

MANET Routing

- Introduction to MANETs
- Fundamentals of Wireless Ad Hoc Networks
- Routing in MANETs
- Properties of MANETs

Graphics on MANET routing taken in parts from: Nitin H. Vaidya

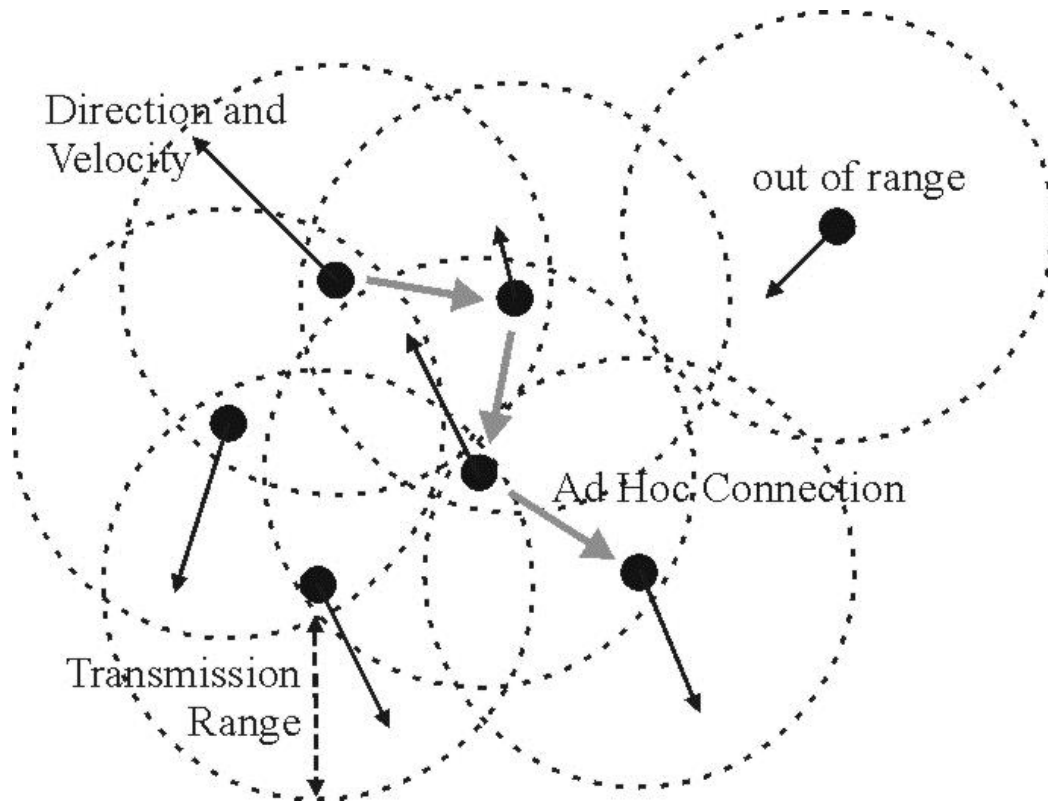


Distributed Systems in Mobile Environments

- ▶ Scenario 1: Mobile Overlay Members
 - ▶ Walking users at roaming devices ...
 - ▶ Issues: Transfer of personal context, location-based context
 - ▶ Networking solution: application transparency of **Mobile IP(v6)**
- ▶ Scenario 2: Spontaneous Application Overlays
 - ▶ Collaborative applications in (local) **Mobile Ad-Hoc Networks**
 - ▶ Machine-to-machine settings in the Internet of Things (IoT)
 - ▶ Issues: Adapt to efficiency & proximity needed in MANETs, cope with unreliable, mobile underlay networks
 - ▶ P2P Systems and MANETs both void infrastructure



Ad Hoc Networks (WLAN, Bluetooth)

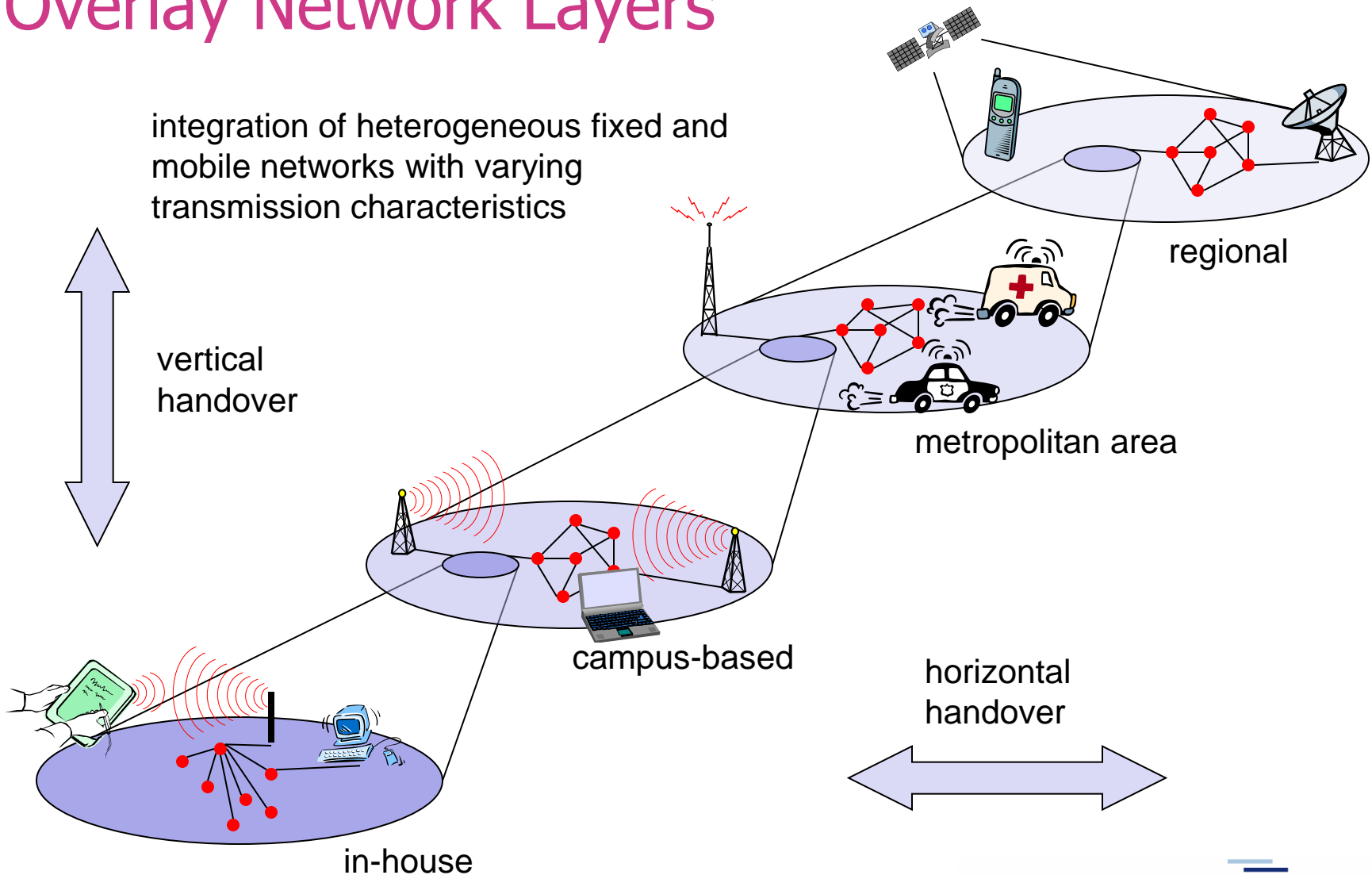


Characteristics:

- ▶ Self configuring
- ▶ Infrastructure free
- ▶ Wireless
- ▶ Unpredictable terminal mobility
- ▶ Limited radio transmission range
- ▶ Often: Low power & lossy
- ▶ Goal: provide communication between nodes



The Global View: Overlay Network Layers



Application Examples

- ▶ Sensors, Actuators & Relaying Nodes
- ▶ Single & Multiple Dedications of Nodes
- ▶ Common Examples:
 - ▶ Military, Rescue Services
 - ▶ Regional Mesh Networks
 - ▶ Collaborative Inter-Vehicular Communication
 - ▶ Sensor Networks
 - ▶ Personal Area Networking / Local Device Networks
 - ▶ Gaming, Edu-/Info-/Sociotainment
 - ▶ Home Automation



Mobile Ad Hoc Networks

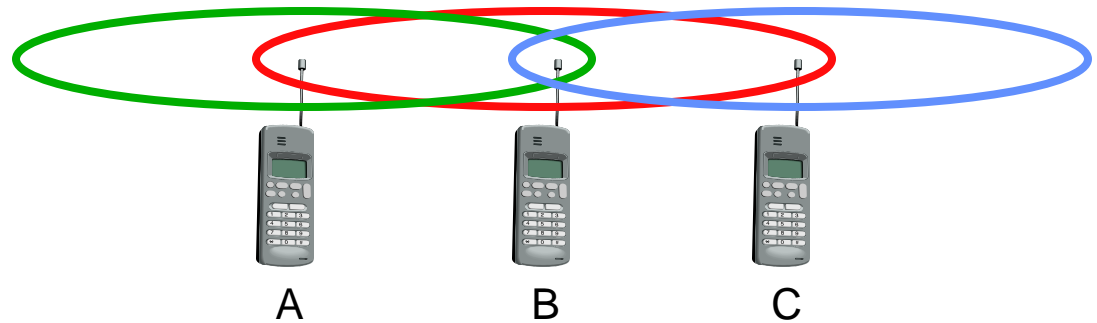
- ▶ Formed by wireless hosts which may be mobile
- ▶ Without (necessarily) using a pre-existing infrastructure
- ▶ Routes between nodes may potentially contain multiple hops
- ▶ Motivations:
 - ▶ Ease of deployment, low costs
 - ▶ Speed of deployment
 - ▶ Decreased dependence on infrastructure



Hidden and exposed terminals

► Hidden terminals

- A sends to B, C cannot receive A
- C wants to send to B, C senses a "free" medium (CS fails)
- collision at B, A cannot receive the collision (CD fails)
- A is "hidden" for C



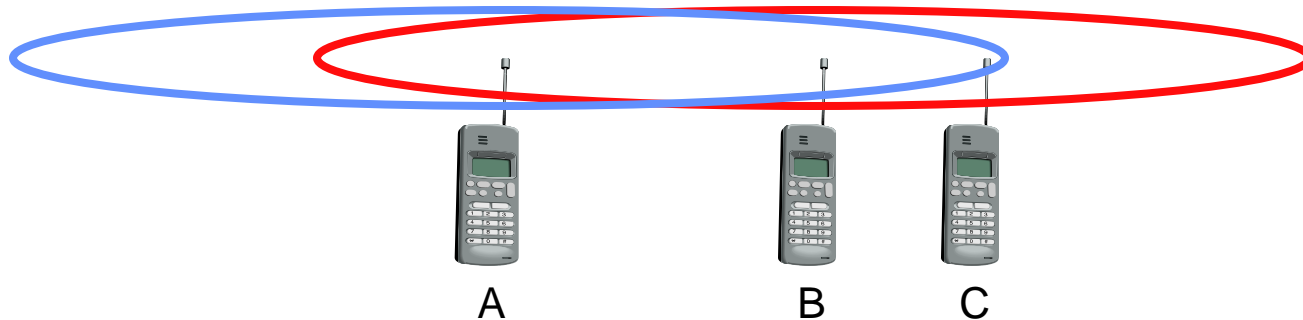
► Exposed terminals

- B sends to A, C wants to send to another terminal (not A or B)
- C has to wait, CS signals a medium in use
- but A is outside the radio range of C, therefore waiting is not necessary
- C is "exposed" to B



Near and far terminals

- ▶ Terminals A and B send, C receives
 - ▶ signal strength decreases proportional to the square of the distance
 - ▶ the signal of terminal B therefore drowns out A's signal
 - ▶ C cannot receive A

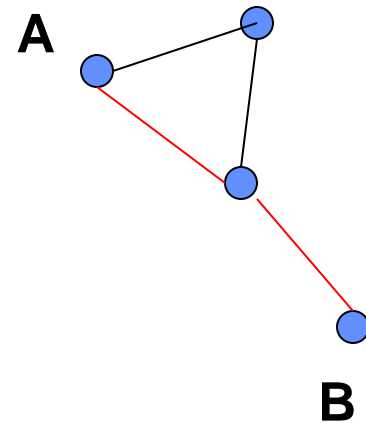
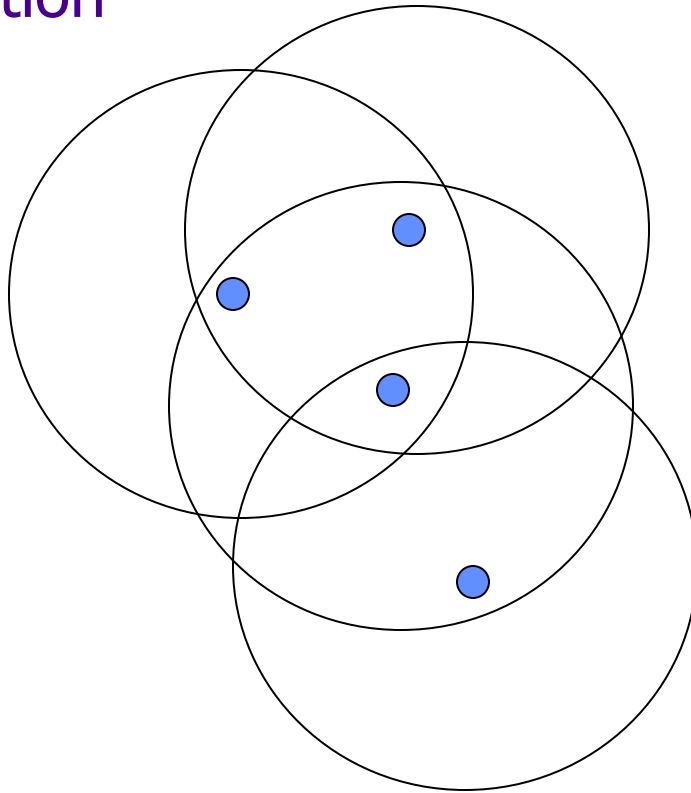


- ▶ If C for example was an arbiter for sending rights, terminal B would drown out terminal A already on the physical layer
- ▶ Also severe problem for CDMA-networks - precise power control needed!



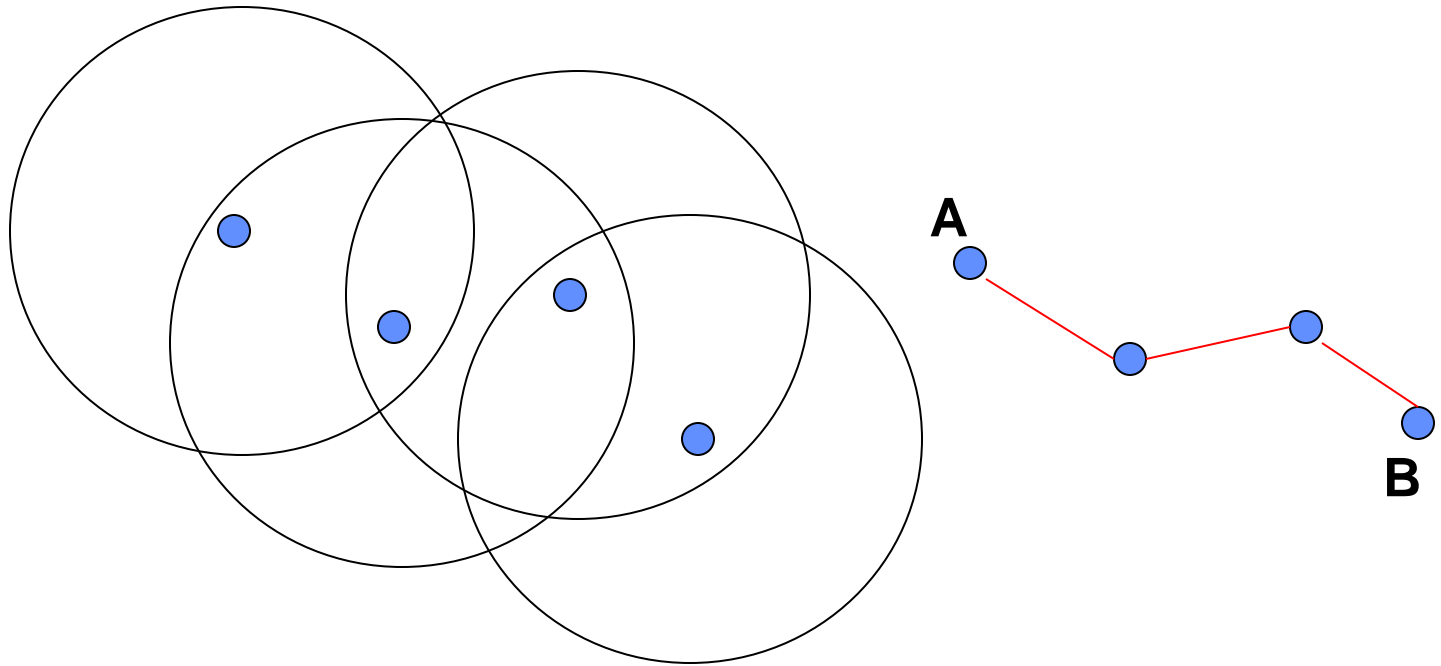
Mobile Ad Hoc Networks

- May need to traverse multiple links to reach a destination



Mobile Ad Hoc Networks (MANET)

- Mobility causes route changes



Many Variations

- ▶ Fully Symmetric Environment
 - ▶ all nodes have identical **capabilities** and **responsibilities**
- ▶ Asymmetric Capabilities
 - ▶ transmission ranges and radios may differ (→ asymmetric links)
 - ▶ battery life at different nodes may differ
 - ▶ processing capacity may be different at different nodes
 - ▶ speed of movement
- ▶ Asymmetric Responsibilities
 - ▶ only some nodes may route packets
 - ▶ some nodes may act as **leaders** of nearby nodes (e.g., cluster head)
- ▶ Varying Traffic Characteristics



Unicast Routing in MANETs - Why is it different ?

- ▶ Host mobility
 - ▶ link failure/repair due to mobility may have different characteristics than those due to other causes
- ▶ Rate of link failure/repair may be high when nodes move fast
- ▶ New performance criteria may be used
 - ▶ route stability despite mobility
 - ▶ energy consumption
- ▶ Many routing protocols proposed – no universal solution



Routing Protocols

- Proactive protocols
 - Determine routes independent of traffic pattern
 - Traditional link-state and distance-vector routing protocols are proactive

- Reactive protocols
 - Maintain routes only if needed

- Hybrid protocols



Trade-Off

- ▶ Latency of route discovery
 - ▶ Proactive protocols may have lower latency since routes are maintained at all times
 - ▶ Reactive protocols may have higher latency because a route from X to Y will be found only when X attempts to send to Y
- ▶ Overhead of route discovery/maintenance
 - ▶ Reactive protocols may have lower overhead since routes are determined only if needed
 - ▶ Proactive protocols can (but not necessarily) result in higher overhead due to continuous route updating
- ▶ Which approach achieves a better trade-off depends on the traffic and mobility patterns

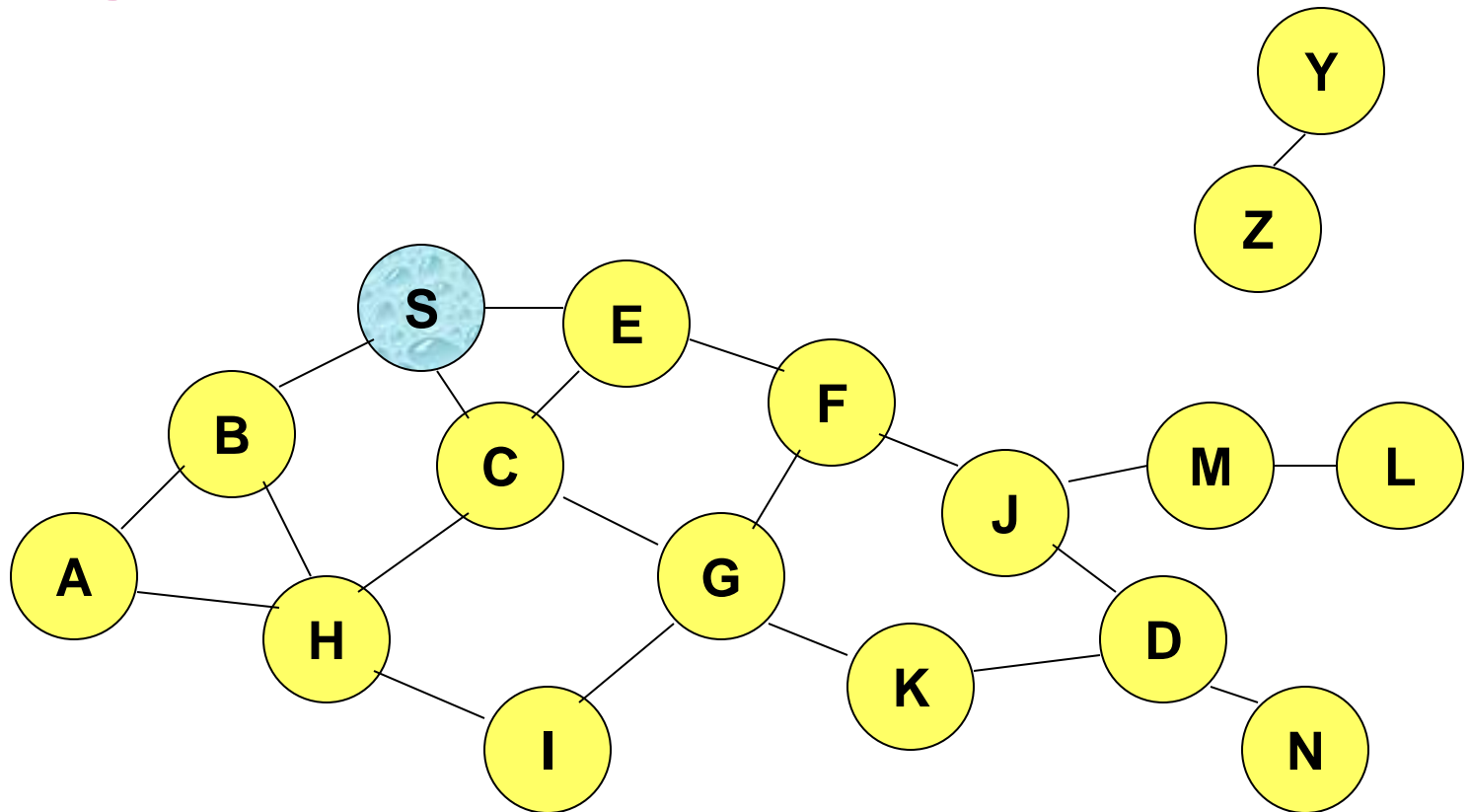


Flooding for Data Delivery

- Sender S broadcasts data packet P to all its neighbors
- Each node receiving P forwards P to its neighbors
- Sequence numbers used to avoid the possibility of forwarding the same packet more than once
- Packet P reaches destination D provided that D is reachable from sender S
- Node D does not forward the packet



Flooding for Data Delivery

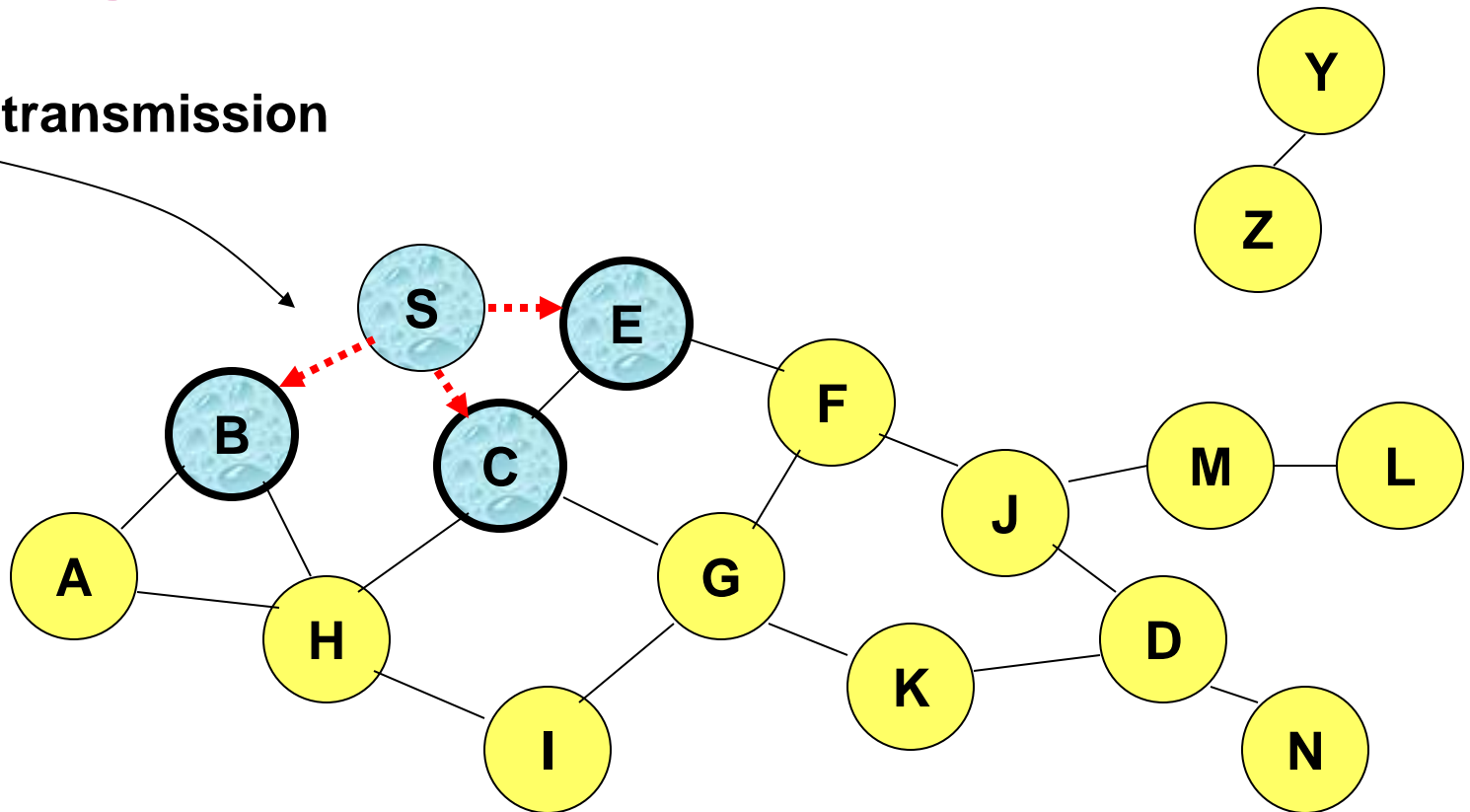


Represents a node that has received packet P
Represents that connected nodes are within each other's transmission range



Flooding for Data Delivery

Broadcast transmission



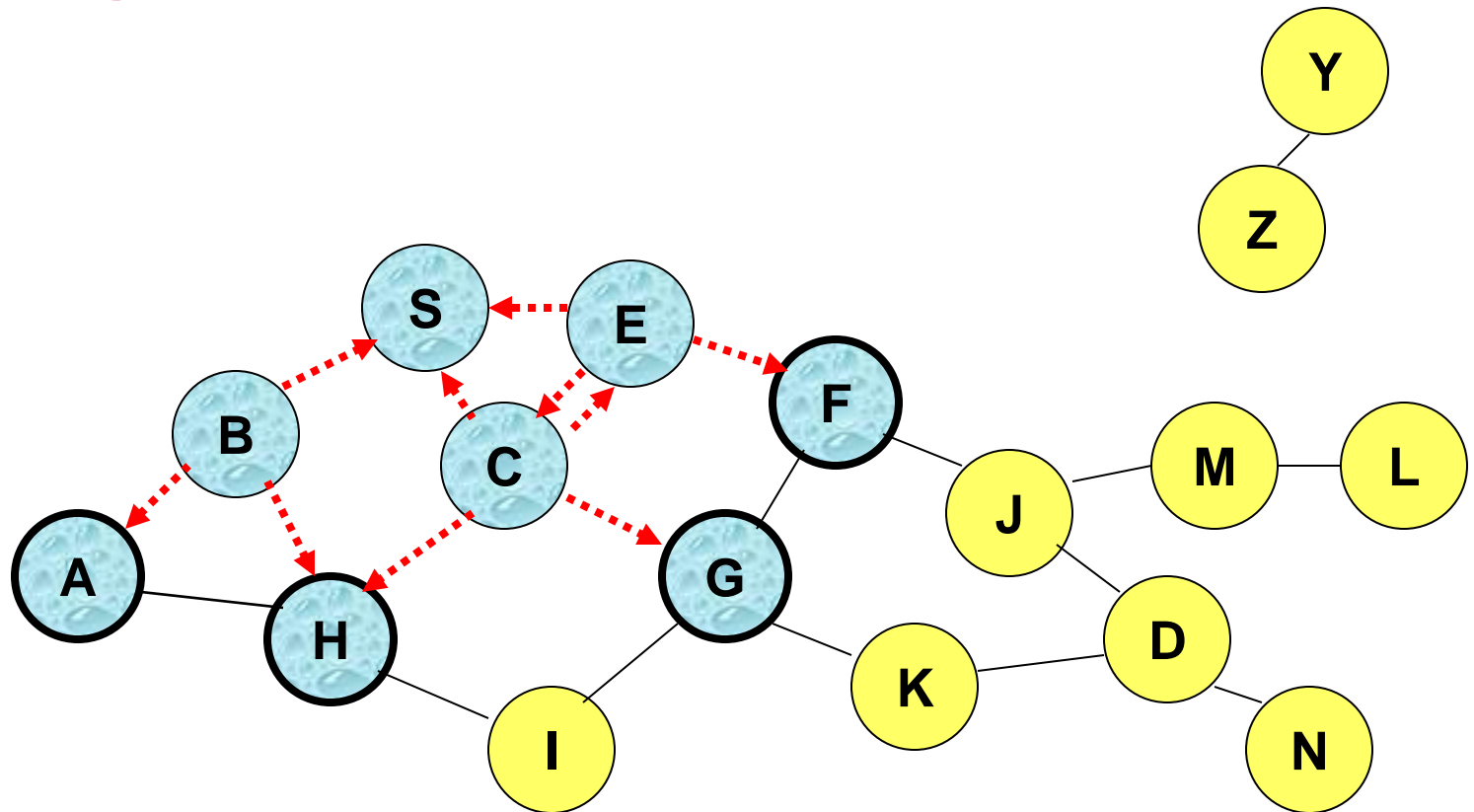
Represents a node that receives packet P for the first time



Represents transmission of packet P



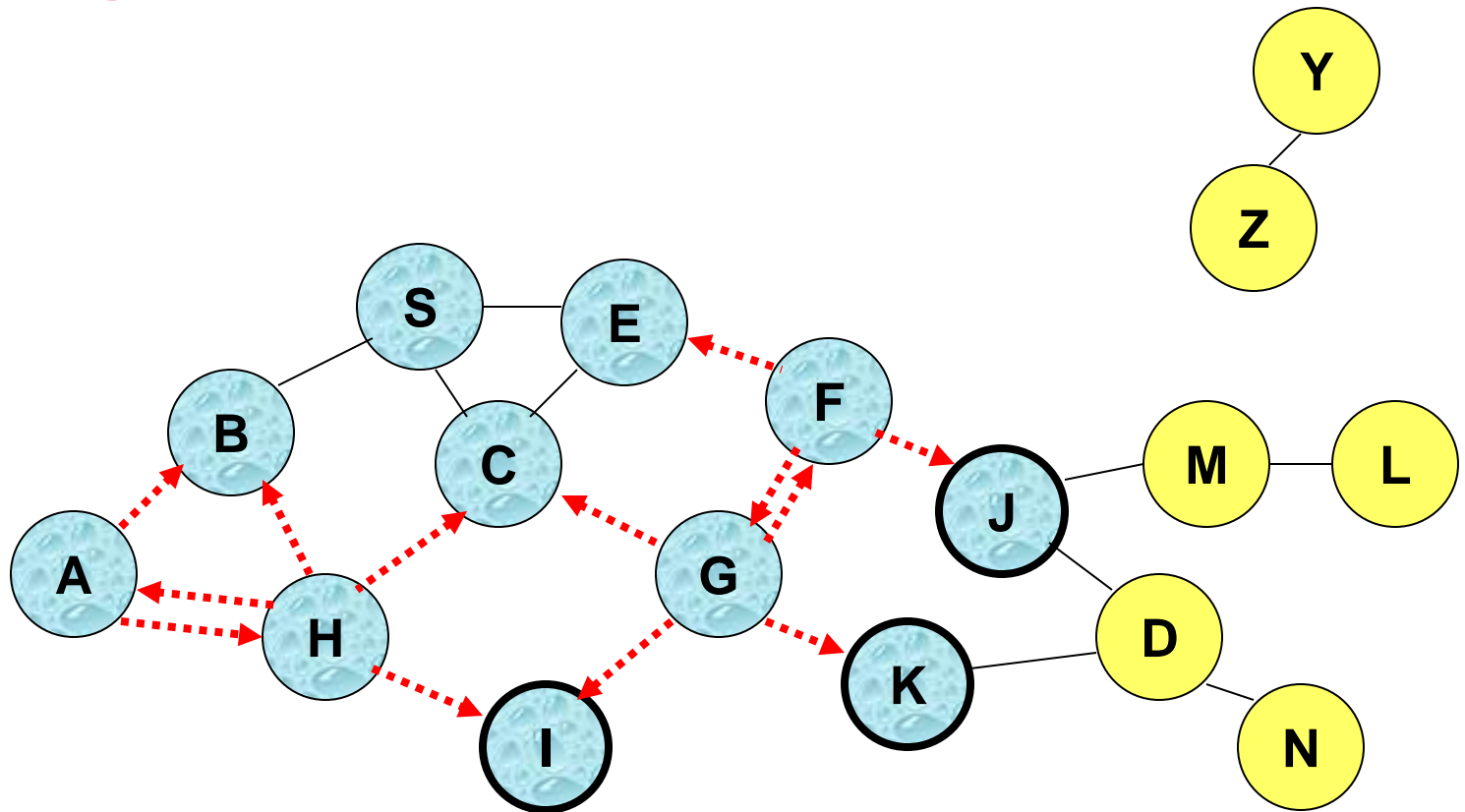
Flooding for Data Delivery



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



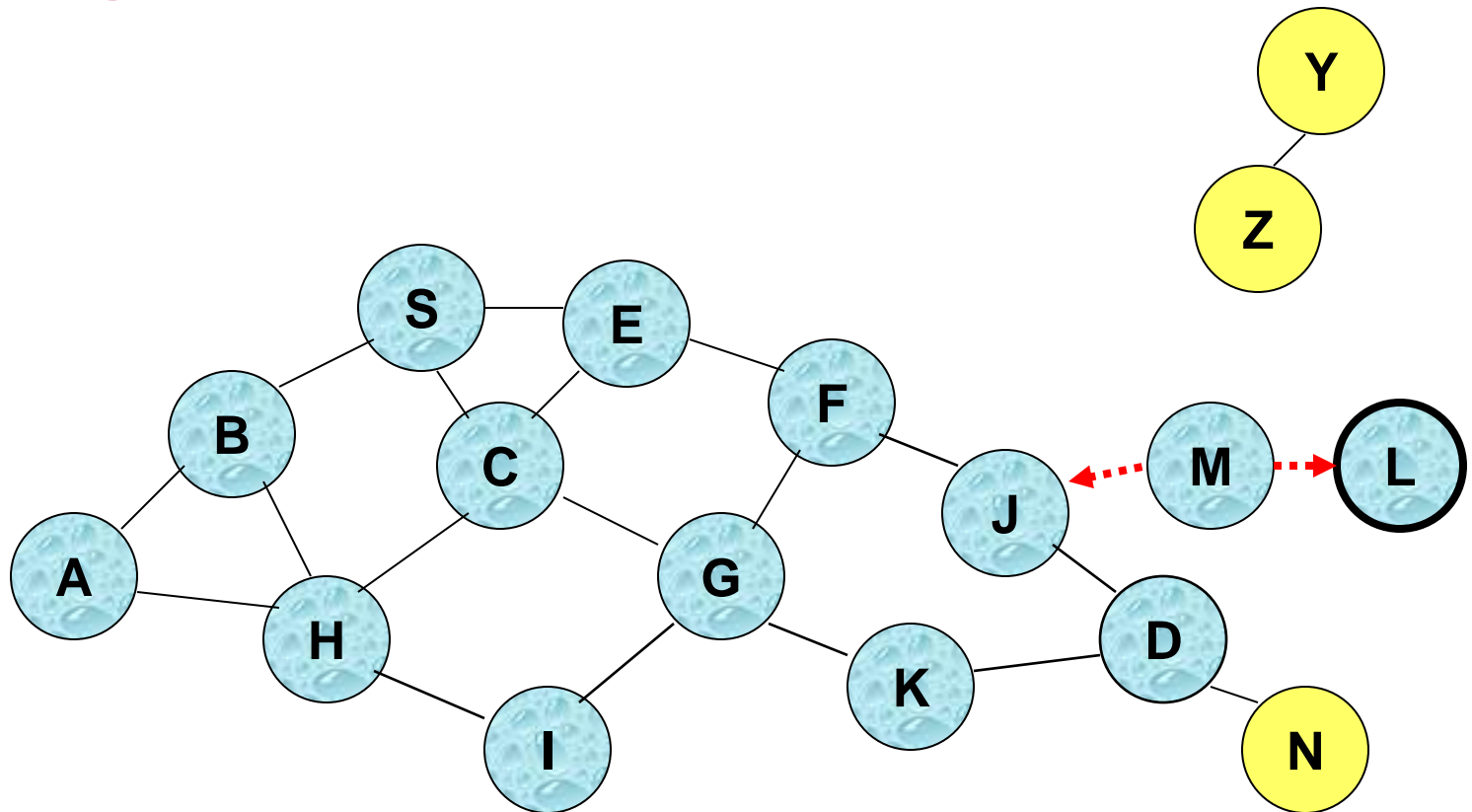
Flooding for Data Delivery



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



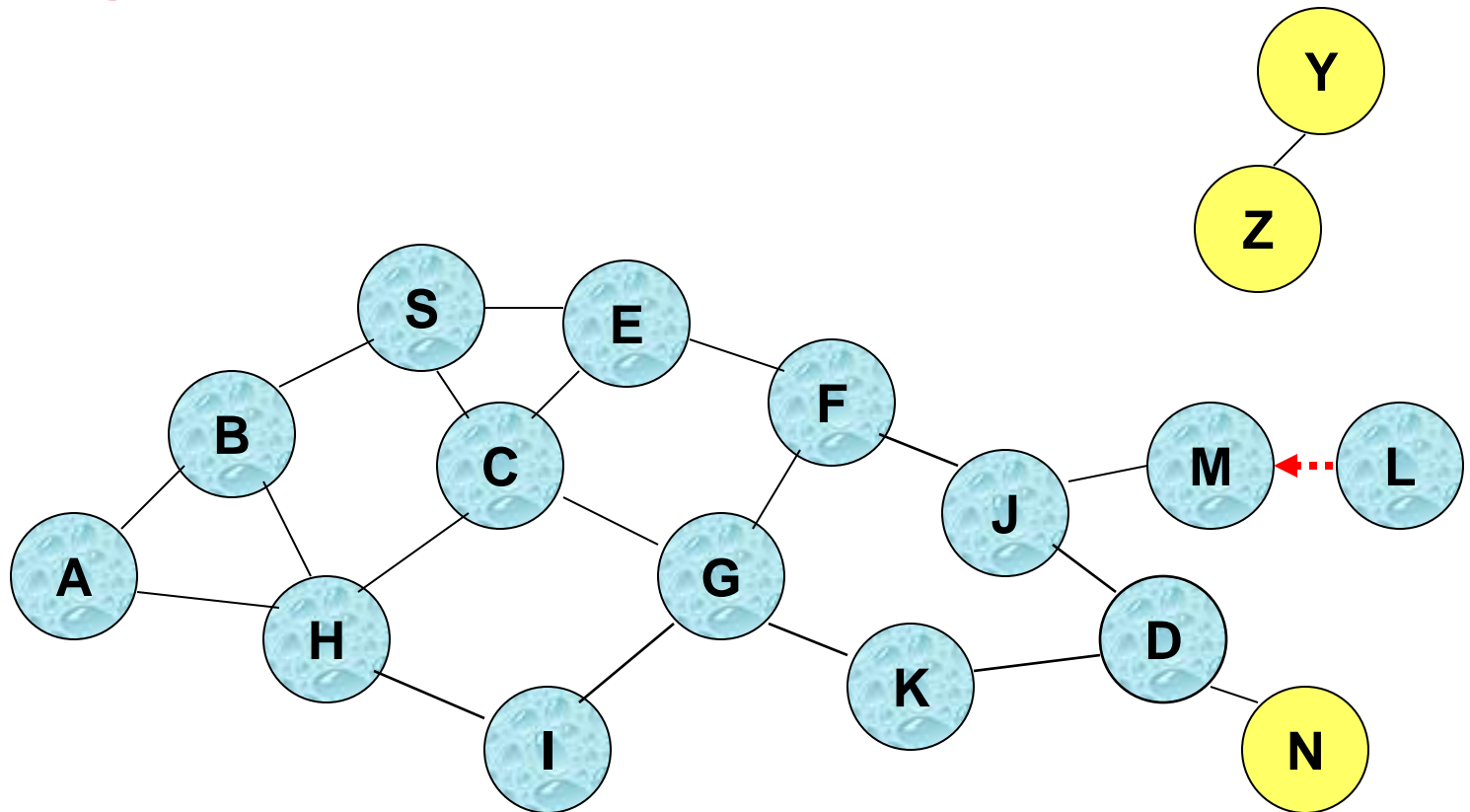
Flooding for Data Delivery



- Node D **does not forward** packet P, because node D is the **intended destination of packet P**



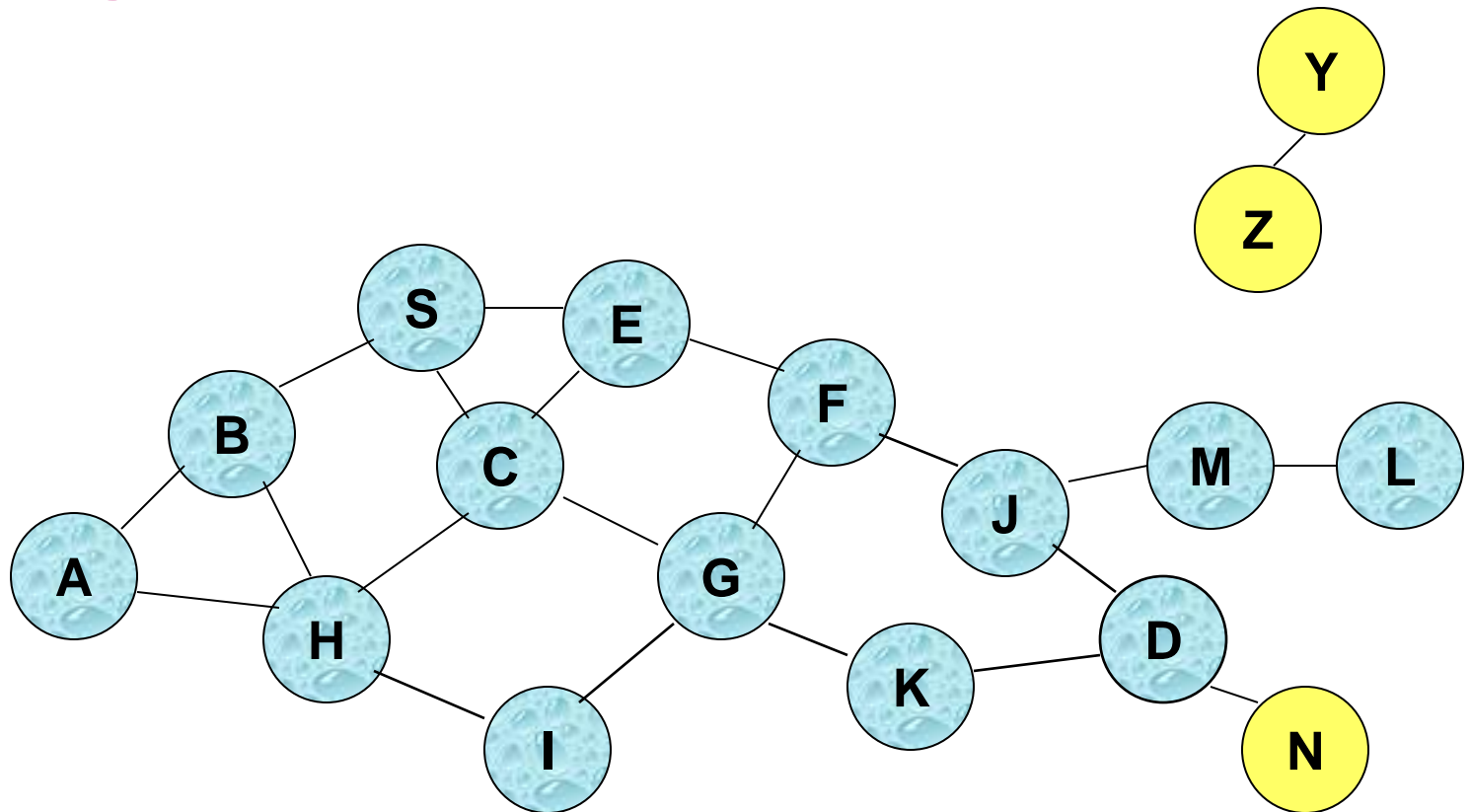
Flooding for Data Delivery



- Flooding completed
- Nodes **unreachable** from S do not receive packet P (e.g., node Z)
- Nodes for which all paths from S go through the destination D also do not receive packet P (example: node N)



Flooding for Data Delivery



- Flooding may deliver packets to too many nodes (in the **worst case**, all nodes reachable from sender may receive the packet)



Flooding for Data Delivery: Advantages

- ▶ Simplicity
- ▶ May be more efficient than other protocols when rate of information transmission is low enough that the overhead of explicit route discovery/maintenance incurred by other protocols is relatively higher
 - ▶ this scenario may occur, for instance, when nodes transmit **small data packets** relatively infrequently, and many topology **changes occur** between consecutive packet transmissions
- ▶ Potentially higher reliability of data delivery
 - ▶ Because packets may be delivered to the destination on **multiple paths**



Flooding for Data Delivery: Disadvantages

- ▶ Potentially, very high overhead
 - ▶ Data packets may be delivered to too many nodes who do not need to receive them
- ▶ Potentially lower reliability of data delivery
 - ▶ Flooding uses broadcasting -- hard to implement reliable broadcast delivery without significantly increasing overhead
 - ▶ Broadcasting in IEEE 802.11 MAC is unreliable
 - ▶ In our example, nodes J and K may transmit to node D simultaneously, resulting in loss of the packet
 - ▶ in this case, destination would not receive the packet at all



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



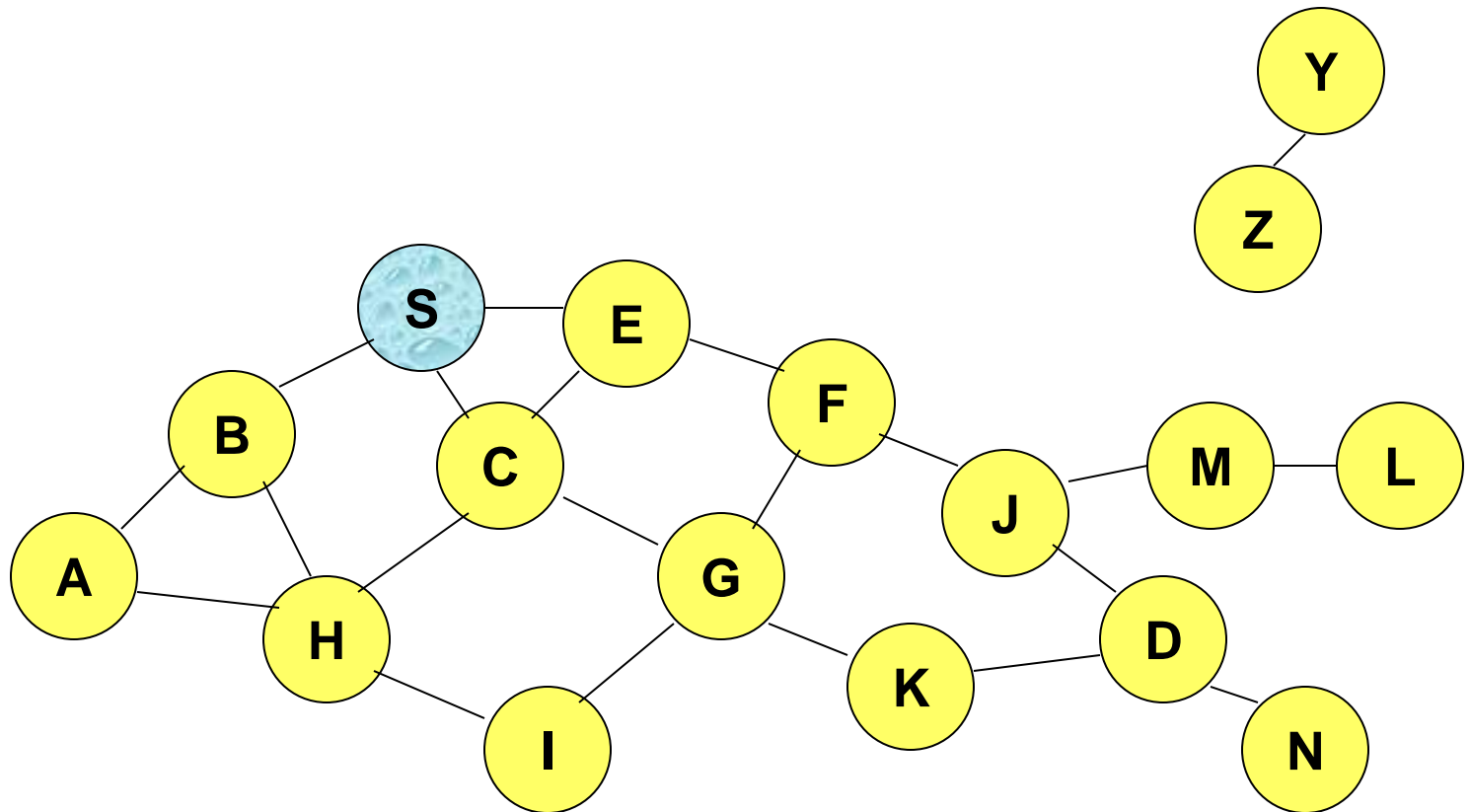
Dynamic Source Routing (DSR)

[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

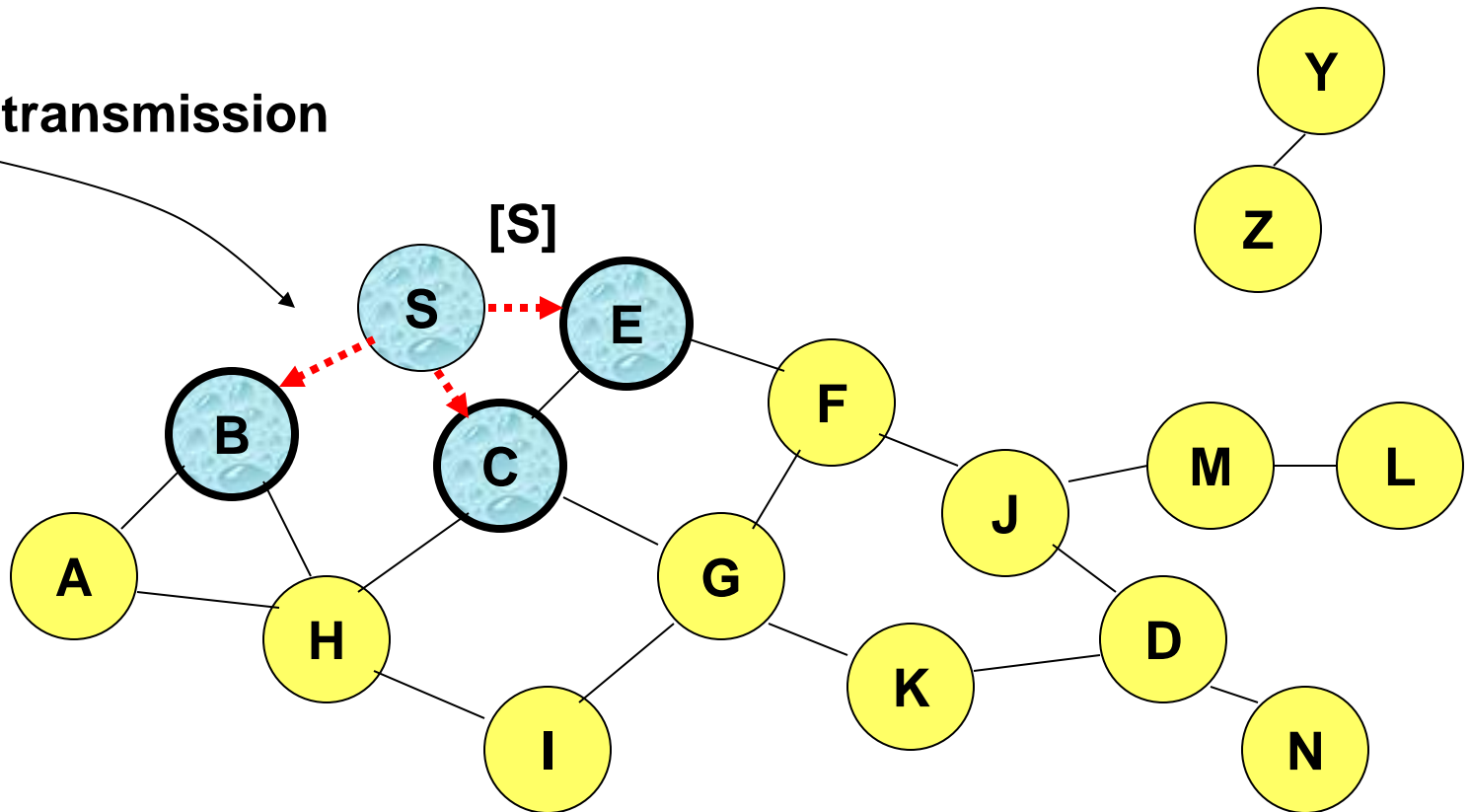


Represents a node that has received RREQ for D from S



Route Discovery in DSR

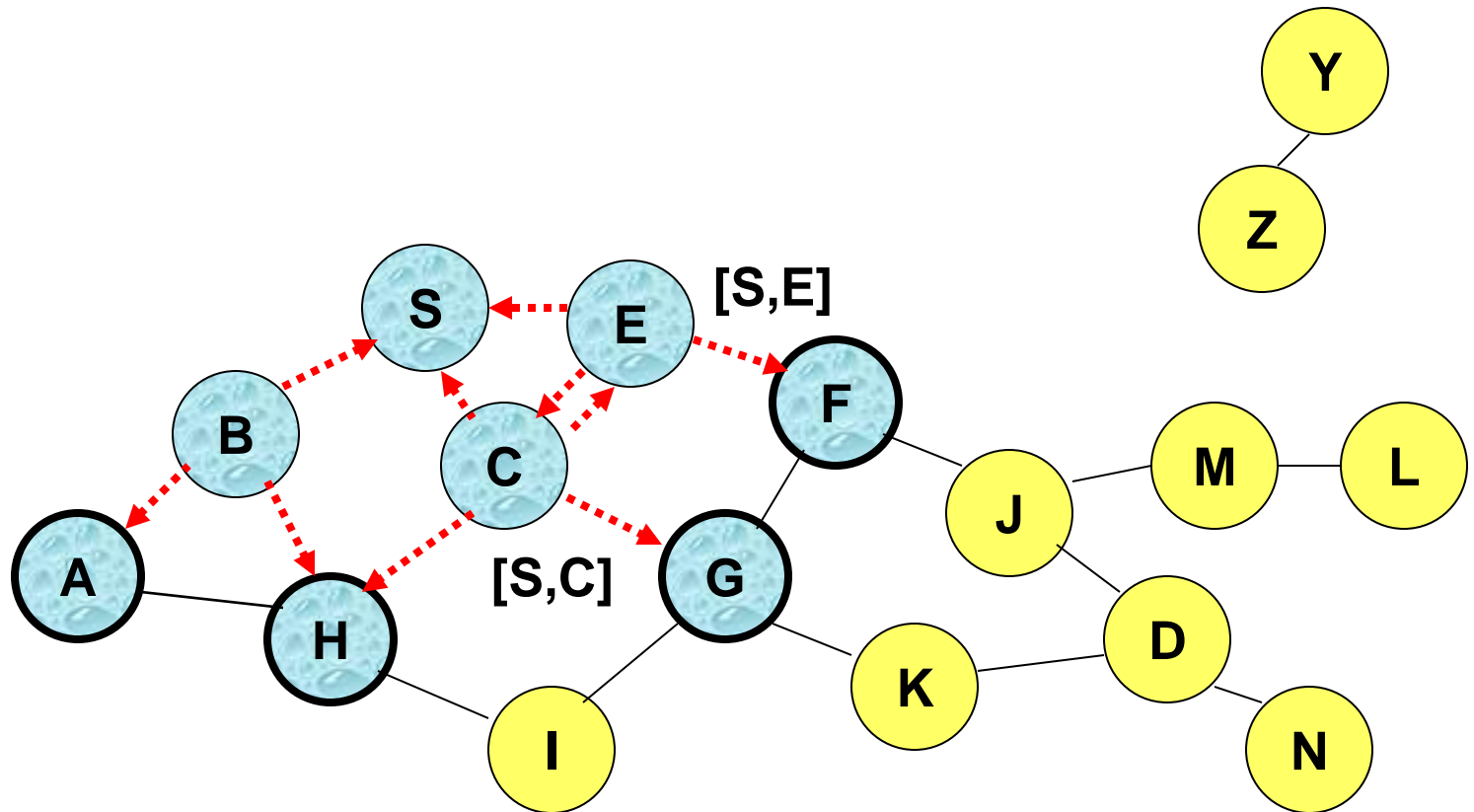
Broadcast transmission



.....→ Represents transmission of RREQ

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

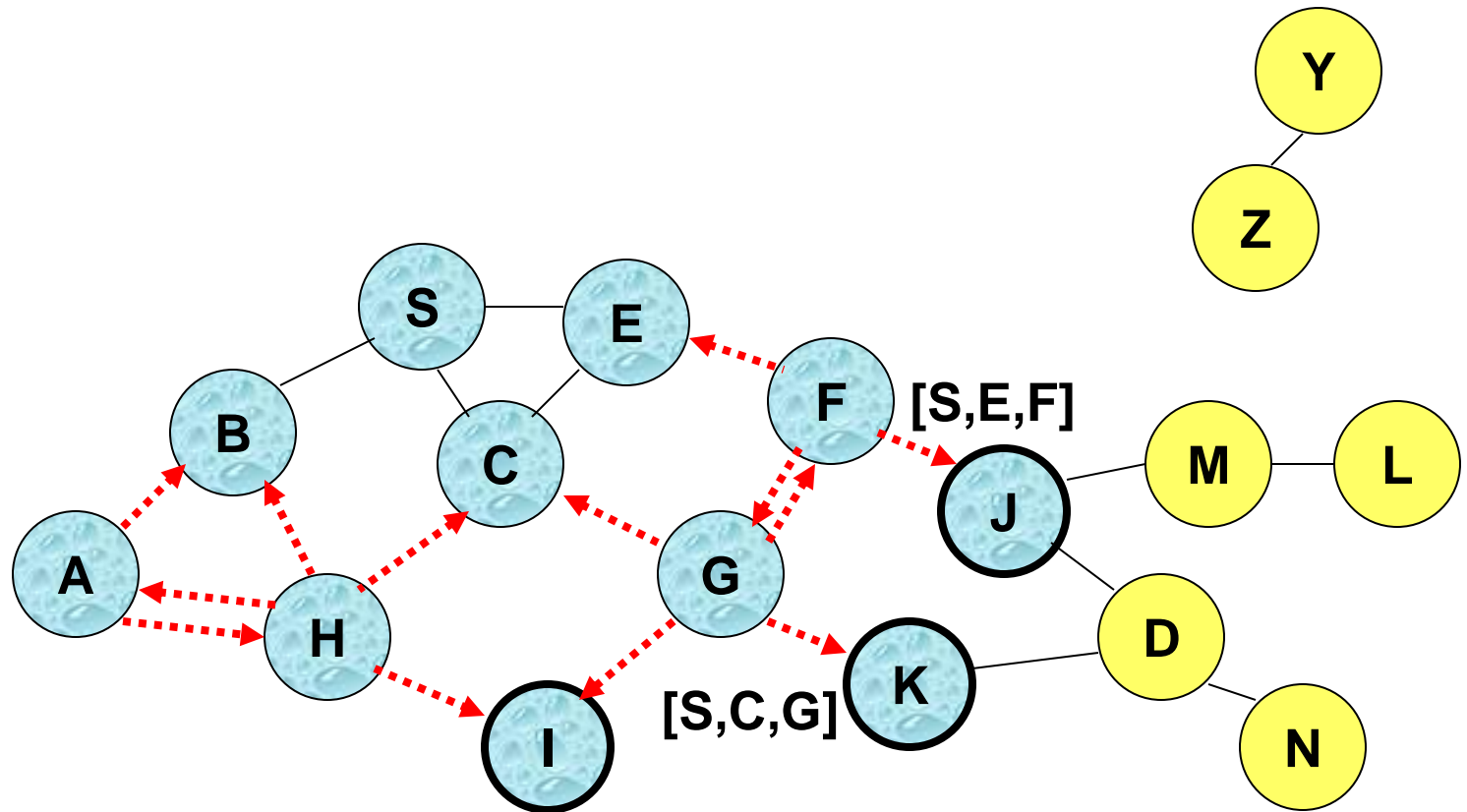
Route Discovery in DSR



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



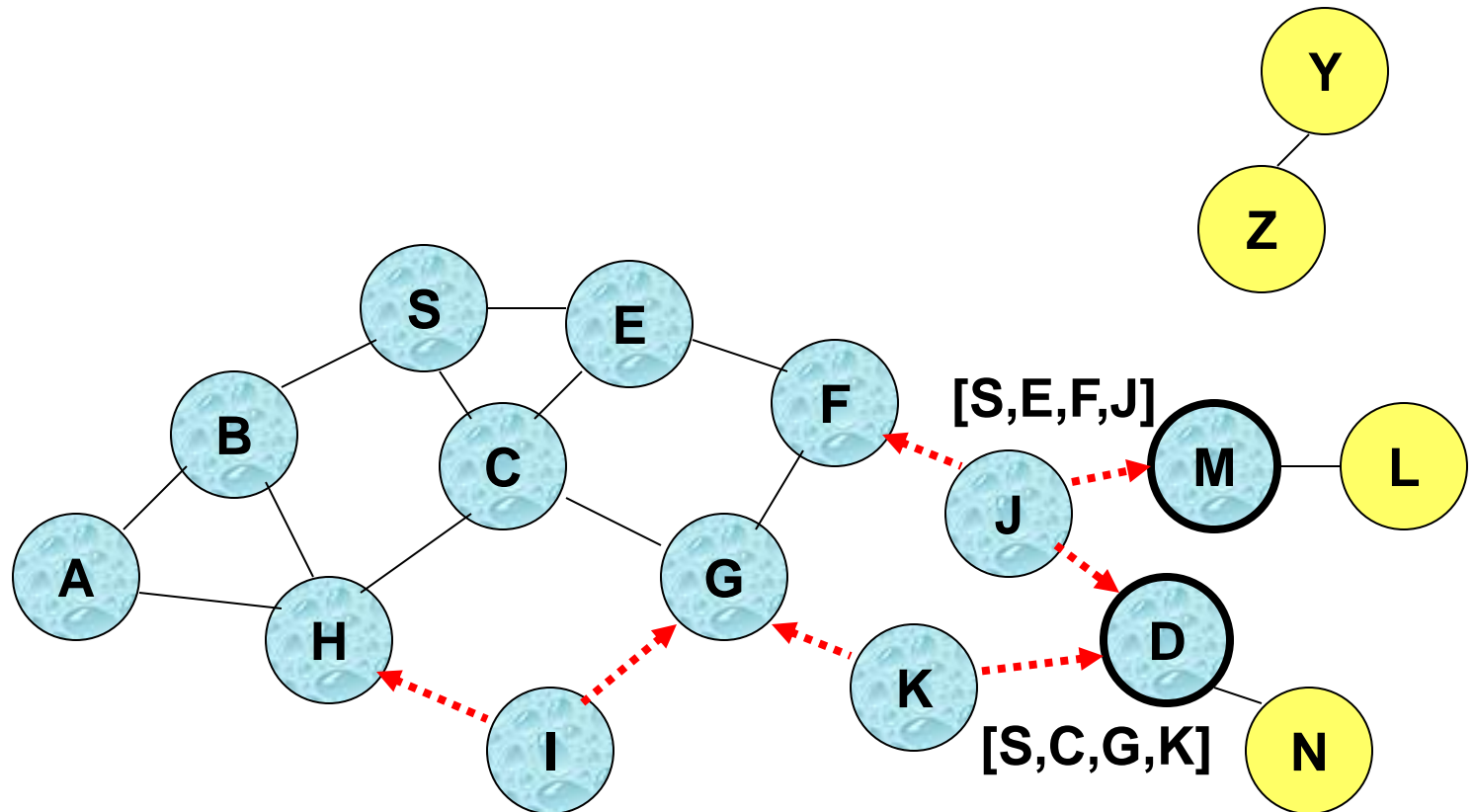
Route Discovery in DSR



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



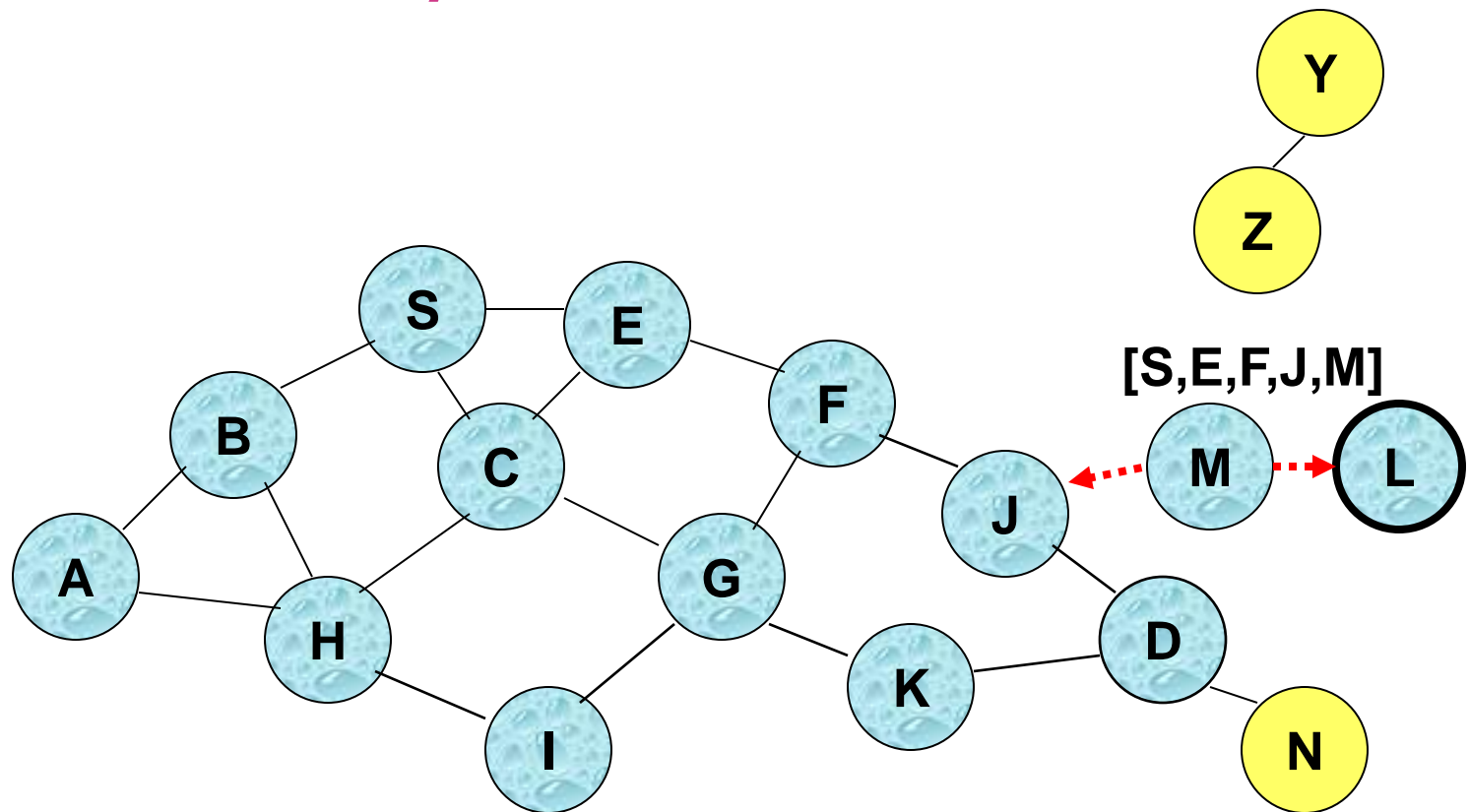
Route Discovery in DSR



- 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

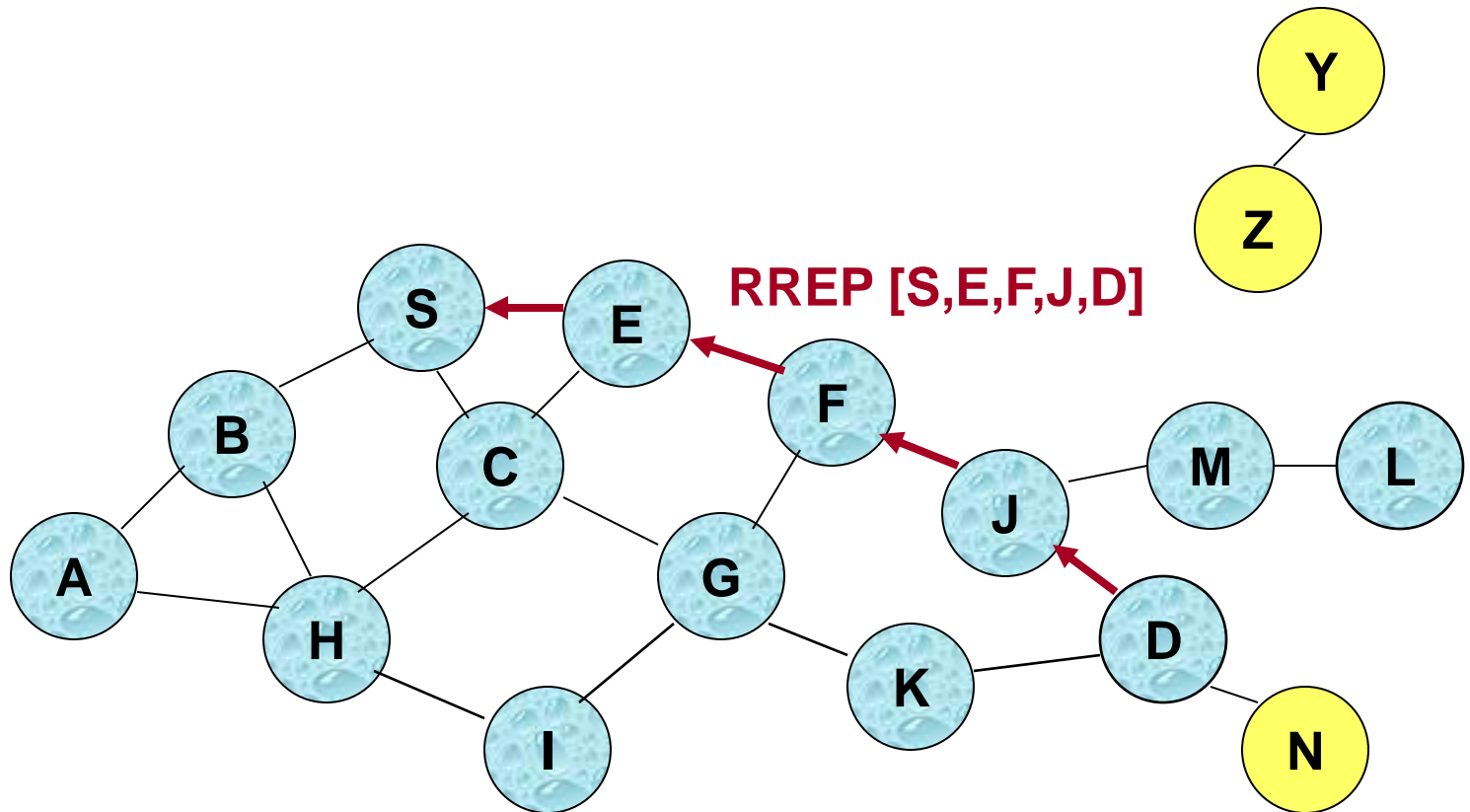


Route Discovery in DSR

- ▶ 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



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)

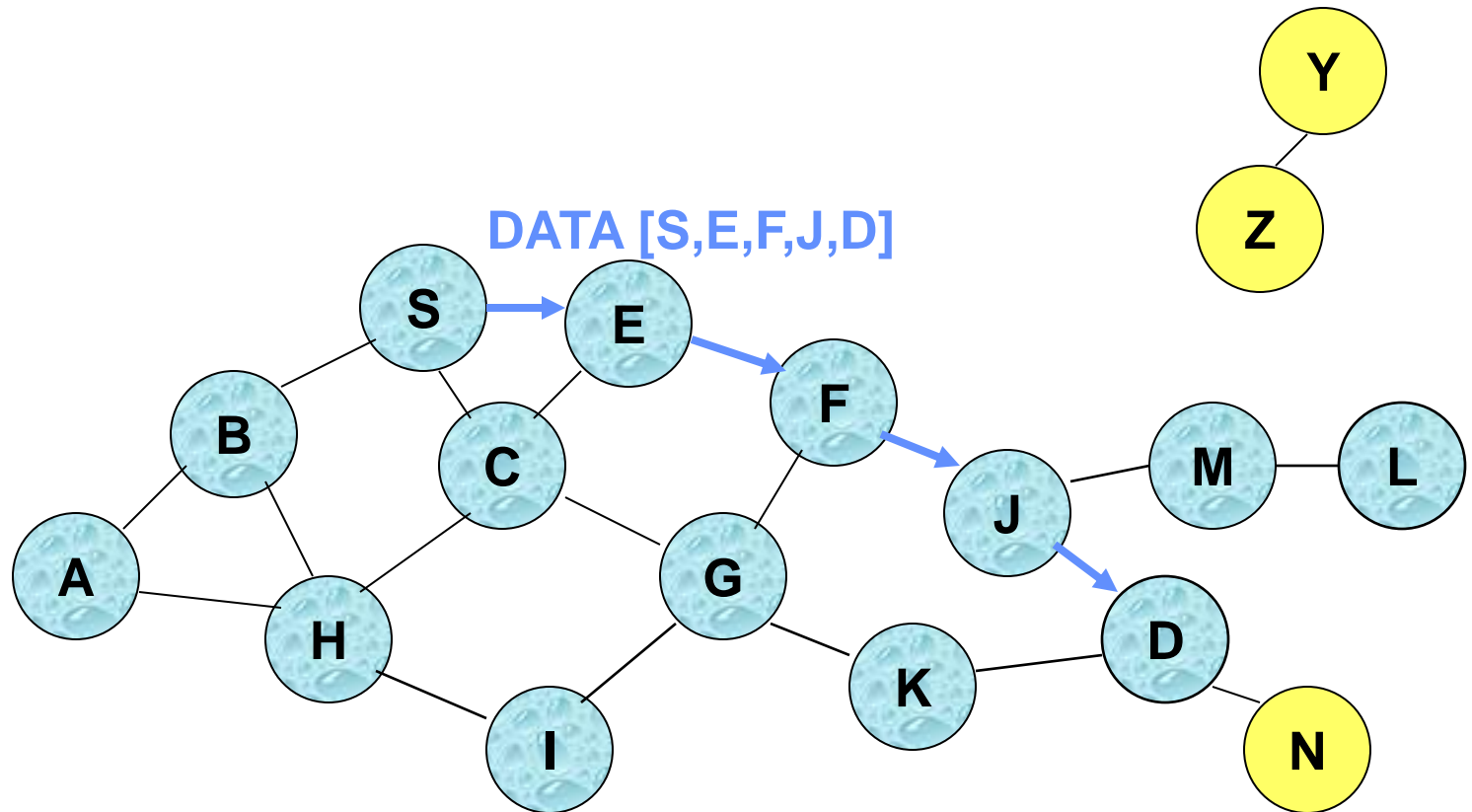


Dynamic Source Routing (DSR)

- ▶ 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



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



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

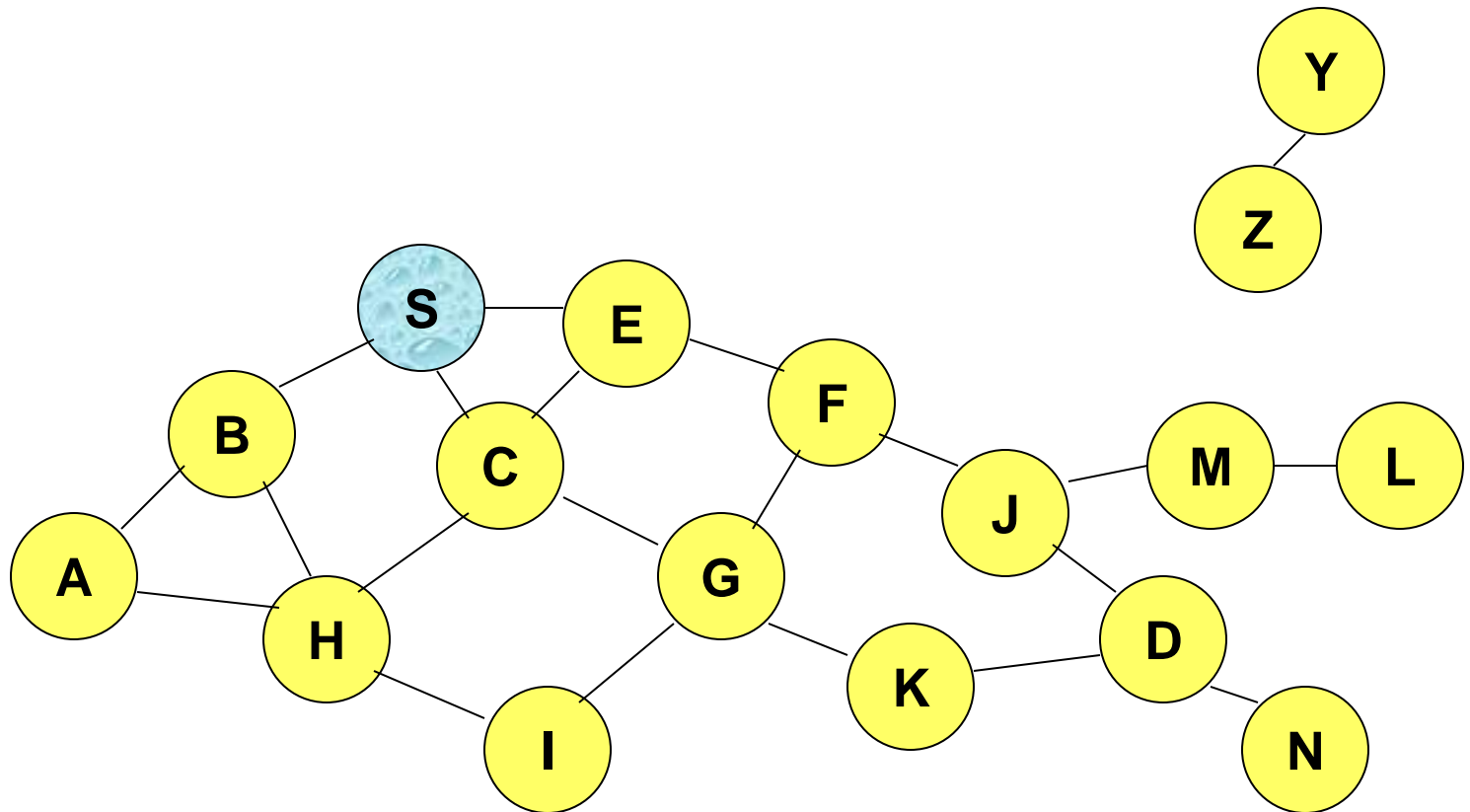


AODV

- 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

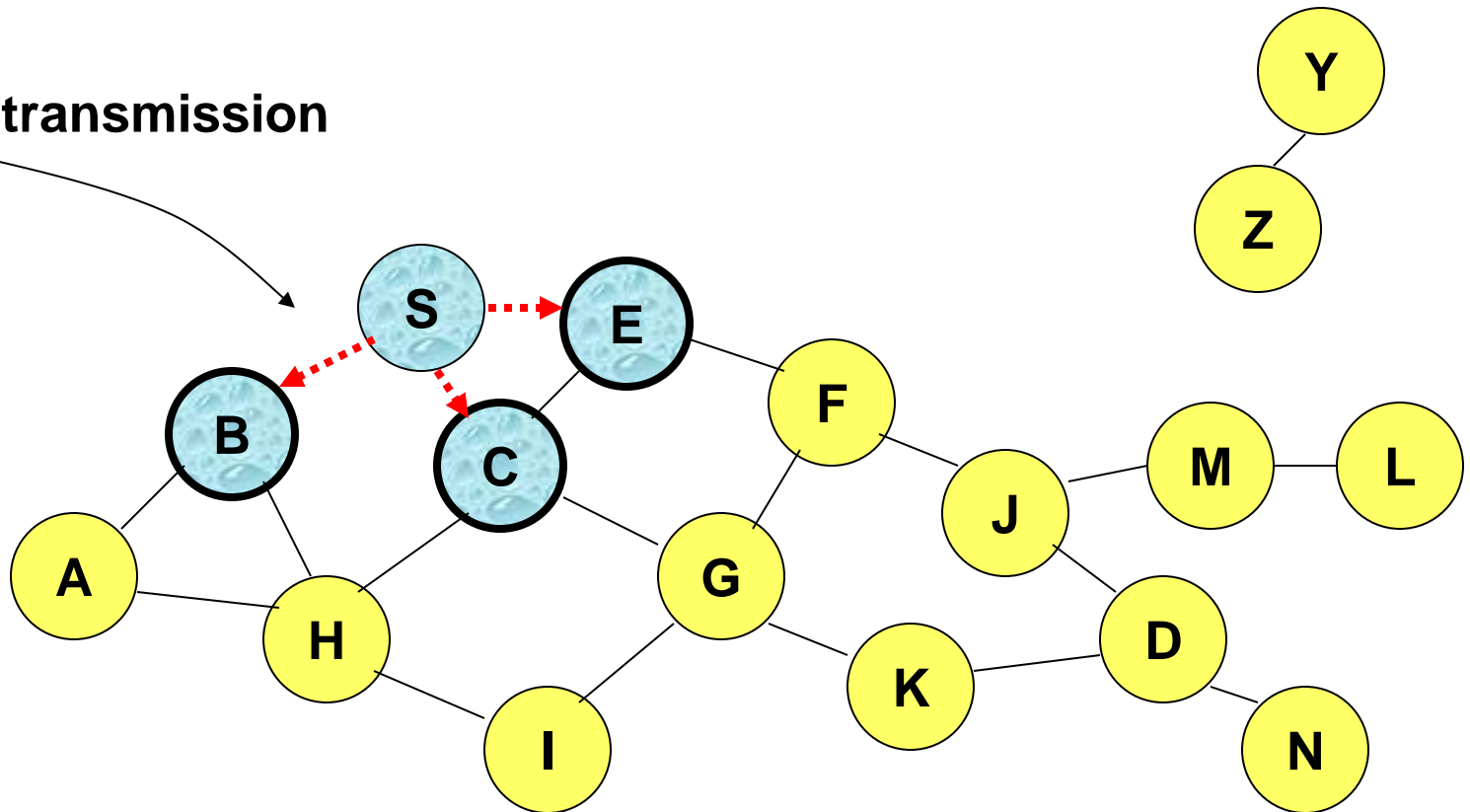


Represents a node that has received RREQ for D from S



Route Requests in AODV

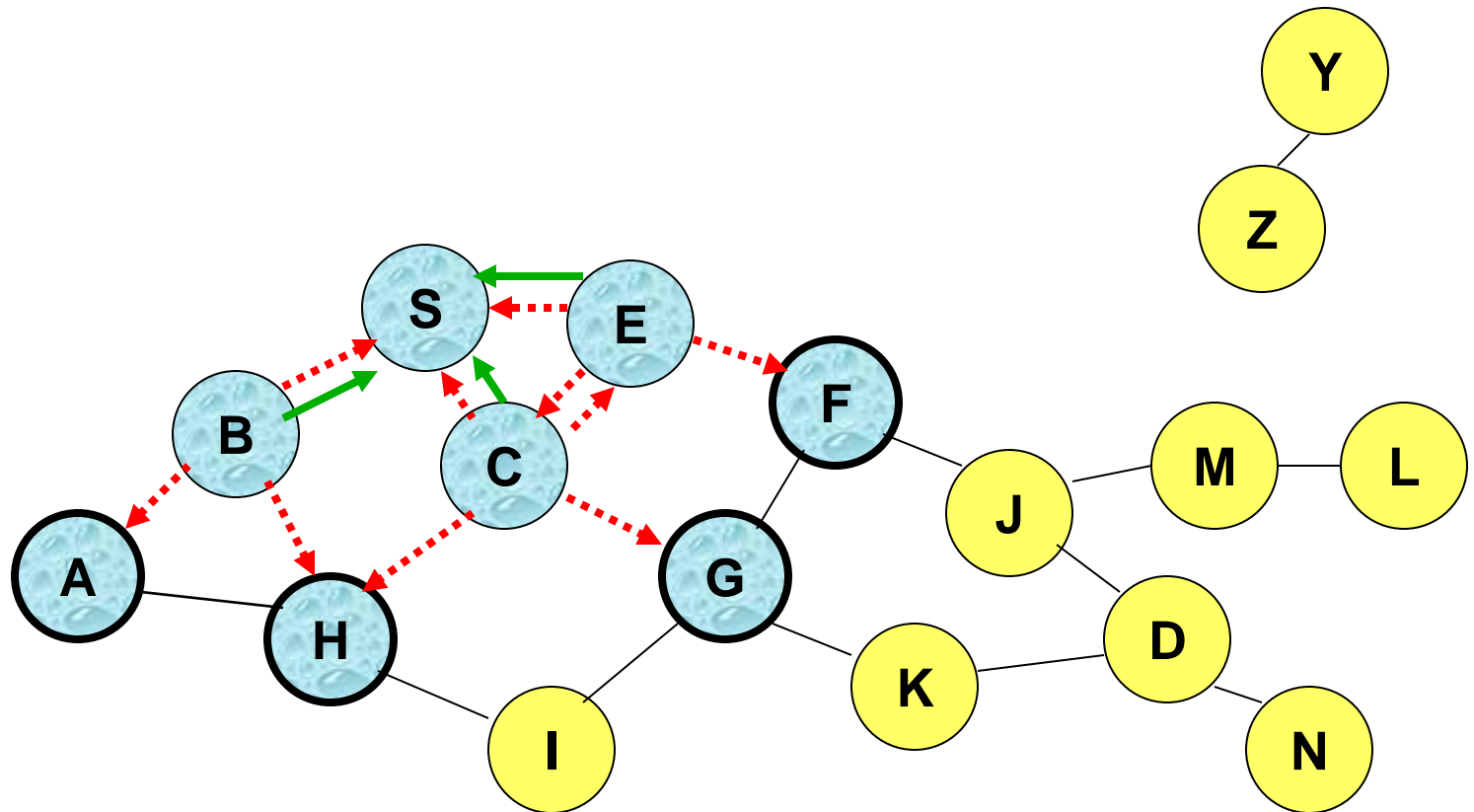
Broadcast transmission



.....→ Represents transmission of RREQ



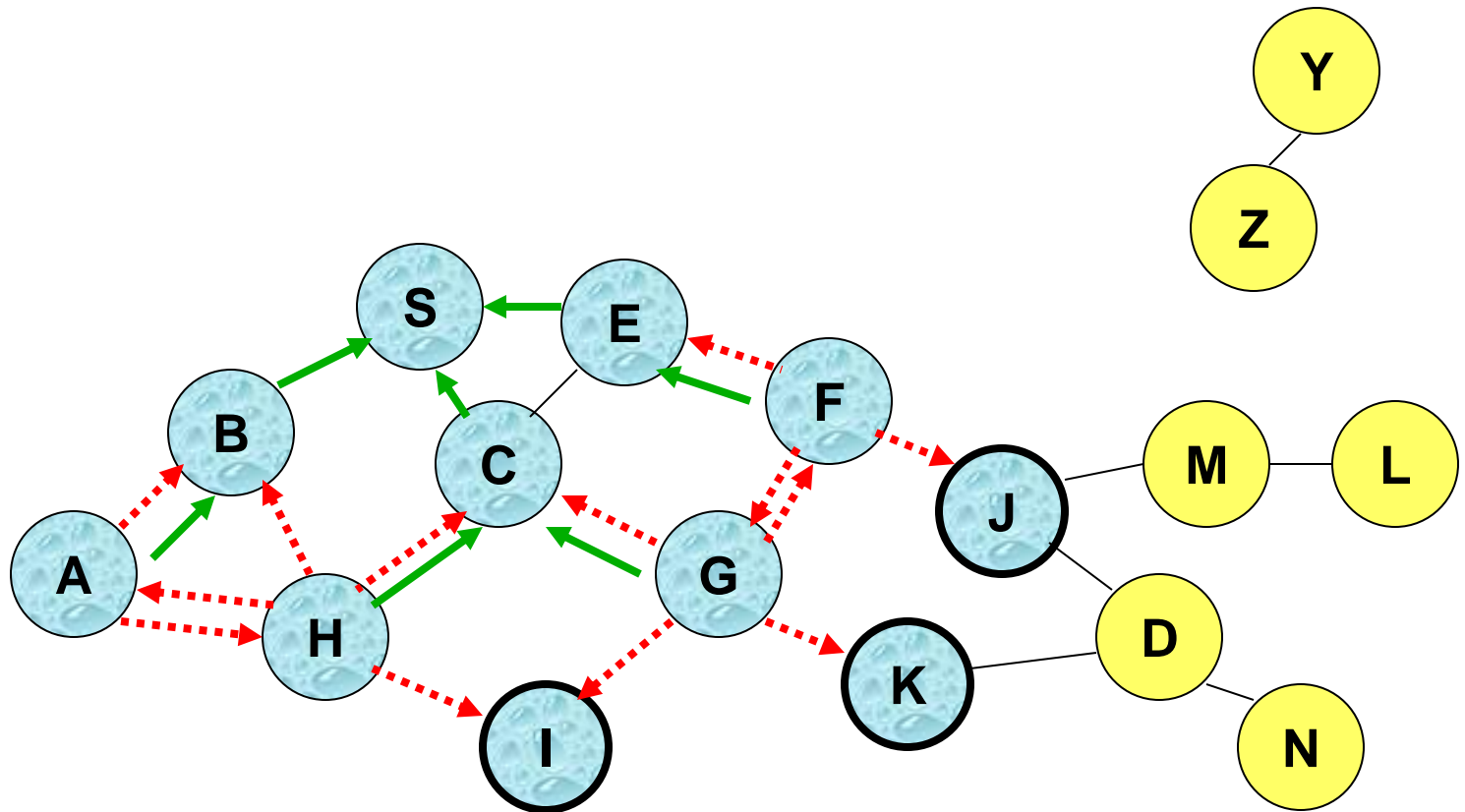
Route Requests in AODV



← Represents links on Reverse Path



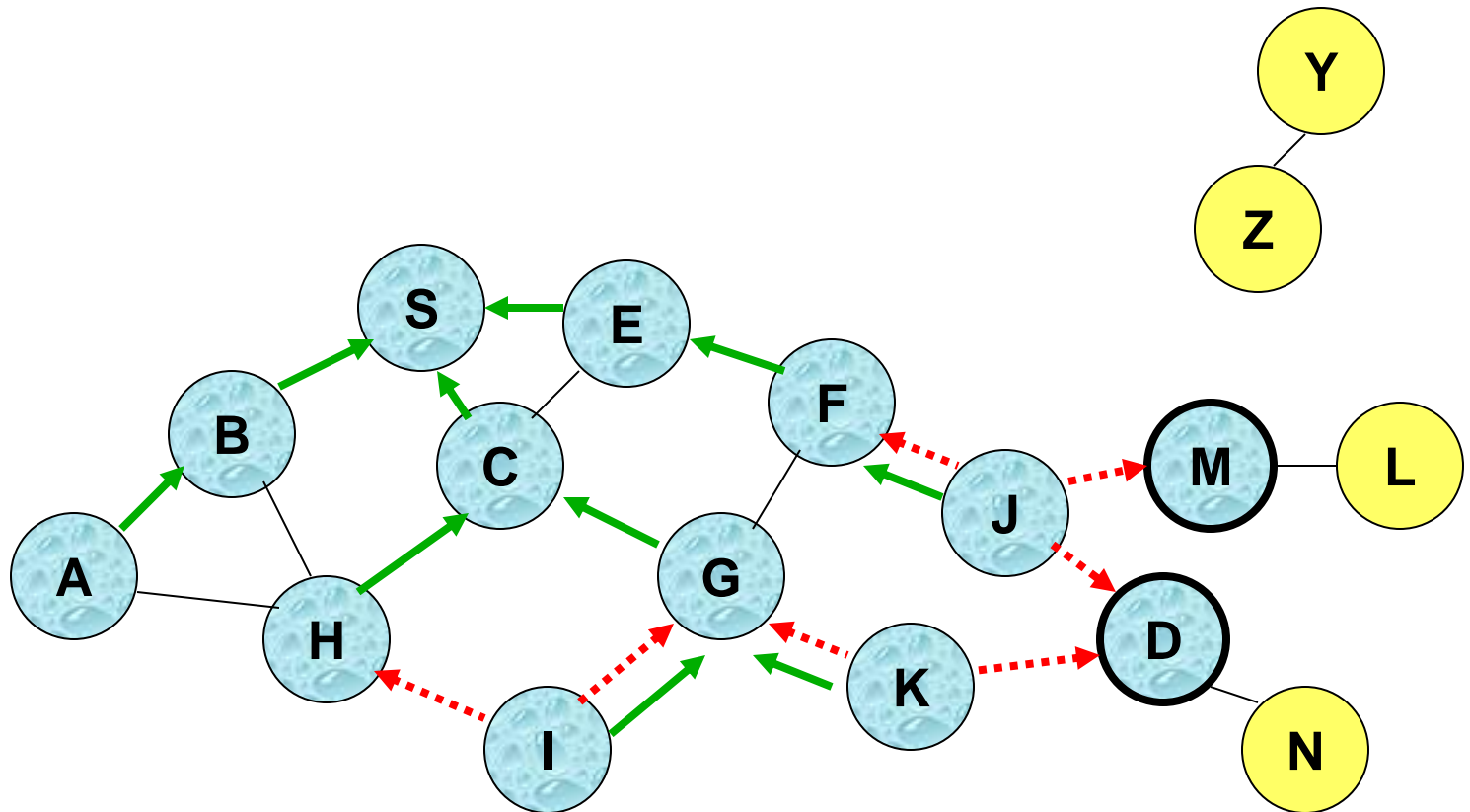
Reverse Path Setup in AODV



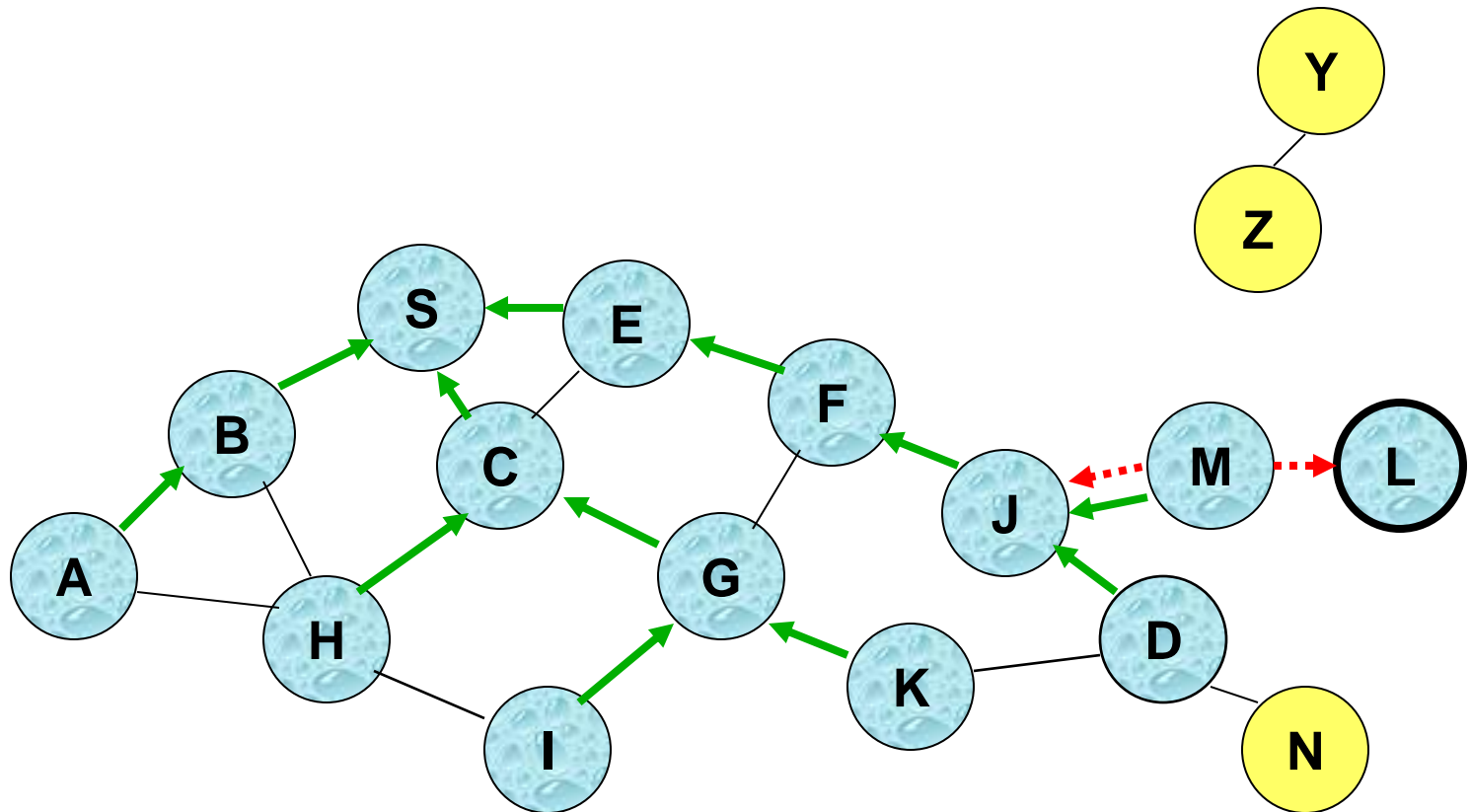
- **Node C** receives RREQ from G and H, but does not forward it again, because node C has **already forwarded RREQ once**



Reverse Path Setup in AODV



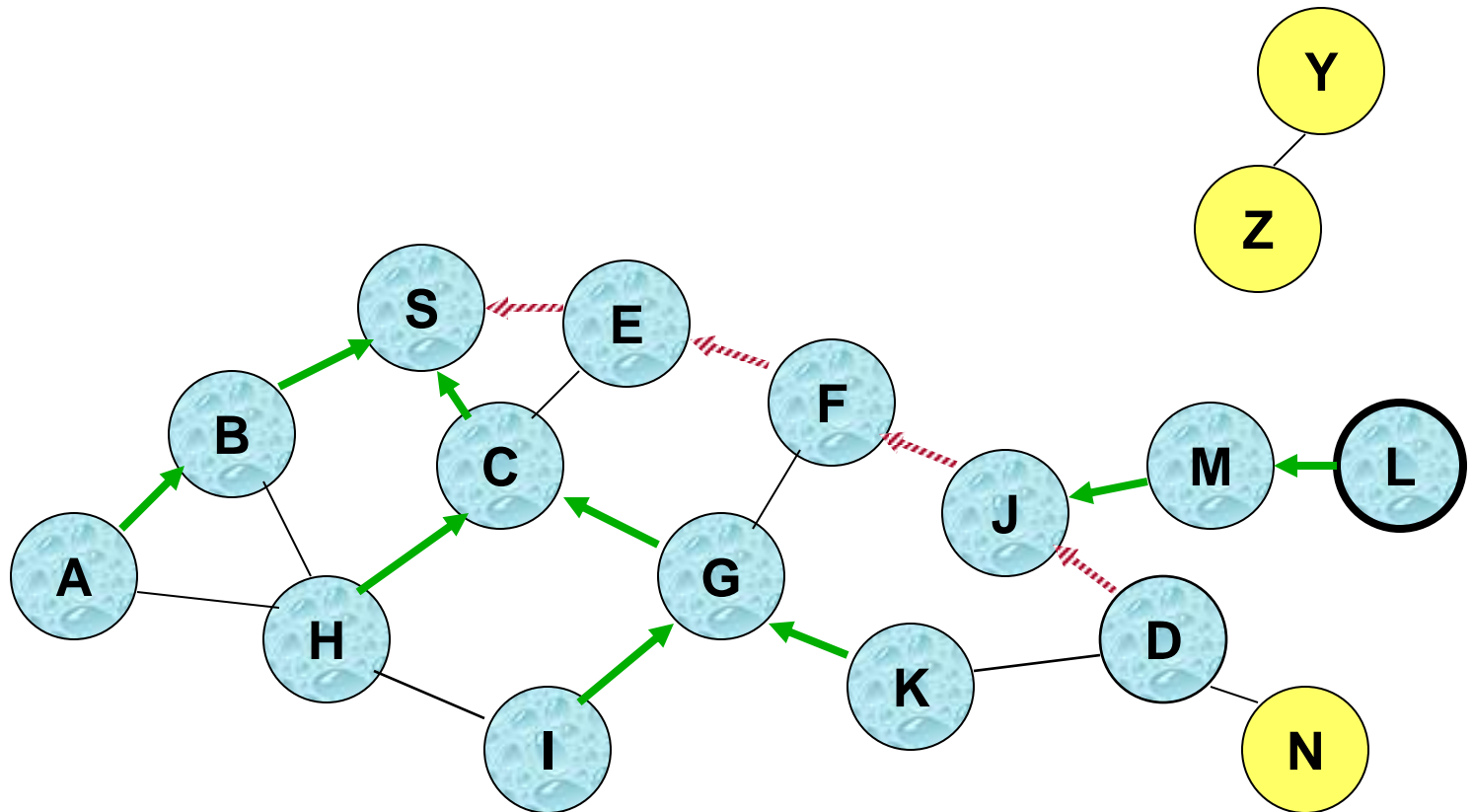
Reverse Path Setup in AODV



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



Route Reply in AODV



 Represents links on path taken by RREP

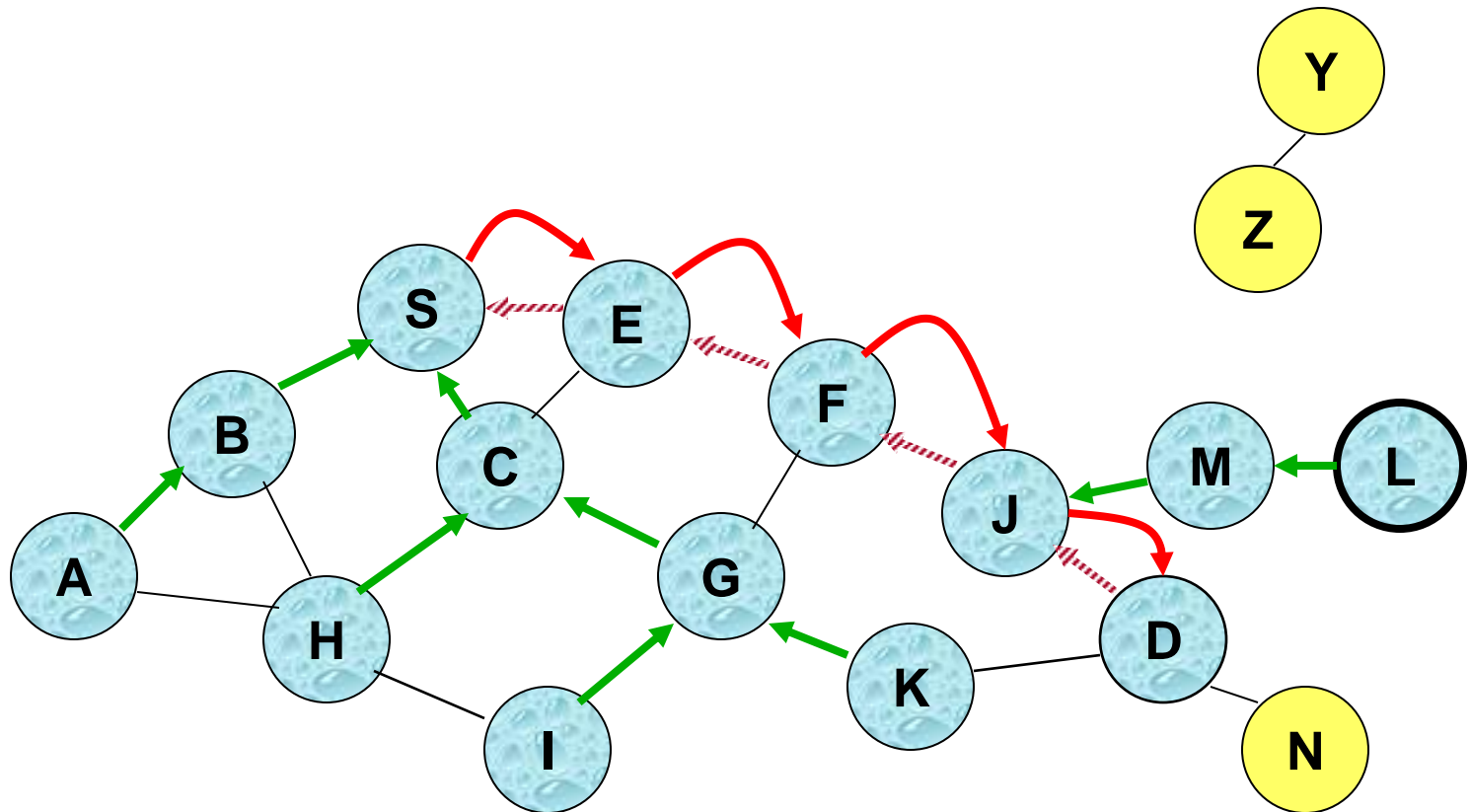


Route Reply in AODV

- ▶ 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 is 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



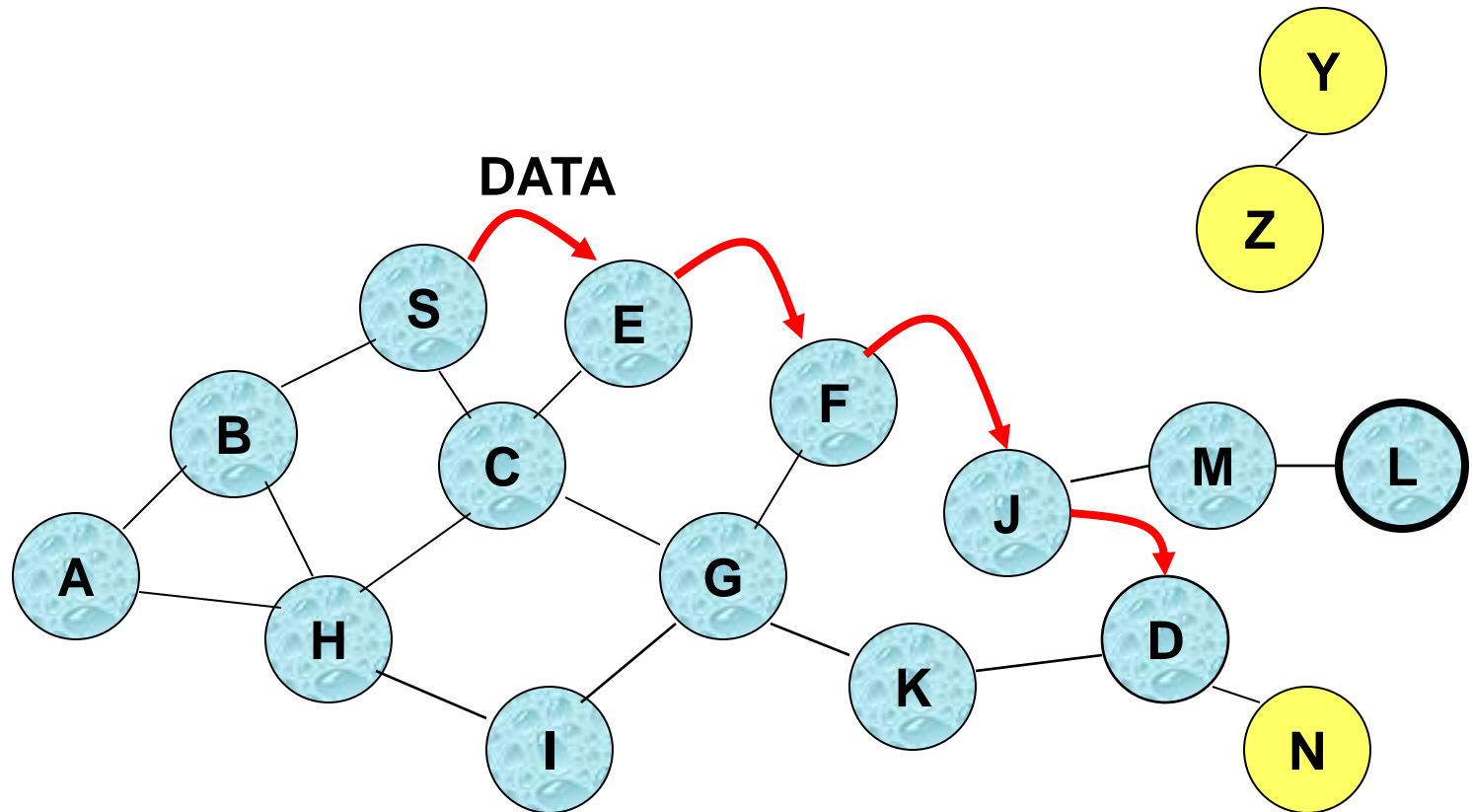
Forward links are setup when RREP travels along the reverse path



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.



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



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



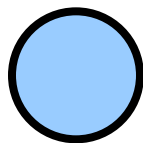
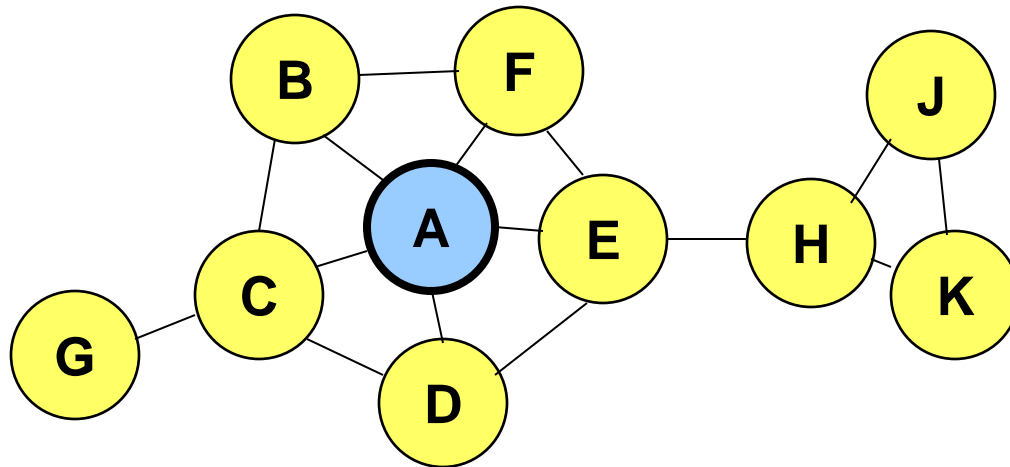
Optimized Link State Routing (OLSR)

- ▶ 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*
- ▶ 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

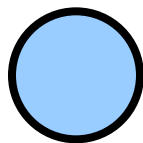
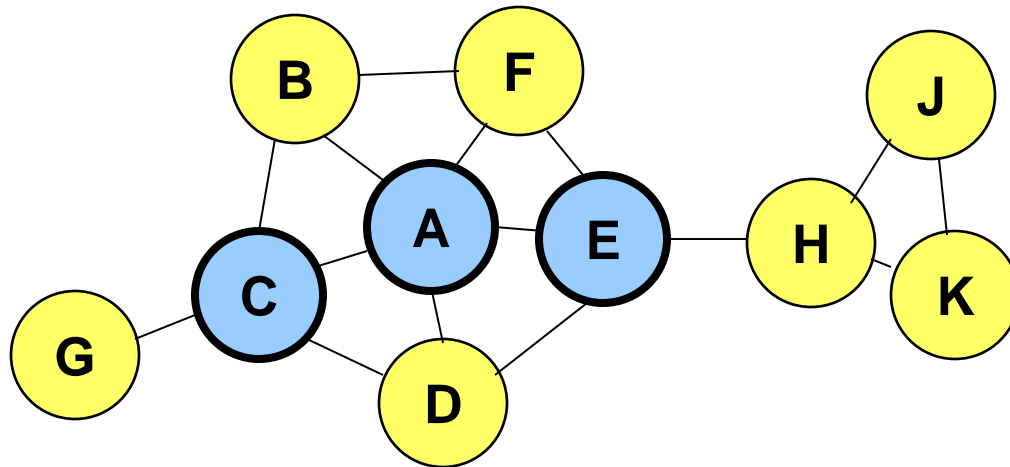


Node that has broadcast state information from A



Optimized Link State Routing (OLSR)

- Nodes C and E forward information received from A

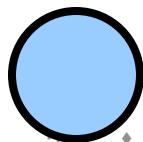
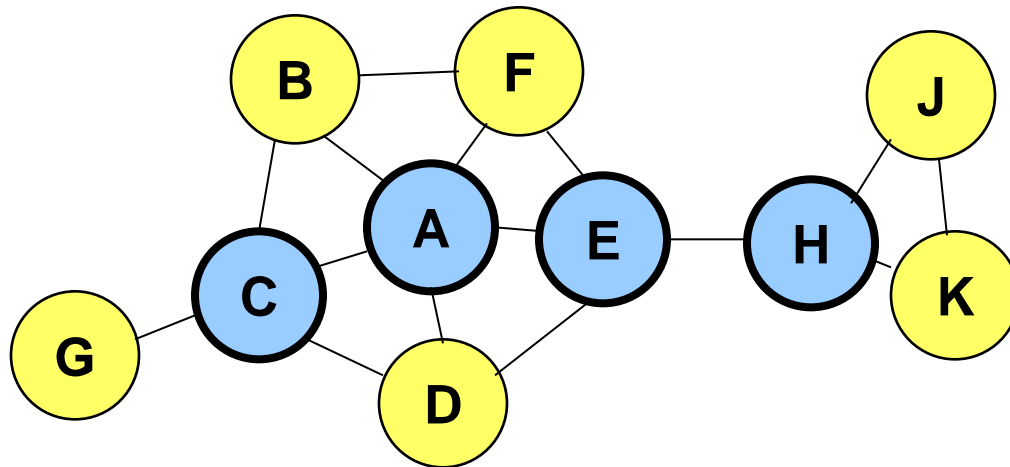


Node that has broadcast state information from A



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



Node that has broadcast state information from A



Summary: OLSR

- ▶ OLSR floods information through the multipoint relays
- ▶ The flooded information itself is for links connecting nodes to respective multipoint relays
- ▶ Nodes need to calculate routes (shortest path trees) based on link-state knowledge, typically using the Dijkstra algorithm
- ▶ Routes used by OLSR only include multipoint relays as intermediate nodes



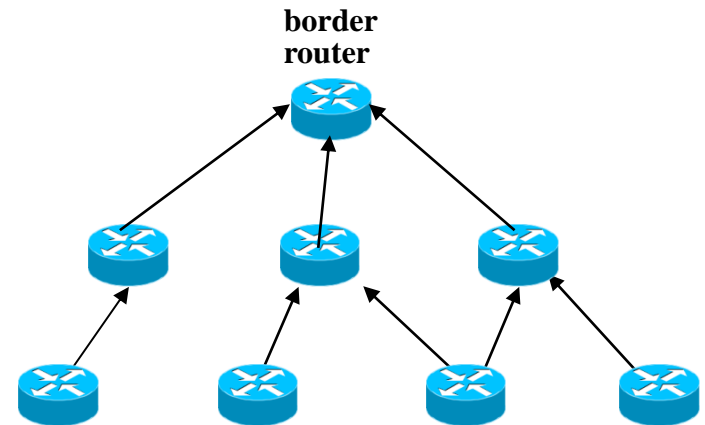
Further Routing Approaches

- ▶ Improvements & Optimisations of Previous Protocols
- ▶ Location Aided Routing
- ▶ Clustering after Landmarking
- ▶ Hierarchic / Anchored Routing
- ▶ Power-Aware Routing
- ▶ ...

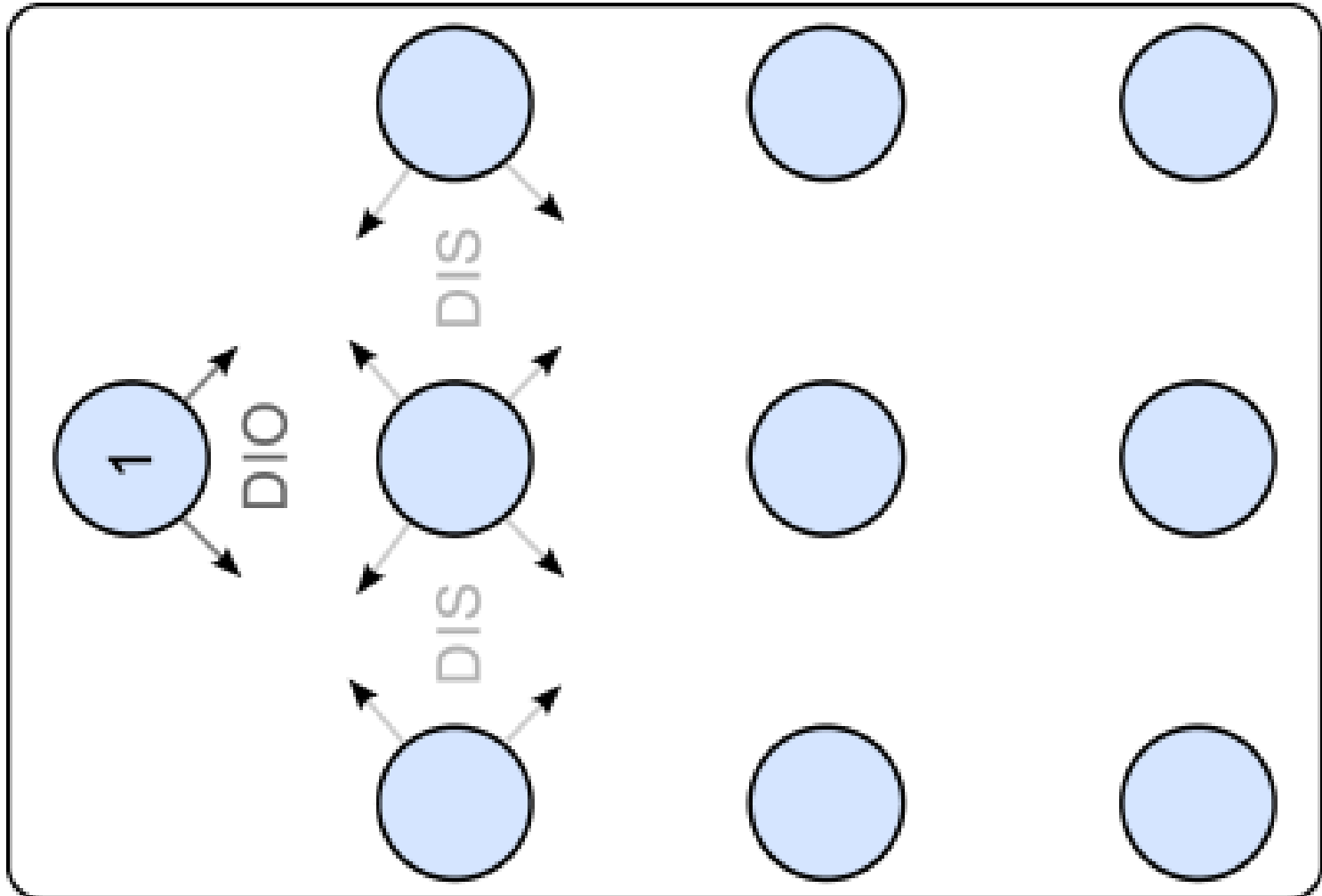


RPL - Routing Protocol for Low Power and Lossy Networks (LLN) – RFC 6550

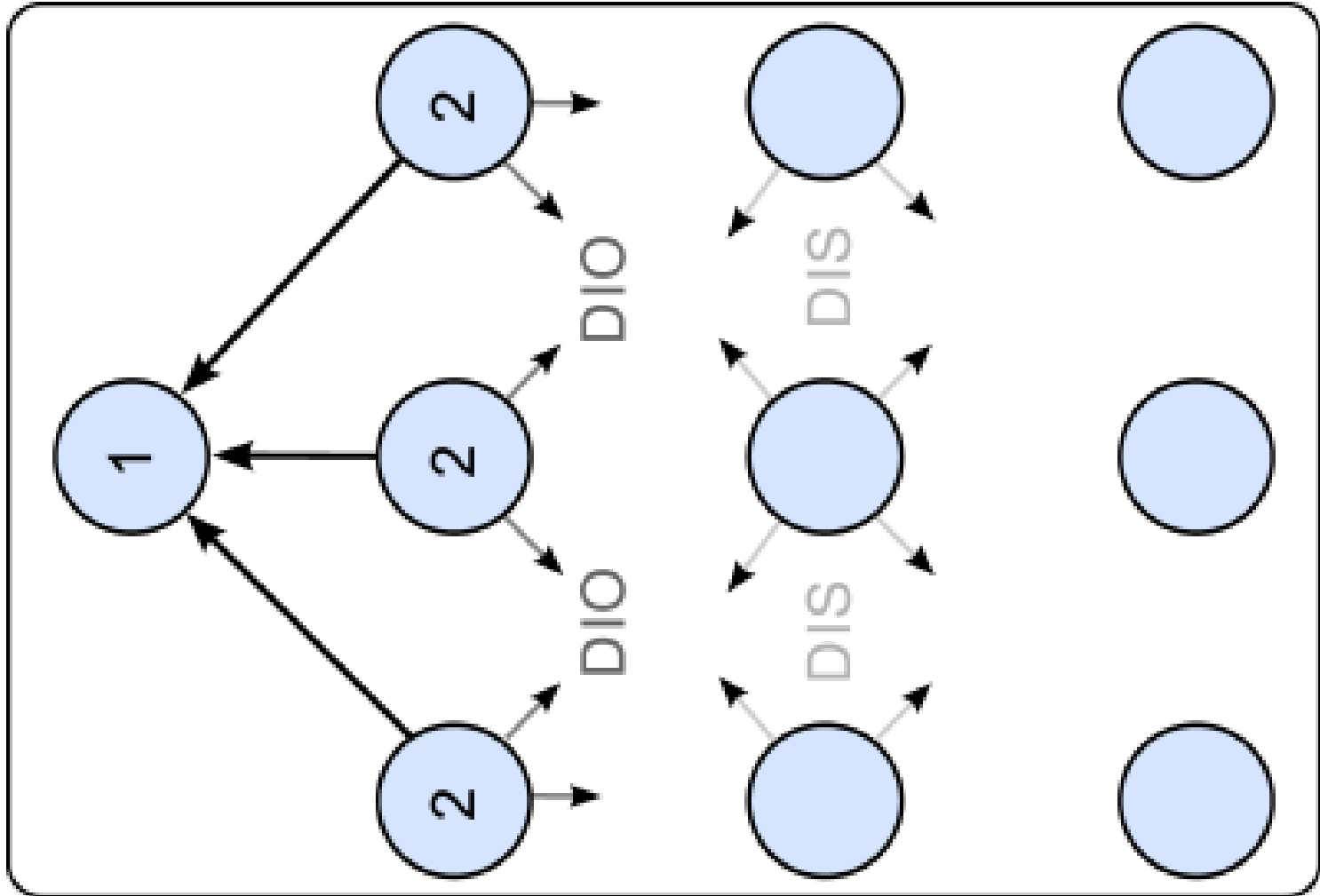
- ▶ Optimized for low-energy networks (without mobility)
- ▶ Builds Directed Acyclic Graphs (DAGs)
- ▶ Routing state propagation
 - ▶ Conventional:
 - ▶ Link-state: scoped flooding
 - ▶ Distance-vector: periodic routing beacons
 - ▶ Trickle:
 - ▶ adaptive exchange rate
- ▶ Spatial diversity
 - ▶ A router maintains multiple potential parents
- ▶ Expressive link metrics
 - ▶ ETX: Estimated Number of Transmissions



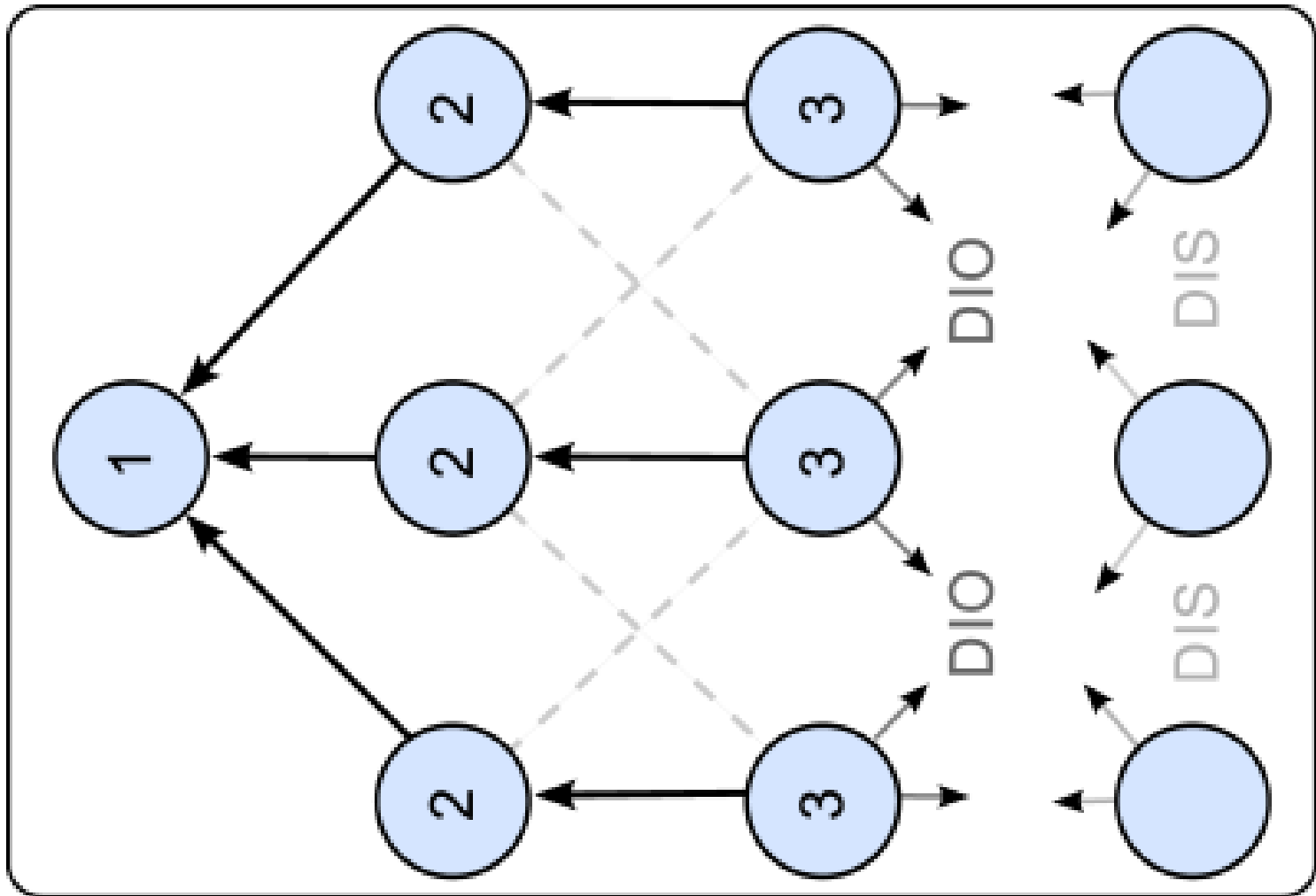
RPL Topology Creation - Upward



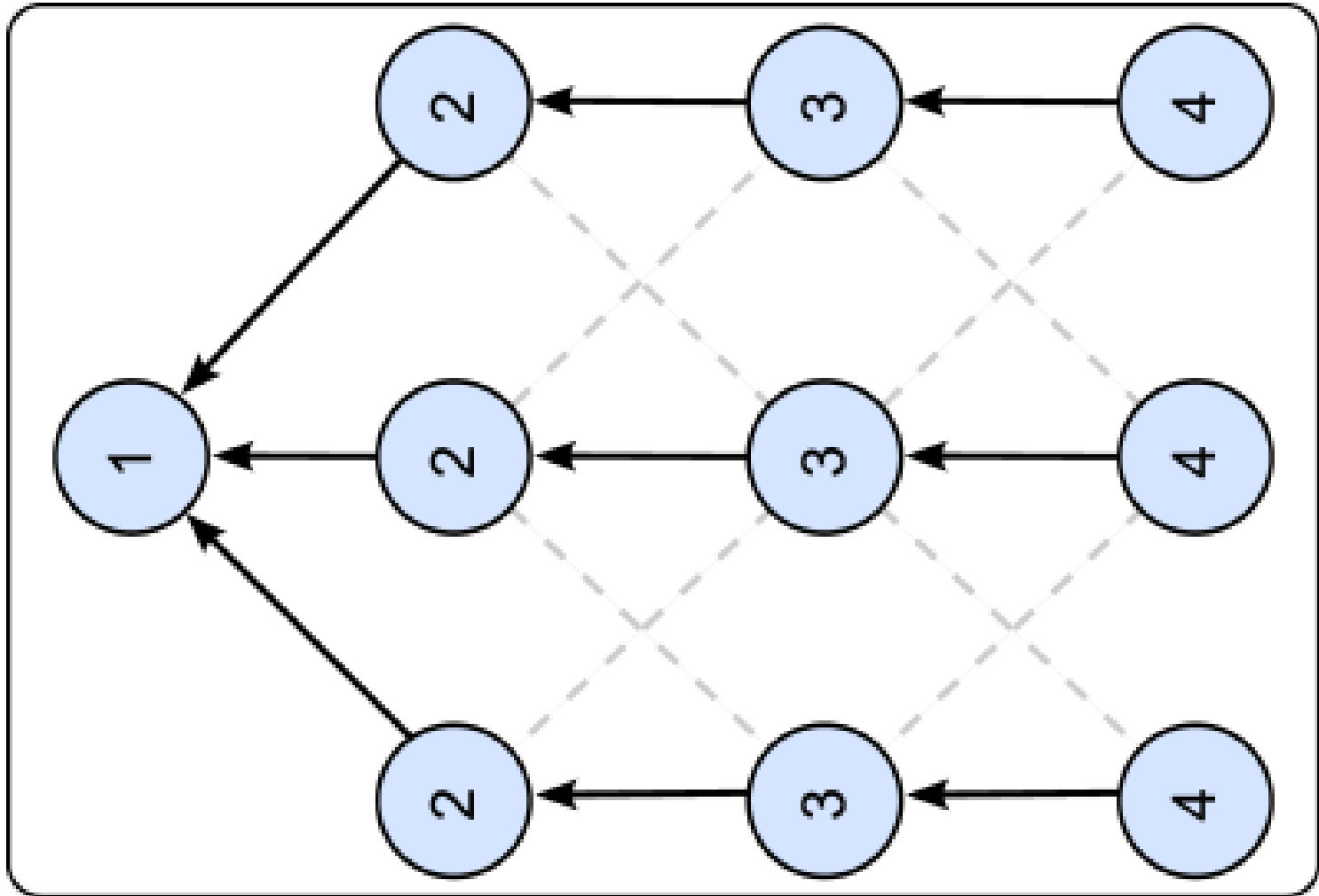
RPL Topology Creation - Upward



RPL Topology Creation - Upward



RPL Topology Creation - Upward



RPL Topology

- ▶ Downward routes created analogously
- ▶ Two routing modes
 - ▶ Non-storing: Via root, without local routing tables
 - ▶ Storing: Directly into subtrees, with local routing tables
- ▶ Topology maintenance: New DAG version created on request



Performance Properties of MANETs

- ▶ One-Hop Capacity:
Consider MANET of n equal nodes, each acting as router, with constant node density. Then the One-Hop Capacity grows linearly $\rightarrow O(n)$
- ▶ Total Capacity surprisingly low:
 - ▶ Consider MANET of n equal nodes, each acting as router in an *optimal* set-up, then the Node Capacity to reach an arbitrary destination reads $\rightarrow O(1/\sqrt{n})$
 - ▶ Node Capacity further decreases under wireless transmission $\rightarrow O(1/\sqrt{(n \ln(n))})$



Aspects in P2P over MANETs

- Manets consist of moving, unstable components
 - ➔ unsuitable for client-server, but P2P applications
- P2P applications built for failure tolerance
 - ➔ potential for compensating Manet drop-outs
- P2P and Manets cope with member mobility
 - ➔ provide capabilities of self-restructuring
- But: P2P routing (mainly) regardless of underlay capacities
 - ➔ Manet limitations require optimising adaptation
- P2P and Manet changes may amplify
 - ➔ Issues of cross-layer synchronisation



References

- C. Murthy and B. Manoj: *Ad Hoc Wireless Networks*, Pearson Prentice Hall, 2004.
- Charles Perkins: *Ad Hoc Networking*, Addison-Wesley, 2001.
- S. Sarkar, T. Basavaraju, C. Puttamadappa: *Ad Hoc Mobile Wireless Networks*, Auerbach Publications, 2008.
- Nitin H. Vaidya: *Mobile Ad Hoc Networks*, Tutorial at InfoCom 2006, <http://www.crhc.uiuc.edu/wireless/talks/2006.Infocom.ppt>.
- P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.
- www.rfc-editor.org

