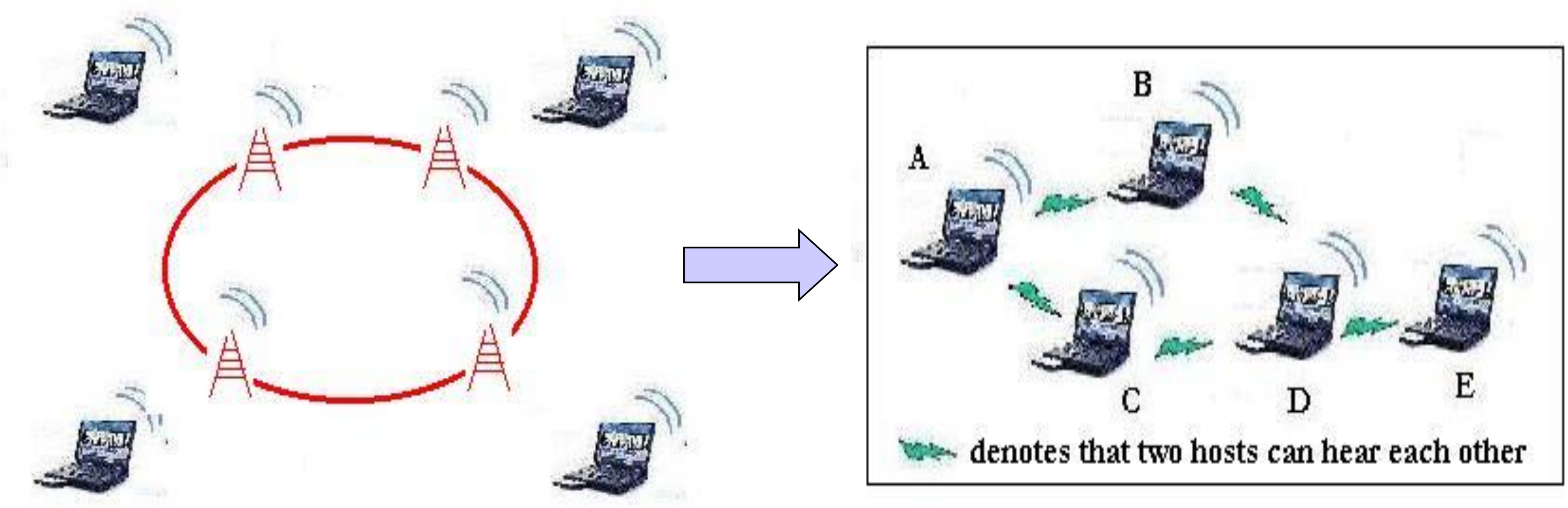
Routing Protocols in MANETs

JYOTHISHMATHI INSTITUTE OF TECHNOLOGY & SCIENCE

B.UMARANI
ASST. PROFESSOR
DEPARTMENT OF CSE

What is a MANET

- Mobile nodes, wireless links
- Infrastructure-less: by the nodes, …
- Multi-hop routing: …, and for the nodes
- Minimal administration: no hassles



What's unique about a MANET ?

- Moving nodes → ever changing topology
- Wireless links
  - → various and volatile link quality
- Pervasive (cheap) devices
  - → Power constraints
- Security
  - Confidentiality, other attacks

Challenges in MANET Routing

- Need dynamic routing
  - Frequent topological changes possible.
  - Very different from dynamic routing in the Internet.
  - Potential of network partitions.
- Routing overhead must be kept minimal
  - Wireless → low bandwidth
  - Mobile → low power
  - Minimize # of routing control messages
  - Minimize routing state at each node

Other Challenges

- Auto configuration issues
  - Address assignment
  - Service discovery
- Security issues
  - Ease of denial-of-service attack
  - Misbehaving nodes difficult to identify
  - Nodes can be easily compromised
- New Applications/services
  - Location based: Distribute some information to all nodes in a geographic area (geocast).
  - Content based: Query all sensors that sensed something particular in the past hour.

MANET Protocol Zoo

- Topology based routing
  - Proactive approach, e.g., DSDV.
  - Reactive approach, e.g., DSR, AODV, TORA.
  - Hybrid approach, e.g., Cluster, ZRP.
- Position based routing
  - Location Services:
    - DREAM, Quorum-based, GLS, Home zone etc.
  - Forwarding Strategy:
    - Greedy, GPSR, RDF, Hierarchical, etc.

Routing Protocols

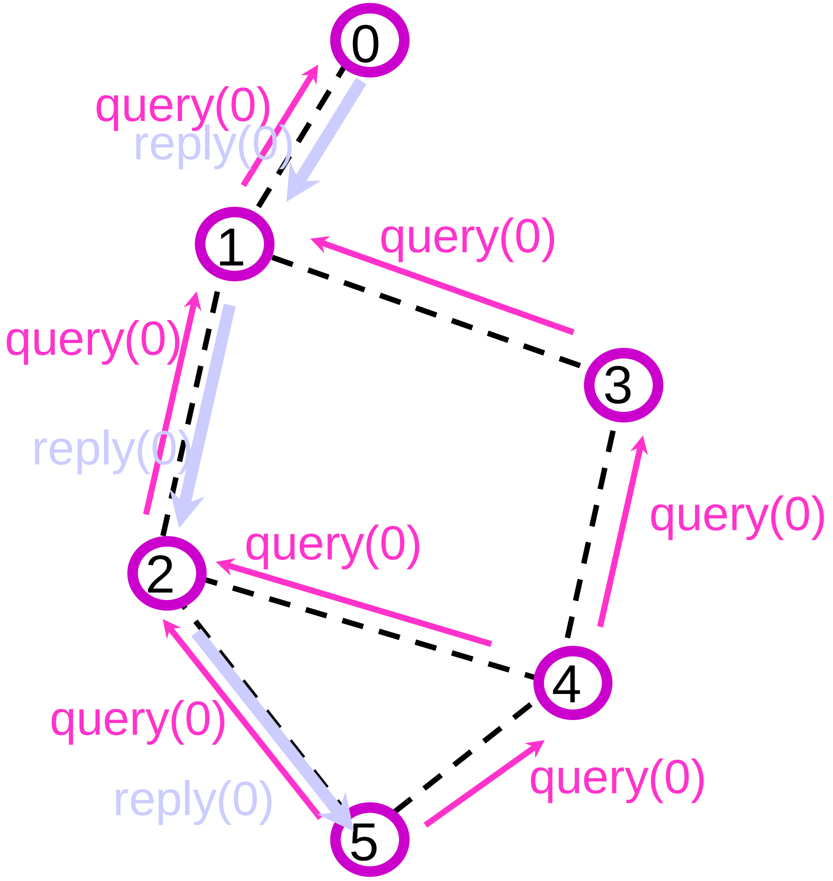
- Reactive (On-demand) protocols
  - Discover routes when needed
  - Source-initiated route discovery
- Proactive protocols
  - Traditional distributed shortest-path protocols
  - Based on periodic updates. High routing overhead
- Tradeoff
  - State maintenance traffic vs. route discovery traffic
  - Route via maintained route vs. delay for route discovery

Reactive Routing

- Key Goal: Reduction in routing overhead
  - Useful when number of traffic sessions is much lower than the number of nodes.
- No routing structure created *a priori*. Let the structure emerge in response to a need
- Two key methods for route discovery
  - source routing
  - backward learning (similar to intra-AS routing)
- Introduces delay

Reactive (on-demand) routing:

- Routing only when needed



Advantages:

- eliminate periodic updates
- adaptive to network dynamics

Disadvantages:

- high flood-search overhead with
  - mobility, distributed traffic
- high route acquisition latency

Reactive Routing – Source initiated

- Source floods the network with a *route request* packet when a route is required to a destination
  - Flood is propagated outwards from the source
  - Pure flooding = every node transmits the request only once
- Destination *replies* to request
  - Reply uses reversed path of route request
  - sets up the forward path
- Two key protocols: DSR and AODV

Dynamic Source Routing (DSR)

- Cooperative nodes
- Relatively small network diameter (5-10 hops)
- Detectable packet error
- Unidirectional or bidirectional link
- Promiscuous mode (optional)

Route Discovery

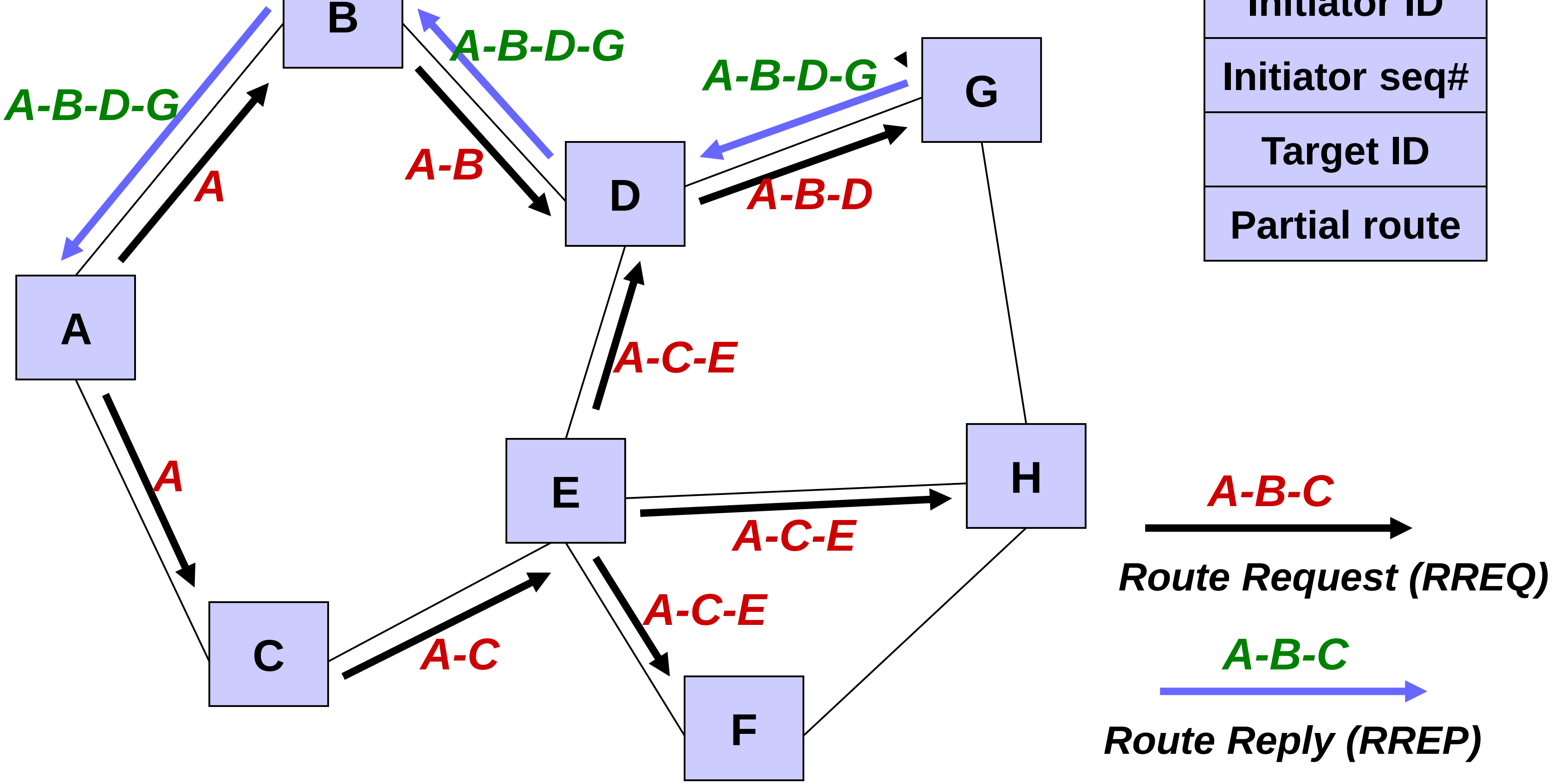
RREQ FORMAT

Initiator ID

Initiator seq#

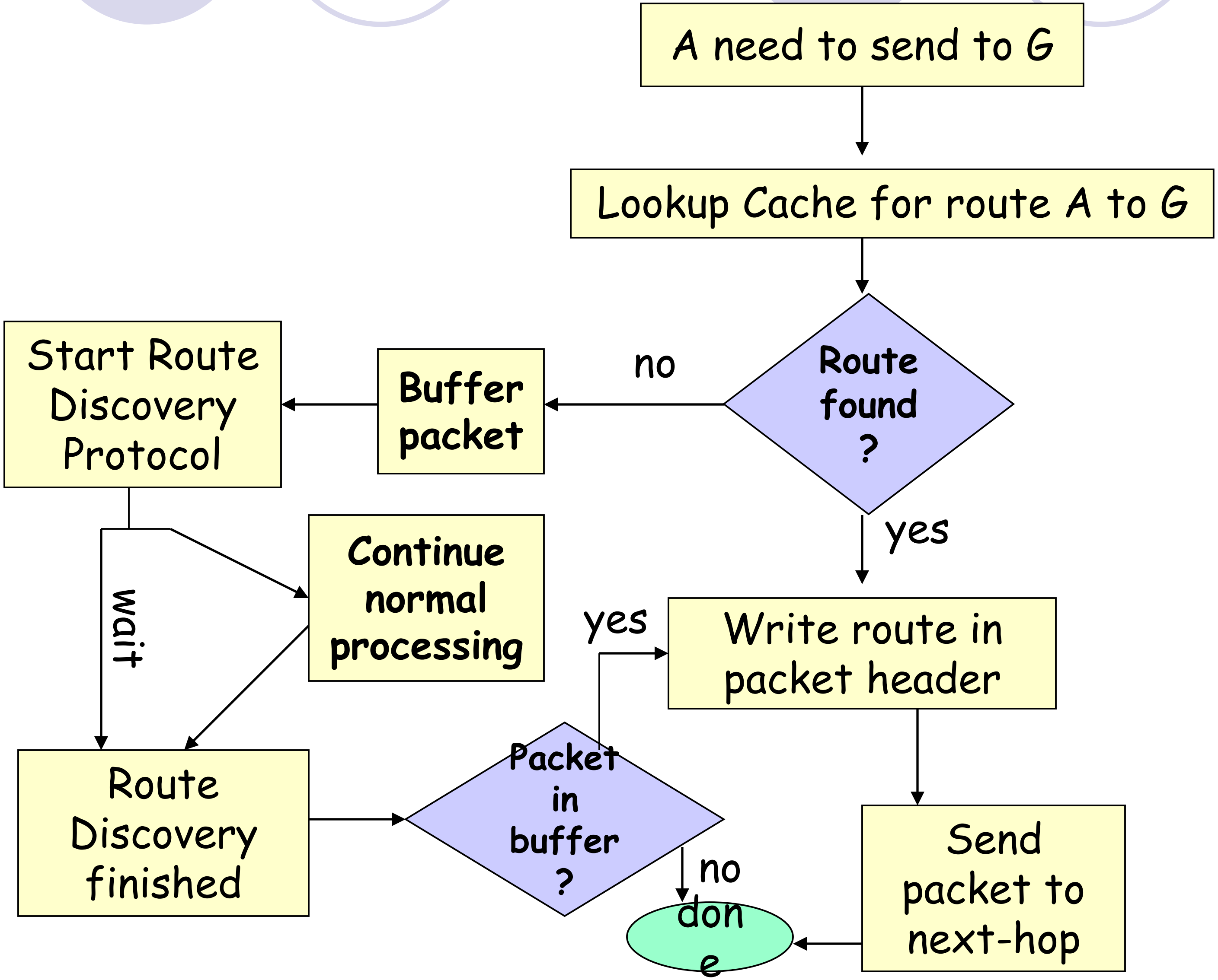
Target ID

Partial route

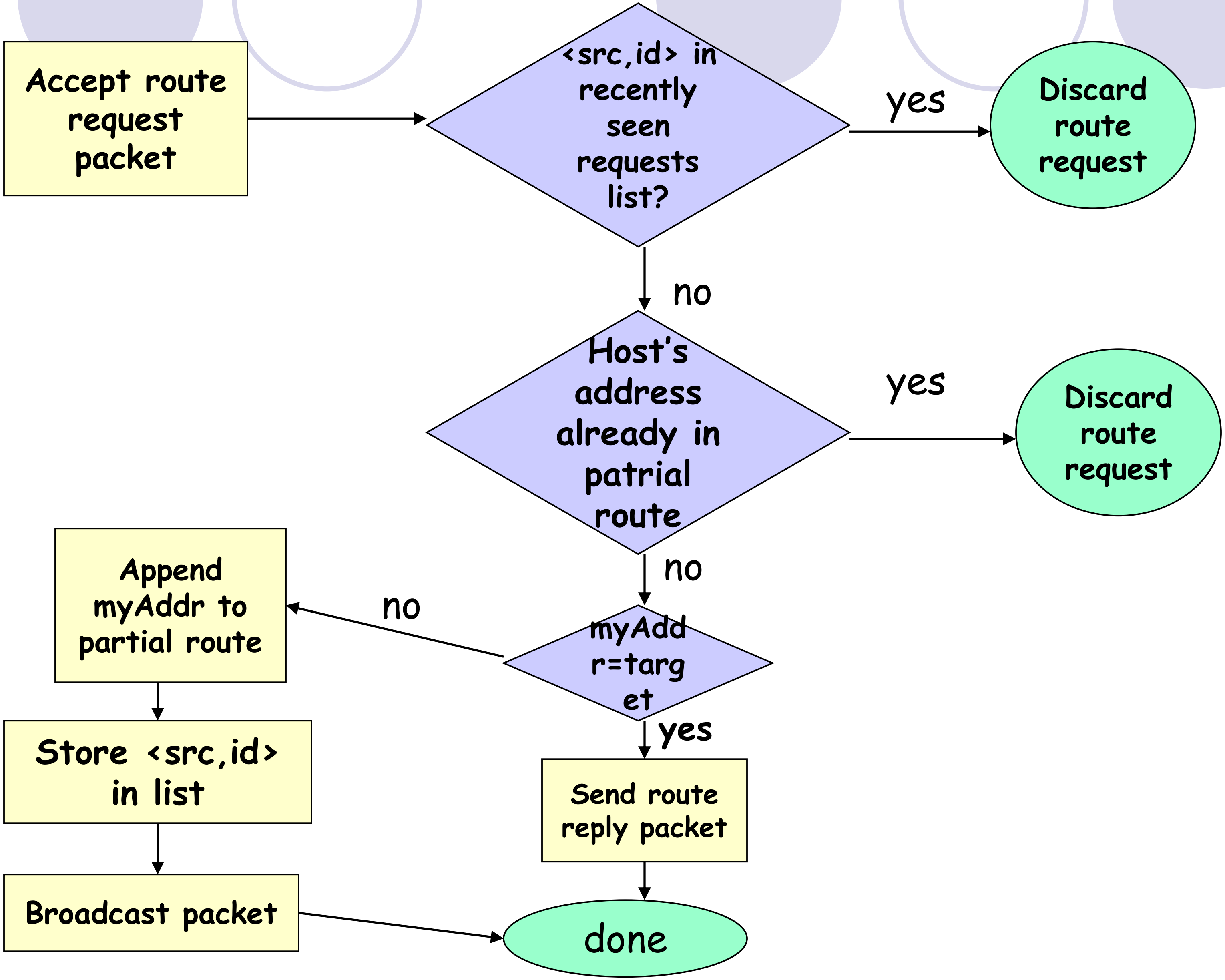


Route Discovery is issued with exponential back-off intervals.

Route Discovery: at source A



Route Discovery: At an intermediate node



Accept route request packet

<src,id> in recently seen requests list?

yes

Discard route request

no

Host's address already in patrial route

yes

Discard route request

no

myAddr=target

no

Append myAddr to partial route

yes

Send route reply packet

Store <src,id> in list

Broadcast packet

done

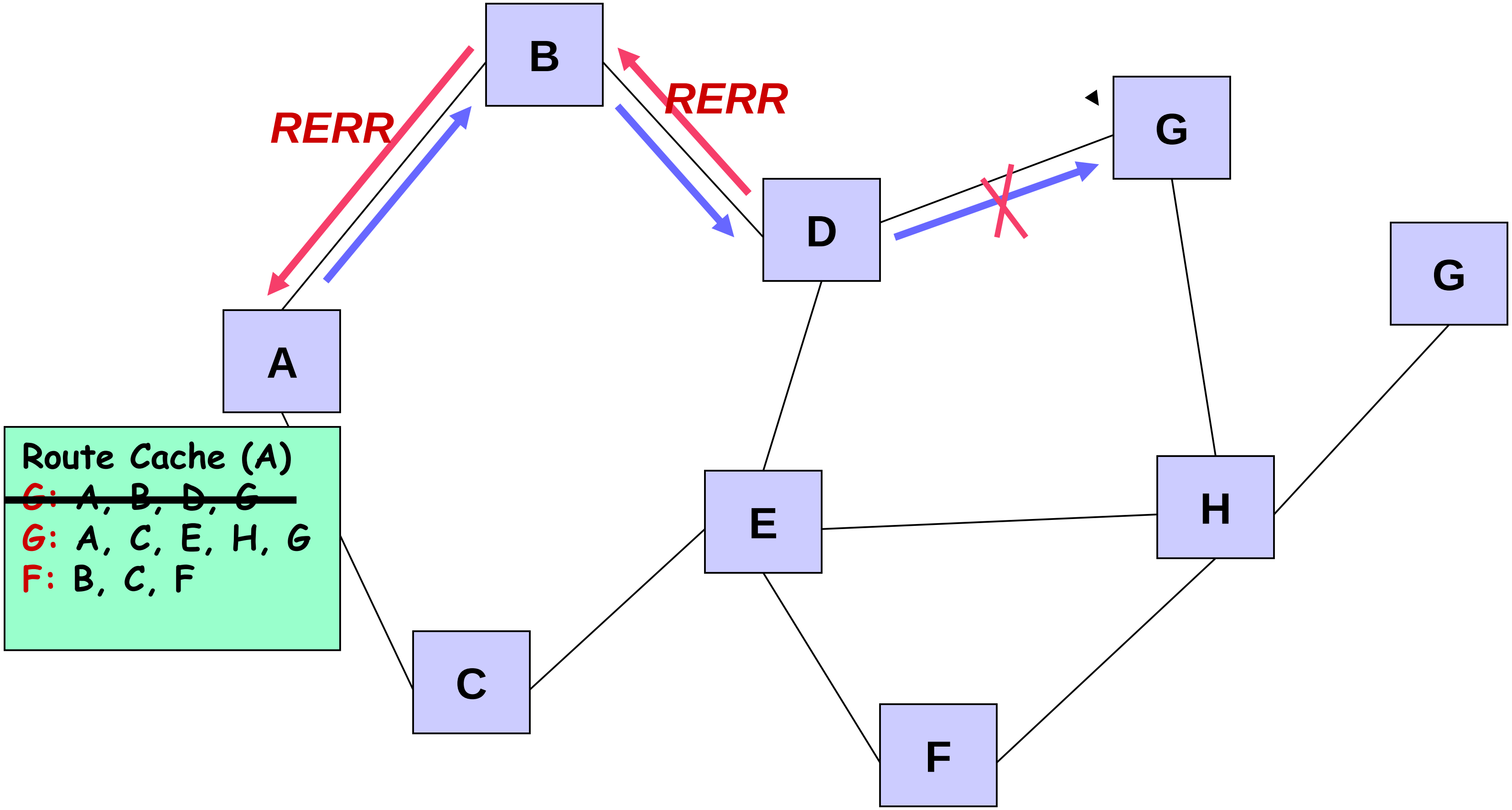
DSR - Route Discovery

- *Route Reply* message containing path information is sent back to the source either by
  - the destination, or
  - intermediate nodes that have a route to the destination
  - Reverse the order of the route record, and include it in Route Reply.
  - Unicast, source routing
- Each node maintains a *Route Cache* which records routes it has learned and overheard over time

Route Maintenance

- Route maintenance performed only while route is in use
- Error detection:
  - Monitors the validity of existing routes by *passively* listening to data packets transmitted at neighboring nodes
  - Lower level acknowledgements
- When problem detected, send *Route Error* packet to original sender to perform new route discovery
  - Host detects the error and the host it was attempting;
  - *Route Error* is sent back to the sender the packet – original src

Route Maintenance



Route Cache (A)
~~G: A, B, D, G~~
G: A, C, E, H, G
F: B, C, F

A Summary of DSR

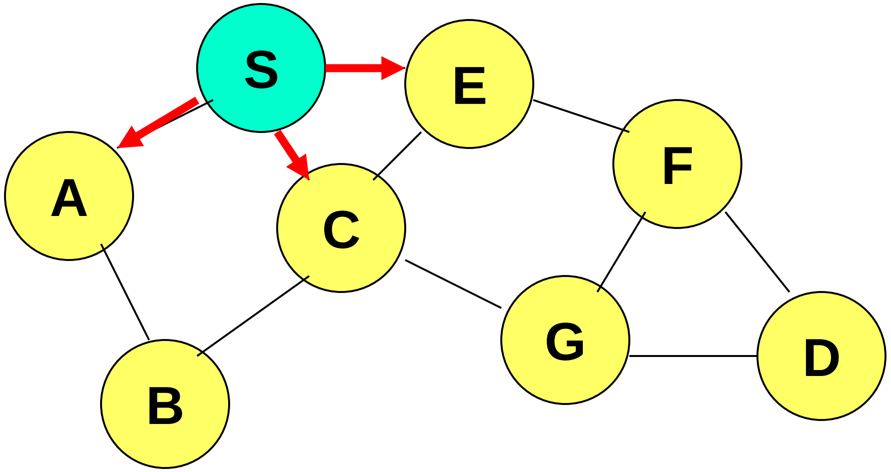
- Entirely on-demand, potentially zero control message overhead
- Trivially loop-free with source routing
- Conceptually supports unidirectional links as well as bidirectional links

- High packet delays/jitters associated with on-demand routing
- Space overhead in packets and route caches
- Promiscuous mode operations consume excessive amount of power

Break…

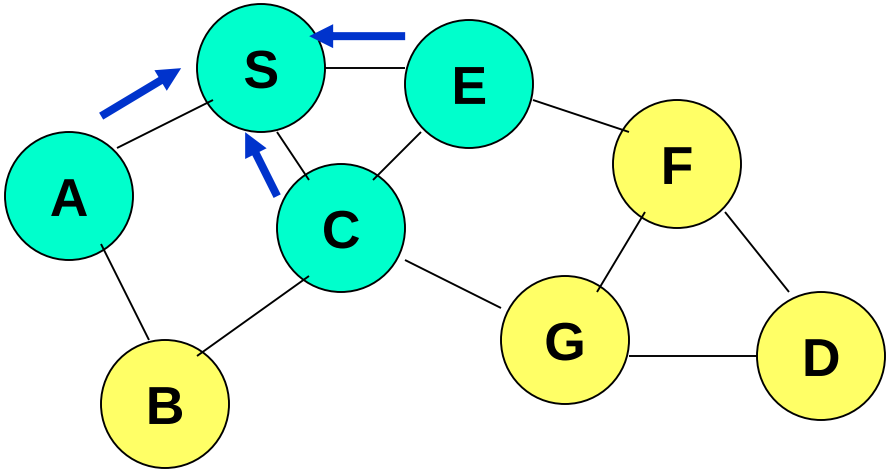
- Then AODV

AODV Routing Protocol



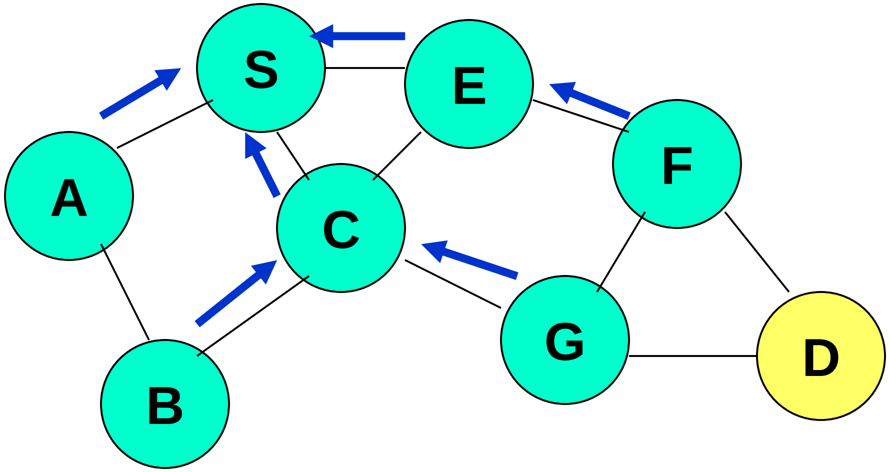
- AODV = Ad Hoc On-demand Distance Vector
- Source floods route request in the network.
- Reverse paths are formed when a node hears a route request.
- Each node forwards the request only once (pure flooding).

AODV Route Discovery



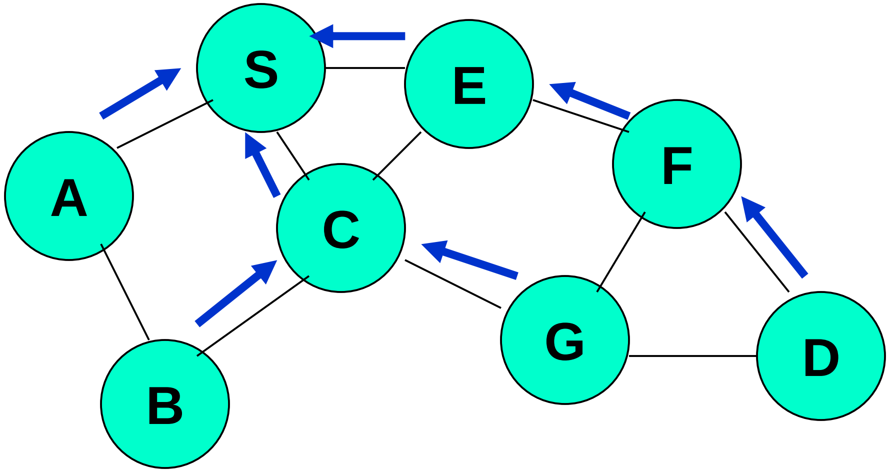
- Source floods route request in the network.
- Each node forwards the request only once (pure flooding).

AODV Route Discovery



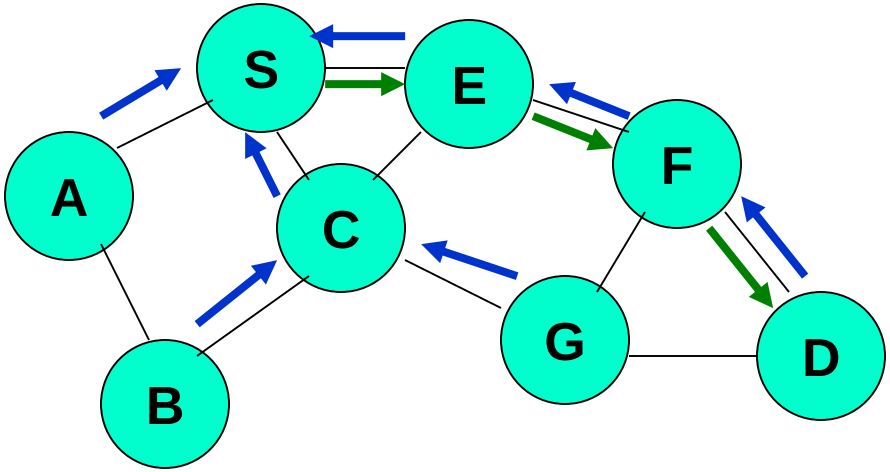
- Uses hop-by-hop routing.
- Each node forwards the request only once (pure flooding).
- Reverse paths are formed when a node hears a route request.

AODV Route Discovery



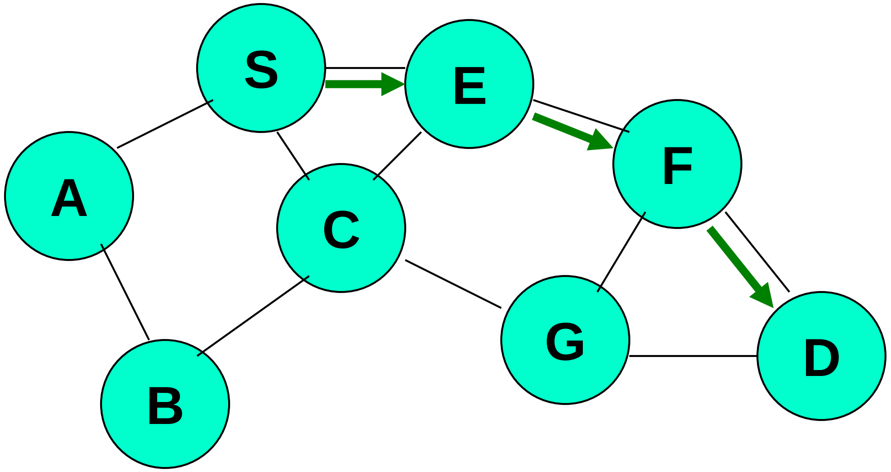
- Route reply forwarded via the reverse path.

AODV Route Discovery



- Route reply is forwarded via the reverse path … thus forming the forward path.
- The forward path is used to route data packets.

Route Expiry

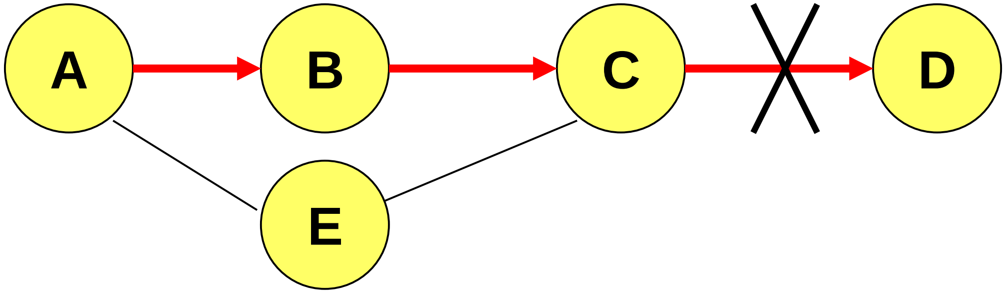


- Unused paths expire based on a timer.

AODV – Optimization

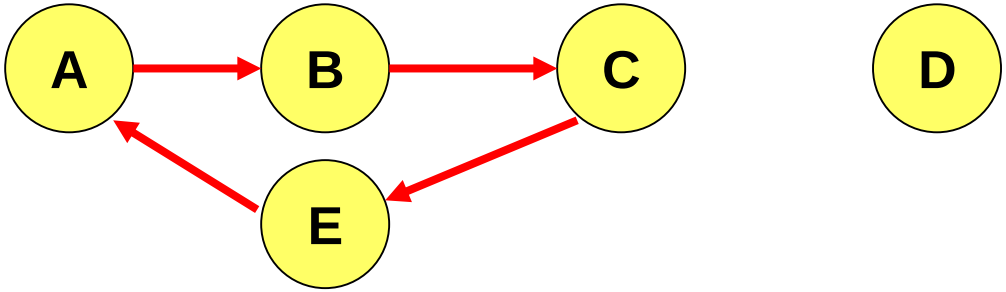
- Useful optimization: An intermediate node with a route to D can reply to route request.
  - Faster operation.
  - Quenches route request flood.
- Above optimization can cause loops in presence of link failures

AODV: Routing Loops



- Assume, link C-D fails, and node A does not know about it (route error packet from C is lost).
- C performs a route discovery for D.
- Node A receives the route request (via path C-E-A)
- Node A replies, since A knows a route to D via node B
- Results in a loop: C-E-A-B-C

AODV: Routing Loops

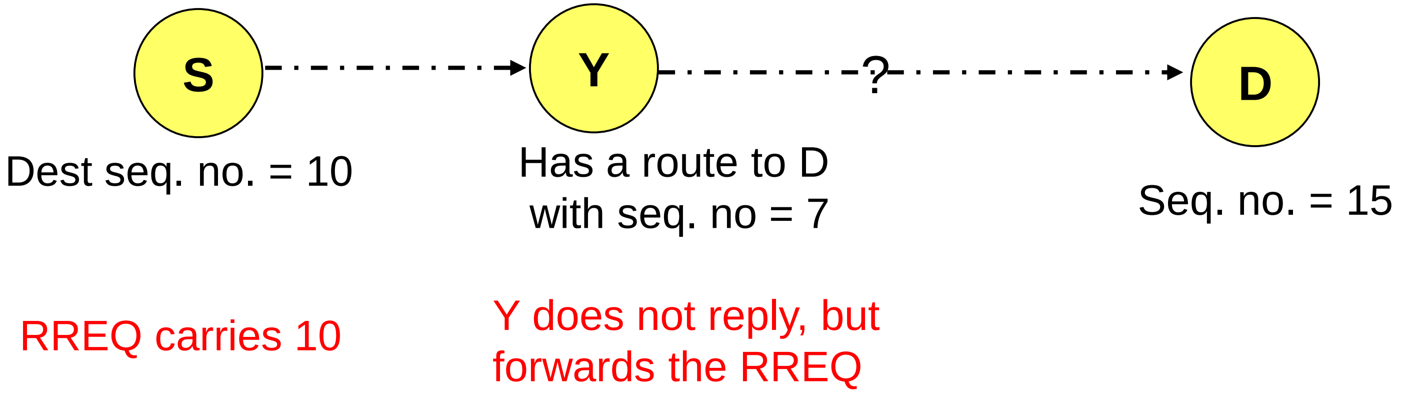


- Assume, the link C-D fails, and node A does not know about it (route error packet from C is lost).
- C performs a route discovery for D.
- Node A receives the route request (via path C-E-A)
- Node A replies, since A knows a route to D via node B
- Results in a loop: C-E-A-B-C

AODV: Use Sequence Numbers

- Each node X maintains a sequence number
  - acts as a time stamp
  - incremented every time X sends any message)
- Each route to X (at any node Y) also has X's sequence number associated with it, which is Y's latest knowledge of X's sequence number.
- Sequence number signifies 'freshness' of the route – higher the number, more up to date is the route.

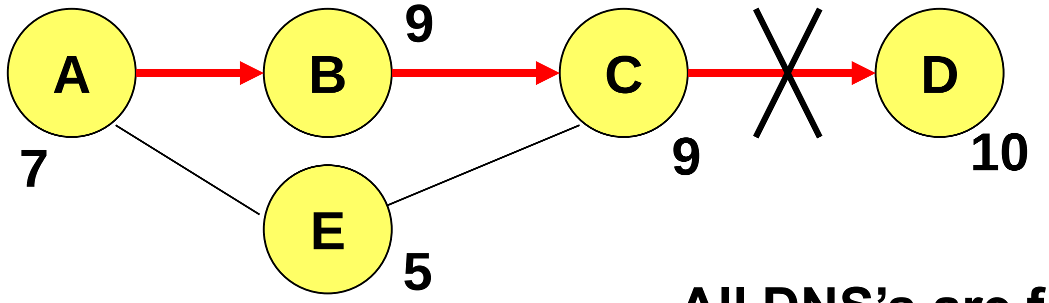
Use of Sequence Numbers in AODV



- Loop freedom: Intermediate node replies with a route (instead of forwarding request) only if it has a route with a higher associated sequence number.

Avoidance of Loop

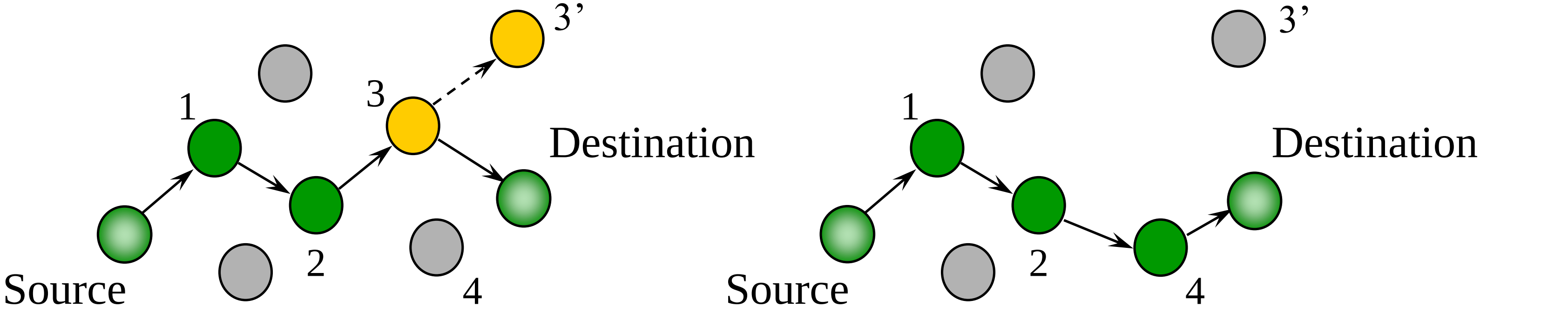
DSN = Destination Sequence Number.



All DNS's are for D

- Link failure increments the DSN at C (now is 10).
- If C needs route to D, RREQ carries the DSN (10).
- A does not reply as its own DSN is less than 10.

Path Maintenance



- Movement not along active path triggers no action
  - If source moves, reinitiate route discovery
- When destination or intermediate node moves
  - upstream node of break broadcasts *Route Error* (RERR)
  - RERR contains list of all destinations no longer reachable due to link break
  - RERR propagated until node with no precursors for destination is reached

Summary: AODV

- At most one route per destination maintained at each node
  - After link break, all routes using the failed link are erased.
- Expiration based on timeouts.
- Use of sequence numbers to prevent loops.
- Optimizations
  - Routing tables instead of storing full routes.
  - Control flooding (incrementally increase ‘region’)

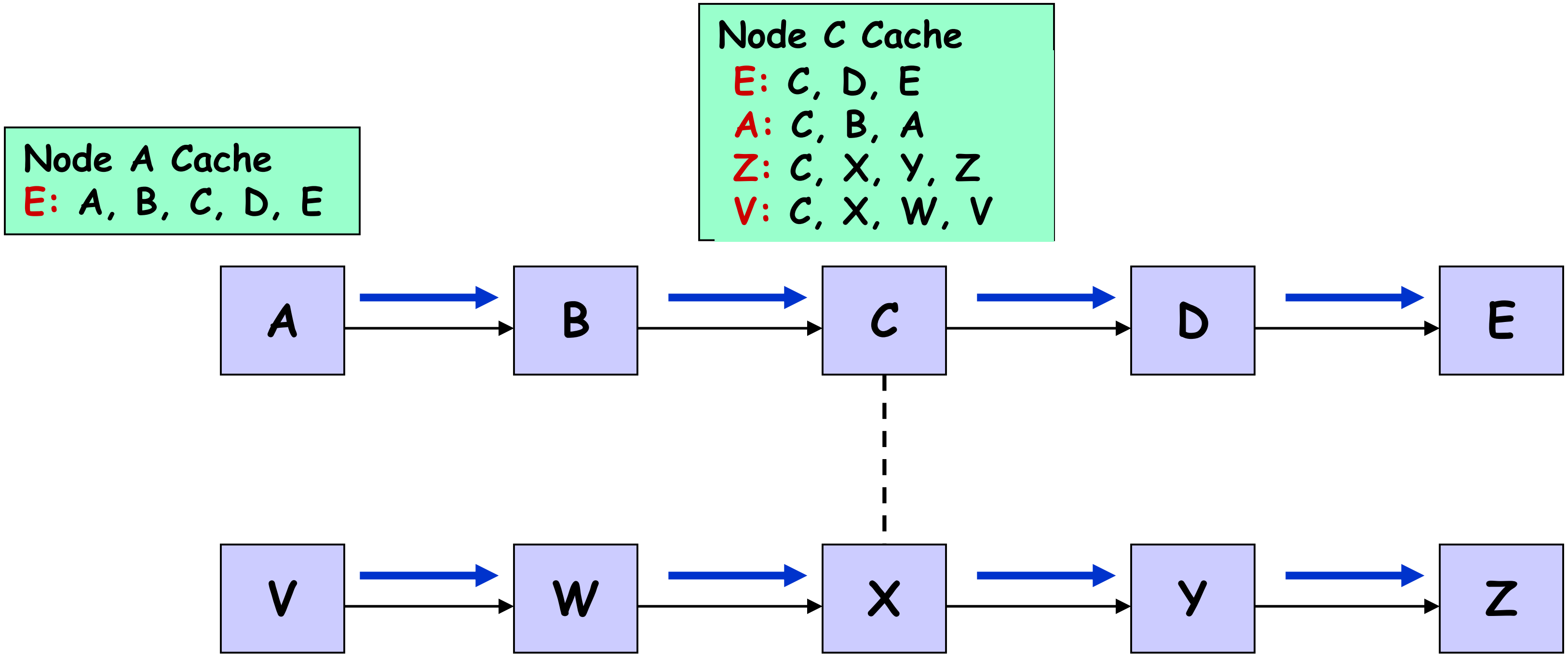
Questions…

- Other notes

Acknowledgements

- DSR Slides:
  - Yinzhe Yu (umn.edu)

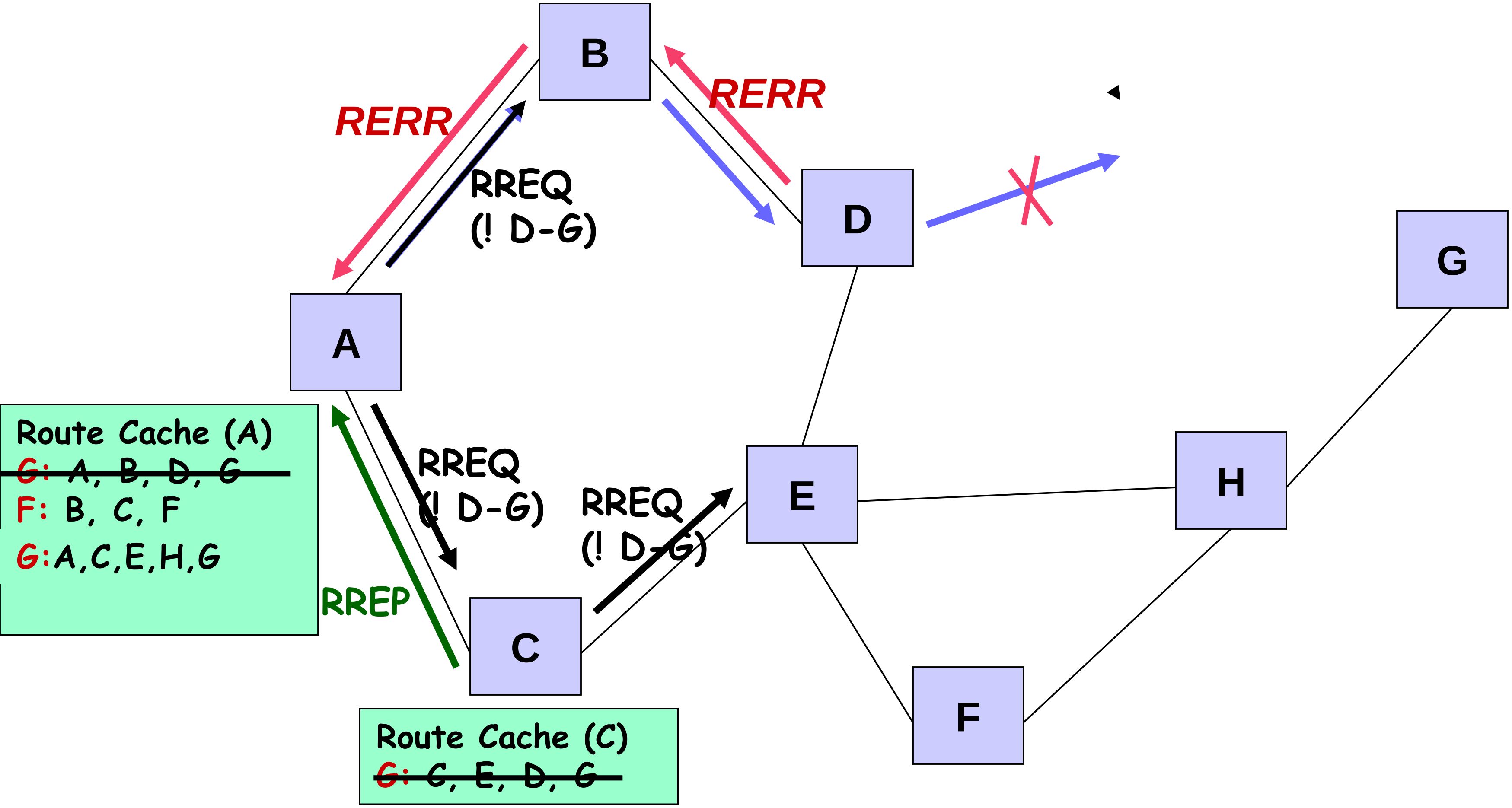
Additional feature #1: Caching Overheard Routes



Node C Cache
E: C, D, E
A: C, B, A
Z: C, X, Y, Z
V: C, X, W, V

Node A Cache
E: A, B, C, D, E

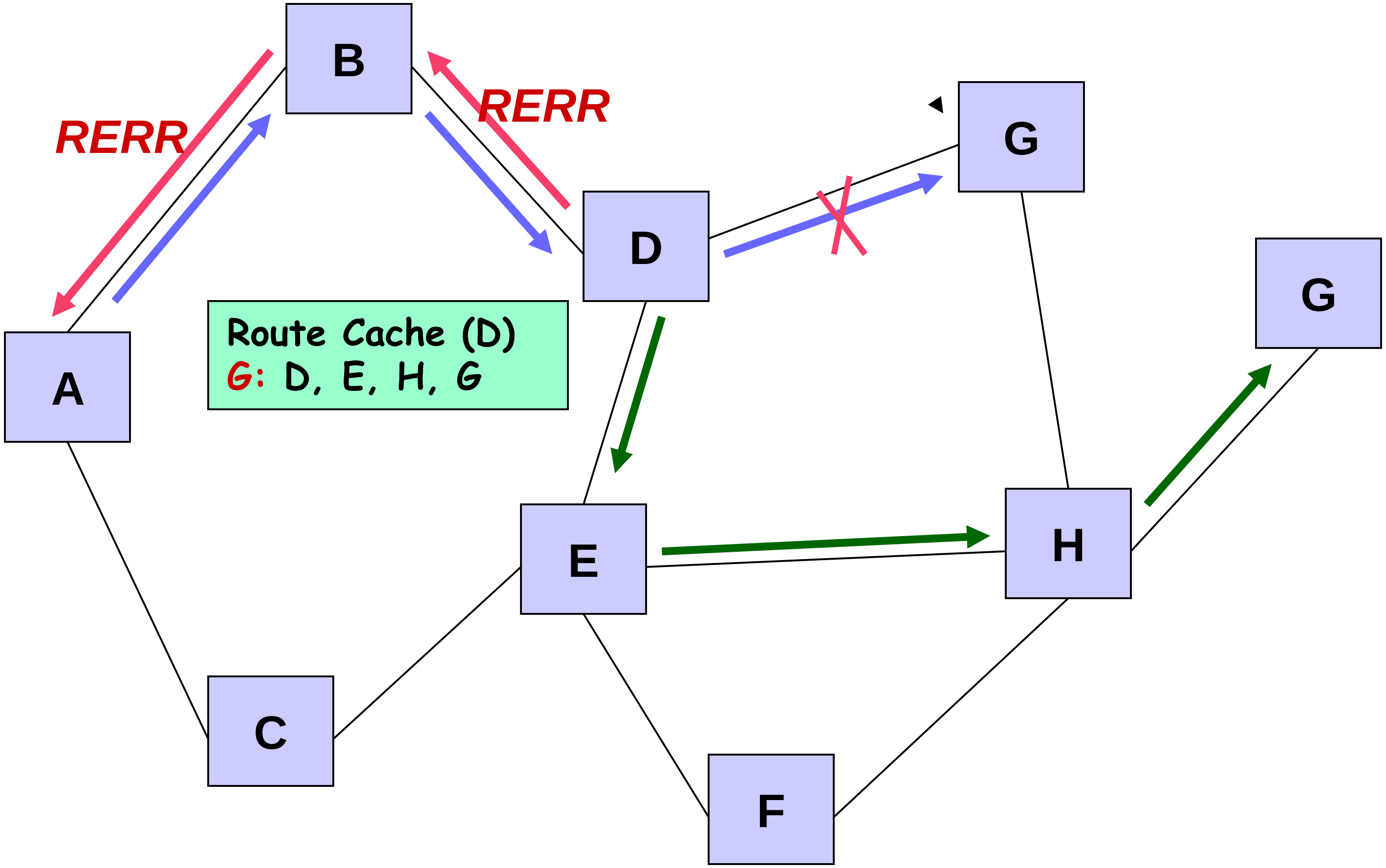
Additional feature #2: RREP with Cached Routes



Route Cache (A)
~~G: A, B, D, G~~
F: B, C, F
G:A,C,E,H,G

Route Cache (C)
~~G: C, E, D, G~~

Additional feature #3: Packet Salvage



Route Cache (D)
G: D, E, H, G

Caution: No double salvage allowed !!!

Proposed Routing Approaches

- Conventional wired-type schemes (global routing, proactive):
  - Distance Vector; Link State
- Hierarchical (global routing) schemes:
  - Fisheye, Hierarchical State Routing, Landmark Routing
- On- Demand, reactive routing:
  - Source routing; backward learning
- Location Assisted routing (Geo-routing):
  - DREAM, LAR etc

Conventional wired routing limitations

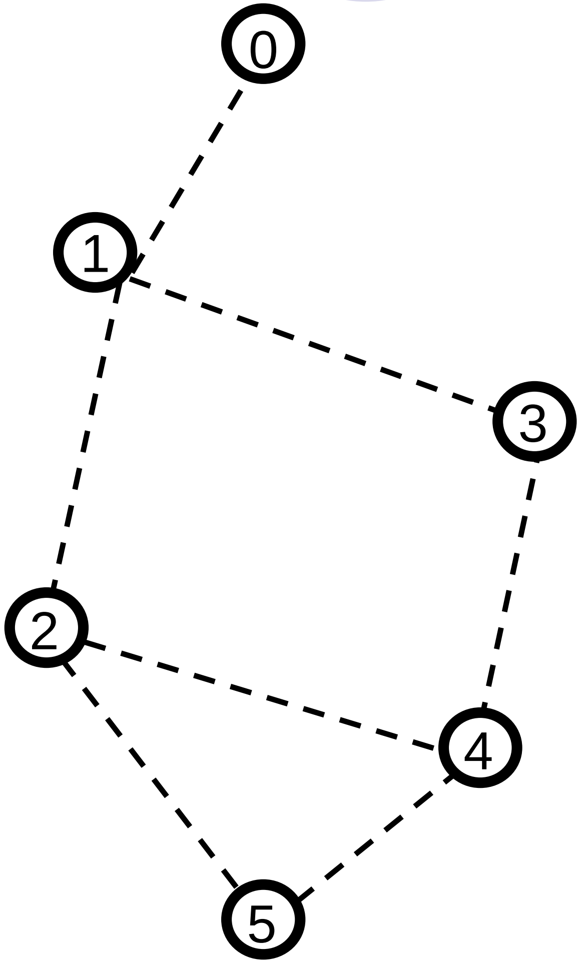
- Distance Vector (eg, Bellman-Ford, DSDV):
  - routing control O/H linearly increasing with net size
  - convergence problems (count to infinity); potential loops
- Link State (eg, OSPF):
  - link update flooding O/H caused by frequent topology changes

CONVENTIONAL ROUTING DOES NOT SCALE TO SIZE AND MOBILITY

Distance Vector

Routing table at node 5 :

Destination	Next Hop	Distance
0	2	3
1	2	2
…	…	…



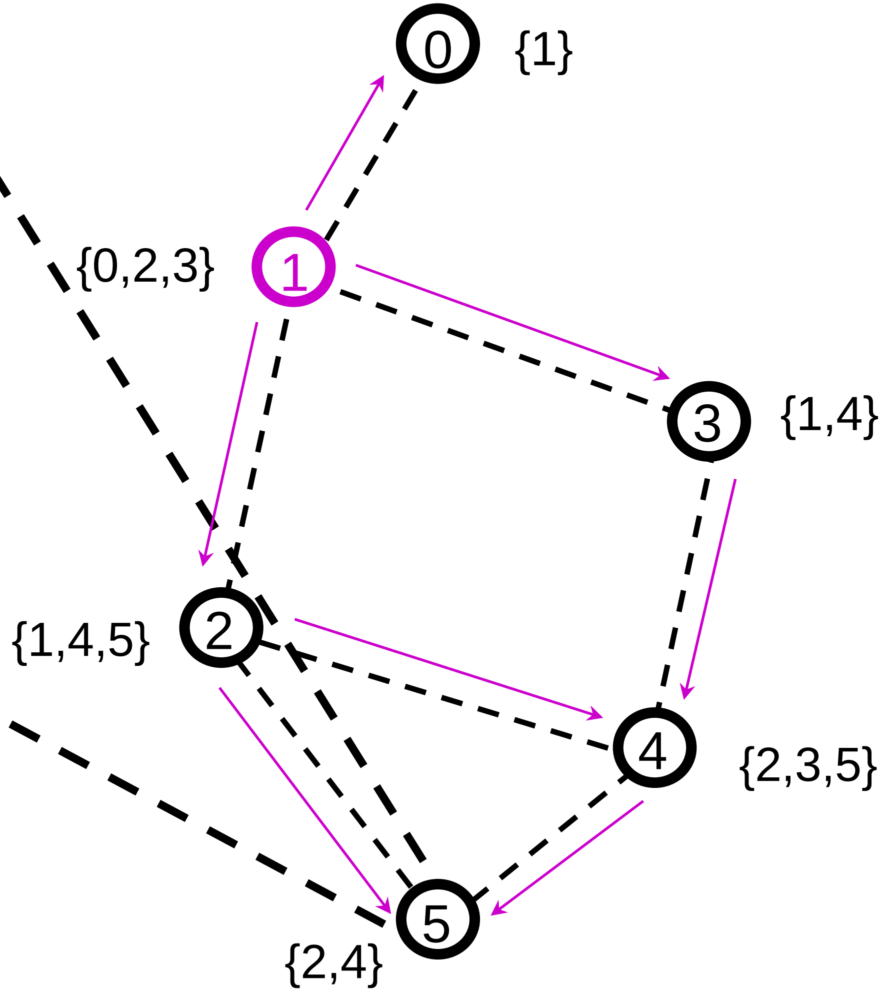
Tables grow linearly with # nodes

Control O/H grows with mobility and size

Link State Routing

- At node 5, based on the link state packets, topology table is constructed:

	0	1	2	3	4	5
0	1	1	0	0	0	0
1	1	1	1	1	0	0
2	0	1	1	0	1	1
3	0	1	0	1	1	0
4	0	0	1	1	1	1
5	0	0	1	0	1	1



- Dijkstra's Algorithm can then be used for the shortest path

Existing On-Demand Protocols

- Dynamic Source Routing (DSR)
- Associativity-Based Routing (ABR)
- Ad-hoc On-demand Distance Vector (AODV)
- Temporarily Ordered Routing Algorithm (TORA)
- Zone Routing Protocol (ZRP)
- Signal Stability Based Adaptive Routing (SSA)
- On Demand Multicast Routing Protocol (ODMRP)
- …