Slide 1

18-452/18-750
Wireless Networks and Applications
Lecture 12: Ad Hoc Networks

Peter Steenkiste

Spring Semester 2020
http://www.cs.cmu.edu/~prs/wirelessS20/

1

Slide 2

Ad Hoc Routing

- **Definition and challenges**
- **Routing:**
  - » **Classes: proactive versus reactive routing**
  - » **Proactive, table based routing: DSDV**
  - » **Reactive routing DSR**
  - » **Geographic routing: GPSR**
- **Discussion**

- **Structured as a survey**
  - » **But a bit longer since it covers more material**

2

Slide 3

Ad Hoc Networking

- **Goal: Communication between wireless nodes**
  - » **No infrastructure – network must be self-configuring**
- **It may require multiple hops to reach a destination**
  - » **Nodes are traffic sources, sinks and forwarders**

3

Slide 4
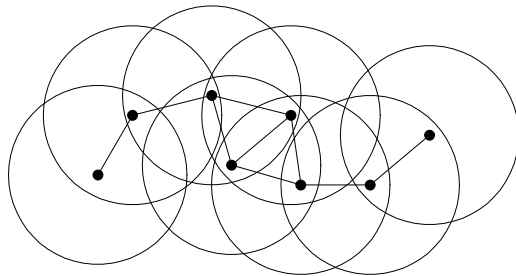
Ad Hoc Networking Challenging

- **All the challenges of wireless, and more:**
  - » **Decentralized: nobody is in charge – no planning!**
  - » **No fixed infrastructure, random network design**
  - » **Generic ad hoc can be arbitrarily bad: limited batteries, malicious nodes, high mobility, low density, ..**
- **Precise challenges depend on the application domain, e.g., vehicular networks versus first-responder networks versus sensor networks**
  - » **Domain focus typically simplifies the problem**
  - » **Most research papers address the complex general case**
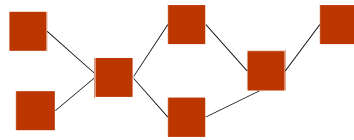- **The big challenge: Routing**

4

Traditional Routing vs Ad Hoc

- **Traditional wired network:**
  - » **Well-structured**
  - » **~O(N) nodes & links**
  - » **All links work equally well**
  - » **Sensible topology**
  - » **Links are independent**
- **Ad Hoc wireless network**
  - » **N^2 links - but many stink!**
  - » **Topology may be really weird**
  - » **Links interfere with each other**
  - » **Dynamic link conditions affect link quality unpredictably**
    - – **So the topology also changes constantly!**
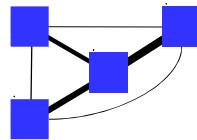
Forwarding Packets is expensive

- **Assume link throughput is X**
  - » **X depends on the WiFi version, distance, fading, …**
- **What is the throughput of a chain?**
  - » **Basic:** $A \rightarrow B \rightarrow C$
  - » **Or:** $A \rightarrow B \rightarrow C \rightarrow D$
  - » **Or:** $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \ldots$
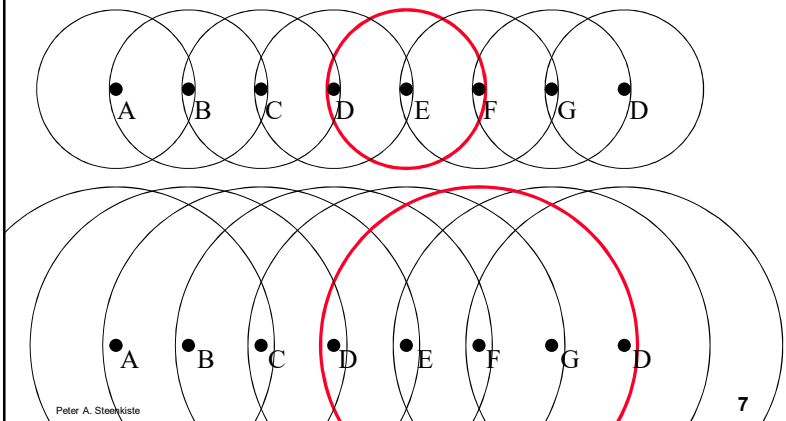- **Considering:**
  - » **Wired versus wireless**
  - » **Assume minimum power for radios.**
  - » **Now assume a dense network, i.e., all radios can hear each other**
  - » **Now assume dynamic link conditions, mobility, ..**

---

2 Simple Examples

$$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G \rightarrow D$$

Ad Hoc Routing

- **Classes: proactive versus reactive routing**
- **Proactive, table based routing: DSDV**
- **Reactive routing DSR**
- **Geographic routing: GPSR**

Proactive or Table-based Protocols

- **Proactive: routers maintain routes independently of the need for communication**
  - » **Similar to wired networking – uses forwarding table**
- **Route update messages are sent periodically or when network topology changes**
- **Low latency – forwarding information is always readily available**
- **Bandwidth might get wasted due to periodic updates**

Reactive or On-Demand Routing

- **Routers discover a route only when there is data to be sent**
- **Saves energy and bandwidth during periods of inactivity or low activity**
- **Route discovery introduces significant delay for the first packet of a new transfer**
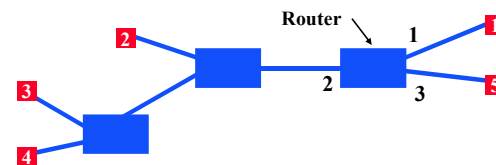- **Bad if many nodes send packets to many destinations at random times or in unstable networks**

Ad Hoc Routing

- **Classes: proactive versus reactive routing**
- **Proactive, table based routing: DSDV**
- **Reactive routing DSR**
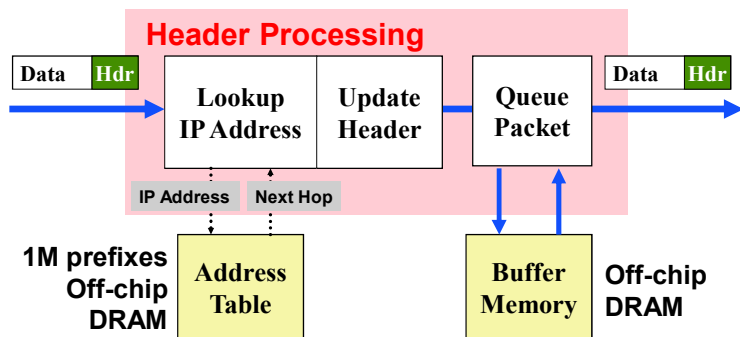- **Geographic routing: GPSR**

Packet Forwarding versus Routing

Destination Address	Port
IP1	1
IP2	2
IP3	2
IP4	2
IP5	3

- **Routing finds a path between two end-points**
- **Forwarding receives a packet and decides which egress port to send it out on**
- **Most networks use a routing protocol to pre-calculate paths between every pair of nodes**
  - » **Routers put the result in a forwarding table**
- **Forwarding only requires a lookup in the forwarding table – fast!**

Generic Router Architecture

Header Processing

| Data | Hdr |

Lookup IP Address → **Update Header** → **Queue Packet** → | Data | Hdr |

IP Address | Next Hop

1M prefixes Off-chip DRAM — **Address Table**

Buffer Memory — **Off-chip DRAM**

13

13

Traditional Routing Solutions

- **Link state routing**
  - » Each router obtains a full topology of the network by having nodes periodically flood connectivity information
  - » Each router then uses Dijkstra's algorithm to locally calculate its forwarding table
  - » Bad fit for ad hoc: LS flooding creates a lot of traffic and relies on all routers having a consistent view of network
- **Distance vector**
  - » Each router tells its neighbors its shortest path to each destination
  - » Routers then use the "best" option provided to them
  - » Based on the Bellman-Ford algorithm
  - » More promising for ad hoc: has lower routing overhead
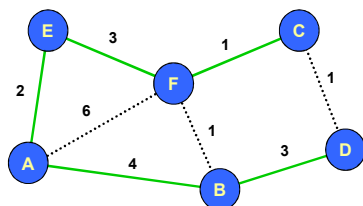 - » Challenge is how to avoid routing loops (details omitted)

Background

14

Routes from Node A

Forwarding Table for A

Dest	Cost	Next Hop
A	0	A
B	4	B
C	6	E
D	7	B
E	2	E
F	5	E

- **Set of shortest paths forms tree**
  - » Shortest path spanning tree
- **Solution is not unique**
  - » E.g., A-E-F-C-D also has cost 7

Background

15

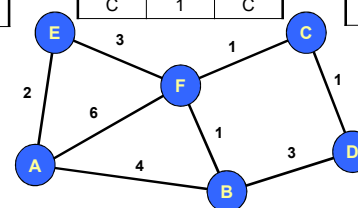## Different View: How to Get to Node C

Forwarding Table for E

Dest	Cost	Next Hop
C	4	F

Forwarding Table for F

Dest	Cost	Next Hop
C	1	C

Forwarding Table for C

Dest	Cost	Next Hop
C	-	-

Forwarding Table for A

Dest	Cost	Next Hop
C	6	E

Forwarding Table for B

Dest	Cost	Next Hop
C	2	F

Forwarding Table for D

Dest	Cost	Next Hop
C	1	C

Background

16

Peter A. Steenkiste

Page 4

Distance-Vector Method

Initial Table for A		
Dest	Cost	Next Hop
A	0	A
B	4	B
C	∞	–
D	∞	–
E	2	E
F	6	F
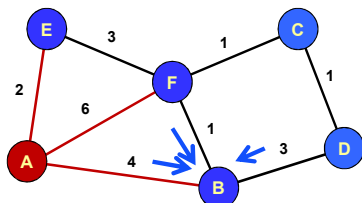
- **Router periodically exchanges tables with neighbors**
  - » **Contains the cost and next hop of best known path to all destination**
- **Routers pick the best of the candidates paths**
  - » **May be the path it is currently using already**

Background

Destination-Sequenced Distance Vector (DSDV)

- **By Perkins and Bhagvat**
- **DV protocol specifically designed for wireless**
  - » **Exchange of routing tables**
  - » **Routing table: the way to the destination, plus the cost**
- **Each node advertises its presence and tables**
  - » **Maintains fresh routes by periodically sending updates to neighbors**
  - » **Update for each destination: hop count, sequence number**
- **Uses sequence number to avoid loops**
  - » **Advertisements for each destination include sequence number that is incremented for each update**
  - » **Since loops are created by advertisements based on old routing information, they can be detected**

DSDV Properties

- **Keep the simplicity of Distance Vector**
- **Guarantees Loop Freeness**
  - » **While adding minimal overhead**
- **Allow fast reaction to topology changes**
  - » **Make immediate route advertisement on significant changes in routing table**
- **Drawbacks:**
  - » **Overhead can be significant under dynamic conditions and mobility**
  - » **Many ad hoc networking scenarios do not require all-all connectivity, so some effort is wasted**

Based on: cone.informatik.uni-freiburg.de/teaching/vorlesung/manet-s07/exercises/DSDV.ppt


Overview

- **Ad hoc networking concept**
- **Proactive versus reactive routing**
- **Proactive, table based routing: DSDV**
- **Reactive routing DSR**
- **Geographic routing: GPSR**
- **Wireless link metrics**
- **Ad hoc networking examples**

Dynamic Source Routing (DSR) Key Features

- **On-demand route discovery finds route only when it is needed**
  - » **Avoid overhead of periodic route advertisements**
- **Uses source routing: path information is stored in the packet header**
- **DSR control functions:**
  - » **Route discovery: senders obtain route to destination**
  - » **Route maintenance: detect changes in topology and update routes that are affected**
  - » **Route caching: nodes cache route information to avoid route discovery for every packet**
    - – **Caching can be done on sender and intermediate routers**
    - – **Flush broken routes from cache**

| Hop 1 |
|-------|
| Hop 1 |
| … |
| Dest |
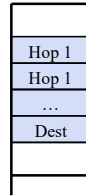
21

DSR Route Discovery

- **Source broadcasts a route-request towards the destination**
  - » **The request includes a (partial) path from source to destination**
- **Each node forwards the request by adding own address to the path and re-broadcasting**
- **Requests propagate outward until:**
  - » **The destination is found, or**
  - » **A node that has a route to the destination is found**

22

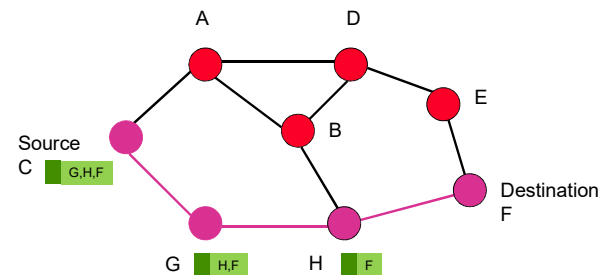
Route Request is Re-Broadcasted until Destination is Reached
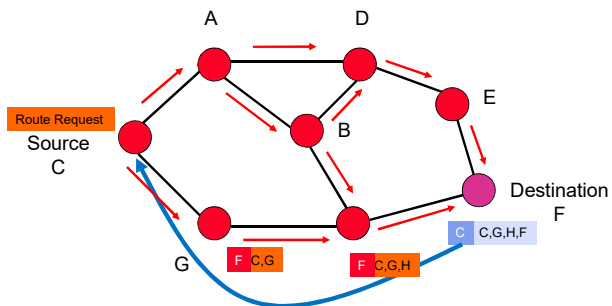
23

C Transmits a Packet to F

24

Forwarding Route Requests

- **A request is forwarded by a node if:**
  - » Node is not the destination
  - » Node not already listed in recorded source route
  - » Node has not seen request with same sequence number
  - » IP TTL field may be used to limit scope
- **Destination copies selected route into a Route-reply packet and sends it back to Source**
  - » I.e., route reply uses reverse path of the route selected by the destination
  - » Destination can choose one of the paths, e.g., first path (with shortest delay)

Route Cache

- **All source routes learned by a node are kept in a Route Cache**
  - » Routes are learned by overhearing route requests/responses
  - » Reduces cost of route discovery
- **If an intermediate node receives a route request for a destination and has an entry for the destination in its route cache, it**
  - » Responds to the request using the cached information
  - » Does not propagate the request any further
- **Nodes use their local route cache when asked to send a data packet**
  - » If route is missing, they initiate a route request

Basic Route Maintenance

A → B → C ✖ D → E

- **When forwarding a packet, each sender must get an acknowledgement from the next hop**
  - » Will retransmit the packet up to a limit if needed
- **If no ACK is received it drops the packet and notifies the sender A of the broken link**
- **A removes the route from its route cache and ..**
  - » Intermediate nodes also remove any cached entries to E
- **Will do a new route discovery when it sends another packet to E**
  - » It is left up to TCP to recover from the packet loss

Discussion

- **Source routing is good for certain types of networks and traffic loads**
  - » For example, stable traffic flows and/or a small number of sender-receiver pairs
  - » Stable network topologies, e.g., with limited mobility
- **Periodic messages are avoided**
- **Significant delay for the first packet to a destination**
  - » Also, need to buffer packets

Overview

- **Ad hoc networking concept**
- **Proactive versus reactive routing**
- **Proactive, table based routing: DSDV**
- **Reactive routing DSR**
- **Geographic routing: GPSR**
- **Wireless link metrics**
- **Ad hoc networking examples**

## Greedy Perimeter Stateless Routing (GPSR)

- **Use *positions* of neighboring nodes and packet destination to forward packets**
  - **No connectivity or global topology is assumed – no forwarding or path information anywhere!**
  - **Nodes are assumed to know their location**
  - **Need a mechanism for address-to-location look up**
- **Two forwarding techniques is used**
  - ***Greedy forwarding*, if possible**
  - ***Perimeter forwarding*, otherwise**

GPSR – Greedy forwarding

- **A sender/forwarder *x* chooses to forward to a neighbor *y* such that $\{d_{xy} + d_{yD}\}$ is minimum**
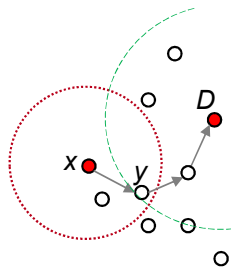
GPSR – Perimeter forwarding

- **What happens if a node does not have a neighbor that is closer to the destination?**
- **Right Hand Rule: you forward the packet to your first neighbor clockwise around yourself**
  - **Traverse an interior region in *clockwise* edge order**
  - **Guaranteed to reach a (reachable) destination for planar graph**
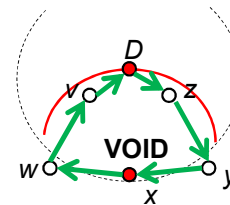
This sequence of edges traversed is called a *PERIMETER*

Discussion

- **Many variants:**
  - » Hybrid approaches mix different solutions
  - » Hierarchical: create a hierarchy of clusters
- **Many proposals for optimizations**
  - » Links use different frequencies, multiple radios, etc.
  - » Link metrics that consider interference level, …
- **Best solutions is highly context dependent: density, traffic load, degree of mobility, …**
- **Practical applications are rare by exist:**
  - » Mesh networking: wireless, last mile Internet access
  - » Challenging conditions: first responders, military, ..
  - » In the future maybe vehicular, drones, …

33

Summary

- **Ad hoc networks face many challenges**
  - » Bad links, interference, mobility, …
  - » Makes routing very challenging
  - » Limited support: hardware and driver limitations
- **Many proposals!**
  - » Proactive routing: variants of "wired" routing protocols
  - » Reactive routing: only establish a path when it is needed
  - » Geographic routing: use destination location info only
  - » Many variants and extensions
- **Specific challenges depend on the application domains**
  - » Mesh versus vehicular
  - » Active area of research

34

Some References

- **DSR:**
  - » www.cs.rice.edu/~dbj/pubs/aw-dsr.pdf
- **DSDV:**
  - » www.cs.jhu.edu/~cs647/class-papers/Routing/p234-perkins.pdf
- **GPSR:**
  - » www.eecs.harvard.edu/~htk/publication/2000-mobi-karp-kung.pdf

- **ETX:**
  - » pdos.csail.mit.edu/papers/grid:mobicom03/paper.pdf
- **ETT**
  - » http://www.cs.jhu.edu/~cs647/class-papers/Routing/p114-draves.pdf

35