Cairo University
Faculty of Engineering
Computer Engineering Department

Cairo University

Wireless Networks

Dr. Sandra Wahid

Proactive vs Reactive Protocols

Proactive Routing

- Is based on **periodic exchanges** that update the routing tables to all possible destinations, even if no traffic goes through.
- Better for **stable** networks.

Reactive Routing

- Is based on **on-demand** route discoveries that update routing tables only for the destination that has traffic going through.
- Better for **highly dynamic** networks.

1. Dynamic Source Routing (DSR)
2. Ad Hoc On-Demand Distance Vector Routing (AODV)

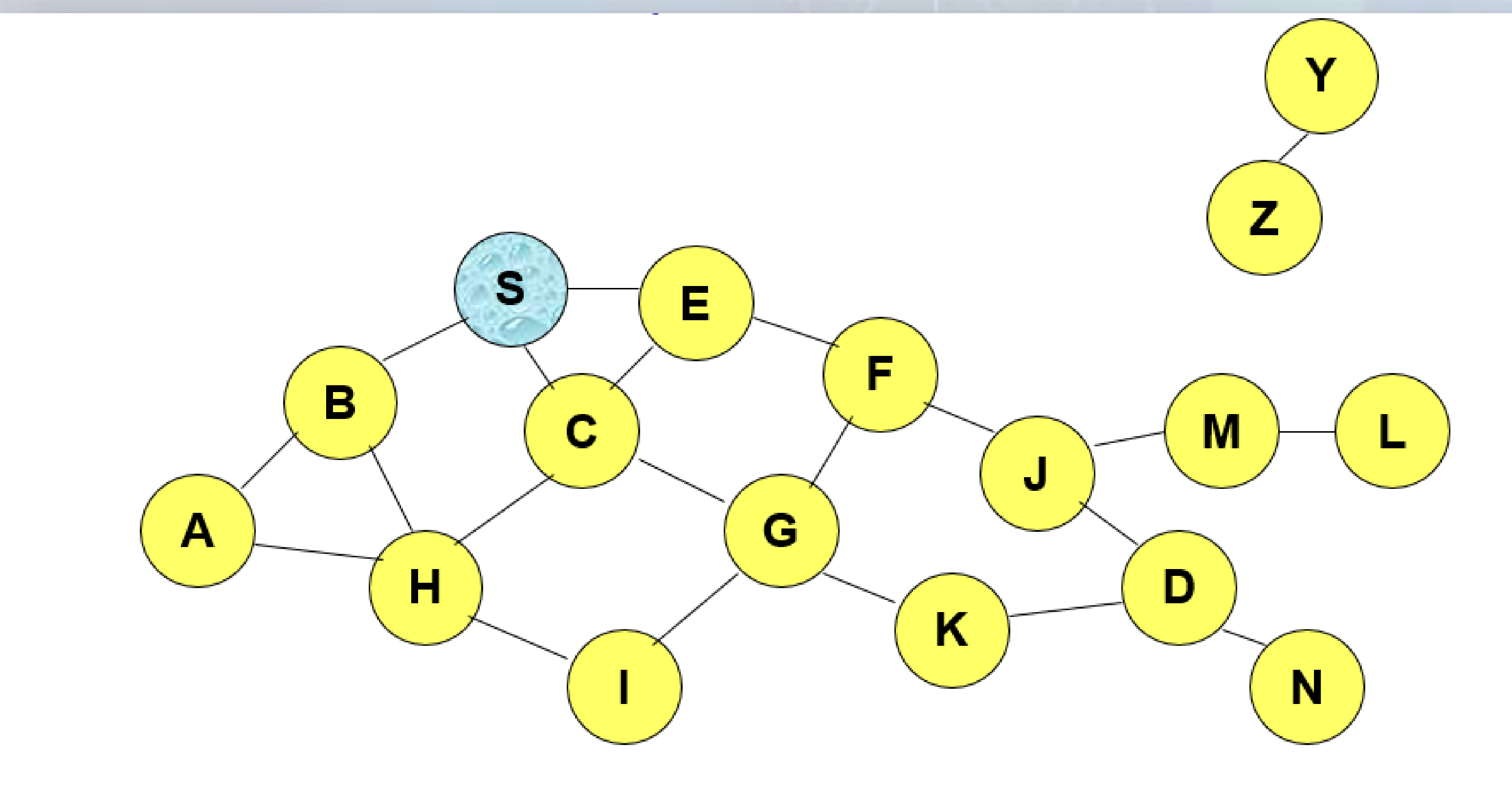
Dynamic Source Routing (DSR)

- When S wants to send a packet to D, but does not know a route to D, S initiates a **route discovery.**

- Source node S floods **Route Request (RREQ).**

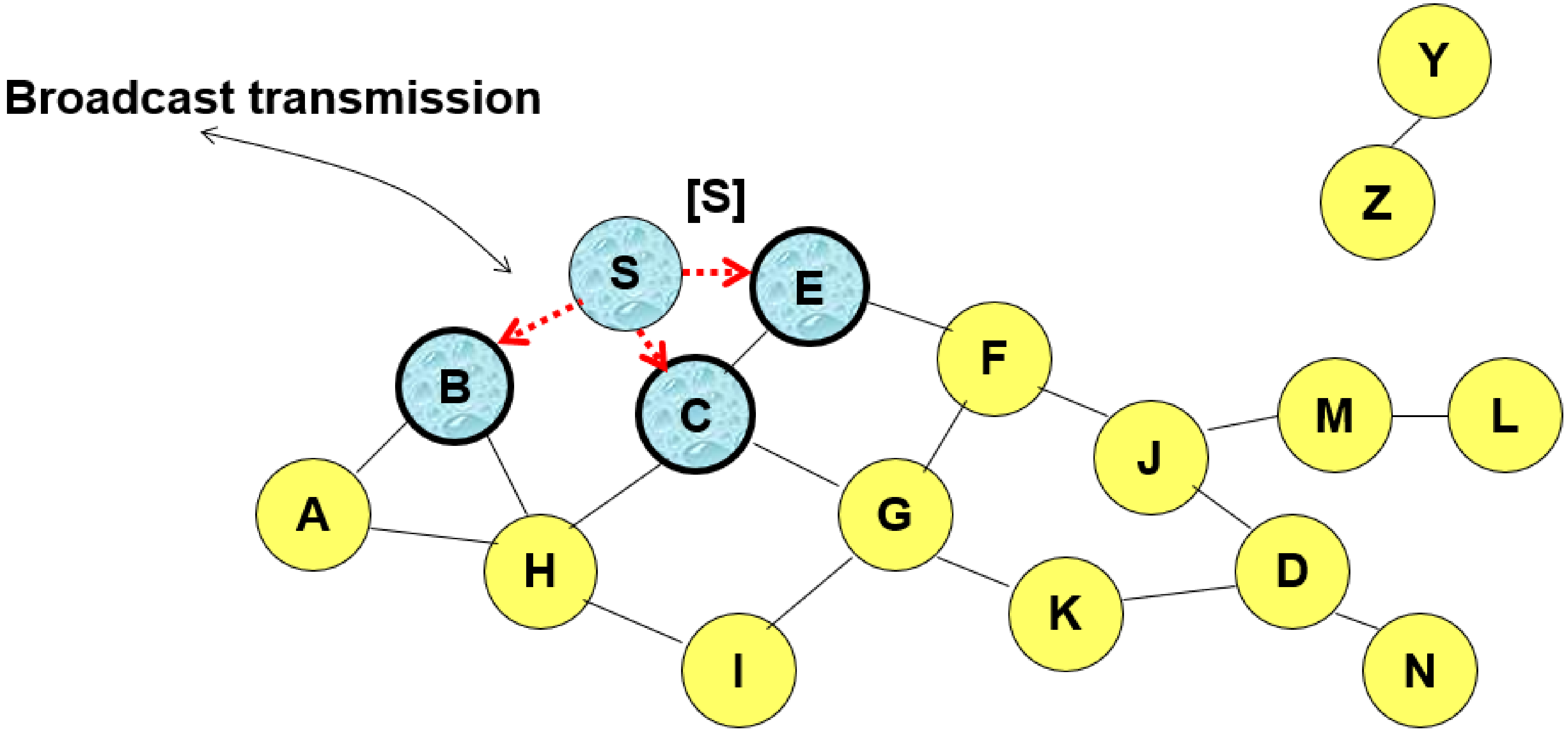
- Each node **appends own identifier** when forwarding RREQ. ***Why??***
  - **To limit flooding**, when a node receives RREQ having its own ID then it does not flood the RREQ again since it knows it already did.

DSR: Route Discovery



● Represents a node that has received RREQ for D from S

DSR: Route Discovery

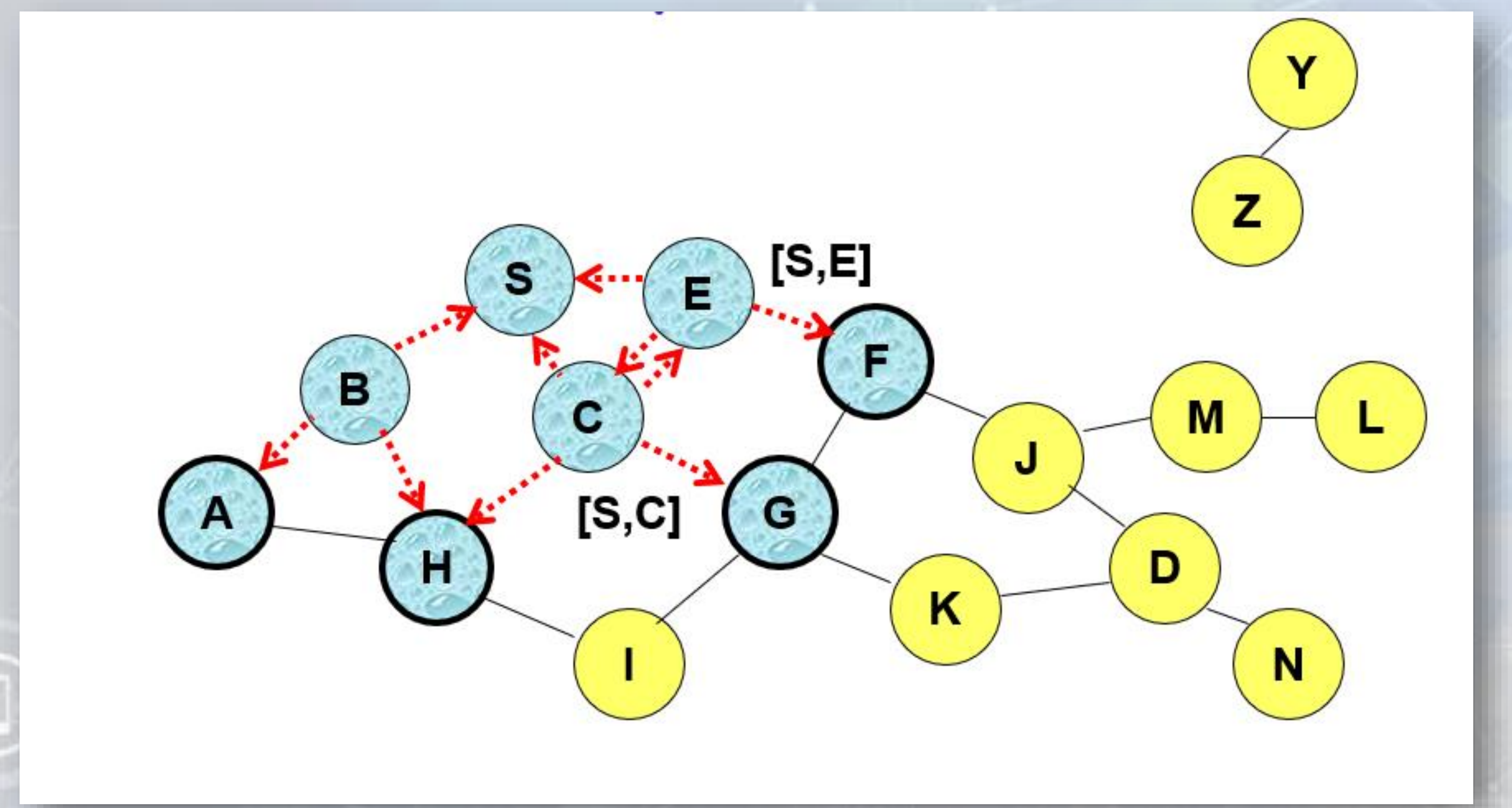


Broadcast transmission

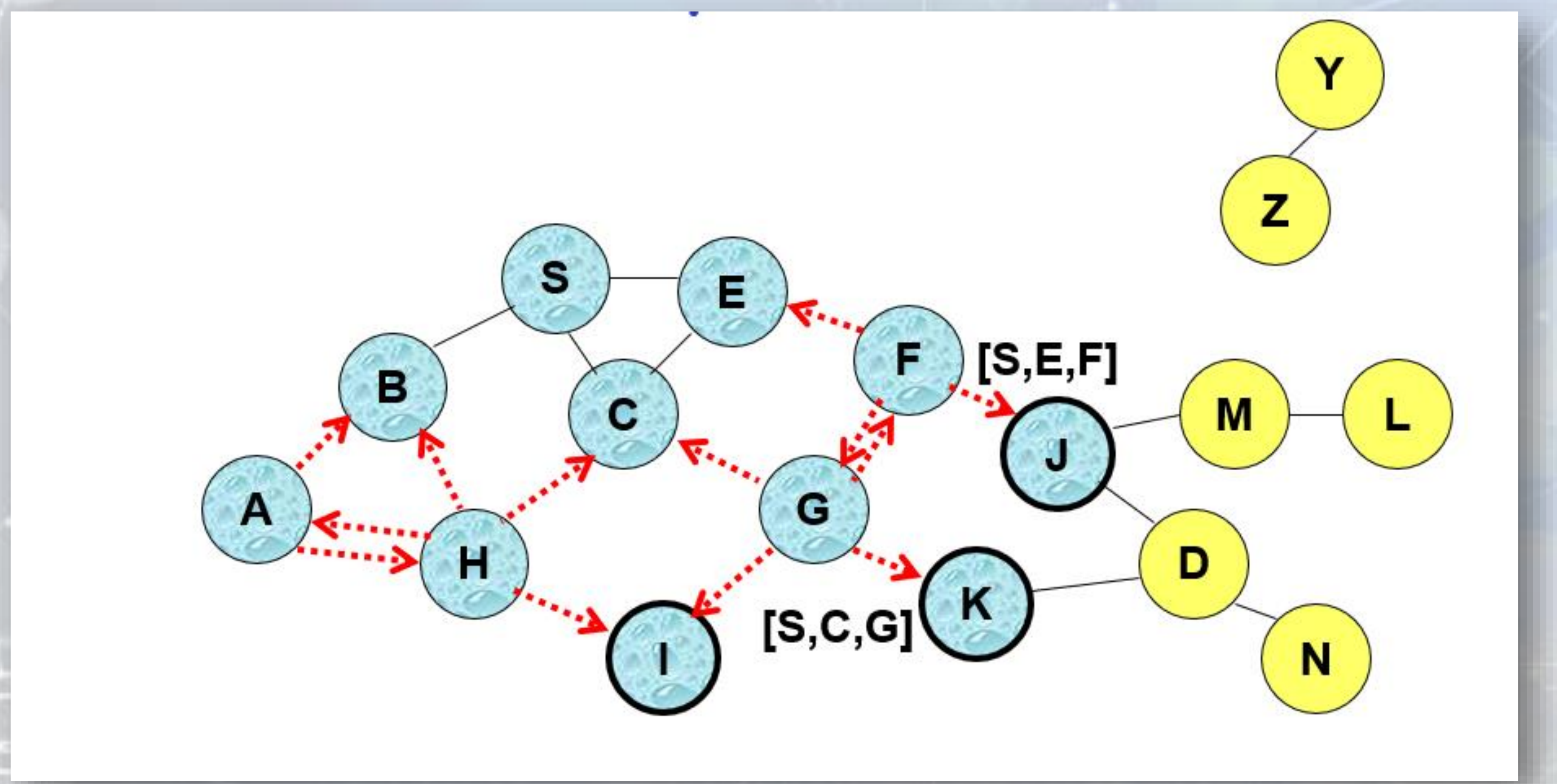
⇢ Represents transmission of RREQ

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

DSR: Route Discovery

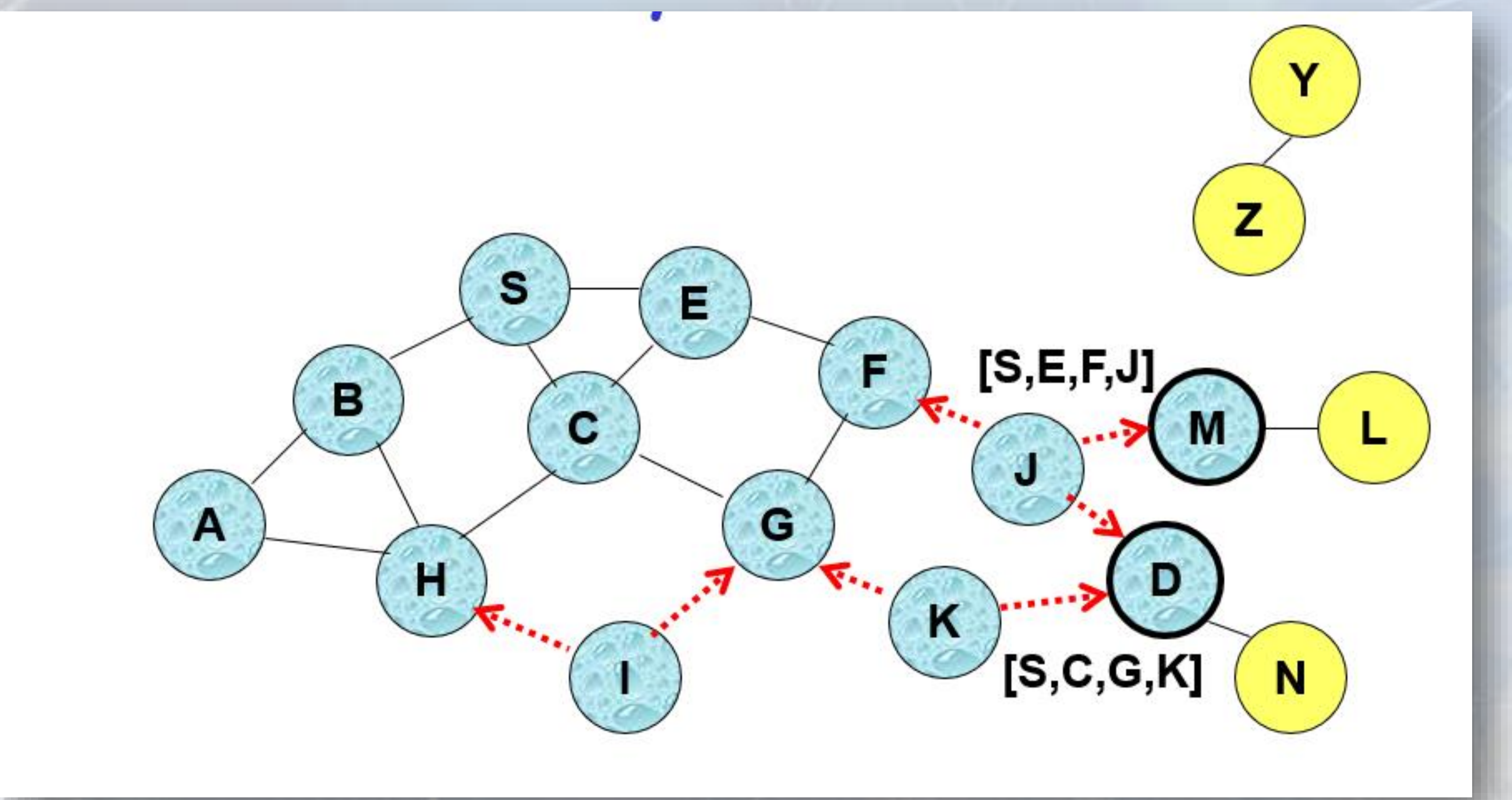


DSR: Route Discovery



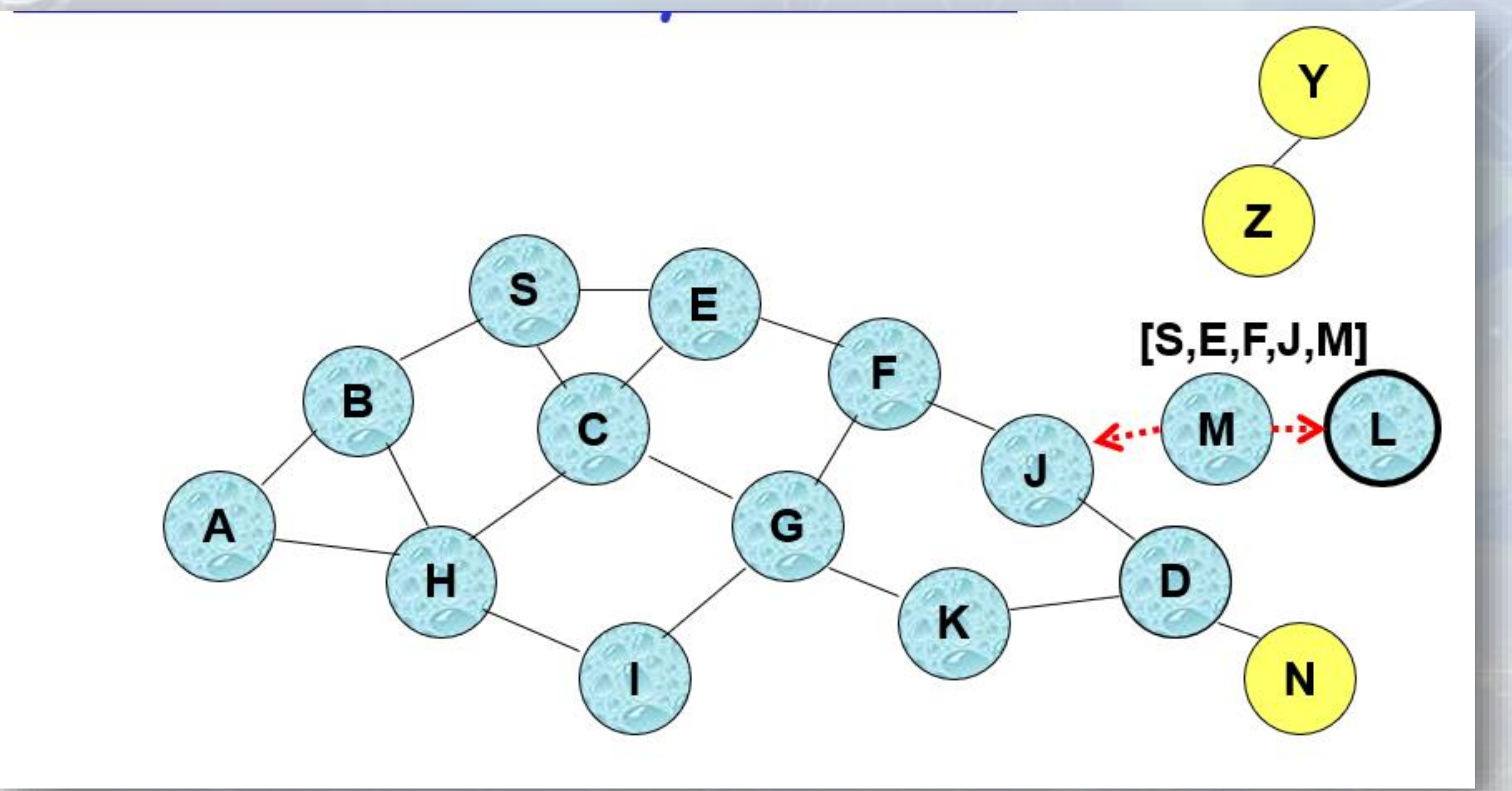
Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once

DSR: Route Discovery



Nodes J and K both broadcast RREQ to node D

DSR: Route Discovery



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

DSR: Route Reply

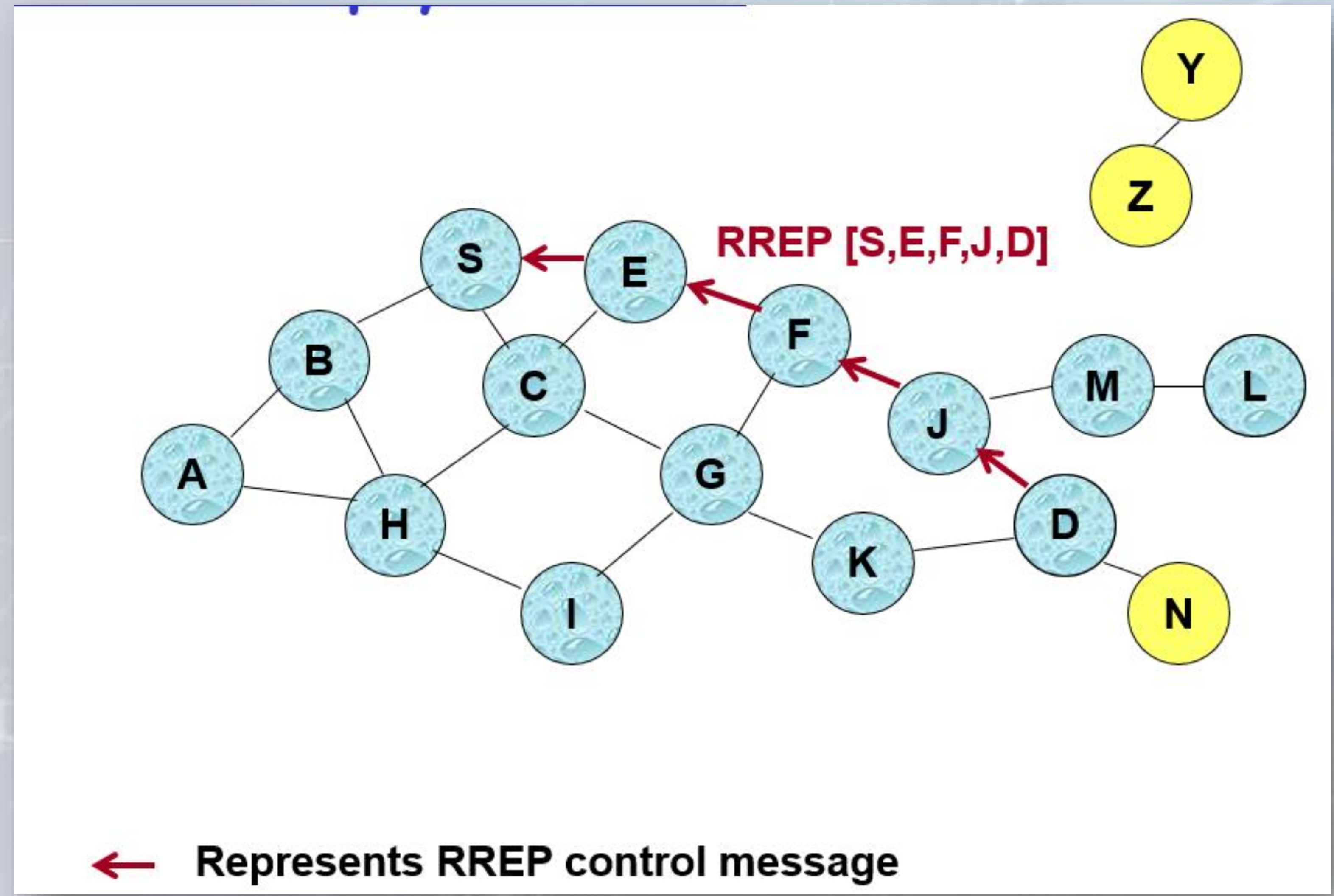
- Destination D on receiving the **first RREQ**, sends a **Route Reply (RREP)**

- Route of RREP is the **reverse** of the route in received RREQ.

- RREP includes the route from S to D on which RREQ was received by node D.

- Reverse route assumes **bi-directional** links. ***How to ensure this??***
  - To ensure this, node only forward RREQ if its link is bi-directional.

DSR: Route Reply



← Represents RREP control message

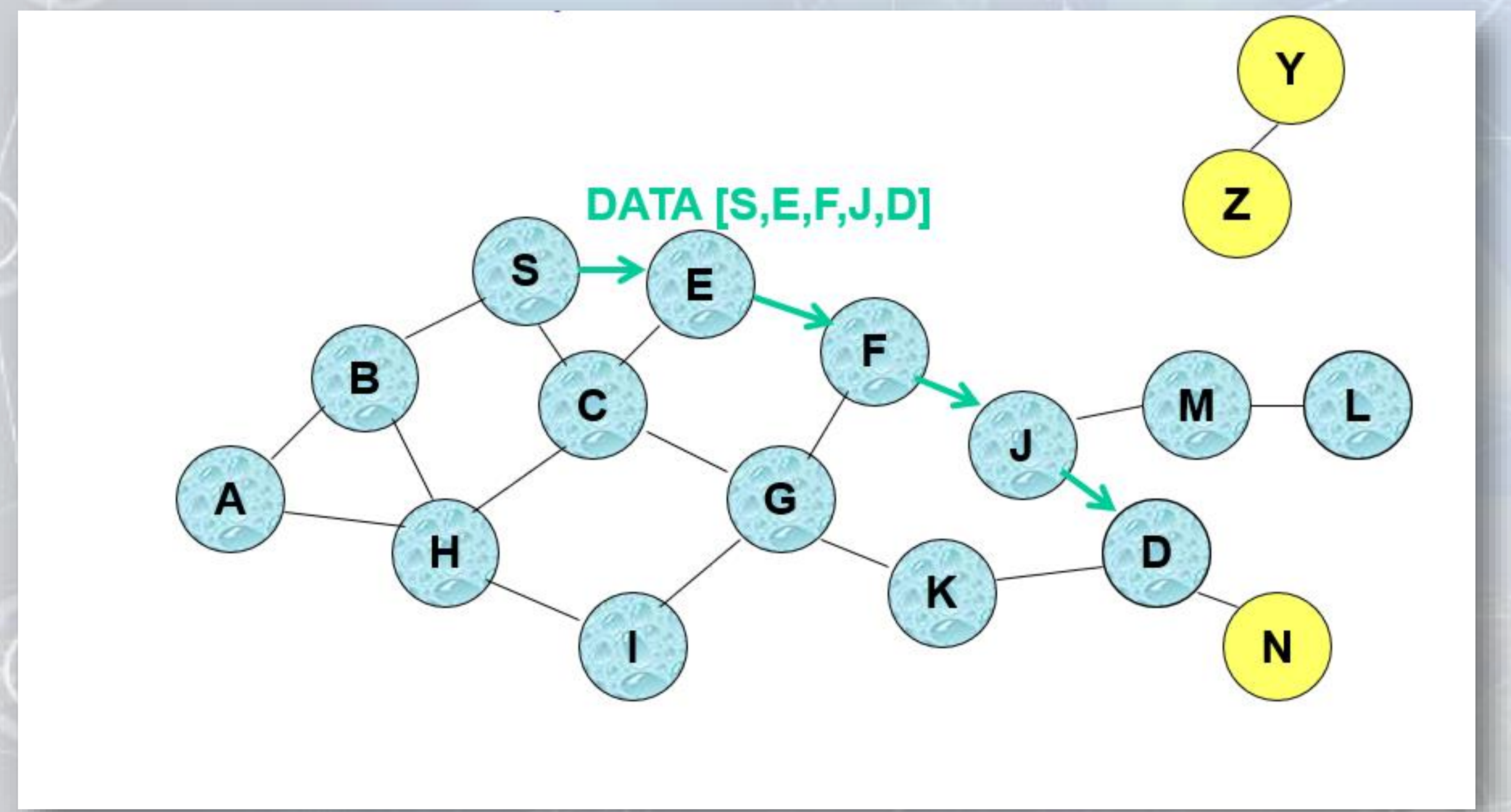
DSR: Data Delivery

- 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

DSR: Data Delivery

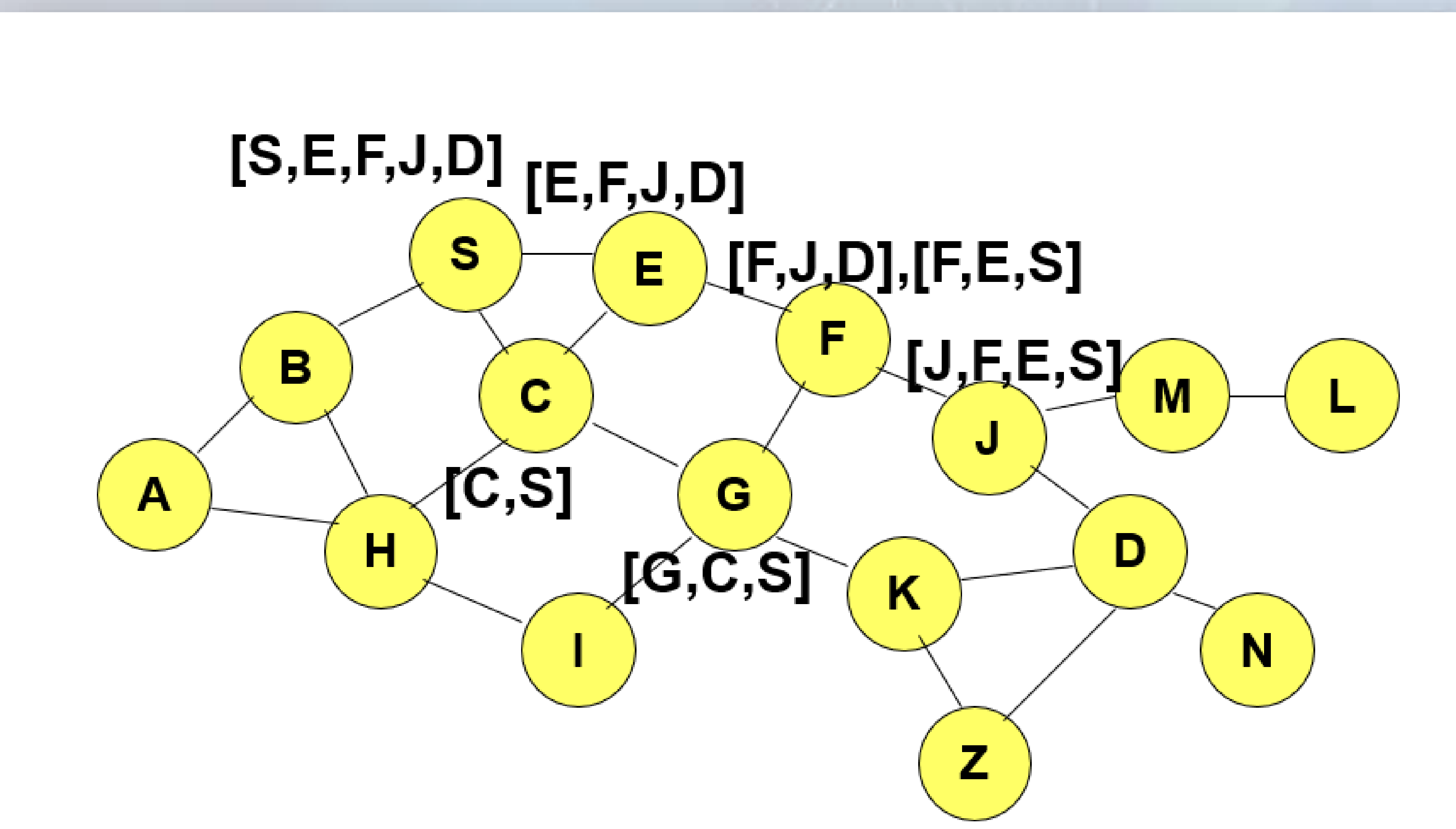


Packet header size grows with route length

DSR: Optimization by Route Caching

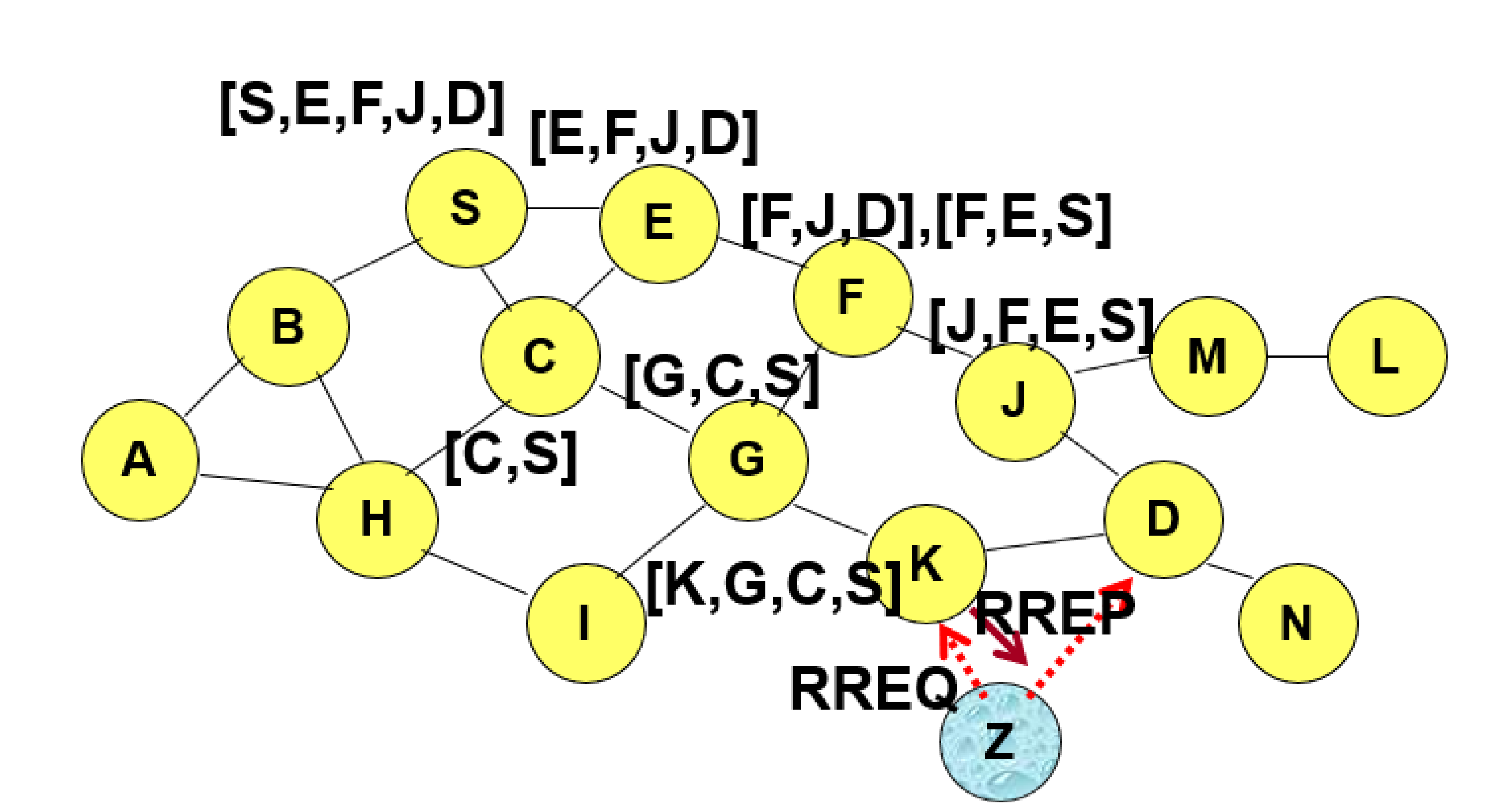
- Each node **caches a new route it learns** by any means.
  - S finds route [S,E,F,J,D] to D, S also learns route [S,E,F] to F, …
  - K receives Route Request [S,C,G], K learns route [K,G,C,S] to S, …
  - F forwards Route Reply [S,E,F,J,D], F learns route [F,J,D] to D, …
  - E forwards Data [S,E,F,J,D], E learns route [E,F,J,D] to D, …
- Discover route from S to D
  - First look up cache.
  - If no route is found, initiate a route discovery.
- When Node X receives a Route Request
  - First look up cache.
  - If X knows a route, send a Route Reply instead of flooding the Route Request.
- Using route cache
  - can **speed up route discovery.**
  - can **reduce propagation of route requests.**

DSR: Optimization by Route Caching



[ ] Represents cached route at a node

DSR: Optimization by Route Caching

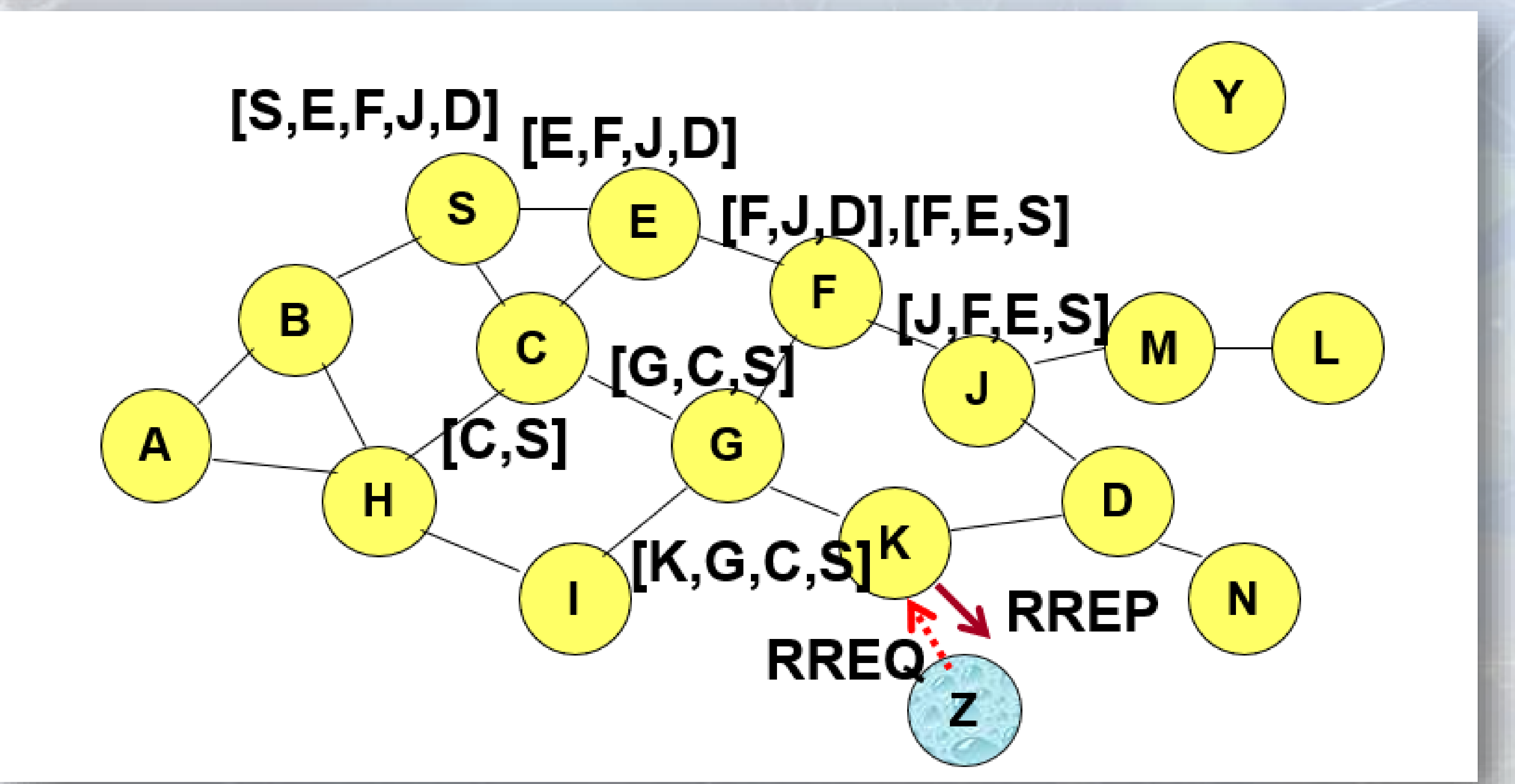


[ ] Represents cached route at a node.

Z sends a route request for C

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

DSR: Optimization by Route Caching

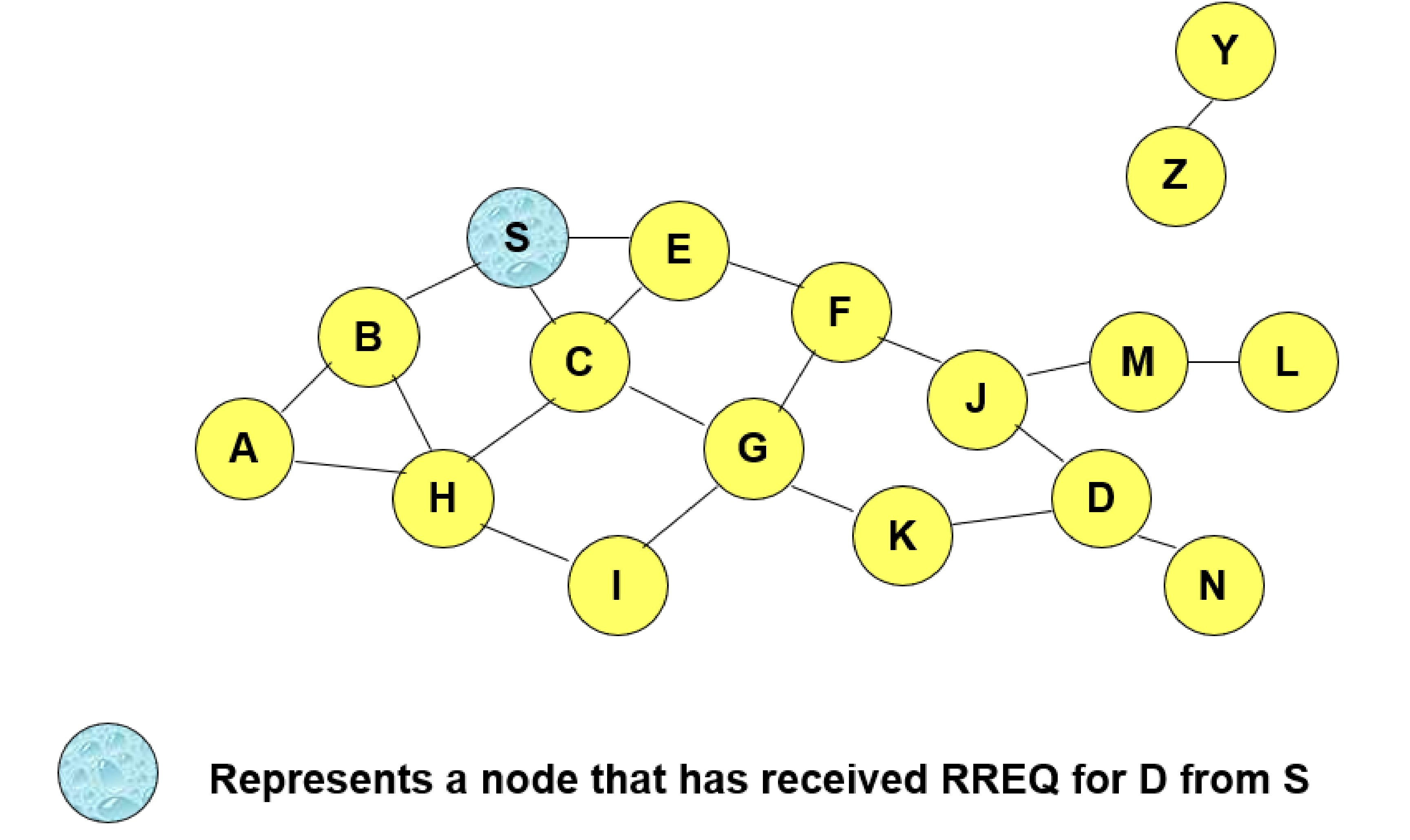


Assume no link between D and Z.
Route Reply (RREP) from node K limits flooding of RREQ.

Ad Hoc On-Demand Distance Vector Routing (AODV)

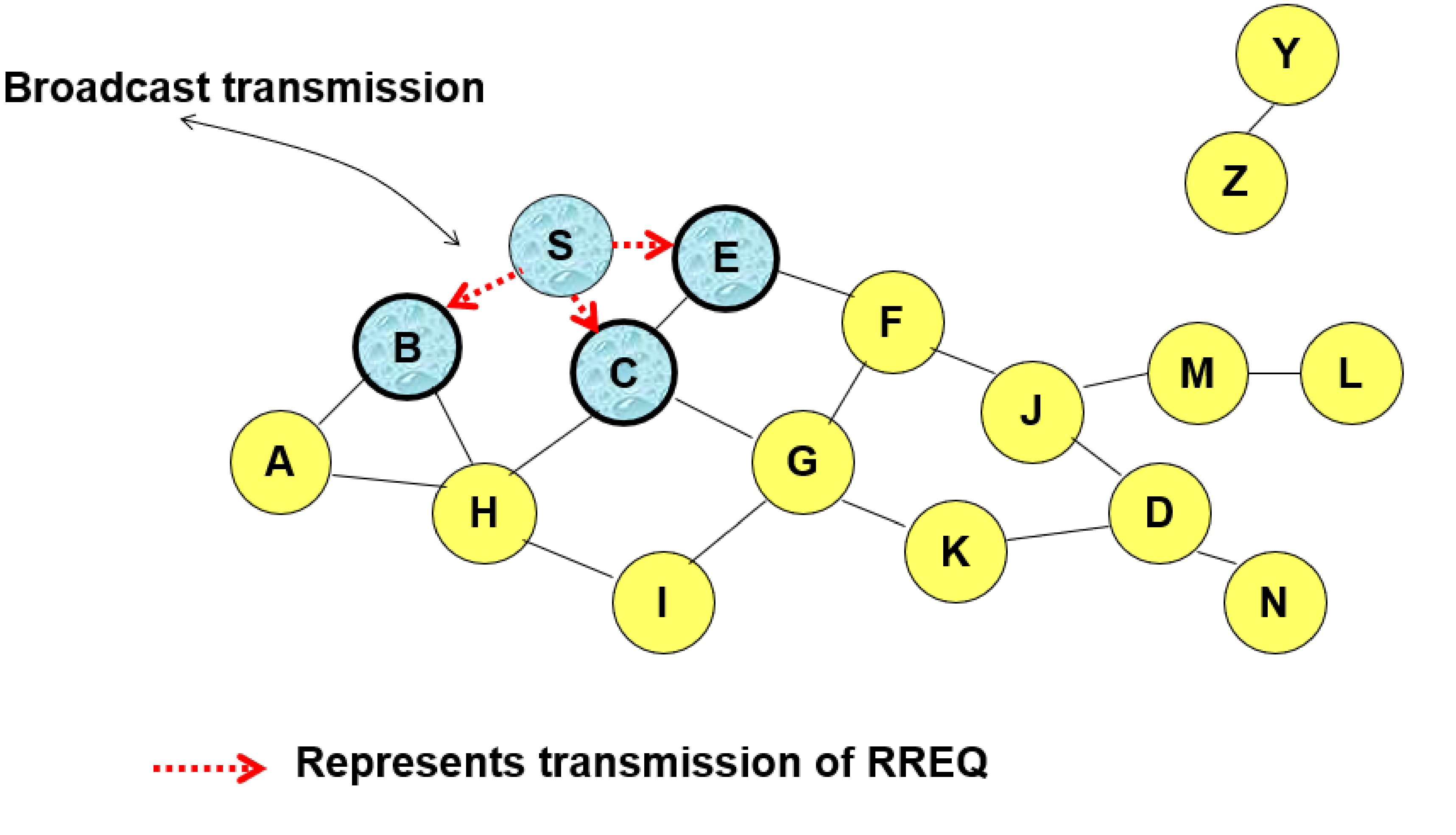
- Route Requests (RREQ) are forwarded in a manner similar to DSR.
- When a node **re-broadcasts** a Route Request, it **sets up a reverse path pointing towards the source**.
- Each node keeps the following information:
  - Destination of the RREQ.
  - **Next Hop.**
- When destination receives a Route Request, it replies by sending a Route Reply.
- Route Reply travels along the reverse path set-up when Route Request is forwarded.

AODV

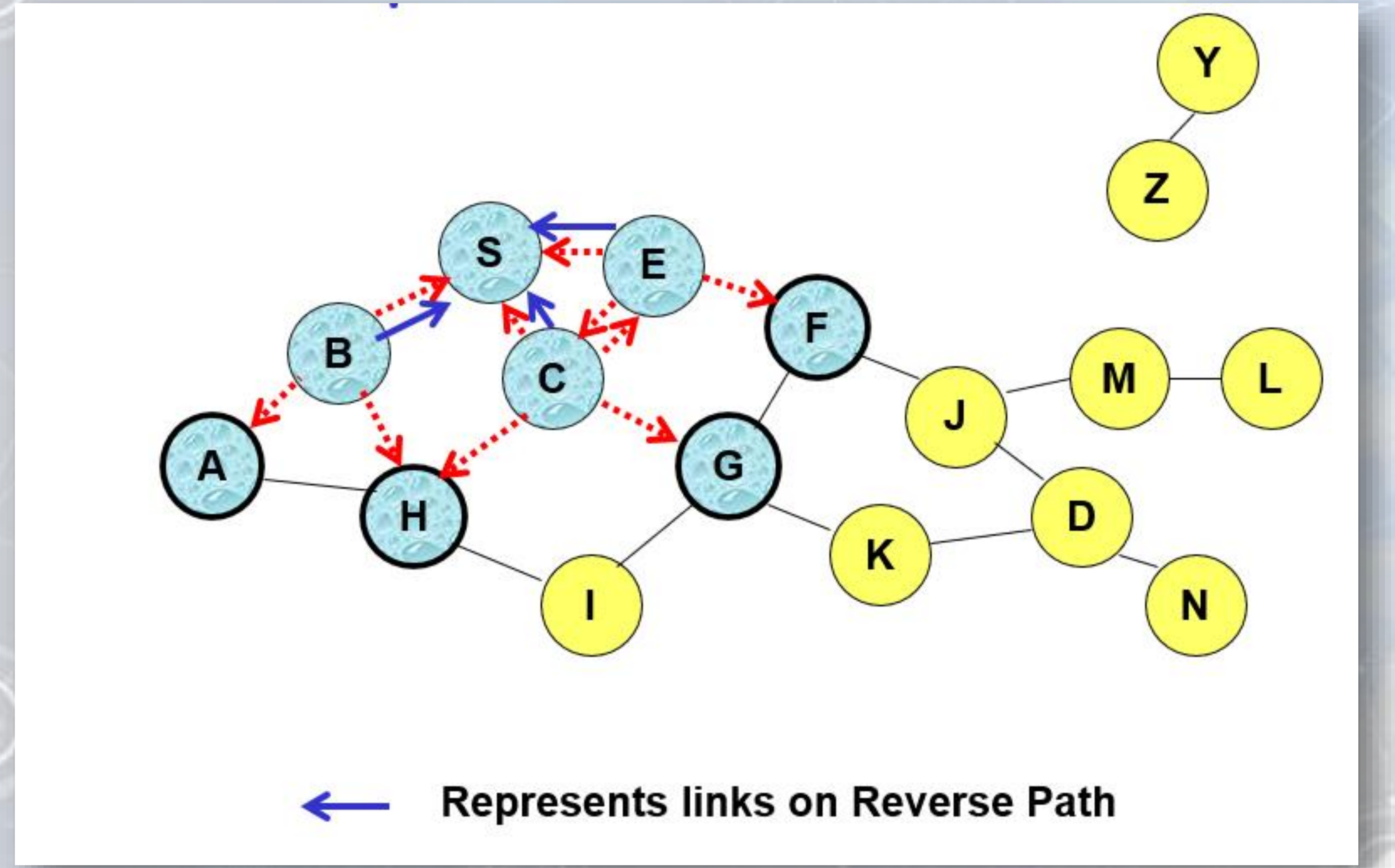


Represents a node that has received RREQ for D from S

AODV

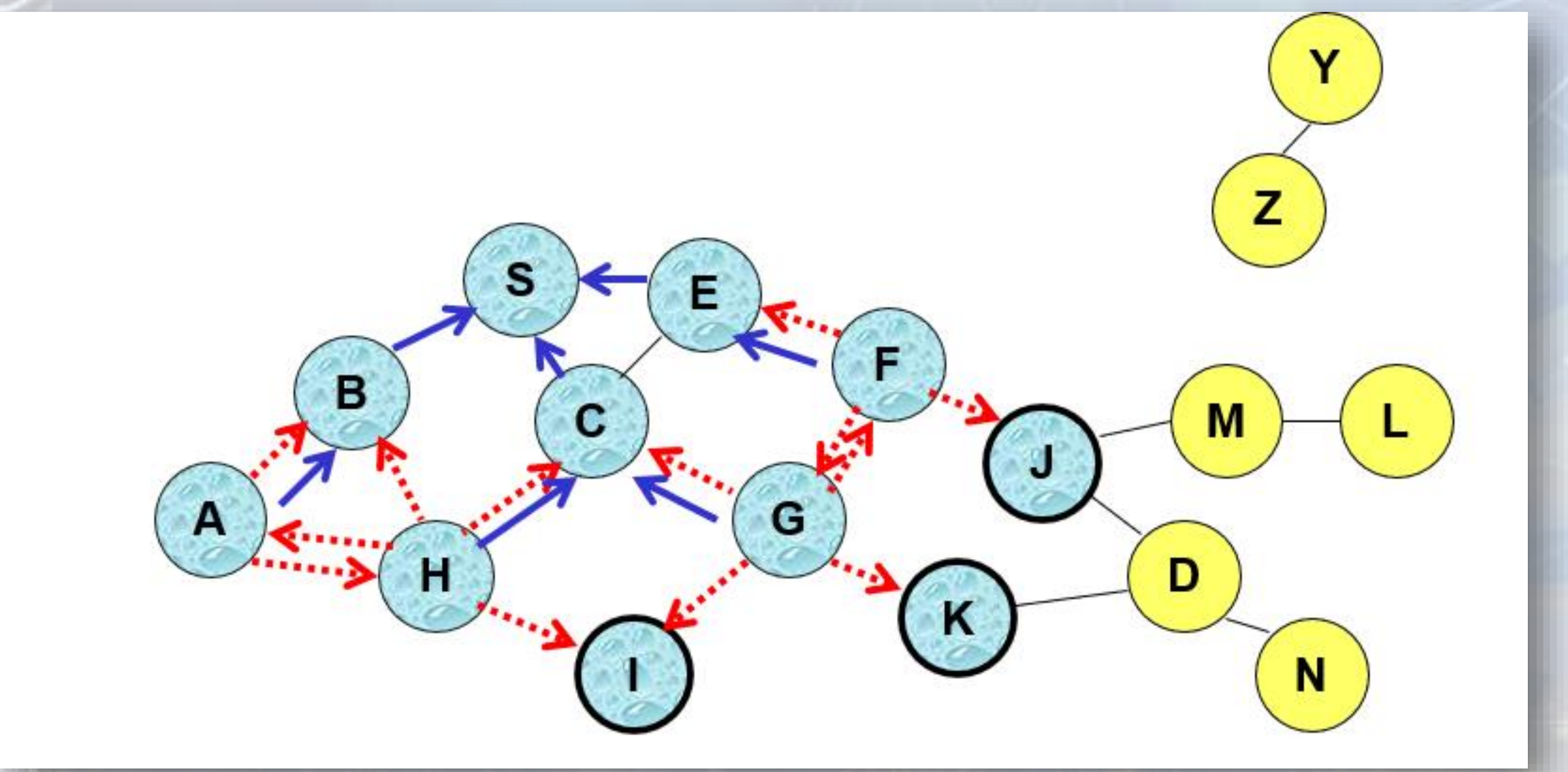

Broadcast transmission
Y
Z
S
E
B
C
F
J
M
L
A
H
G
D
K
I
N
Represents transmission of RREQ

AODV



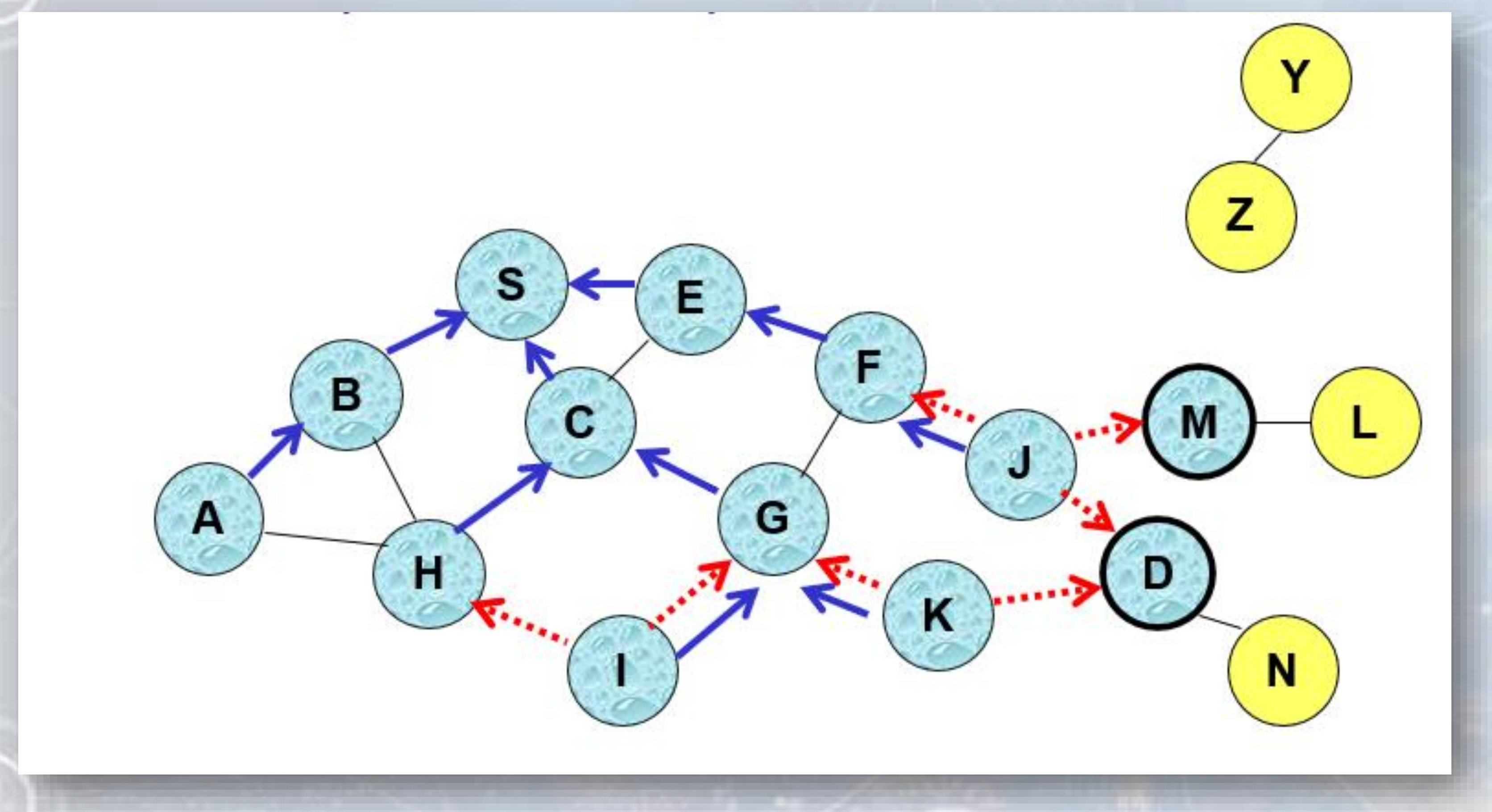
← Represents links on Reverse Path

AODV

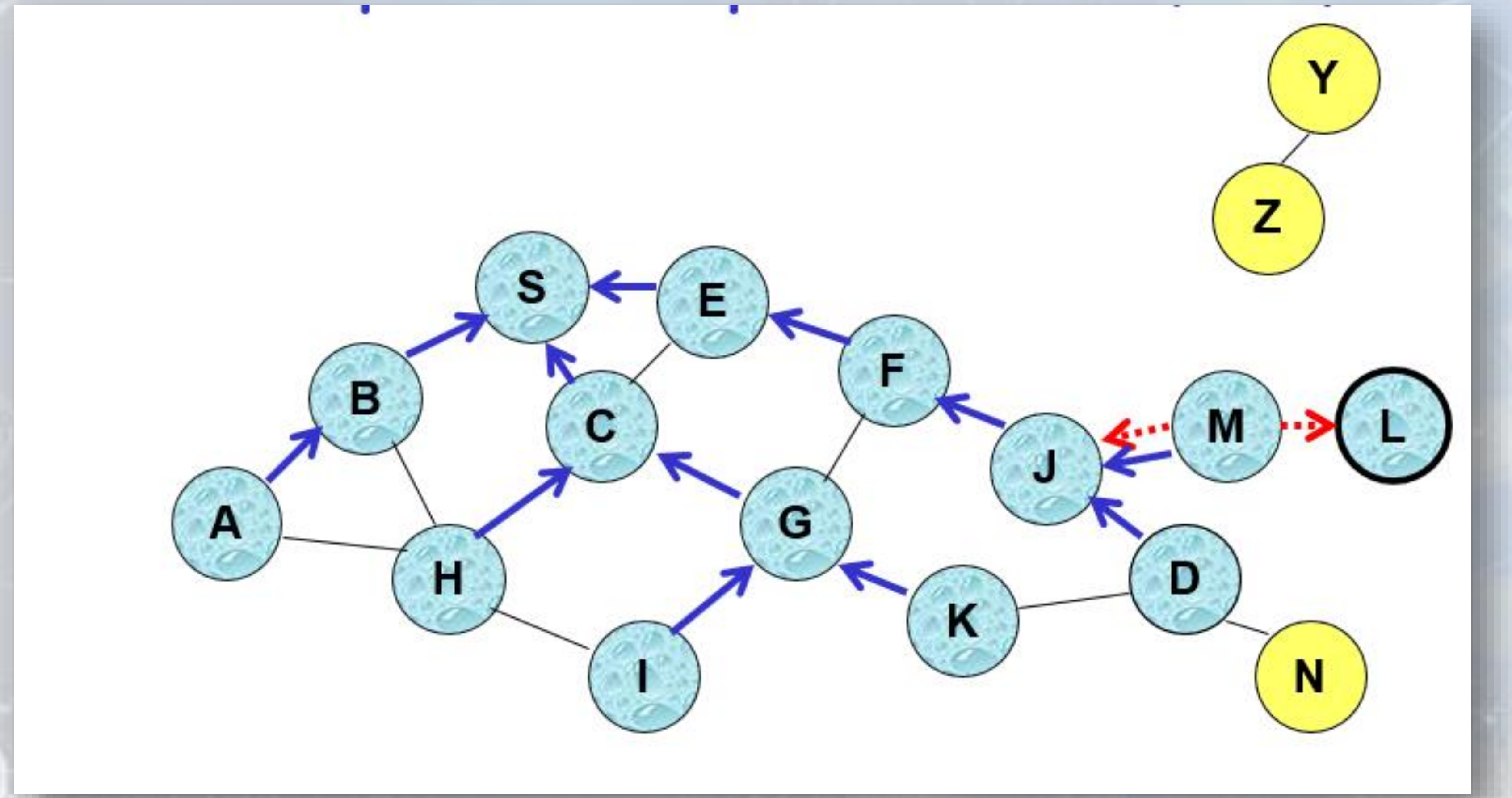


Node C receives RREQ from G and H, but does not forward it again nor sets-up a reverse pointer, because node C has already forwarded RREQ once.

AODV

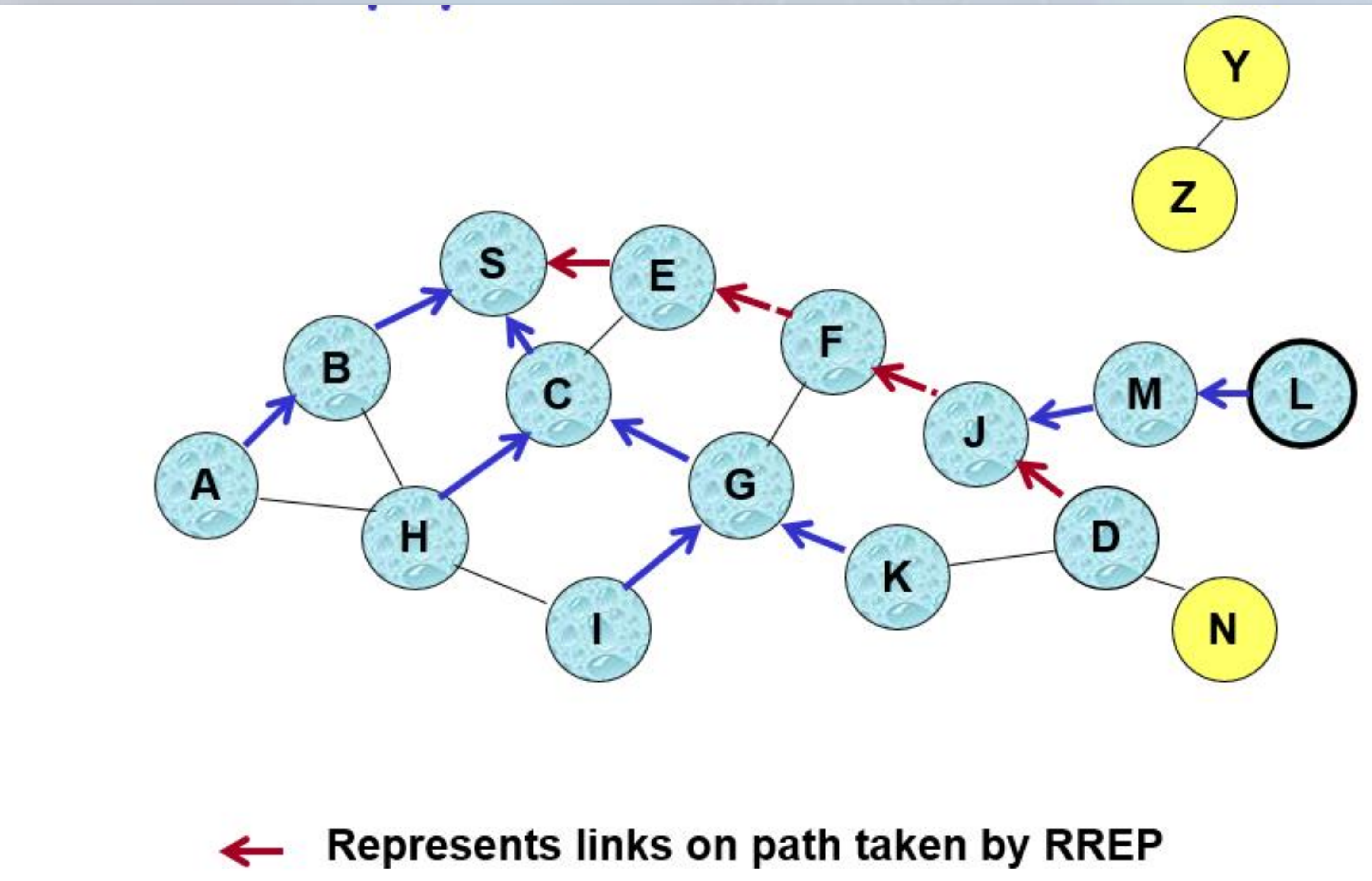


AODV



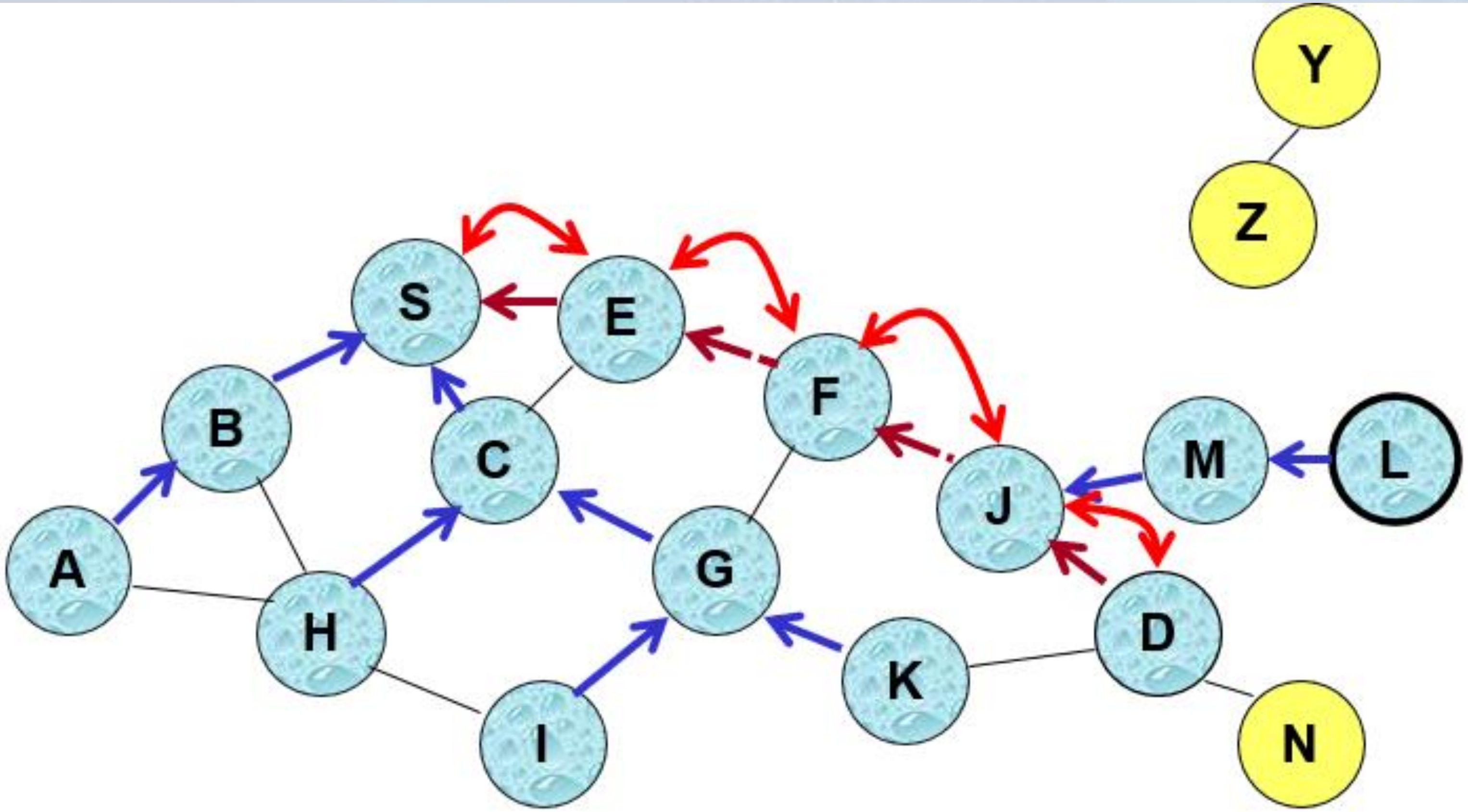
Node D does not forward RREQ, because node D is the intended target of the RREQ

AODV: Route Reply



← Represents links on path taken by RREP

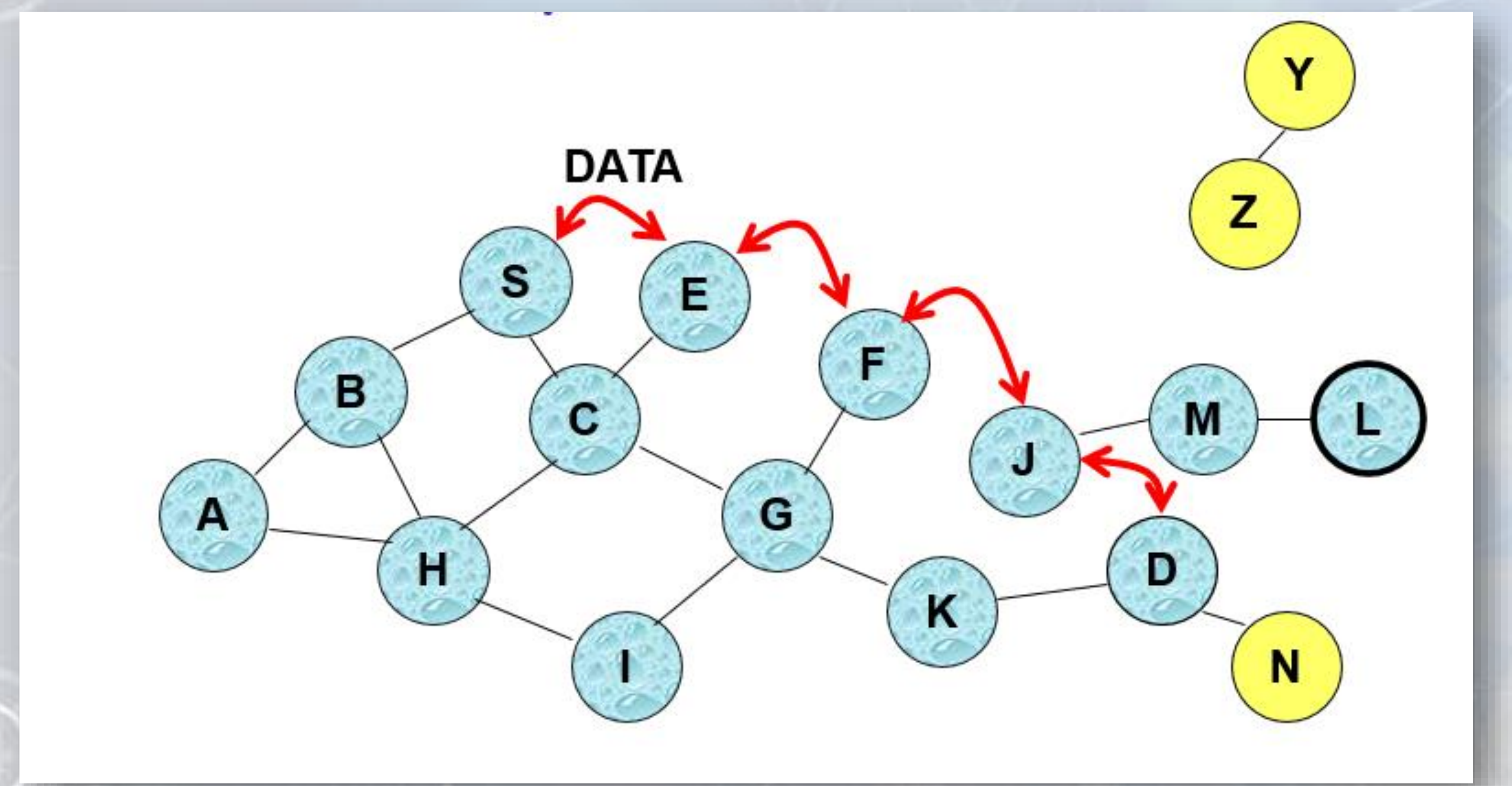
AODV: Forward Path Setup



Forward links are setup when RREP travels along the reverse path

↔ Represents a link on the forward path

AODV: Data Delivery



Routing table entries used to forward data packet.
Route is *not* included in packet header.

Thank You