EE [redacted] 597
April 10, 2012

<u>Recap</u>: Backpressure Scheduling

How to support any flow arrival rate vector $\vec{\lambda} \in \Lambda$?

- maxWeight algorithm
  pick independent sets $S(t)$ of edges & commodities over those edges at each time to maximize

$$\sum_{(i,j) \in S(t), c} \omega_{ij}^{c}(t) = \sum_{i,j \in S(t), c} \underbrace{\left(Q_i^c(t) - Q_j^c(t)\right)}_{\text{Queue differential "backpressure"}} R_{ij}$$

<u>BCP</u>: distributed implementation of backpressure scheduling for convergecast

$$W_{ij} = \left(Q_i - Q_j - \underbrace{V \cdot ETX_{ij}}\right) R_{ij}$$

other key elements:
- gradient
  - LIFO gives better delay
  - overall benefits:
    - higher capacity / throughput
    - responsiveness to dynamics (interference, sink mobility)

MANET's.

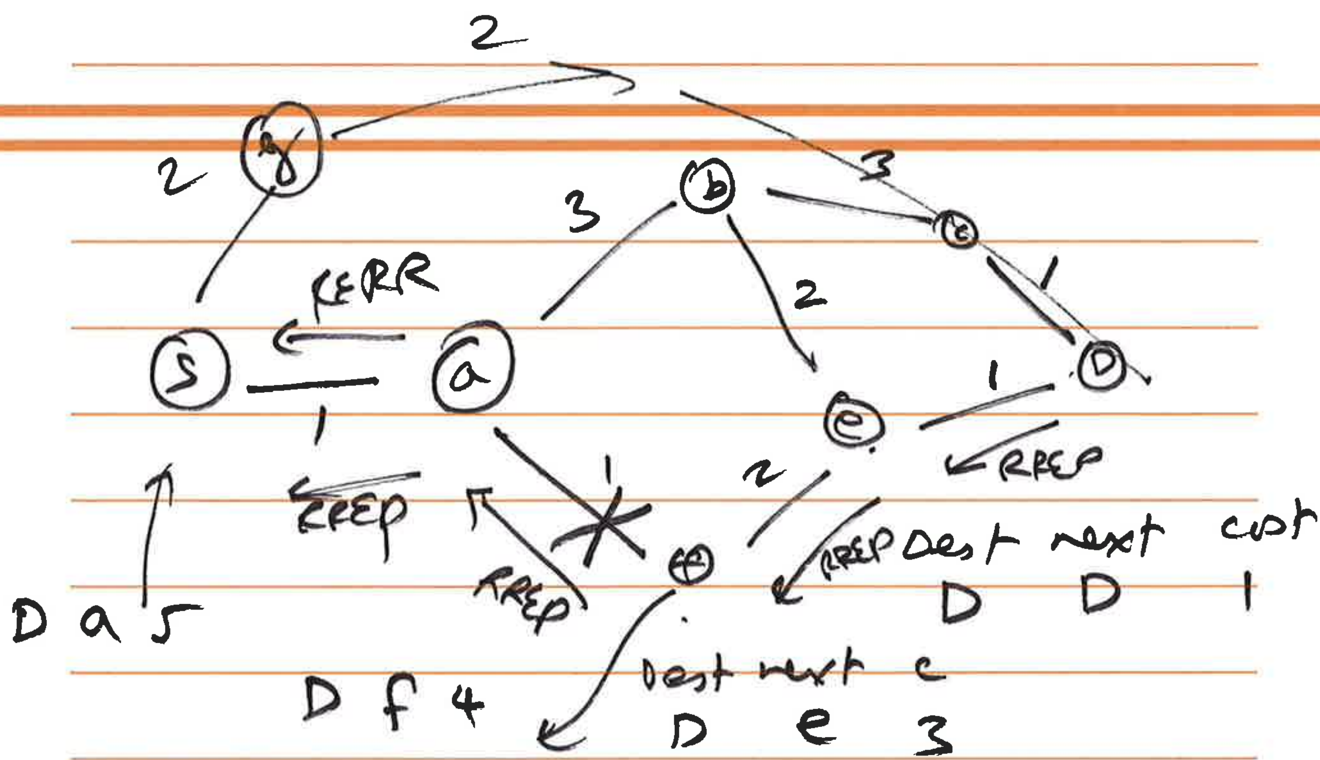
Reactive Routing Protocols:
build routes to destinations when needed / on demand.

AODV - Ad Hoc On Demand Distance Vector

Route Discovery:

- source floods a route request (RREQ) packet through the network looking for the destination.
- Intermediate nodes accumulate costs & forward only pkts w/ lower cost than earlier frwded packets.
- Destination, generates a RREP (route reply) packet to its neighbor that sent the lowest cost RREQ. Each node does the same till the RREP reaches back to the source.
- Source can start sending packets once it receives RREP packet.

While the RREP packets were being forwarded back to the source, the intermediate nodes install the routing entry for the destination.



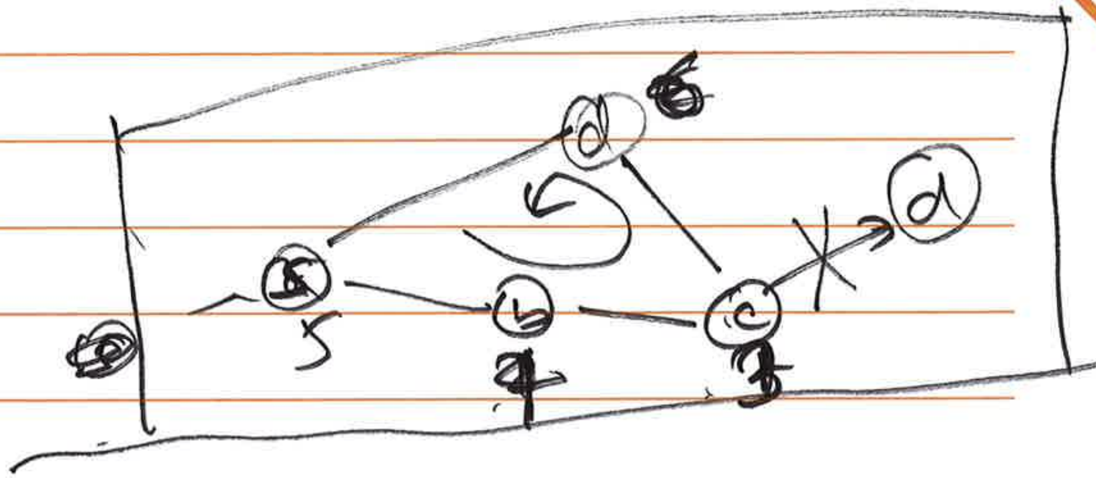
<u>note</u>: AODV will not guarantee the shortest path at all times. A new shorter path will not be discovered.

Route Repair: Link breaks are detected based on link layer beacons / acks.

the node immediately upstream of the broken link recognizes that ~~this~~ has happened & either floods or sends along the reverse path a RERR (route error) packet.

Upon receipt of the RERR packet the source reinitiates route discovery.

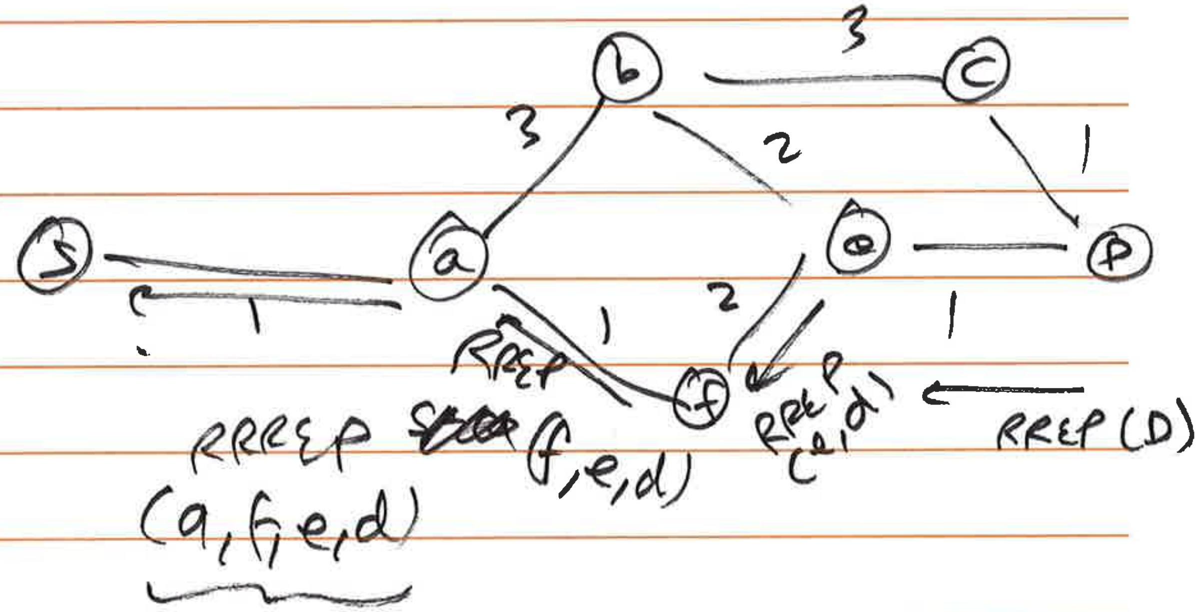
Implementation also uses sequence numbers in all the RREQ packets & RREP packets to ensure freshness & prevent looping.

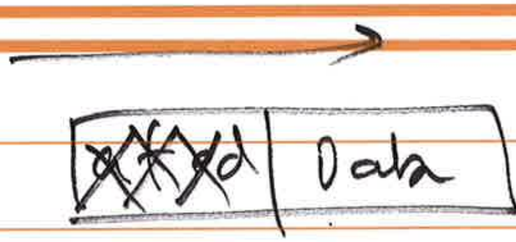


Count to infinity problem of distance vector protocols (stale information in intermediate routers)

Alternative reactive routing protocol: DSR

Dynamic Source Routing





no state is maintained in the intermediate routers ; the state is described in the source routing header of each packet.

Trellisware: MANET protocol using ~~se~~ transmitter diversity.

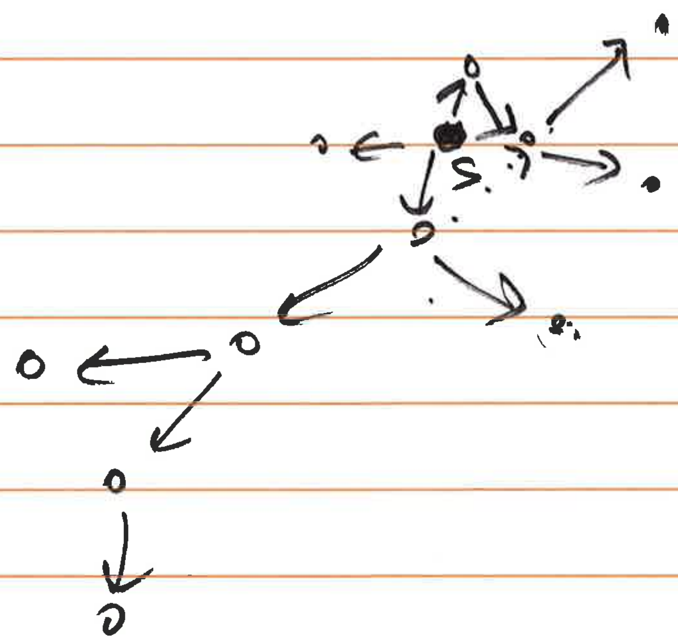
Barrage Relay Networks

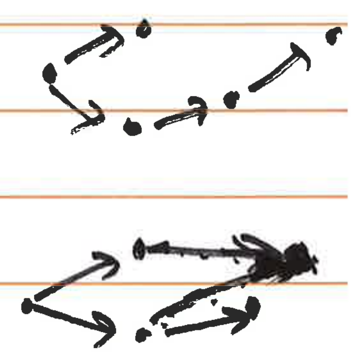
- The nodes in the network are all time-synchronized
- At any given time slot there is only 1 pkt in the air in the entire network
- Each packet is broadcast/flooded to all nodes in the network.
- Does not require any routing state to be maintained

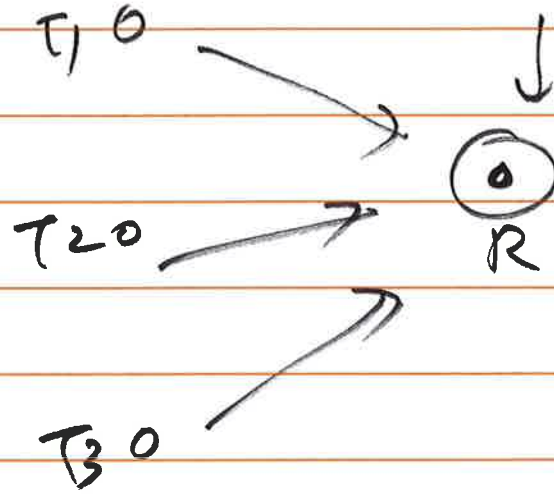
completely eliminate the contend-before-send MAC protocol typical of MANET's or multi-hop networks (e.g. 802.11 CSMA)

- the first time a node hears a packet it relays it at the very next time slot.

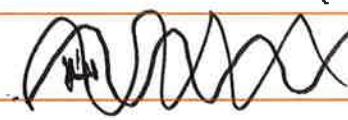
Transmit diversity: increases range & reliability







at the receiver need to be able to handle time shifted versions of the signal.

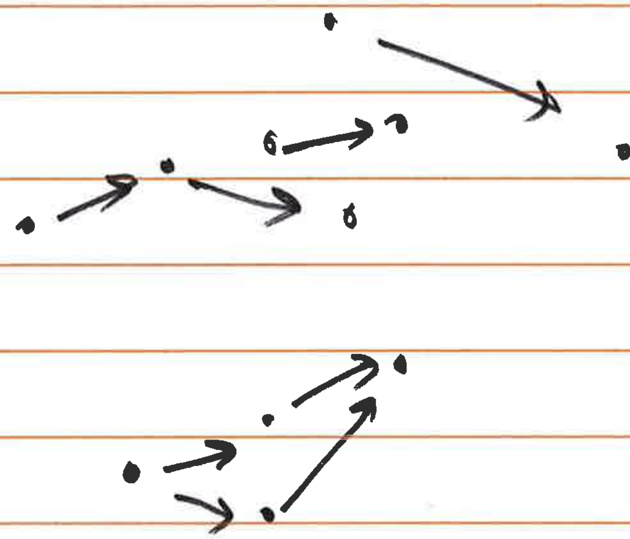


2.4GHz = 2.4×10^{9}.

$\lambda = \frac{c}{f} = 3 \times 10$ ×

Delay Spread factors = multipath + transmit synchronization

At the Phy layer these radios can handle the multi-delayed components of the signal.



1. Static Wireless Networks: Mesh / WSN

- ETX metric (OSPF w/ ETX)

collection tree protocol → CTP ETX-based shortest path tree (distance-vector)

- Anypath (ExOR)
- Backpressure scheduling & routing (BCP)

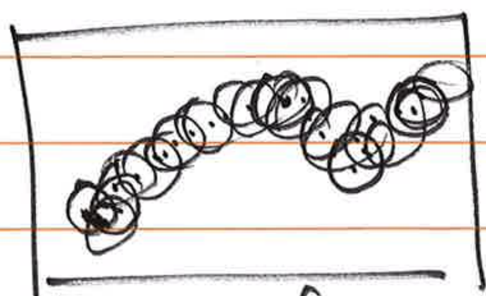
2. Mobile Ad Hoc Networks

- Reactive Routing Protocols AODV & DSR
- Barrage Relay Networks

3. Intermittently connected Mobile Networks

ICMN : encounters between nodes are sparse. The network is rarely if ever instantaneously connected end to end.

radio range << mobility range
& # of nodes is not high with respect to the mobility range.



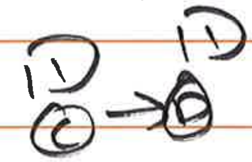
Routing schemes for such a network

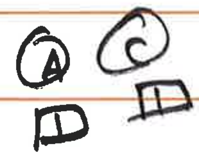
Wait to Deliver: when A wants to send data to B, it waits till B is in range.

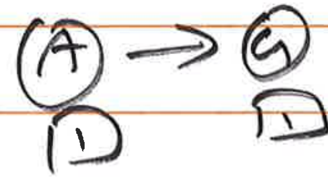
Pros: good throughput / link quality, low energy usage

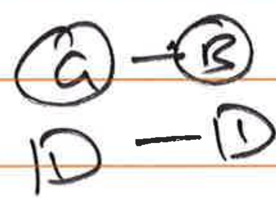
Con: Potentially very high delay

Epidemic Routing (multicopy): whenever another node is encountered, copy the packet.









Pros: · Guaranteed delivery
· Delay optimal
} at low loads.

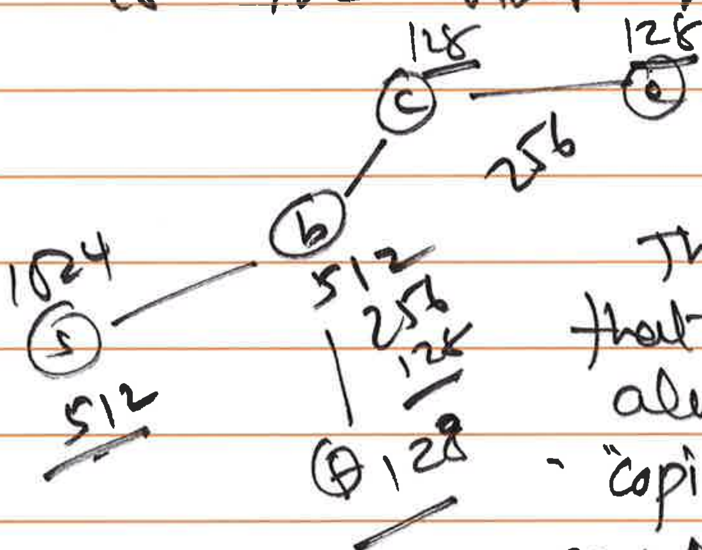
Con: · may need large buffer size
· throughput could be affected (at high loads: high delay & lost pkts)
· high energy usage

Spray & wait : K-copy routing.

Initially source maintains "K" (virtual) copies of the packet

(spray phase) any node that has "m" (virtual) copies of a packet that encounters a node w/ no copies of that packet transfers m/2 of them to the other node.

e.g.



This ensures that there are always "1024" "copies" f.f the packet. i.e. actual # of copies always less than 1024.

goal: (d)

no copying takes place except to destination node if $m = 1$. (wait phase)

---

① Wait to Deliver

② Epidemic Routing

k-copy Spray & Wait

how is it related?

$K = 1$ Spray & wait $\equiv$ is wait to deliver

$k = 2^{N+1}$ " is epidemic

↑ # of nodes in network

~~$\log_2 K_{init}$~~ ~~[illegible]~~

$K_{init} \rightarrow \frac{K_{init}}{2} \; . \; . \; . \; . \; .$

after n encounters
still > 1

$K_{init} > 2^N$.

or

$\log_2 K_{init} > N$