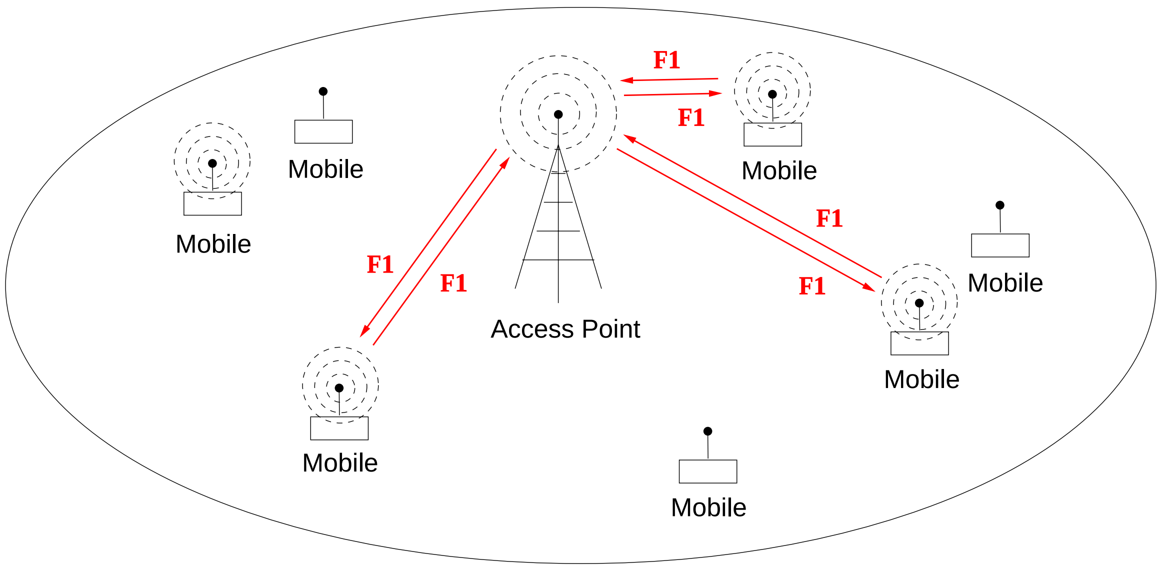
Wireless LAN (WLAN): infrastructure mode

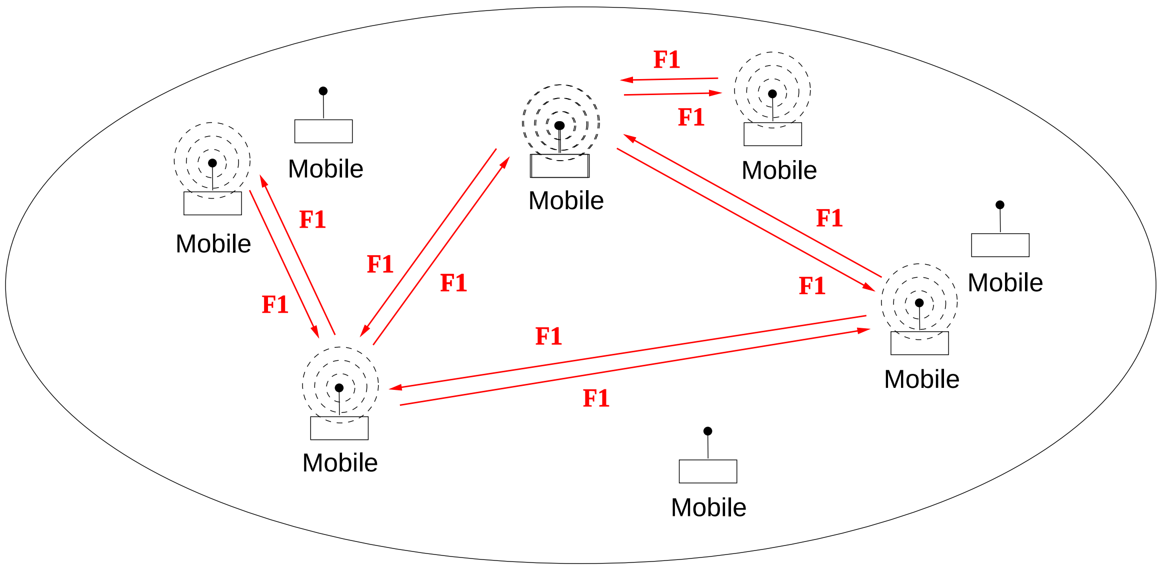


`WLAN: Infrastructure Network`

$\longrightarrow$ shared uplink & downlink channel $F1$

- basic service set (BSS)
- SSID (service set identifier): name/label of BSS
- base station: access point (AP)
- mobile stations must communicate through AP

WLAN: ad hoc mode



WLAN: Ad Hoc Network

⟶ homogeneous: no base station

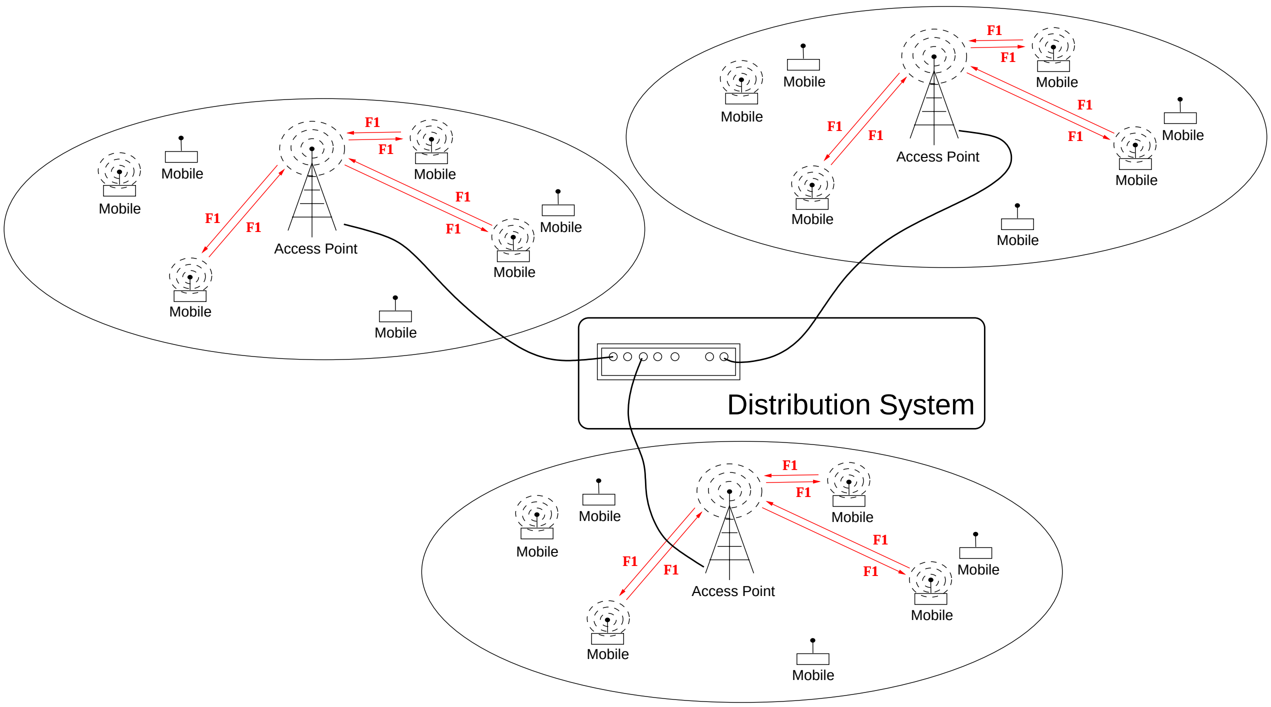
⟶ everyone is the same

⟶ share forwarding responsibility

- independent basic service set (IBSS)
- mobile stations communicate peer-to-peer

→ also called peer-to-peer mode

WLAN: internetworking



`WLAN: Extended Service Set`

$\longrightarrow$ internetworking between BSS's through APs

$\longrightarrow$ mobility and handoff

- extended service set (ESS): shared SSID
- APs are connected by distribution system (DS)

  $\rightarrow$ typically: Ethernet switch

- How do APs and Ethernet switches know where to forward frames?

  → spanning tree

  → IEEE 802.1 (Perlman's algorithm)

  → learning bridge: source address discovery

  → per interface: log source MAC address of incoming frames

  → initially or if unclear: broadcast

  → a very simple form of routing

  → adequate for small systems

Additional headache: mobility

$\longrightarrow$ also called roaming

$\longrightarrow$ how to perform handoff

$\longrightarrow$ mobility management at MAC vs. IP

Mobility between BSS's in an ESS

- association

  $\rightarrow$ registration process

  $\rightarrow$ AP sends out periodic beacon frame

  $\rightarrow$ mobile station (MS) associates with one AP

- disassociation

  $\rightarrow$ upon permanent departure: notification

Handoff from old to new AP:

- reassociation

  → movement of MS from one AP to another

  → client initiated

  → e.g., AP's signal strength is low

  → passive (beacon) or active (probe) scanning to find alternate AP

  → go through association process

  → inform new AP of old AP

  → forwarding of buffered frames from old to new AP in ESS

Note: when and parts of how to perform handoff are not part of IEEE standard

→ vendor dependent

IEEE 802.11b/g WLAN spectrum 2.4–2.4835 GHz:

$\longrightarrow$ 11 channels (U.S.)

$\longrightarrow$ 2.412 GHz, 2.417 GHz, . . ., 2.462 GHz

$\longrightarrow$ unlicensed ISM (Industrial, Scientific, Medical) band

$\longrightarrow$ global: 2.4–2.4835 GHz

$\longrightarrow$ up to 14 channels (e.g., Japan)

IEEE 802.11a: 5.15–5.35 GHz and 5.725–5.825 GHz

$\longrightarrow$ UNNI (unlicensed National Information Infrastructure)

$\longrightarrow$ non-global

IEEE 802.11n: both 2.4 and 5 GHz

$\longrightarrow$ 2.4 GHz: backward compatible

$\longrightarrow$ also uses multiple antennae

$\longrightarrow$ called MIMO (multiple input multiple output)

$\longrightarrow$ e.g., Apple's 802.11n has 3 antennae

IEEE 802.11 WLAN MAC: uses CSMA

→ multi-user bandwidth sharing

However:

- 802.11b: uses DSSS CDMA

  → 11-bit chip sequence (Barker sequence)

  → single-user DSSS

  → why?

- 802.11a/g/n: uses OFDM

  → single-user OFDM (i.e., not OFDMA)

  → also called single-carrier (vs. multi-carrier)

  → 802.11g: 48 carrier frequencies

  → subcarrier separation: 312.5 KHz

  → bits of single frame are distributed across 48 sub-carriers

  → first bit on subcarrier 1, second bit on subcarrier 2, etc.

  → but: transmission is sequential—not parallel!

  → similar to FHSS

  → why use OFDM without parallel speed-up?

Why not use OFDMA?

IEEE 802.11 MAC

$\longrightarrow$ CSMA/CA with exponential backoff

$\longrightarrow$ almost like CSMA/CD

$\longrightarrow$ drop CD

$\longrightarrow$ explicit positive ACK frame

$\longrightarrow$ added optional feature: CA (collision avoidance)

Two modes for MAC operation:

- Distributed coordination function (DCF)

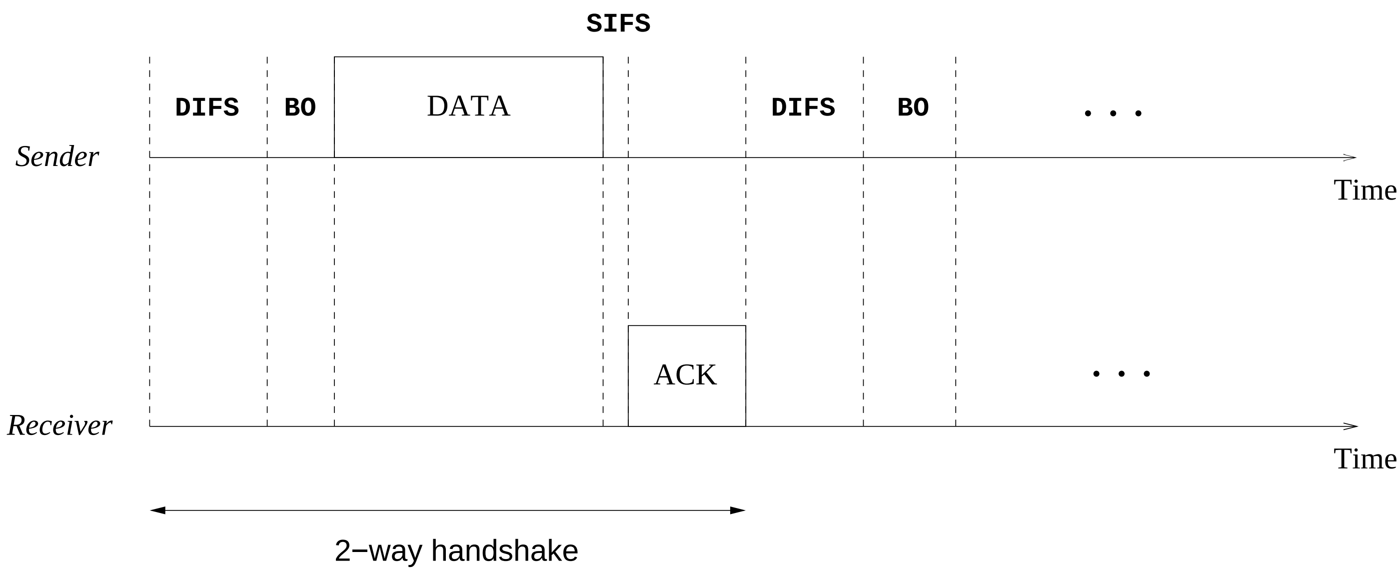
  $\rightarrow$ multiple access (default mode)

- Point coordination function (PCF)

  $\rightarrow$ polling-based priority

. . . neither PCF nor CA used in practice

Timeline without collision:



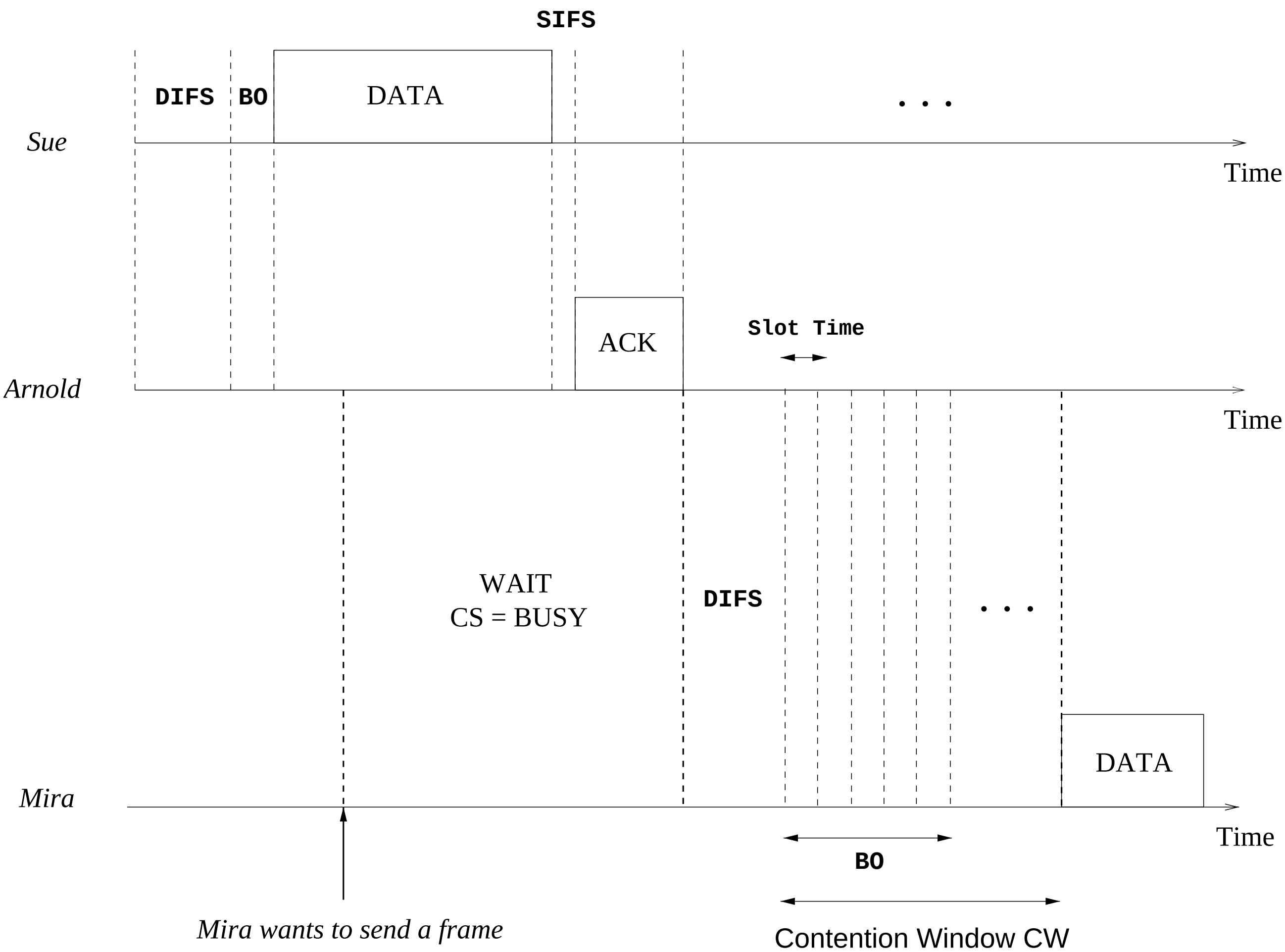
- SIFS (short interframe space): 10 μs
- Slot Time: 20 μs
- DIFS (distributed interframe space): 50 μs

  $\rightarrow$ DIFS = SIFS + 2 $\times$ slot time
- BO: variable back-off (within one `CW`)

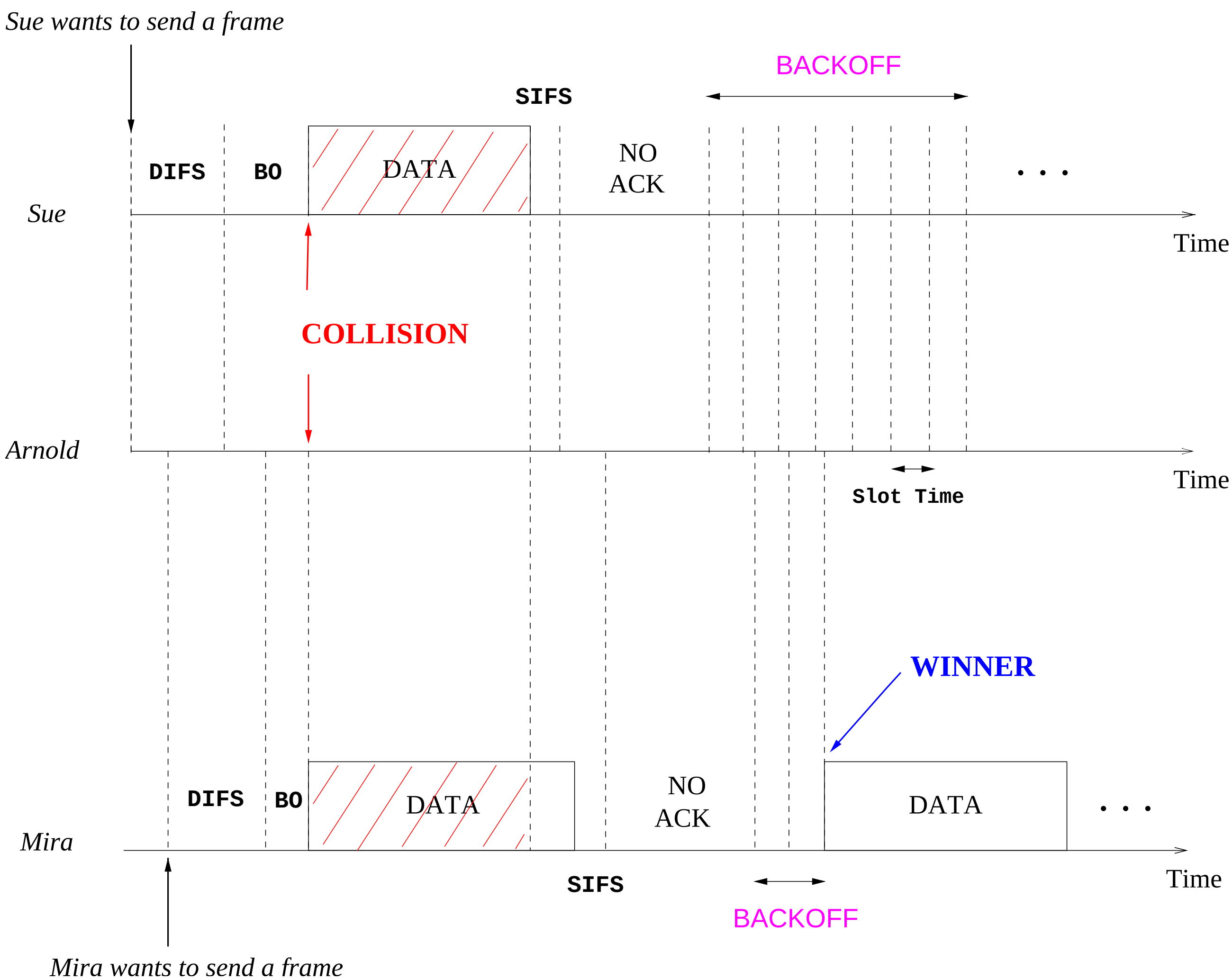
  $\rightarrow$ `CWmin`: 31; `CWmax`: 1023

Time snapshot with Mira-come-lately:

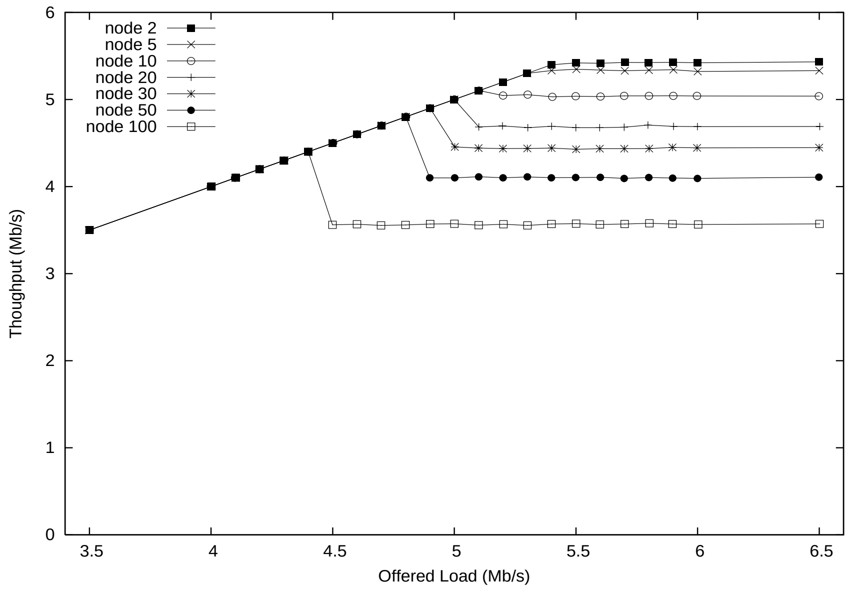
$\longrightarrow$ Sue sends to Arnold

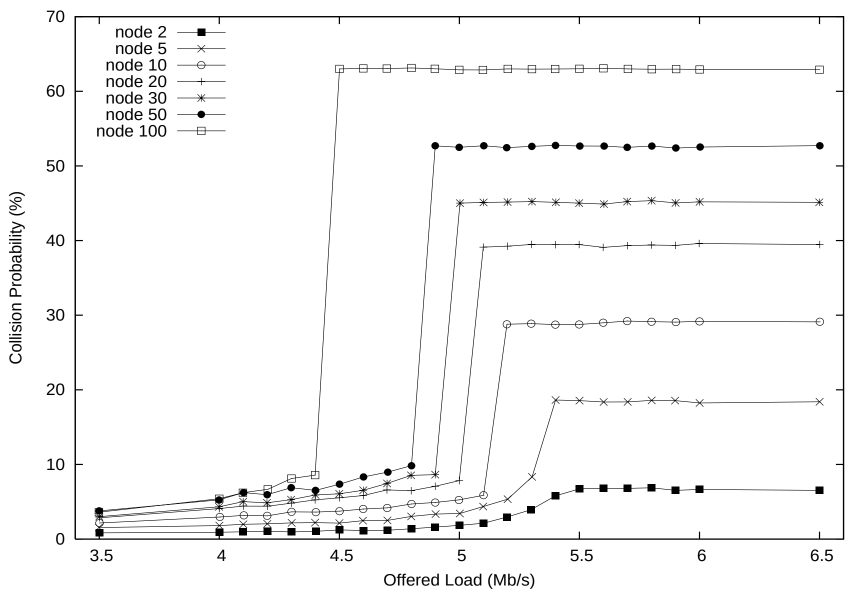


Time snapshot with collision (Sue & Mira):



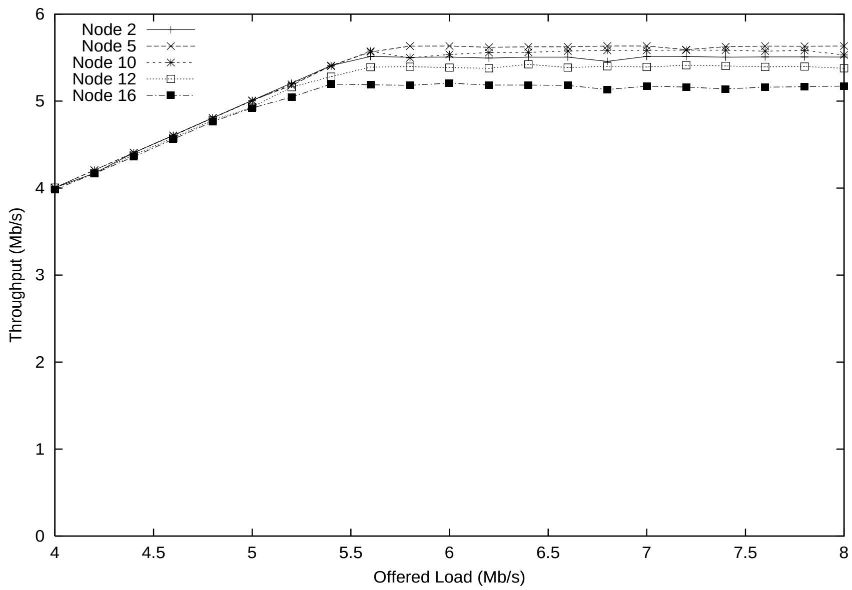
MAC throughput and collision (simulation):





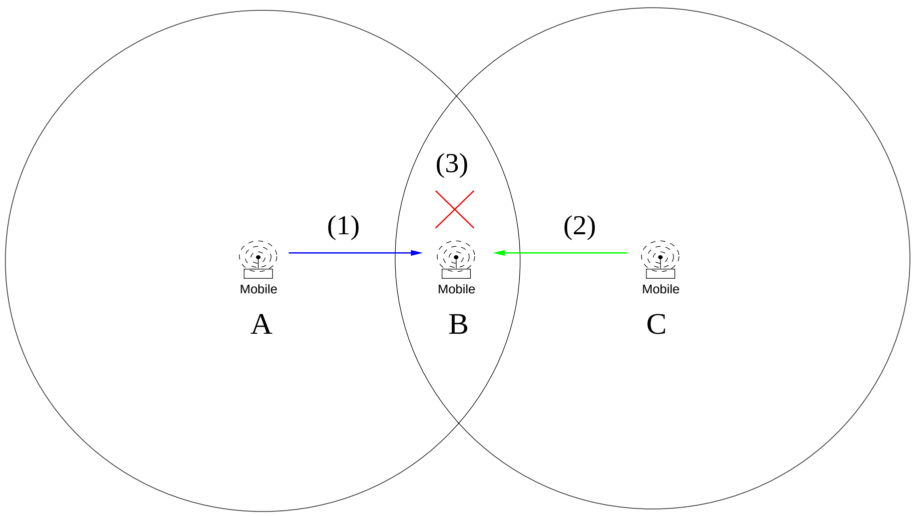
MAC throughput (experiment):

⟶ HP iPAQ pocket PC running Linux



Additional issues with CSMA in wireless media:

Hidden station problem:



Hidden Station Problem

(1) A transmits to B

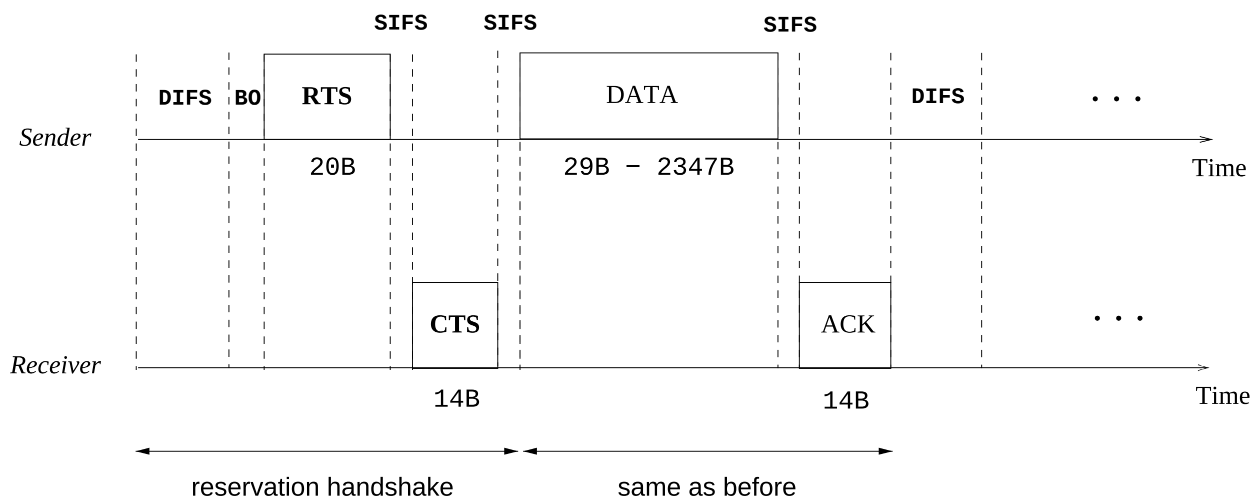
(2) C does not sense A; transmits to B

(3) interference occurs at B: i.e., collision

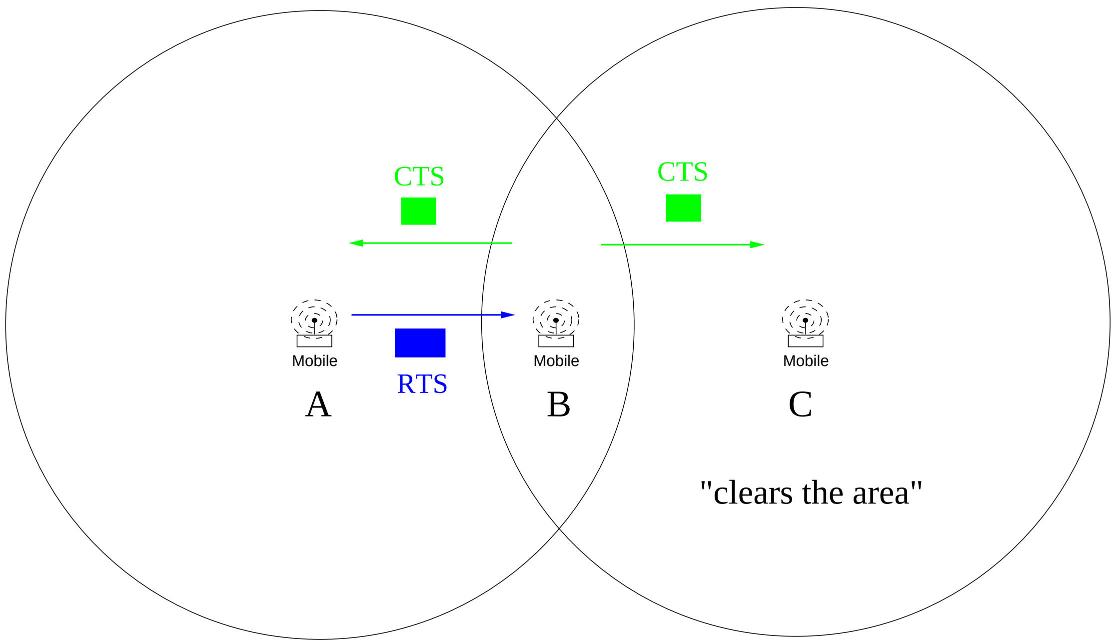
Hidden station problem: CA (congestion avoidance)

$\longrightarrow$ RTS/CTS reservation handshake

- Before data transmit, perform RTS/CTS handshake
- RTS: request to send
- CTS: clear to send



Hidden station problem: RTS/CTS handshake "clears" hidden area



RTS/CTS Handshake

RTS/CTS perform only if data > RTS threshold

$\longrightarrow$ why not for small data?

. . . feature available but not used

Another problem: starvation

→ related to hidden station problem

→ A cannot hear C, C cannot hear A

→ B can hear both A and C

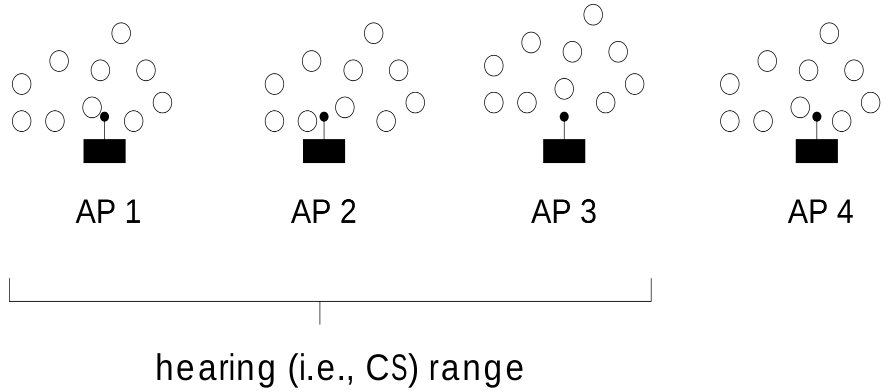
→ by CS: B gets less chance to speak

→ "sandwiched" between A and C

→ may even lead to near-starvation

Example: four 802.11 hot spots, each with 10 clients

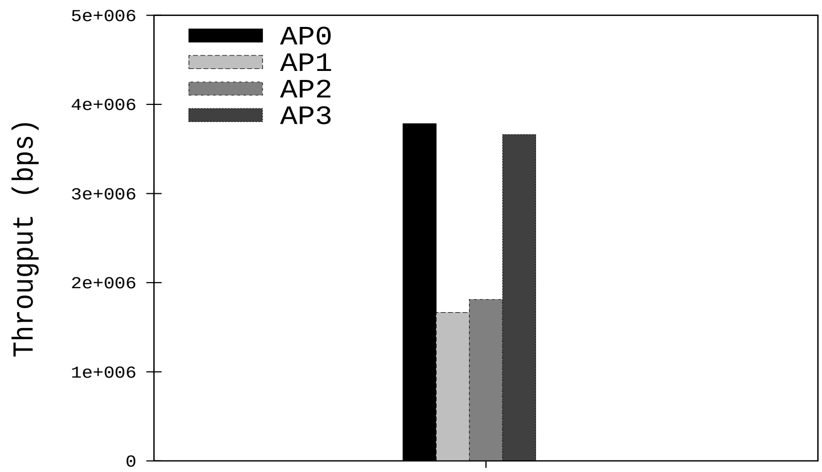
→ e.g., 4 neighboring coffee shops on a street



→ 3 neighboring hot spots (BSS's) are within hearing range of each other

→ AP1 and AP4 are outside CS range

Throughput at four hot spots:



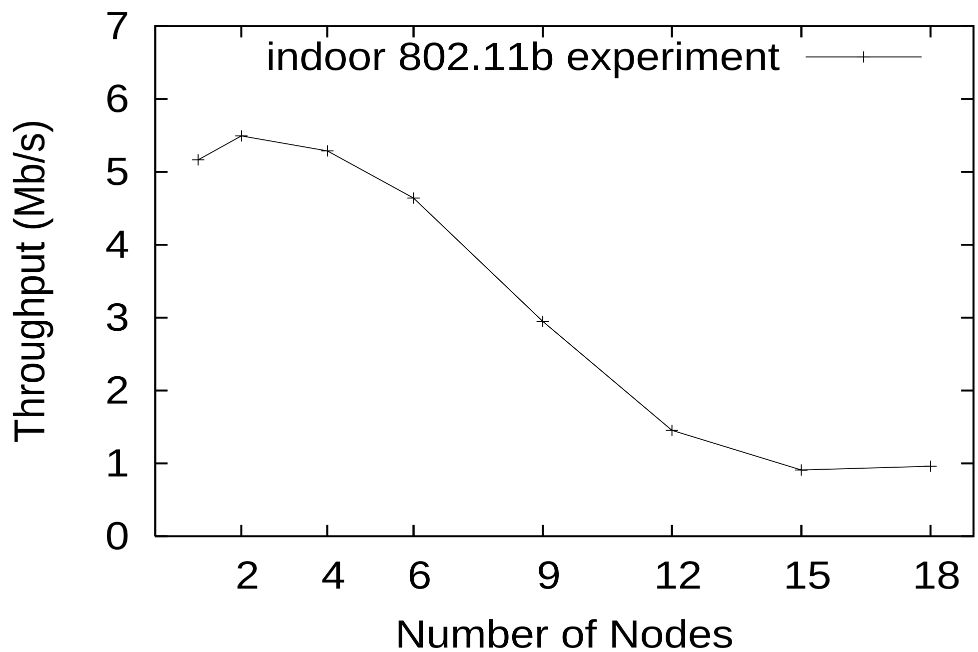
→ middle two get half the throughput

→ depending on configuration, can be even less

WLAN throughput collapse

→ IEEE 802.11b hot spot experiment

→ similar for 802.11g/n



→ throughput collapse to 1 Mbps

→ only moderate contention

IEEE 802.11b defines four data rates

→ 1, 2, 5.5, 11 Mbps

→ 802.11g defines 8 rates: 6, 9, 12, 18, 24, 36, 48, 54 Mbps

→ difference: amount of FEC protection

Note: the higher the data rate, the smaller the frame size

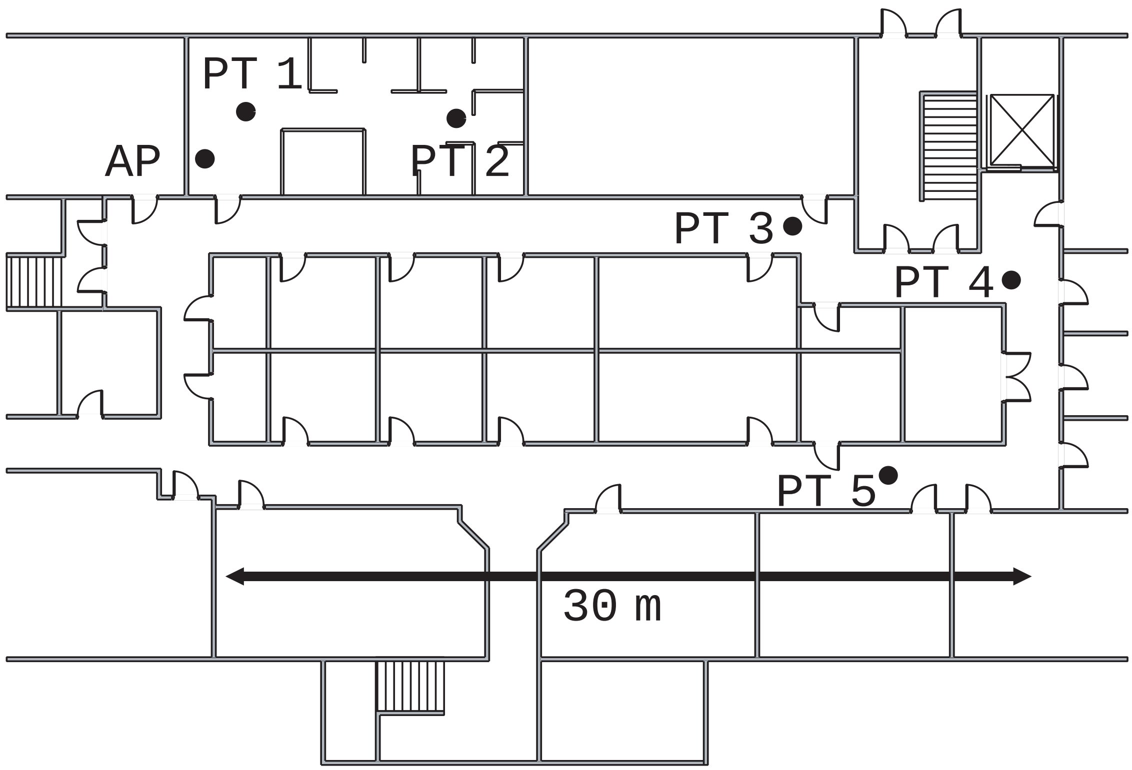
→ why?

→ multiple data rates needed due to noise

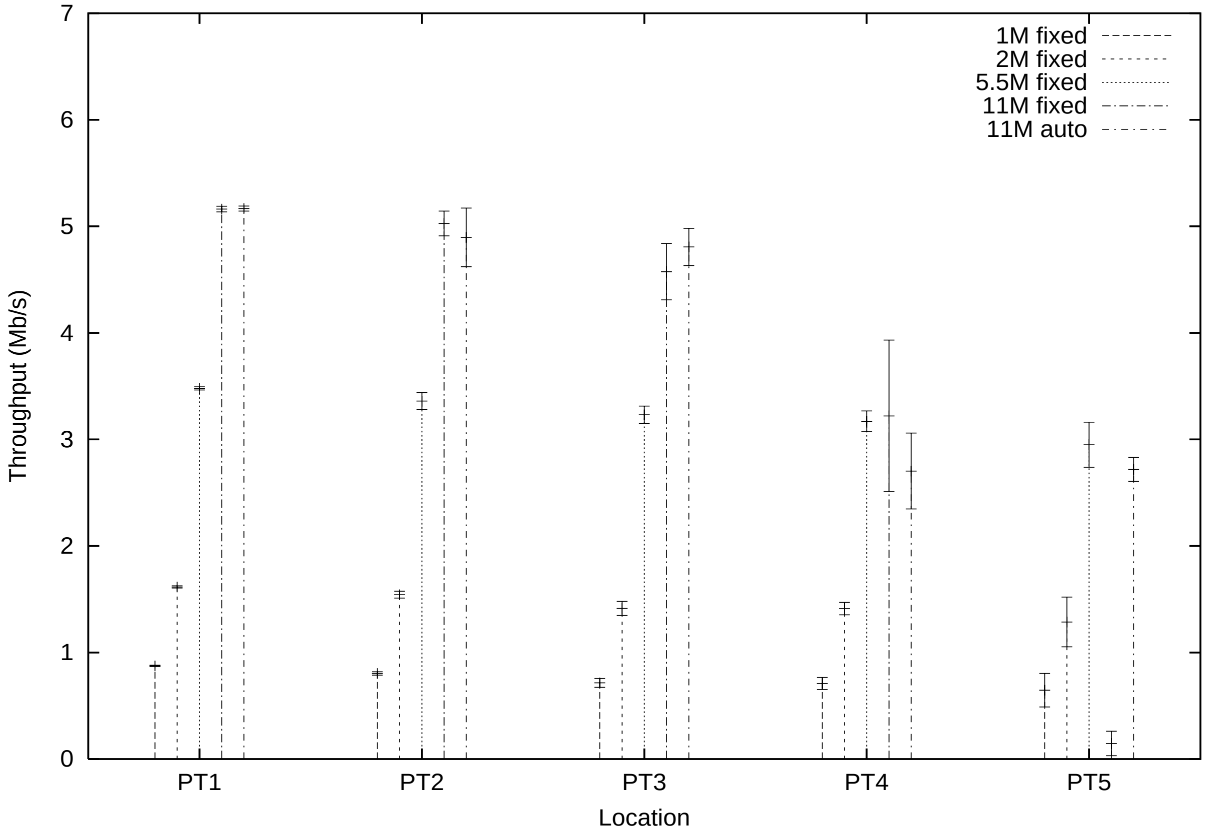
→ same in cellular networks

Ex.: HAAS basement corridor experiment

→ single wireless client



Throughput at different locations:



→ through driver can instruct NIC to fix data rate

→ auto: adaptive method implemented in WLAN cards

→ default mode

→ note inversion: 5.5 vs. 11 Mbps code rates at PT5

How does auto mode work?

→ called automatic rate fallback (ARF)

→ not part of IEEE 802.11 standard

→ vendor could implement different method (most implement ARF)

ARF protocol:

- if 2 successive 802.11 ACK frames are not received, downshift
- if 10 successive 802.11 ACKs are received, upshift

→ origin: Bell Labs WaveLAN (late '90s)

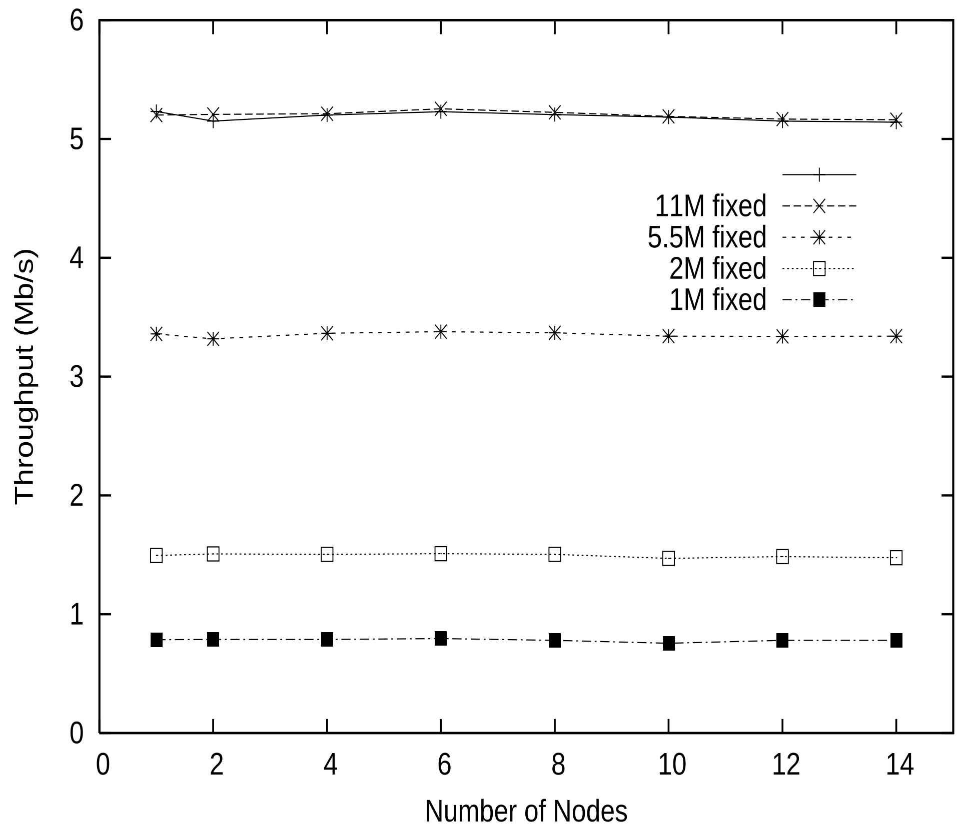
→ note: up/down thresholds are asymmetric

ARF: causes the WLAN throughput collapse

→ how?

WLAN performance without ARF

→ fix data rates



→ no more throughput collapse

→ if throughput is bad, try fixing data rate

Huge problem but no good solutions yet

→ implementation: firmware fix

→ good problem for start-up company ...

Cellular networks

→ use multiple data rates for FEC as in WLAN

→ throughput collapse doesn't arise

→ why?

802.16 (WiMax)

→ also part of 4G

802.15

→ OFDM based wideband communication

→ may be mixed: e.g., TDMA over OFDM

MIMO (multiple input multiple output)

→ space division multiple access (SDMA)

→ send parallel bit streams over multiple antennas using single carrier frequency

→ spatial diversity principle

→ 2x2: up to 2-fold potential throughput increase

RFID (radio frequency identification)

→ tag, reader

→ passive (no battery), active

→ passive: EM principle

→ protocol: variant of ALOHA