol: which pro-actively updates network state and maintains route regardless of whether any data traffic exists or not

- Reactive protocol: which only determines route to a destination if there is some data to be sent to the destination

ZRP

- 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

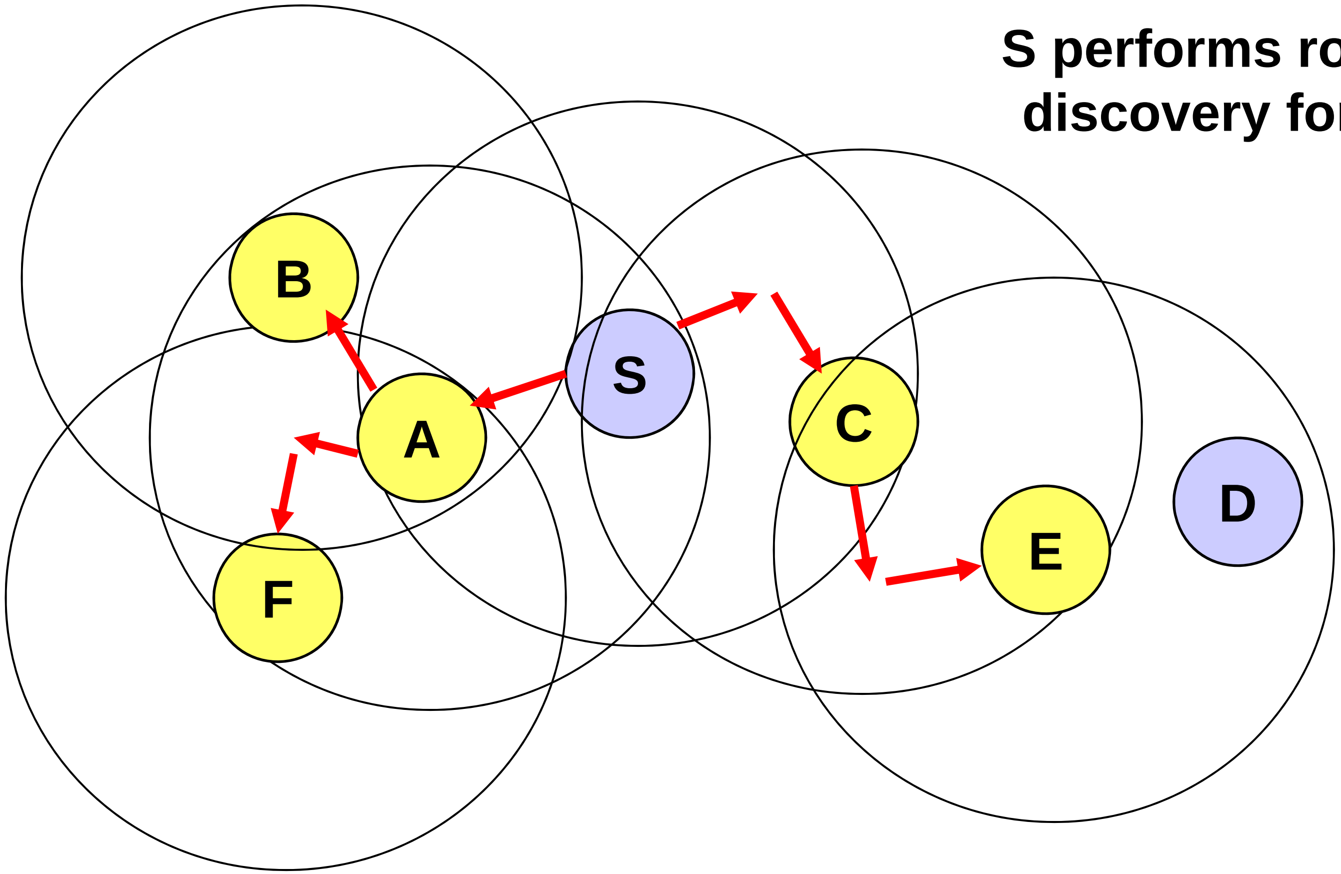
ZRP

- Intra-zone routing: Pro-actively maintain state information for links within a short distance from any given node
  - Routes to nodes within short distance are thus maintained proactively (using, say, link state or distance vector protocol)

- Inter-zone routing: Use a route discovery protocol for determining routes to far away nodes. Route discovery is similar to DSR with the exception that route requests are propagated via peripheral nodes.

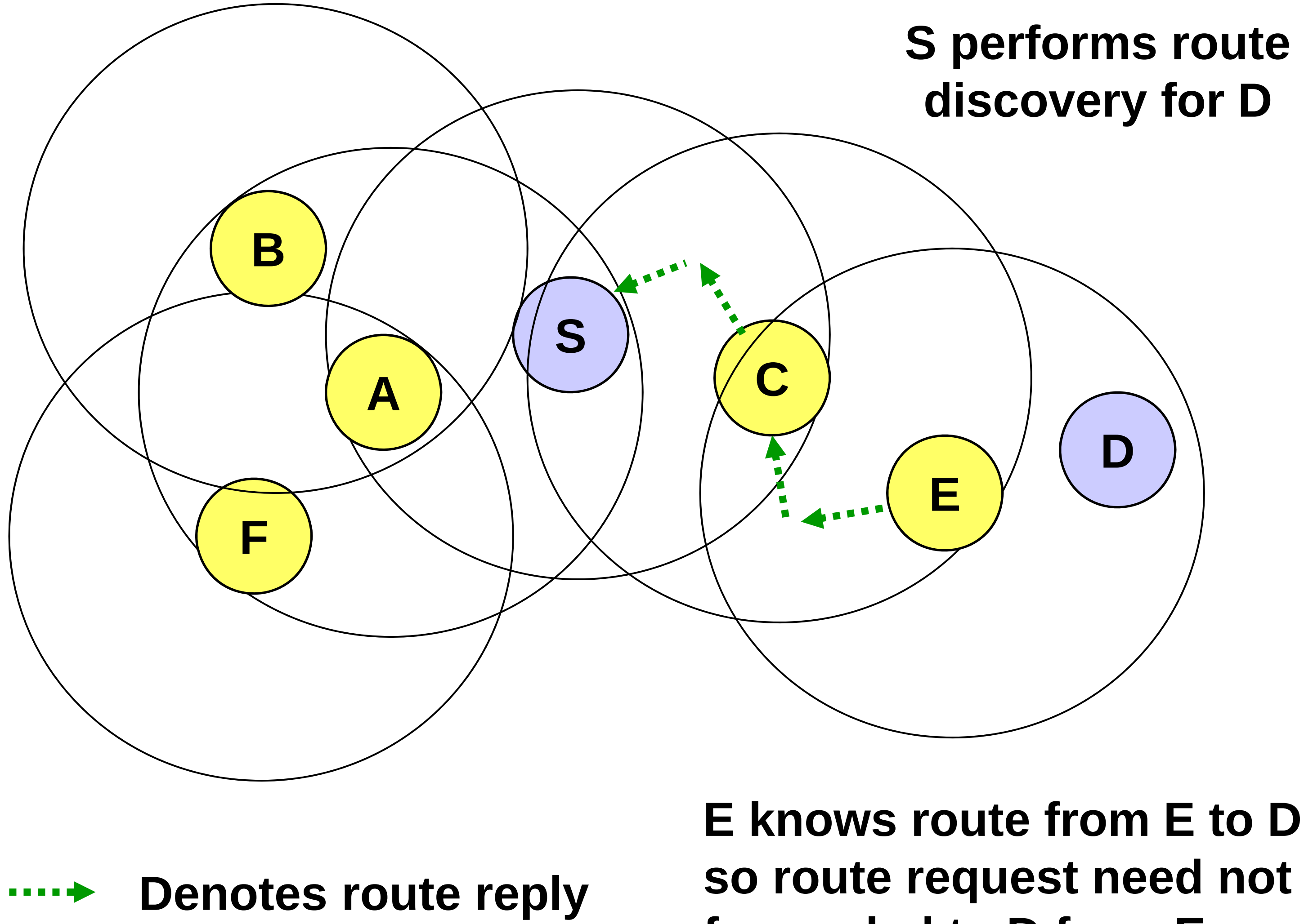
ZRP: Example with Zone Radius = d = 2

S performs route discovery for D



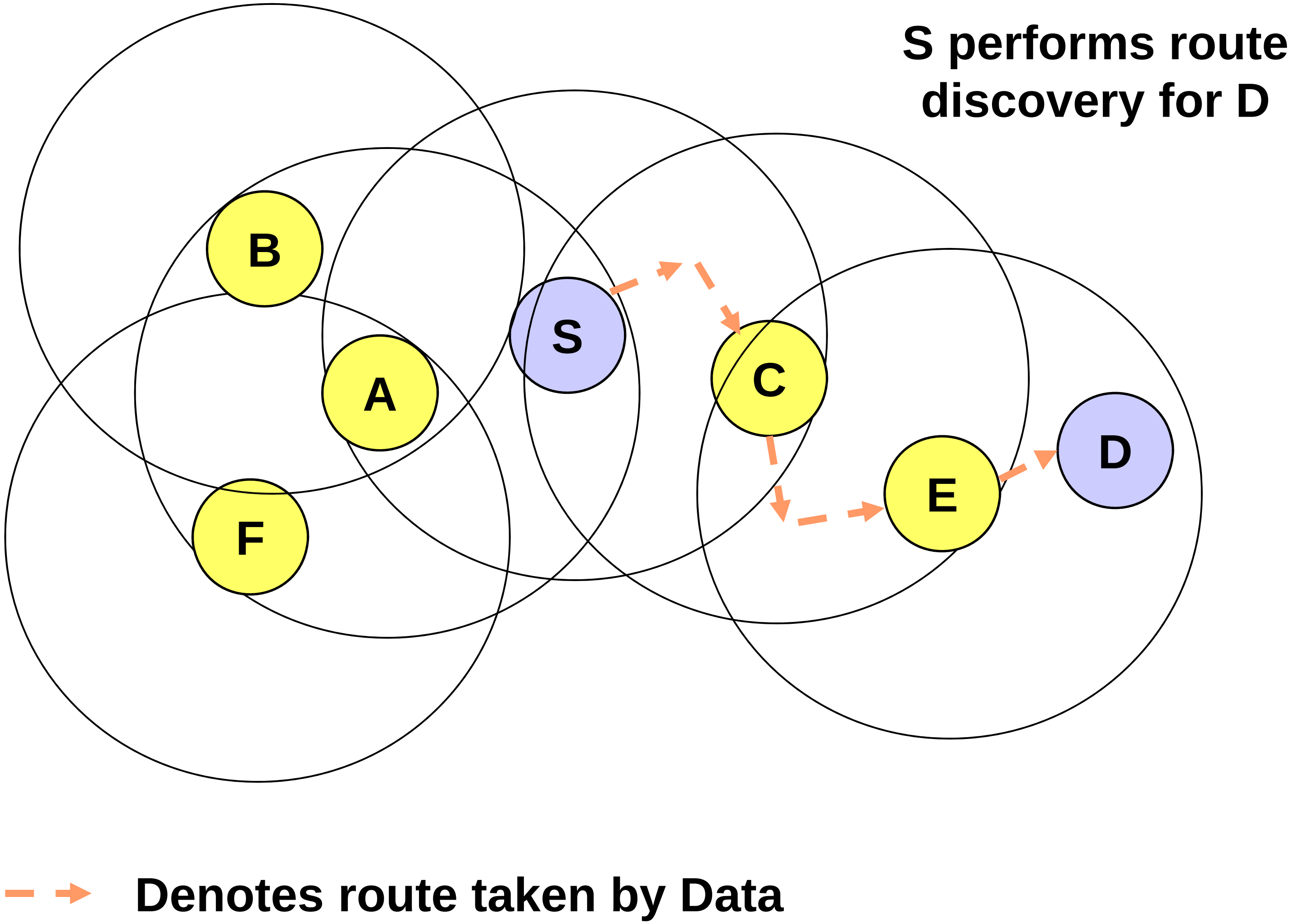
Denotes route request

ZRP: Example with $d = 2$



E knows route from E to D, so route request need not be forwarded to D from E

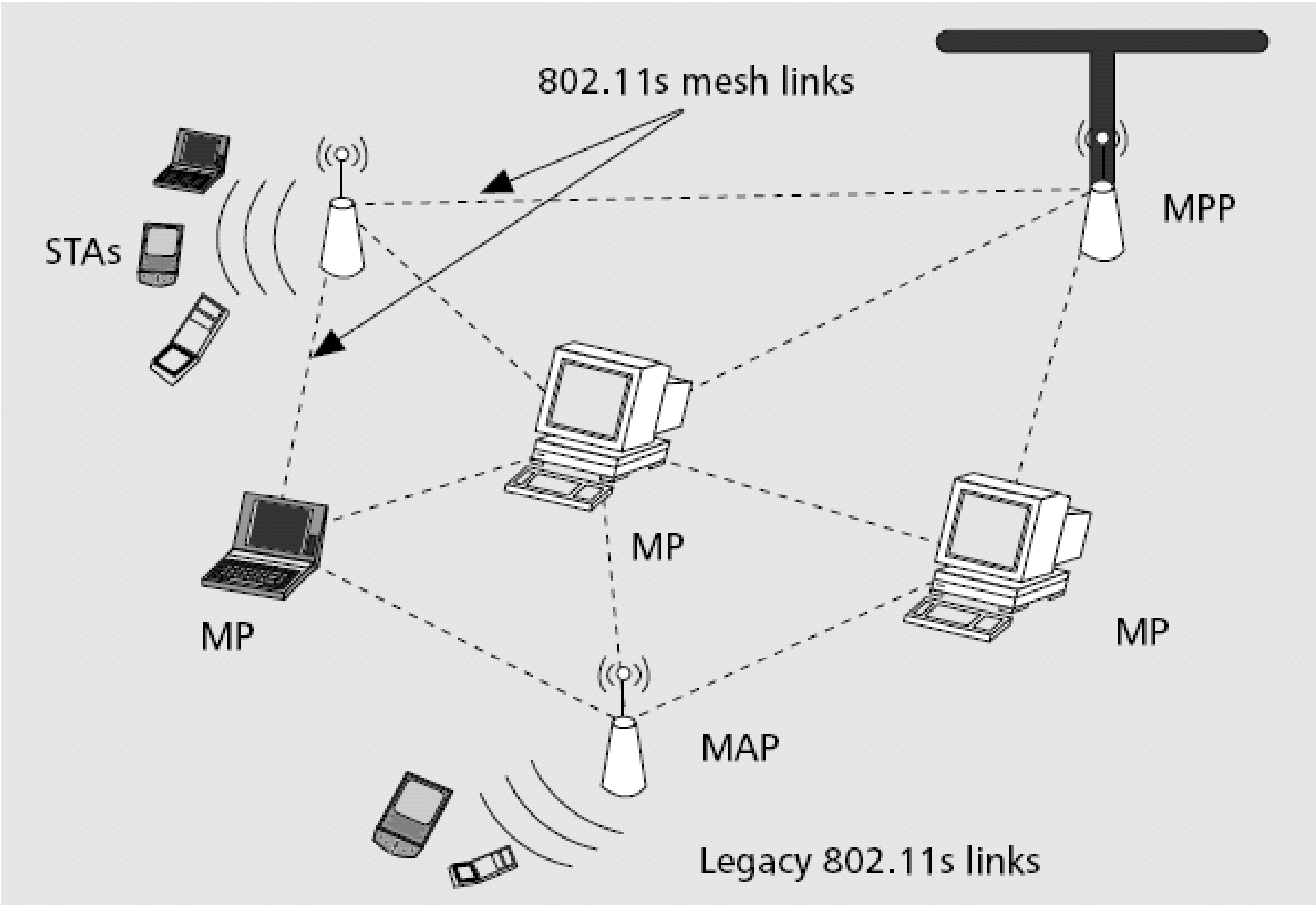
ZRP: Example with *d* = 2



Research Methodologies

- Analytical models
  - Queuing theory
  - Asymptotic Methods in Probability
  - Graph theory
  - Game theory
- Simulation
  - Ns-2, OPNET, Qualnet
- Implementation
  - MIT Roofnet → outdoor wireless mesh
  - In-building mesh testbed
  - Sensor networks testbed
  - Outdoor vehicular testing

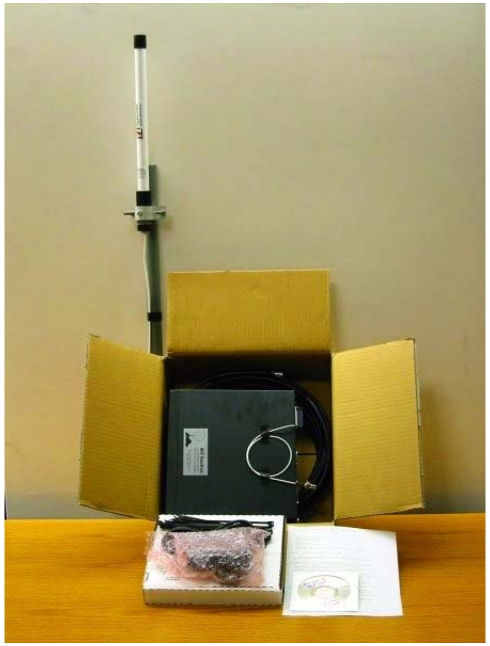
802.11s



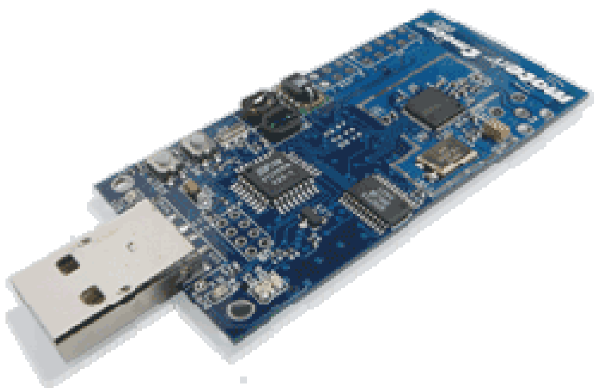
MIT Roofnet


Rate
(KB/s)
1
10
100
200
300
400
500
600
16 active nodes
Last Updated: Thu Sep 15 03:52:13 EDT 2005

Wireless Mesh Network Nodes



Wireless Sensor Nodes



WSN: Applications

- Environmental monitoring
  - Traffic, habitat, security
- Industrial sensing and diagnostics
  - Manufacturing, supply chains
- Context-aware computing
  - Intelligent homes
- Military applications:
  - Multi-target tracking
- Infrastructure protection:
  - Power grids

IoT and M2M

- IoT: Internet of Things
  - Internet of Everything
- M2M
  - Machine-to-machine communications
    - LTE-based MTC (machine-type-communications)
    - IEEE 802.11ah
    - Zigbee
    - Bluetooth
    - OneM2M