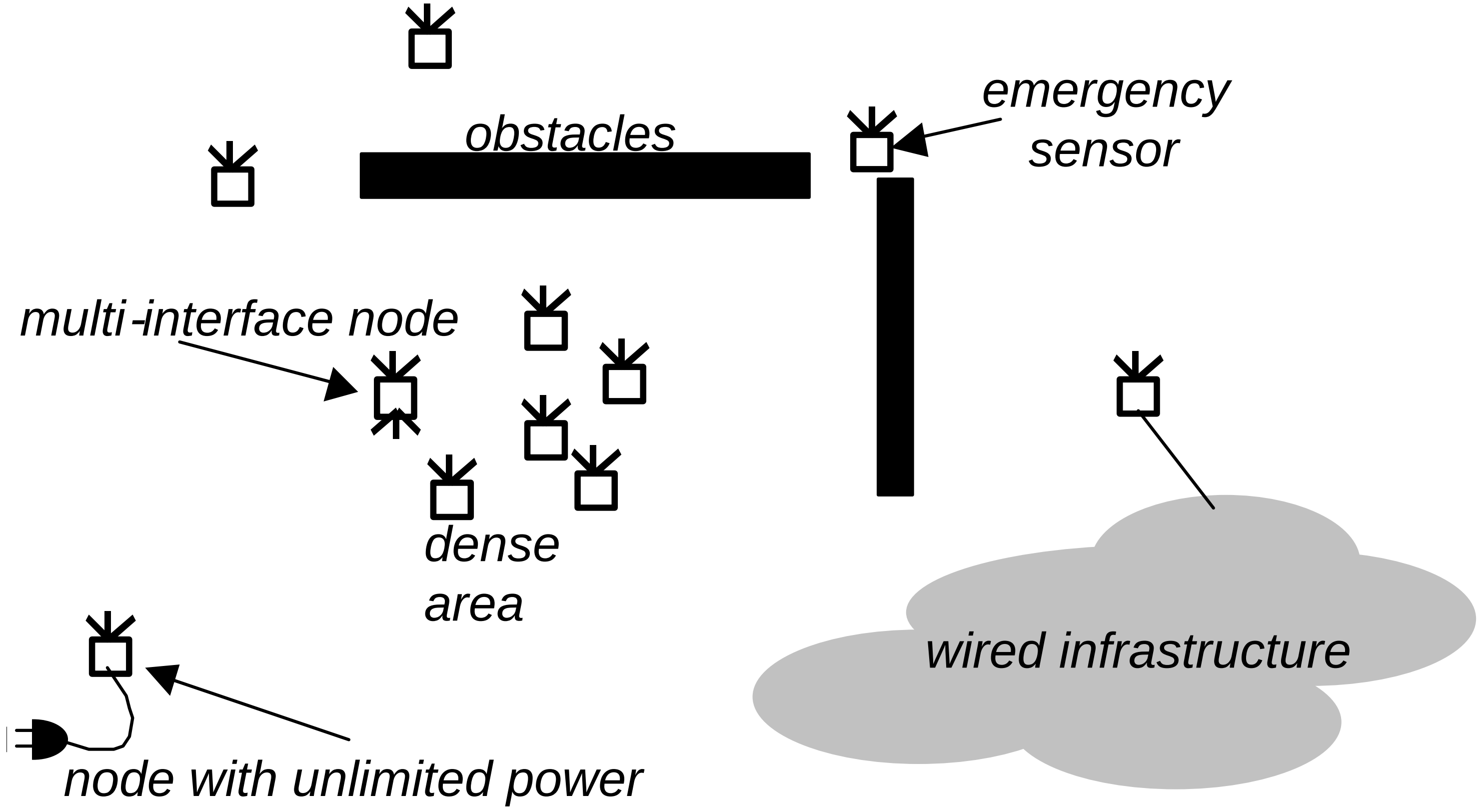
Wireless Multihop Ad Hoc Networks

Guevara Noubir
noubir@ccs.neu.edu

Infrastructure vs. Ad Hoc Wireless Networks

- Infrastructure networks:
  - One or several Access-Points (AP) connected to the wired network
  - Mobile nodes communicate through the AP
- Ad hoc network:
  - Mobile nodes communicate directly with each other
  - Multi-hop ad hoc networks: all nodes can also act as routers
- Hybrid (nodes relay packets from AP):
  - Goal: increase capacity, reduce power consumption, and guarantee a minimum service



obstacles

emergency
sensor

multi-interface node

dense
area

wired infrastructure

node with unlimited power

Constraints

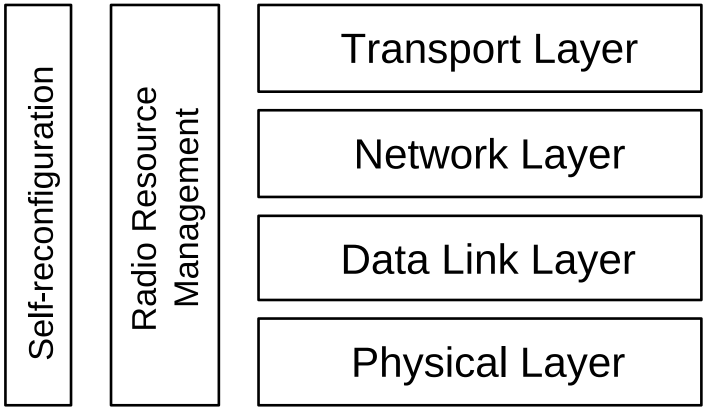
- Limited radio spectrum
- Broadcast Medium (collisions)
- Limited power available at the nodes
- Limited storage memory
- Connection QoS requirements (e.g., delay, packet loss)
- Unreliable network connectivity (depends on the channel)
- Dynamic topology (i.e., mobility of nodes, density)
- Need to provide a full coverage
- Need to enforce fairness

Parameters

- Use of various coding/modulation schemes
- Use of packets fragmentation
- Use of various transmission power level
- Use of smart antennas and MIMO systems
- Use of multiple RF interfaces (multiple IF characteristics)
- Clustering and backbone formation
- Planning of the fixed nodes location
- Packets scheduling schemes
- Application adaptivity

Adaptivity and Cooperation

- Classical networking stacks have only minimum interaction between adjacent layers



- Multi-hop wireless ad hoc networks require more cooperation between layer because:
  - Channel variation and network topology changes affect the application
  - Routing versus single hop communication considerably affects the medium access control (MAC) performance
  - Collisions versus channel fading affects both the physical layer and the MAC
  - Battery power has implications on all layers

Adaptive Coding

- Example:
  - ½ rate convolutional code (K=5) versus uncoded communication
  - Channel with two states: E_b/N_0 = 6.8 dB or 11.3 dB (AWGN), L=200 Bytes

E_b/N_0	**BER**		**FER**		**Nb_Transmit**		**Total_Tx_Bytes**	
	UC	**½ CC**	**UC**	**½ CC**	**UC**	**½ CC**	**UC**	**½ CC**
6.8dB	10^{-3}	10^{-7}	**0.8**	$1.6\ 10^{-4}$	**5**	**~ 1**	**5*200**	**2*200**
11.3dB	10^{-7}	**~ 0**	$1.6\ 10^{-4}$	**~ 0**	**~ 1**	**~ 1**	**200**	**2*200**

- Need to estimate the channel and adapt to it
- Differentiate between congestions and a bad channel condition
- Use of Hybrid-ARQ?

Adaptive Fragmentation

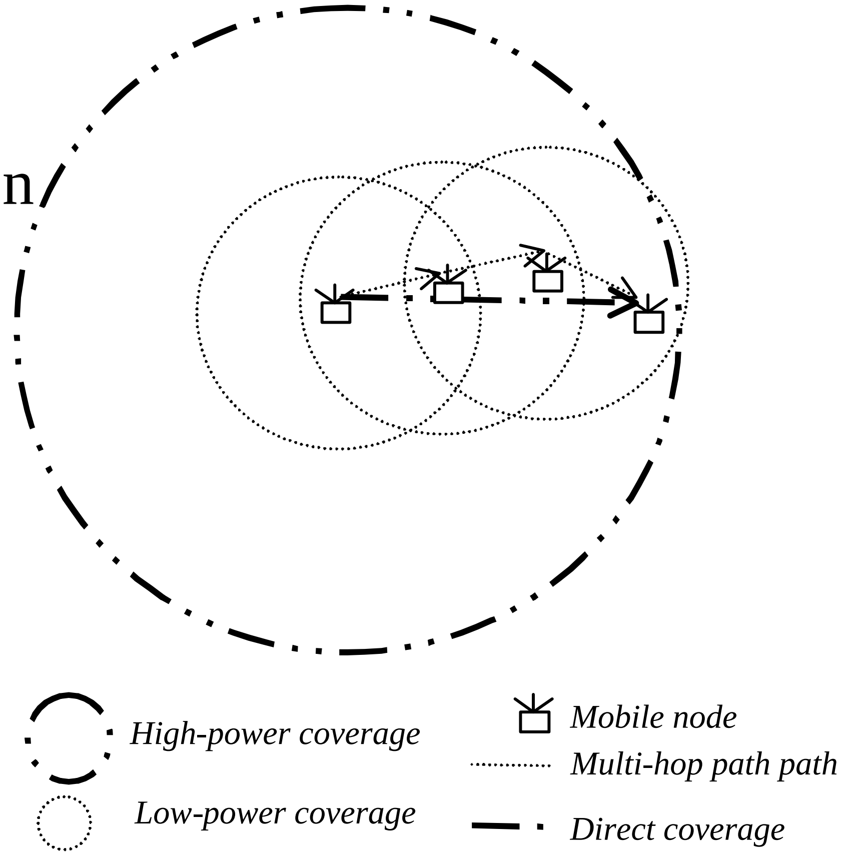
- Example:
  - To transmit a frame of length 200 Bytes, we can fragment into 4 frames of length 50 Bytes (+ 10 Bytes overhead)

BER	**FER**		**Nb_Transmit**		**Total_Tx_Bytes (incl. overhead)**	
	L=60B	**L=200B**	**L=60B**	**L=200B**	**L=60B**	**L=200B**
10^{-3}	**0.38**	**0.8**	**1.6**	**5**	**384**	**1000**
10^{-7}	**~ 0**	**~ 0**	**~ 1**	**~ 1**	**240**	**200**

- Need to estimate the channel and adapt to it

Multiple Power Levels

- Using multi-hop transmission (*h* hops) and reducing the transmission power accordingly
  - Increases capacity (factor of *h*)
  - Reduces overall power consumption (by a factor of *h*)
- In asymmetric environments
  - Low power node can encode data and transmit it at low power
  - Powerful nodes can decode use higher transmission power



Problems of Multi-Hop Routing

- Routing:
  - How to maintain up-to-date information on the network topology? Routing messages overhead
  - How to determine number of hops
  - How to estimate buffers size
- Higher delay
- Risk of congestion on nodes

Practical Approaches

- Solving sub-problems independently
  - Improving TCP to be wireless aware
  - Routing in multi-hop wireless ad hoc networks: DSDV, DSR, AODV, TORA, FSR
    - Not power or resource aware. Single hop whenever possible (no interaction with the MAC of higher layers)
  - Fragmenting packets according to the channel performance
  - Adapting coding/modulation scheme to the channel
  - Adapting transmission power to destination
- There is a need for a global approach:
  1. Combine: transmission power, coding, and fragmentation
  2. Add routing
  3. Add medium access control

- Engineering perspective: what minimal subset of functionalities do we need to implement to achieve near optimal performance?
  - What minimal set of coding/modulation schemes? What power levels do we need?

Routing Protocols: DSDV

- Destination-Sequenced Distance Vector:
  - Each node maintains a routing table listing:

    <next-hop, metric, SeqNum>:

    The favored route is changed if a new route with higher SeqNum is received, or if the new route has equal SeqNum and lower metric
  - Nodes send advertisement with evenly increased SeqNum
  - When a node detects a broken link he sends an advertisement with ∞ metric and SeqNum=PrevSeqNum+1
  - Damping fluctuations:
    - Fluctuations are due to out-of-order arrival of route advertisement
    - Proposed solution: maintain a settling time estimation for routes
    - Route with an ∞ metric are advertised without delay

Dynamic Source Routing I <draft-ietf-manet-dsr-05.txt:2001>

- Split routing into discovering a path and maintaining a path
- Discover a path
  - only if a path for sending packets to a certain destination is needed and no path is currently available
- Maintaining a path
  - only while the path is in use, one has to make sure that it can be used continuously

- No periodic updates needed!

Dynamic Source Routing II

- Path discovery
  - broadcast a *Route Request* packet with destination address and unique ID
  - if a station receives a broadcast packet
    - if the station is the receiver (i.e., has the correct destination address) then return the packet to the sender (path was collected in the packet): *Route Reply*
    - if the packet has already been received earlier (identified via ID) then discard the packet
    - otherwise, append own address and broadcast packet
  - sender receives packet with the current path (address list)
- Optimizations
  - limit broadcasting if maximum diameter of the network is known
  - caching of address lists (i.e. paths) from passing packets (overhearing)
    - stations can use the cached information for path discovery (own paths or paths for other hosts)

Dynamic Source Routing III

- Maintaining paths
  - after sending a packet
    - wait for a layer 2 acknowledgement (if applicable)
    - listen into the medium to detect if other stations forward the packet (if possible)
    - request an explicit acknowledgement
  - if a station encounters problems it can inform the sender of a packet or look-up a new path locally

Routing Protocols: DSR and FSR

- Dynamic Source Routing has two mechanisms:
  - Route Discovery:
    - A *Route Request* packet is flooded through the network
    - *Route reply* is sent back to the source
    - Optimization: cache of source routes (learned or overheard)
  - Route Maintenance:
    - On discovery of a broken link (e.g., nodes moved out of range), a *Route Error* is sent to the source => use an alternate cached route or rerun *Route Discovery*
- Fisheye State Routing:
  - Maintains link state information with a frequency that depends on the fisheye scope distance
  - Nodes do not need to have a very precise information on far away nodes

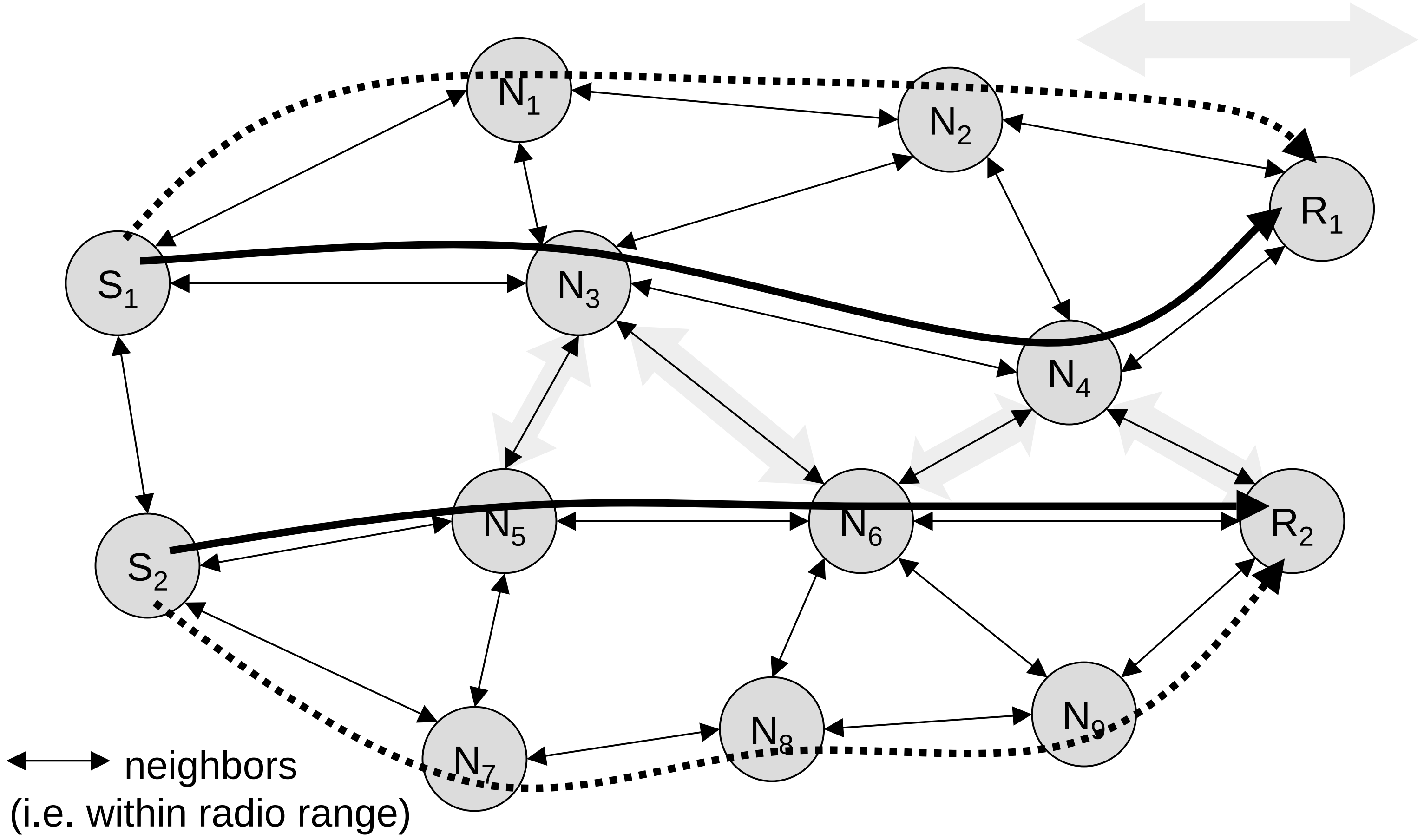
Parameters of IEEE802.11

- IEEE802.11 has three mechanisms that can be used to improve performance under dynamic channels:
  - Fragmentation (also used to avoid collision)
  - Multiple coding/modulation schemes (8 schemes)
  - 8 power levels

- Multiple coding/modulation schemes will be available with 802.11a products over 5GHz
- Currently parameters are statically configured

Interference-Based Routing

- Routing based on assumptions about interference between signals



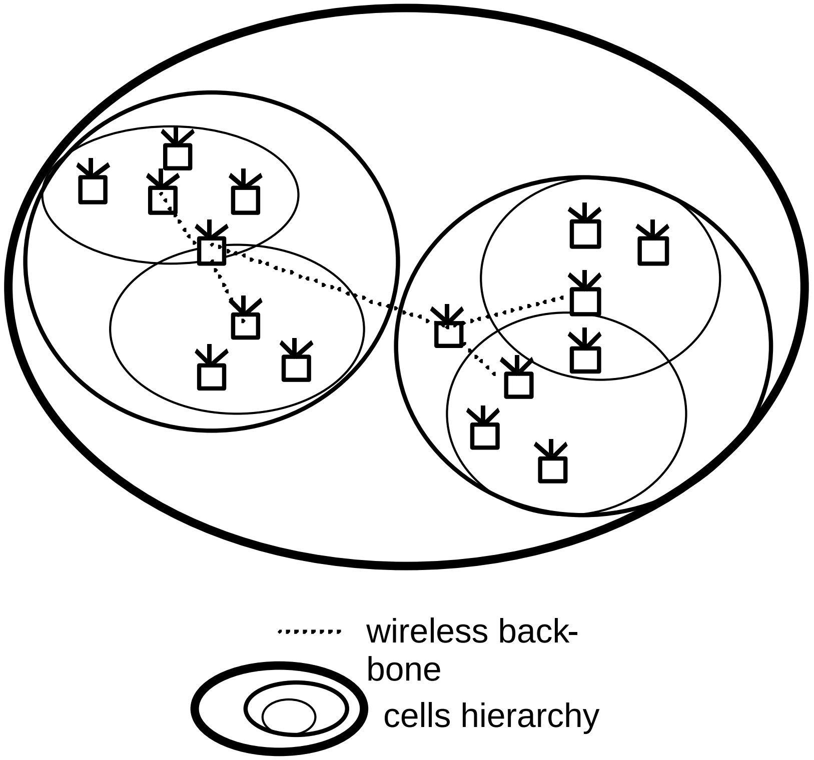
Examples for interference based routing

- Least Interference Routing (LIR)
  - calculate the cost of a path based on the number of stations that can receive a transmission
- Max-Min Residual Capacity Routing (MMRCR)
  - calculate the cost of a path based on a probability function of successful transmissions and interference
- Least Resistance Routing (LRR)
  - calculate the cost of a path based on interference, jamming and other transmissions

- LIR is very simple to implement, only information from direct neighbors is necessary

Clustering

- Goal:
  - Reduce channel contention
  - Form routing backbone to reduce network diameter
  - Abstract network state to reduce its quantity and its variability
- Various approaches to clustering
  - Started in the 70s with Packet Radio Network (PRNet) sponsored by DARPA



Theoretical Results

- Capacity of a wireless network [Gupta & Kumar 2000]
  - n identical randomly located nodes each capable of transmitting W bits can only achieve a throughput of
    $\Theta(\frac{W}{\sqrt{n \log n}})$ bit/sec
  - n optimally placed nodes with an optimal traffic pattern and an optimal transmission power can only achieve
    $\Theta(W\sqrt{n})\ bit\text{-}meters/\mathrm{sec}$

Clustering for Transmission Management

- Goal: reduce contention
- Cluster = clusterhead + gateways + ordinary nodes
- Roles:
  - Clusterhead: schedules traffic, allocates resources (tokens, emits busy tone, etc.). Similar to the master in a Bluetooth piconet.
  - Gateways: interconnect clusters
  - Ordinary nodes are 1-hop away from a clusterhead and 2-hops away from other members in the cluster
- Tasks:
  - Connectivity discovery
  - Election of clusterheads
  - Agree on Gateways

Clustering for Transmission Management (Cont)

- Clusterhead election:
  - Centralized/distributed algorithms
  - Node identifier/degree based
  - Principles:
    - Centralized: (1) elect the highest ID node and create a corresponding cluster, repeat step (1) with nodes not already members of a cluster
    - Distributed:
      - a node elects itself as cluster head if it has the highest ID among its neighbors
      - otherwise elect a neighbor that is not member of another cluster
  - Leads to disjoint clusters
- Gateways:
  - If connected to > 1 cluster => gateway candidate
  - If num_hops (CH_1) = 1 & num_hops(CH_2) = 2 =>GW_1 --- GW_2
  - When multiple candidates to connect two clusters, choose GW with highest ID

Clustering for Transmission Management (Cont)

- Mobility:
  - – Most algorithms do not adjust to mobility. They recompute the cluster.
  - – Recent algorithms:
    - change clusterhead status with two clusterheads become neighbors
    - [Gerla-Lin 1995, 1997]: cluster maintenance
- Routing:
  - – To avoid clusterhead congestion and improve robustness, routing is done over the flat network

Clustering for Backbone Formation

- Wireless multiphop networks have high end-to-end delay:
  - link-layer ARQ, MAC delay, FEC/spreading, tx/rx switching time
- Clustering can reduce the end-to-end delay by allowing faster forwarding through the clusterheads backbone
- Approaches:
  - Near-Term Digital Radio Network (NTDR) [Zavgren 1997]
  - Virtual Subnet Architecture [Sharony 1996]

Near-Term Digital Radio Network (NTDR)

- Goal: support mobile tactical communication
- Architecture: clusterheads = gateways+single-hop nodes
- Communication:
  - All 2-hop packets go through the clusterheads backbone
  - Two frequencies: intra-cluster band + inter-cluster band
- Clusterhead election:
  - Clusterheads send a beacon (organization, members, TxPwrL, PL to nodes)
  - Nodes elect themselves if: (1) do not receive a beacon, (2) receives beacons with different partition numbers
- Preventing multiple clusterhead creation:
  - Wait a random amount of time before self-election as clusterhead
  - After self-election, immediately issue beacons
  - Eliminate superfluous clusterheads without partitioning the network

NTDR (Cont)

- Cluster affiliation criteria:
  - Node and clusterhead are in the same organization
  - Low path loss (conserves energy)
  - Resulting cluster has a small size
- Canceling affiliation:
  - Clusterhead relinquishes its role
  - Broken link (or low quality link)
- On membership change all clusterheads are updated
- Routing:
  - Use a link state protocol: Dijkstra’s Shortest Path First algorithm
  - Resistance metric: minimize interference experienced by packets
  - Does not scale for high mobility networks because of flooding cost

Virtual Subnet Architecture

- Goal: fault-tolerant connectivity and load balancing
- Architecture:
  - Disjoint clusters: *physical subnets* (at most P)
  - Physical subnets are clustered into *virtual subnets* (at most Q)
  - Virtual subnets ideally span all physical subnets
  - Virtual subnets and neighboring subnets operate over different frequencies
  - Nodes addresses: location dependent addresses (physical subnet, virtual subnet) + location independent address
- After affiliation to a physical/virtual subnet, the node advertises its new address to the physical/virtual subnet
- To determine the location dependent address of a node:
  1. Distribute an address query to the physical subnet. Use the virtual subnets membership information of the physical subnet
  2. If unsuccessful distribute the address query to the virtual subnet

Virtual Subnet Architecture

- Routing strategies:
  - Direct routing:
    - Forward the packet to a node *x* that is both in the source physical subnet and destination virtual subnet
    - Forward the packet through the virtual subnet to reach the physical subnet of the destination
  - Long-path routing (under high mobility): randomly distribute the routes over the space of possible routes
    - Forward the packet through a virtual subnet represented by a node in the source physical subnet, or
    - Forward the packet through the physical subnet of a node in the virtual subnet of the source
    - P+Q-1 possible paths

Clustering for Routing Efficiency

- Quasi-hierarchical routing
  - – Each node learns the next node to use to reach all level-*i* clusters within its (*i*+1) ancestral level
  - – Reduces the amount of routing information from $O(N)$ to $O(mC_{max})$
  - – Both distance-vector and link-state approaches
  - – Minimized forwarding tables and optimal routes:
    - Number of level-i clusters in each level-(i+1) = *e*, and ln *N* levels
    - Quasi-hierarchical routing routes -> optimum (under ideal assumptions)
- Strict hierarchical routing
  - – Each node learns the next *level-i* cluster to use in order to reach each *level-i* cluster within its *level-i*+1 cluster
  - – Each node learns which level-*i* clusters lie on the boundary of its level-(*i*+1) cluster
  - – Read page 109 (A Prototypical Scheme)

Building a Clustering Hierarchy

- Objectives:
  - Minimizing the amount of routing information
  - Maximize connectivity within each cluster
  - Localize high-intensity traffic within one cluster
  - Minimizing the number of intercluster links
  - Maximizing the stability of intercluster links
  - Minimizing the difference between the hierarchical route and the optimal route
- Possible constraints:
  - Upper/lower limit of the number of childs
  - Upper limit on the diameter of each cluster
  - Constant number of childs

Limitations of Current Clustering Algorithms

- Most clustering algorithms do not consider the flexibility/constraints of radio frequency communication:
  - Connectivity is not rigid but can be controlled through power-control
  - Leader consume more energy then ordinary nodes
- Relation between MAC, power-control and coding:
  - Coding generates more time interference (collisions?), high power generates more space interference (SNIR)
- Hybrid networks require to consider fairness more seriously:
  - Nodes in the vicinity of the AP would experience more traffic than nodes on the periphery