Announcements

• Project proposals.
  – Feedback.

• Class schedule updated.
  – Exam: May 19th.
  – Project presentations: June 10th.
Student Presentations: Schedule

- May 21
- May 26
- May 28
- June 2
- June 4

- Larissa: user mobility and modeling
- Daniel:
- Ben:
- Armando:
- Alan:
- Sam:
- Anuj: IoT
- Maziar:
Today

• Unicast routing.
  – MANETs
Unicast Routing in MANETs
Why MANET routing is challenging?

• No fixed infrastructure.
  – Nodes can have unlimited mobility.
  – So?
• Multiple (wireless) hops to destination.
• Unreliable communication medium.
• All nodes need to participate in routing/forwarding.
  – Also, security issues.
Mobility

- Mobility patterns may vary widely.
  - Stationary nodes (e.g., sensor nodes).
  - Highly mobile nodes (e.g., vehicles).
  - Discrete versus continuous mobility.
  - Structured versus unstructured mobility.

- Mobility characteristics:
  - Pattern.
  - Speed.
  - Direction.
  - Pause time.
MANET Routing Requirements

• Multi-hop paths.
• Loop freedom.
• Self configuring and adaptive to dynamic topology.
• Efficiency, e.g., low consumption of communication bandwidth, energy.
  - Limited signaling overhead.
  - Scalable with number of nodes.
  - Localized effect of topology or flow change.
MANET Unicast Routing

- Many protocols have been proposed.
- Many have been invented specifically for MANETs.
- Many are adapted from protocols for wired networks.
- Can any one protocol work well in all MANET environments?
DV or LS?
DV or LS?

• **Distance-Vector Algorithm:** Routers exchange their distances to known destinations; a router uses the distance vectors received from its neighbors to compute its own distances. *Distributed computation is problem.*

• **Link-State Algorithm:** Routers exchange information about the state of the links in the network; a router uses this information to compute its distances to destinations. *Distributed database problem.*
Proactive or Reactive?
MANET Unicast Routing Taxonomy

• Proactive protocols:
  – A.k.a, table-driven.
  – Traditional routing protocols are proactive.
  – Compute and maintain routes independent on traffic demand/patterns.
  – E.g., OLSR.

• Reactive protocols:
  – Compute and maintain routes “on-demand”.
  – E.g., DSR, AODV.

• Hybrid protocols.
  – E.g., ZRP.
MANET Unicast Routing Taxonomy

[Feeney, 1999]
Trade-offs?

• 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.
Trade-offs?

- Overhead of route discovery/maintenance.
  - Reactive protocols may have lower overhead because routes are determined only if needed.
  - Proactive protocols may result in higher overhead due to continuous route updating (depends on rate of changes).

- Which approach achieves better trade-offs depends on the traffic and mobility patterns.
What about flooding?
Flooding for Data Delivery

- Sender broadcasts data packet to all its neighbors.
- Each node receiving packet forwards it to its neighbors.
- Sequence numbers used to avoid forwarding packet more than once. Why?
- Packet reaches destination if reachable from source. Destination does not forward P.
Flooding

Flooding for Data Delivery

- Represents nodes that are within each other's transmission range.

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

- Nodes J and K both broadcast packet P to node D
  - Since nodes J and K are hidden from each other, their transmissions may collide
  => Packet P may not be delivered to node D at all, despite the use of flooding

Flooding for Data Delivery

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

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

• Advantages:
  – Simplicity.
  – Efficient when rate of information transmission lower than topology changes.
  – Robustness.

• Disadvantages:
  – High overhead.
  – May result in network congestion.
Flooding for the Control Plane

• Many protocols perform flooding of control packets.
  – E.g., route discovery and maintenance.
• Overhead of control packet flooding may be amortized over data packets transmitted.
Dynamic Source Routing (DSR)  
[Johnson96]

- Reactive protocol.
- When node S wants to send a packet to D, and does not have a route to D, node S initiates a route discovery.
- 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

CMPE 257 Spring’15
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 the received RREQ.  
- RREP includes the route from S to D on which RREQ was received by node D.
Route Reply in DSR

RREP [S,E,F,J,D]

Represents RREP control message
Route Reply in DSR

- RREP can be sent by reversing the route in RREQ only if links are guaranteed to be bi-directional.
- If unidirectional (asymmetric) links are allowed, then RREP may need a route discovery for S from D.
  - Unless D already knows a route to S.
  - If a route discovery is initiated by D for a route to S, then the RREP is piggybacked on D’s RREQ.
Processing RREP

• Node S on receiving RREP, caches the route.
• 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 F forwards Route Reply RREP [S,E,F,J,D], F learns route [F,J,D] to D.
  - When node E forwards Data [S,E,F,J,D] it learns route [E,F,J,D] to node D.
  - Nodes may also learn route when it overhears data.
Use of Route Caching

• When S learns that a route to D is broken, it uses another route from its local cache, if such a route to D exists in its cache; otherwise, S initiates route discovery.
• Node X on receiving a RREQ for some node D can send a RREP if X knows a route to 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
When node Z sends a route request for node S, node K sends back a route reply [Z,K,G,C,S] to node Z using a locally cached route. Route caches at K and J limit the flooding of Z’s RREQ.
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 [S,E,F,J,D]) 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 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.
  – New RREQ may need to be issued.
DSR: Advantages

• Routes maintained only between nodes who need to communicate.
  – Reduces overhead of route maintenance.

• Route caching can further reduce route discovery overhead.

• Single route discovery may yield many routes to the destination, due to intermediate nodes replying from local caches.
  – Multi-path routing.
DSR: Disadvantages

• Packet header size grows with route length (network diameter).

• 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.
DSR: Disadvantages

• Increased contention if too many route replies come back due to nodes replying using their local cache.
  – “RREP” storm problem.
  – Reply storm may be eased by preventing a node from sending RREP if it hears another RREP with a shorter route.
DSR: Disadvantages

• An intermediate node may send RREP using a stale cached route, thus polluting other caches.
  – This problem can be eased if some mechanism to purge (potentially) invalid cached routes is incorporated.
  • Static timeouts.
  • Adaptive timeouts based on link stability [interesting class project].
AODV
AODV

- Route Requests (RREQ) are forwarded similarly to DSR.
  - When a node re-broadcasts a RREQ, it sets up a reverse path pointing towards the source.
  - AODV assumes symmetric (bi-directional) links.
- When the intended destination receives a RREQ, it replies by sending a RREP.
- RREPs travel along the reverse path set-up when RREQ 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
AODV Route Discovery: Observations

- RREQ contains source and destination IP address, current destination seq. number (incremented as a result of loss of prior route), and broadcast id (incremented for every RREQ).
  - Source IP + bcast id uniquely identifies RREQ: nodes do not forward RREQs they have forwarded recently.
  - RREQ processing: node creates reverse route table entry for RREQ source with TTL.
  - If node has “unexpired” route to destination in its table with sequence number >= RREQ’s, it replies to RREQ with Route Reply (RREP) back to source.
  - Otherwise, broadcast RREQ onward.
Destination Sequence Number

• When node D receives route request with destination sequence number N, D sets its sequence number to N, unless it is already larger than N.

• Node’s own sequence number is monotonically increasing.
  – Sequence number is incremented after neighborhood topology change.
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

- An intermediate node has current route to destination, responds to RREQ with RREP.
- RREP contains source and destination IP, current sequence number, number of hops to destination.
  - If destination, then destination seq. #.
  - Else, node’s current record of destination’s seq. #.
- Node receiving RREP sets up forward path to destination.
- If multiple RREPs received, node forwards first one. Later RREPs discarded unless greater seq. # or smaller # of hops.
Route Reply Example

Represents links on path taken by RREP
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.
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.

• Routing table entry maintaining a forward path is purged if not used for active_route_timeout interval.
  – If no data 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).