Wireless Networks: What’s Different?

- Varying channel quality and dynamic conditions
  - Separation (and merging) of networks
  - Asymmetric links
  - Shortest hop may be worst choice
  - Redundant links – routes need to converge quickly

Wireless Networks: What’s Different?

- Varying channel quality and dynamic conditions
  - Separation (and merging) of networks
  - Asymmetric links
  - Shortest hop often worst choice
  - Routes need to converge quickly

- Scarce transmission capacity
  - Overhead conspicuous; No frequent periodic updates

- Low compute power
  - No extensive route computations

- Low energy
  - Low control message overhead
  - Low control message frequency (to realize sleep mode)

- And if that’s not bad enough, add node mobility…
Routing in Wireless Mobile Networks

- Imagine hundreds of hosts moving
  - Routing algorithm needs to cope with varying wireless channel and node mobility
  - Routes may break, and reconnect later

Where’s RED guy

Metrics for Routing?

- Minimal:
  - Number of nodes
  - Loss rate
  - Delay
  - Congestion
  - Interference
  - Overhead
  - ...

- Maximal:
  - Stability of the logical network
  - Battery run-time
  - Time of connectivity
  - ...

HOW TO DO ROUTING IN AD-HOC NETWORKS?

Routing Protocols

- **Proactive protocols**
  - Determine routes independent of traffic pattern
  - e.g., traditional link-state and distance-vector routing

- **Reactive protocols**
  - Maintain routes only if needed

- **Hybrid protocols**
  - Maintain routes to nearby nodes
  - Discover routes for far away nodes
Trade-Off

- Latency of route discovery
  - Proactive protocols may have lower latency
  - Reactive protocols higher because a route discovery from X to Y will be initiated 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 do continuous route updates / maintenance

- Which approach achieves a better trade-off depends on the traffic and mobility patterns

THE CONCEPT OF FLOODING
How well does it work in a wireless network?
Data Delivery Using Flooding

- 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

What can go wrong?

Flooding Example

Broadcast transmission

- Represents a node that receives packet P for the first time
- Represents transmission of packet P
Flooding Example

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

Flooding Example

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

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

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

![Diagram of a network showing nodes and edges]

- 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

- **Advantages:**
  - Simplicity
  - May be more efficient if infrequent communication
    - Route setup / maintenance not worth it
    - Especially, when changing topology / mobility
  - Potentially higher robustness to path failure
    - Because of multi-path redundancy

- **Disadvantages:**
  - Potentially, very high overhead
    - Packets delivered to too many nodes who don’t need them
  - Potentially lower reliability of data delivery
    - Reliable broadcast is difficult
    - Hidden terminal because no channel reservation
Flooding as a Building Block

- 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)
(Johnson 96)

A Reactive Protocol
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”

- **Route Discovery:**
  - Source node S floods Route Request (RREQ)
  - Each node appends own identifier when forwarding RREQ

---

**Route Discovery in DSR**

- Broadcast transmission

- [S] Represents transmission of RREQ

- [X,Y] Represents list of identifiers appended to RREQ
Route Discovery in DSR

- Same issues as in flooding

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

- Same issues as in flooding

- Nodes J and K both broadcast RREQ to node D
  - Since J and K are hidden from each other, their transmissions may collide
- Node D does not forward RREQ
  - Because it is the intended target of route discovery
**Route Discovery: Route Reply**

- **Route Reply:**
  - 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 reversal => links need to be bi-directional**
  - If unidirectional (asymmetric) links are allowed
    - Then RREP may need a route discovery for S from node D
  - 802.11 links always bi-directional (since ACK is used)

---

**Route Reply in DSR**

![Diagram of RREP in DSR]

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

  ![Diagram with routes and nodes labeled]

  Represents RREP control message
Data Delivery in 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
  - Packet header grows with route length

- Intermediate nodes use the source route in packet
  - To determine to whom a packet should be forwarded

DSR Optimization: Route Caching

- 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 D,

- 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 by overhearing Data packets
**Route Caching: Speed up 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 its local cache.

**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!

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

- Stale caches can adversely affect performance
  - Local caching by other nodes may result in stale routes
    - Even on multiple tries by sender host
  - May require several trials before finding a good route

Query Localization

- **Path locality heuristic:**
  - Look for a new path that contains at most $k$ nodes that were not present in the previously known route

- **Old route is piggybacked on a Route Request**

- **Route Request is forwarded only if:**
  - Accumulated route in the RREQ contains at most $k$ new nodes that were absent in the old route

- 😊: Limits propagation of the route request
**Query Localization: Example**

Node F does not forward the RREQ since it is not on any route from S to D that contains at most 2 new nodes.

**DSR: Advantages**

- Routes maintained reactively
  - Reduces overhead of maintenance

- Route caching can reduce route discovery overhead

- Discovery of multiple routes at D
DSR: Disadvantages

- Packet header size grows with route length
- Flood of RREQs may potentially reach all nodes
- 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 RREPs come back
  - Due to nodes replying using their local cache
  - Route Reply Storm problem
    - May be eased by preventing a node from sending RREP
      - If it hears another RREP with a shorter route
- An intermediate node may RREP using a stale cache
  - Thus polluting other caches
  - Can be eased by purging invalid cached routes [Hu00]