Self Organizing and Self-Healing Ad-hoc Networks

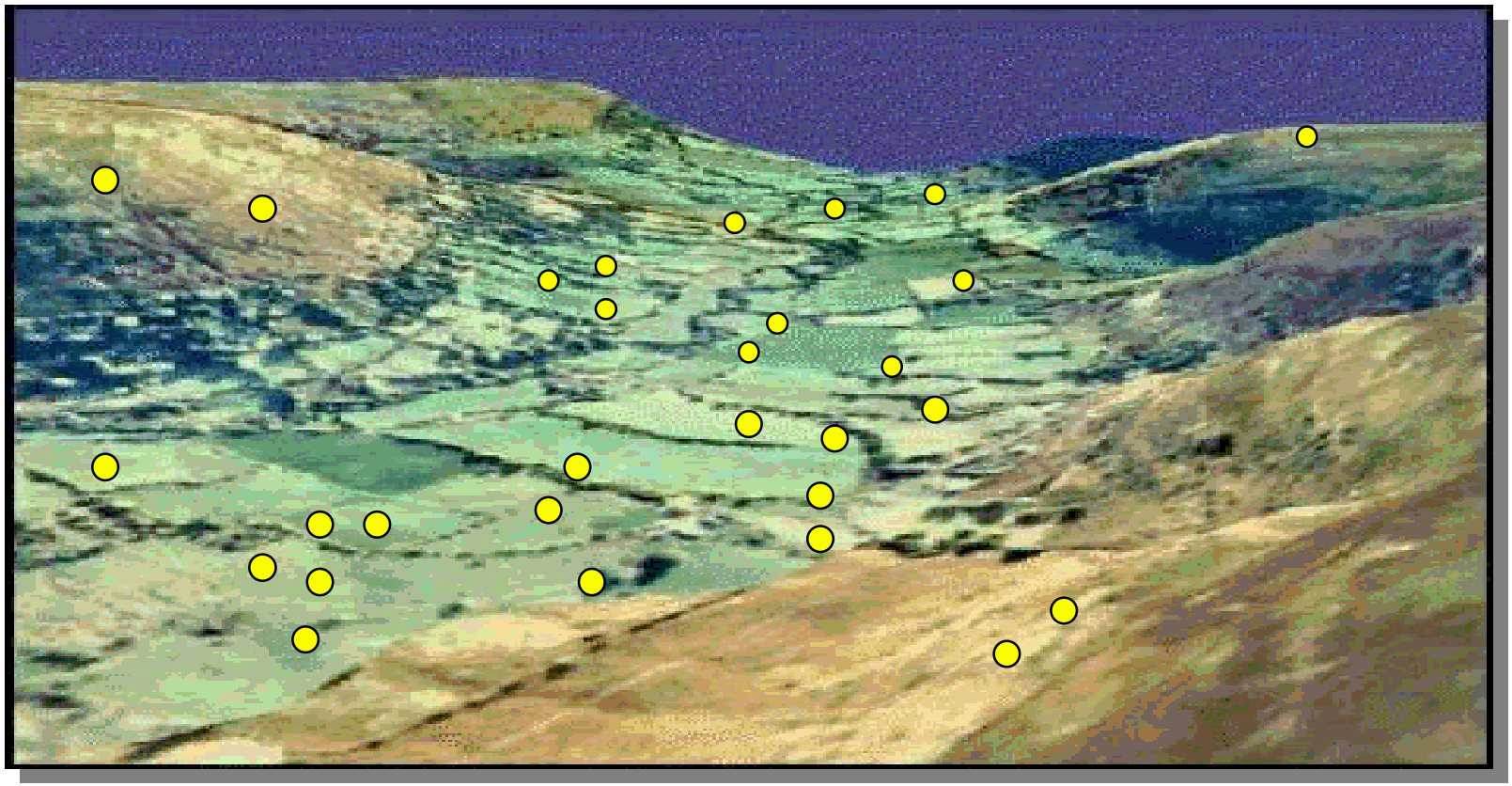
WLAN 57- by Chip (Brig) Elliott,
BBN Technologies

Cambridge, MA, 02138

Presented at WLAN 2001

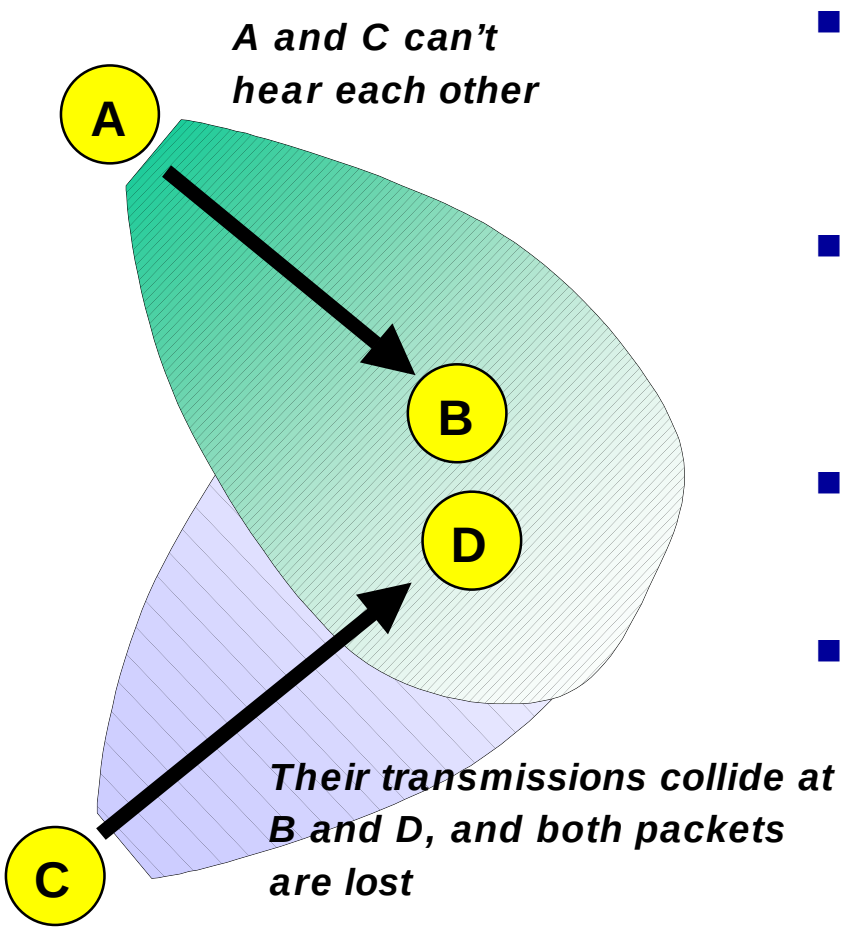
by Jerry Burchfiel

Example Problem



- *N* radios on a terrain
- Each moving
- No Fixed Infrastructure
- High-Speed Data Communication
- Mobile Voice and Video
- Cheap and Reliable

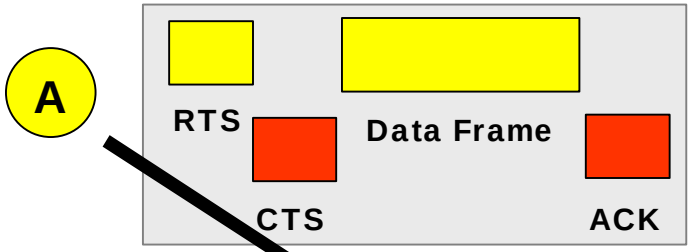
The Hidden Terminal Problem



- A transmits to B, while C transmits to D
- Results in one or both transmissions failing
- Very common event, alas!
- How could A or C know when to transmit?

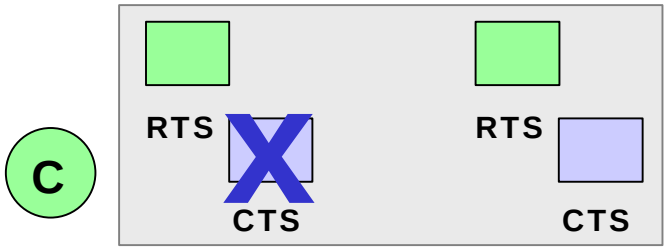
Channel Access Mechanisms

RTS / CTS Approach



Receiver directed
D cues off B's CTS
Complex state machines
Behaves like CSMA
(Doesn't work for broadcast)

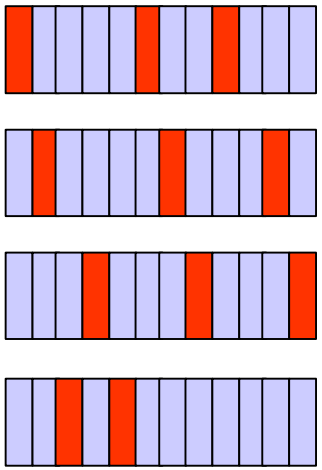
D



Timeslot Approach

A

Transmitter directed
Good features of TDMA
Needs sync'd clocks
Very hard distributed algorithm



B

D

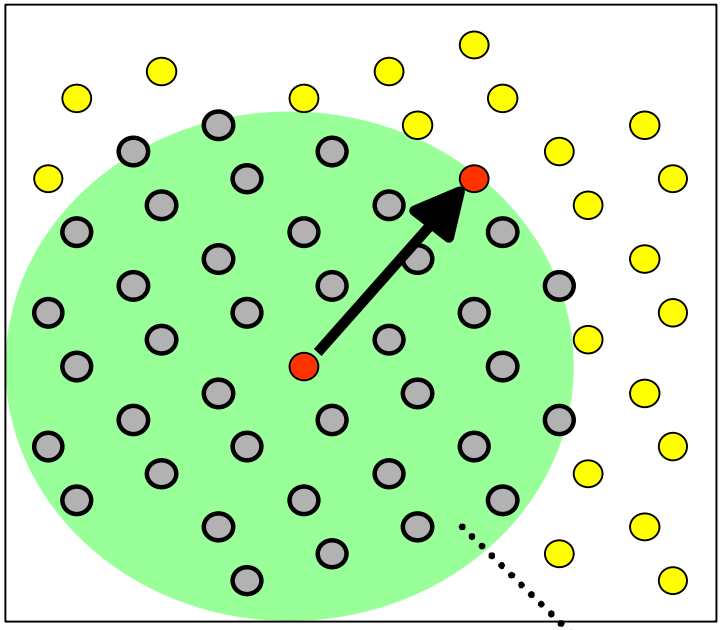
C

Note that A and C must be coordinated!

Delay vs. Throughput Tradeoff

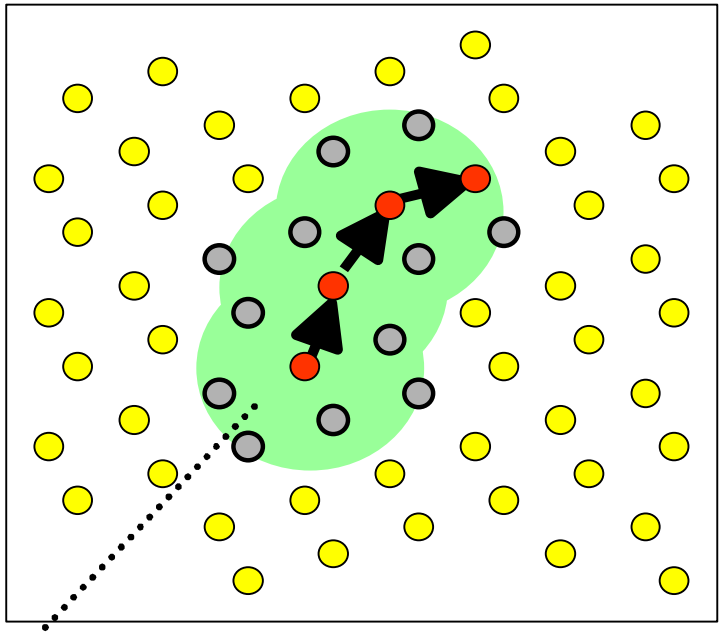
High Power *means low network throughput*
Low Power *means long end-to-end delay*

Single, High-Power Transmission



38 other radios blocked

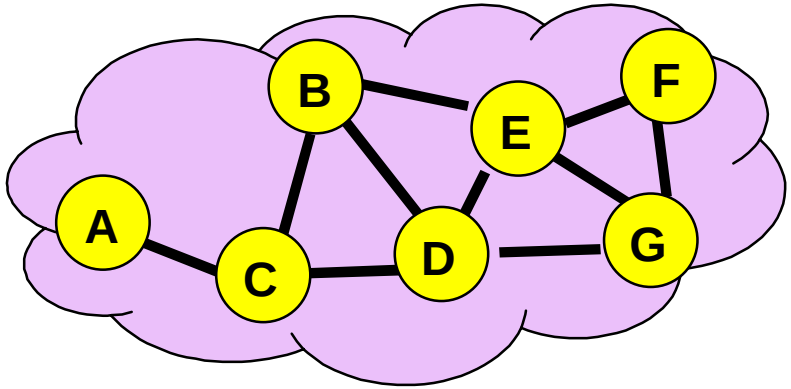
Multi-Hop, Low Power Transmissions

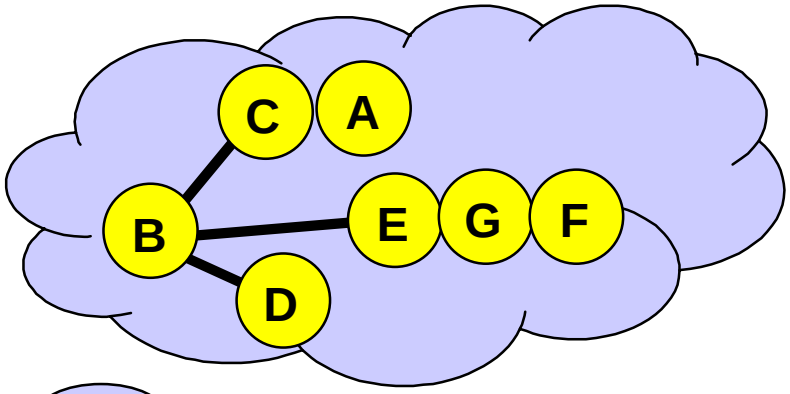


14 other radios blocked

Interference Area

“Proactive” Routing Protocols





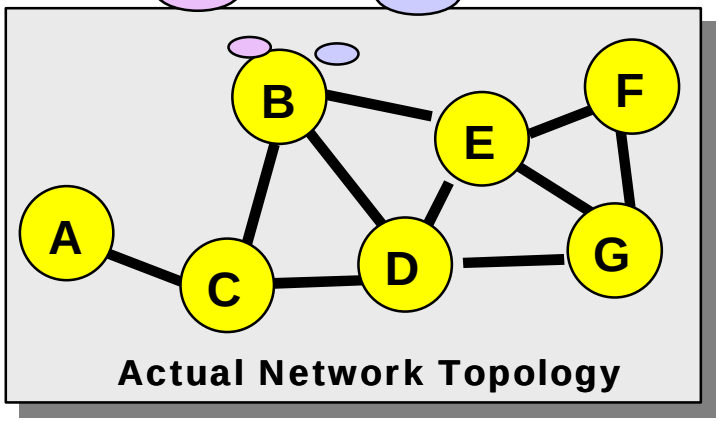
Link State (SPF)

Good Points

- Global Knowledge
- Allows QOS Routing
- Multicast is “easy”

Bad Points

- Scales poorly
- Global bursts of control traffic



Distance Vector

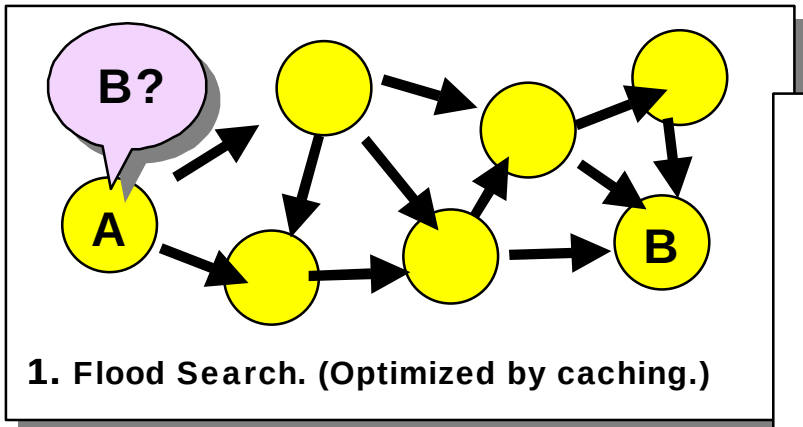
Good Points

- Less Control Traffic
- “Local Repair”
- Simple to Code

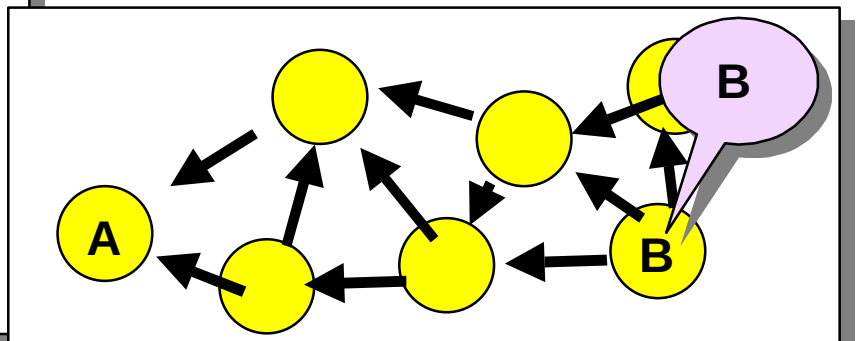
Bad Points

- Control traffic is hard to estimate
- QOS is hard
- Multicast is hard

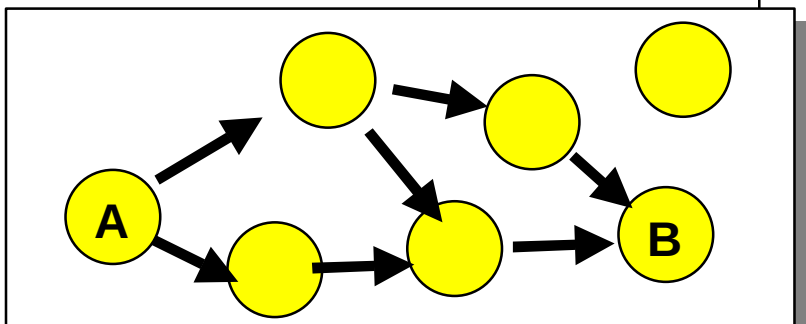
“On-Demand” Routing Protocols



1. Flood Search. (Optimized by caching.)



2. Path Discovery. (Optimized by pruning.)



3. Traffic. (Source routed.)

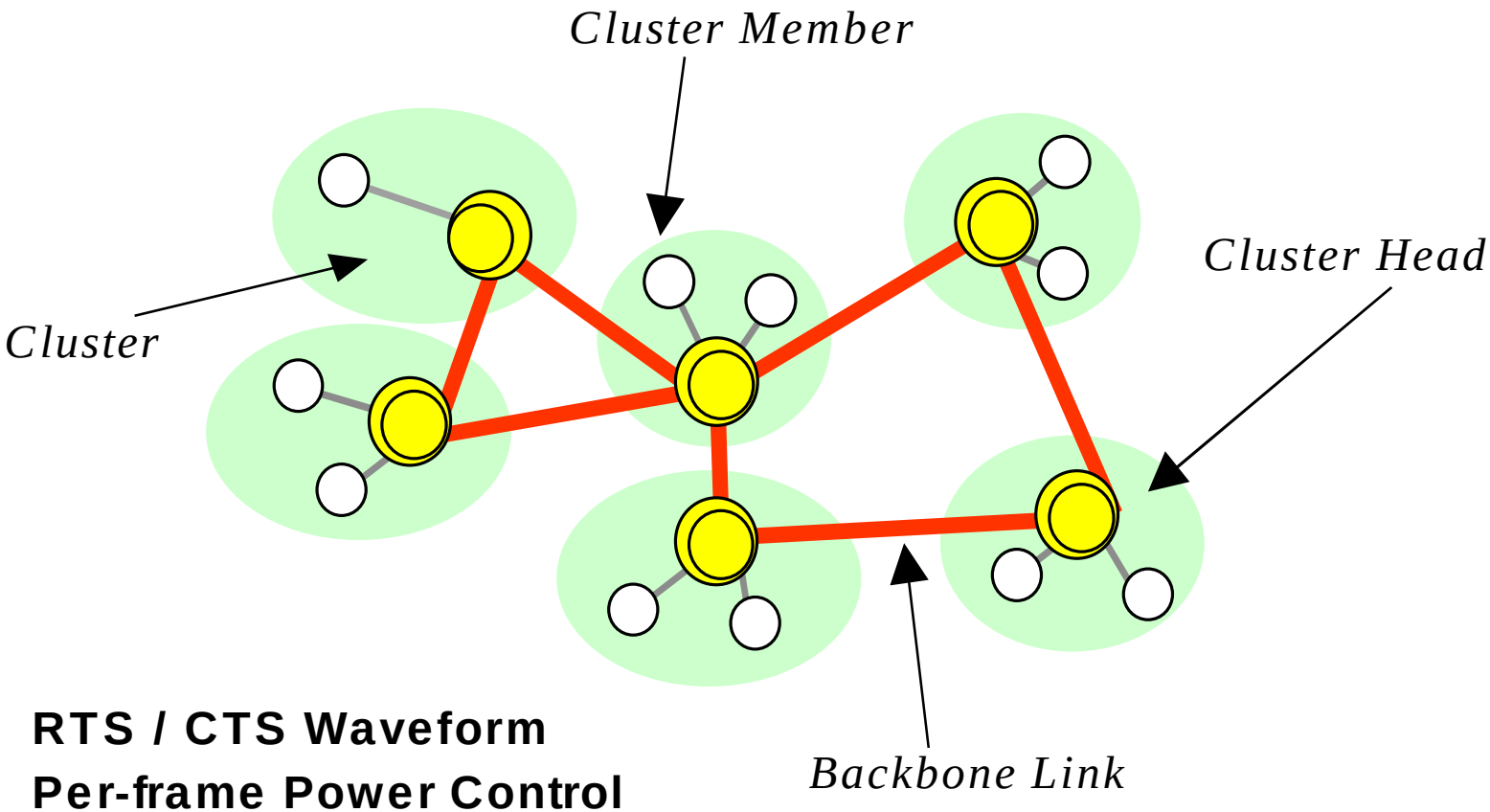
Good Points

- No global knowledge
- Allows local repair
- Scales with Traffic Flows

Bad Points

- Repair routes are suboptimal
- Bad interactions with TCP
- Multicast is complicated
- Denial of Service attacks

A Hierarchical Ad Hoc Network



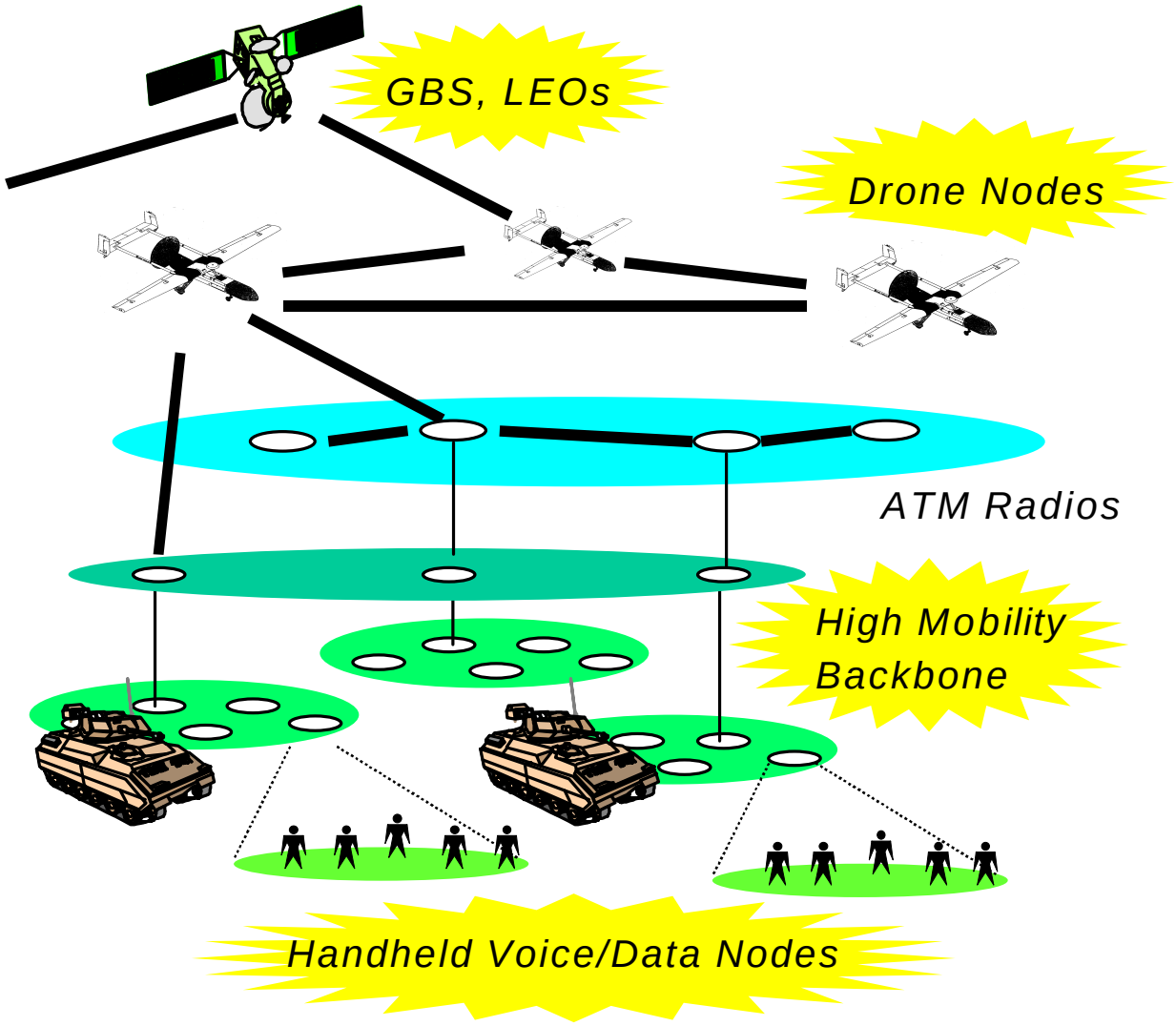
RTS / CTS Waveform

Per-frame Power Control

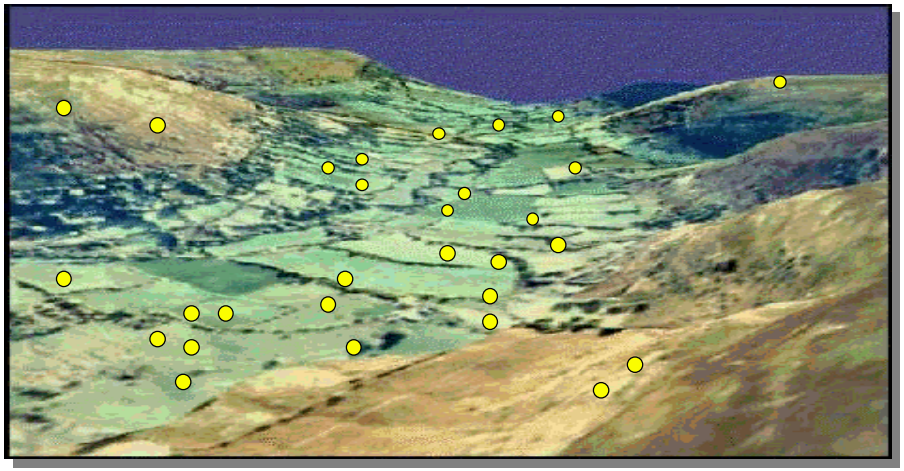
Hierarchical Link-State Routing

Spatial reuse *with* good end-to-end delay

Ad Hoc Network with Unmanned Aircraft

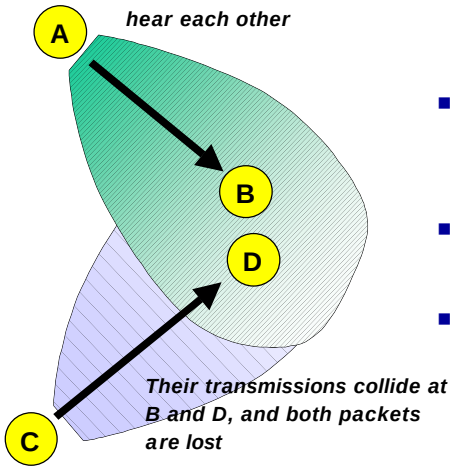


Example Problem



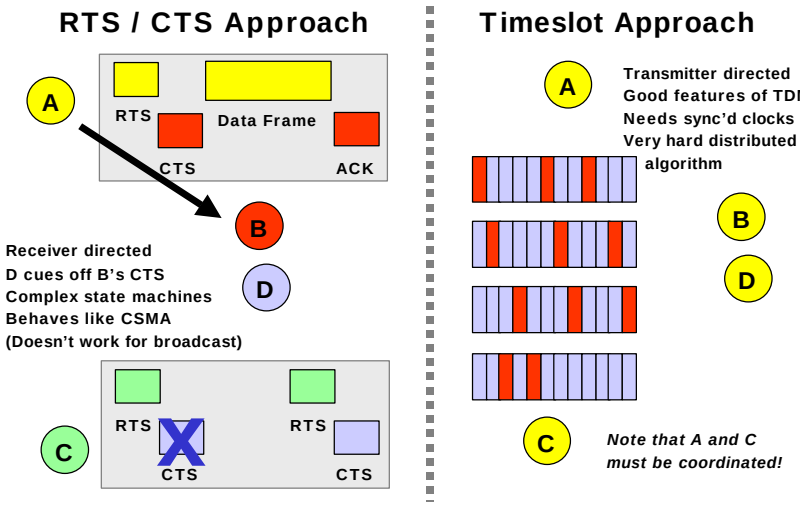
- *N* radios on a terrain
- Each moving
- No Fixed Infrastructure
- High-Speed Data Communication
- Mobile Voice and Video
- Cheap and Reliable

Hidden Terminal Problem



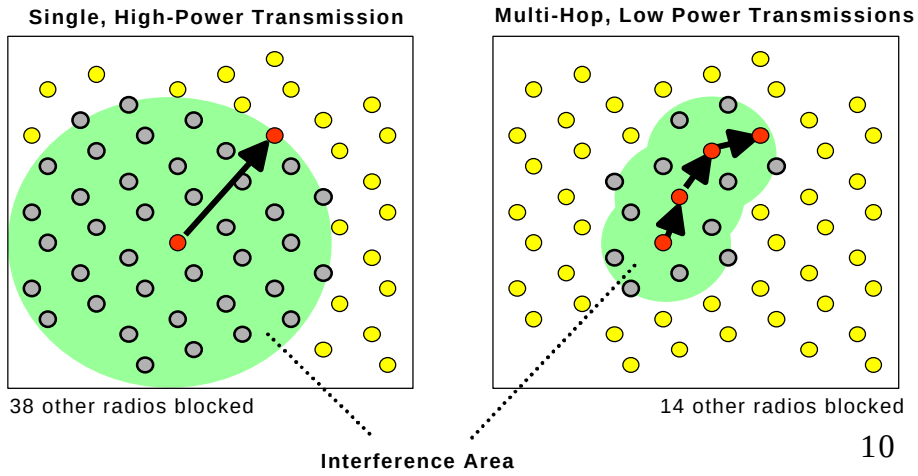
- A transmits to B, while C transmits to D
- Results in one or both transmissions failing
- Very common event, alas!
- How could A or C know when to transmit?

Channel Access Mechanisms

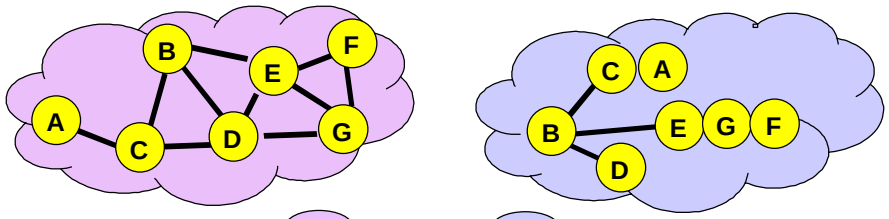


Delay versus Throughput

High Power *means low network throughput*
Low Power *means long end-to-end delay*



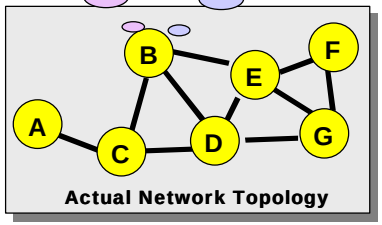
"Proactive" Routing Protocols



Link State (SPF)

Good Points
- Global Knowledge
- Allows QOS Routing
- Multicast is "easy"

Bad Points
- Scales poorly
- Global bursts of control traffic



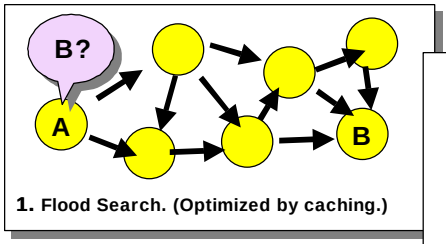
Actual Network Topology

Distance Vector

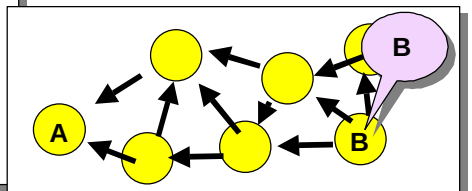
Good Points
- Less Control Traffic
- "Local Repair"
- Simple to Code

Bad Points
- Control traffic is hard to estimate
- QOS is hard
- Multicast is hard

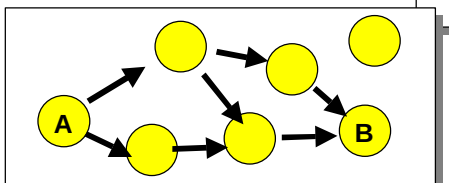
"On-Demand" Routing Protocols



1. Flood Search. (Optimized by caching.)



2. Path Discovery. (Optimized by pruning.)

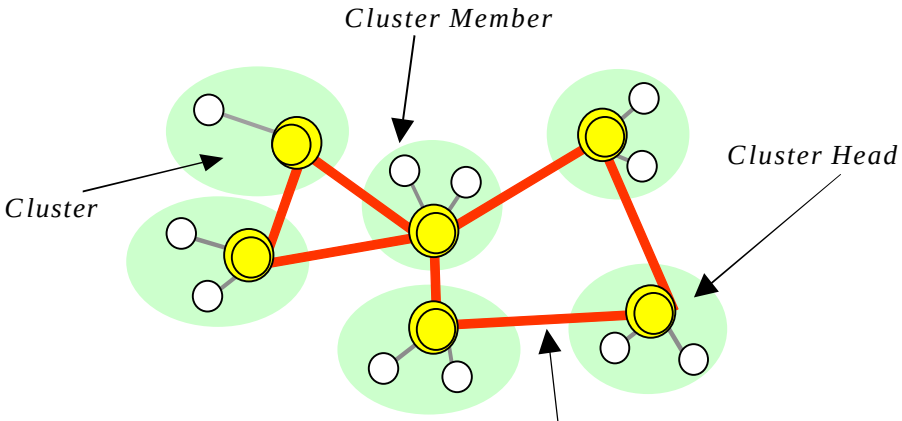


3. Traffic. (Source routed.)

Good Points
- No global knowledge
- Allows local repair
- Scales with Traffic Flows

Bad Points
- Repair routes are suboptimal
- Bad interactions with TCP
- Multicast is complicated
- Denial of Service attacks

A Hierarchical Ad Hoc Network



RTS / CTS Waveform
Per-frame Power Control
Hierarchical Link-State Routing
Spatial reuse *with* good end-to-end delay

Ad Hoc Network with Drone Aircraft