# Lecture 4c

## Wireless LANs

# Evolution

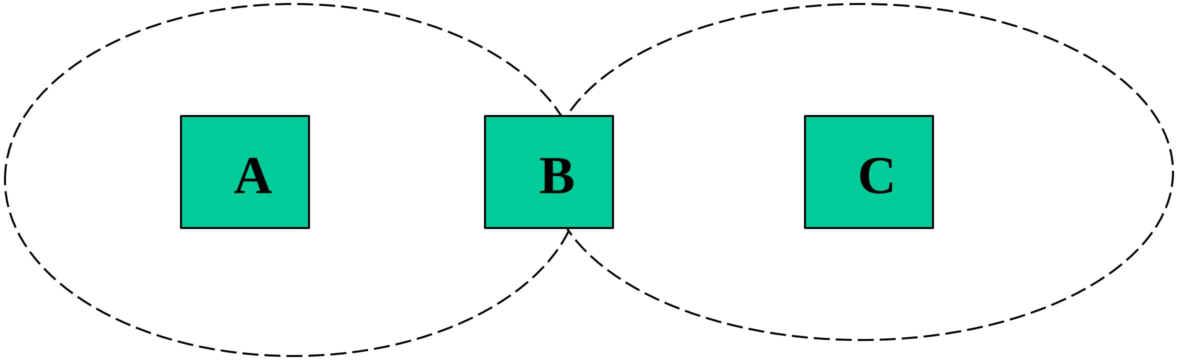
- Mobile Phone System predates / overlaps with wireless LANs
  - AMPS ( Advanced Mobile Phone System): 1G system 1982
    - Analog, cellular, FDMA, voice, low bandwidth, 60kHz duplex
  - GSM (Global System for Mobile Communication) 2G system 1991
    - Digital, TDMA, SIM card, single standard across Europe
    - Also low bit rate data services
    - Competitors: IS-95 (CDMA)
    - All 2G kept evolving, for example GPRS (general packet radio service) used packet switching technologies
  - UMTS (Universal Terrestrial Mobile System) 3G system 200l
    - Requirements: 144 kbps mobile, 384 kbps pedestrian, 2 Mbps indoors
    - Driven by European Telcos
    - WCDMA (Wideband Code Division Multiple Access) air interface technology
    - Competitor in US: CDMA2000 1x EV-DO used CDMA technology
    - Another competing standard is TD-SCDMA
  - Integrating all the above and new technologies 4G, 5G (2010 – 2020)
    - all IP packet switched, higher bandwidth, low mobility and high mobility cases,
    - Maintain compatibility with previous technologies also,
    - LTE (long term evolution - data) - VoLTE (voice)
    - WIMAX (802.16) broadband wireless
    - 5G requirements: 1 Gbps downlink, latency 0.5 ms, Capacity target 15 Tbps/km$^2$, 250K users/km$^2$ for indoor hotspots such as office environments.
      Support for fixed, mobile, wireless and satellite access technologies. Connecting to billions of devices
    - 6/7 G evolving now, integration of AI and multimedia

# Wireless LANs 802.11

- Started in the mid 1990's , Goal: connecting laptops to the Internet, Nickname WiFi, Operates in unlicensed bands and not (expensive) licensed spectrum
- Two modes of operation
  - Using an Access Point which connects to the (wired) internet
  - As an Ad Hoc Network communicating with other mobile devices without an access point
- Usage: more nomadic than mobile
- Environmental concerns
  - Radio signals affected by the weather
  - Signals bounce off of objects resulting reception following multiple paths and fading of the signal
- Bandwidth ! (more recently use of different channels, MIMO spatial streams, etc.)
  - 802.11b: rates up to 11 Mbps (Wi-Fi 1)
  - 802.11g : rates up to 54 Mbps (Wi-Fi 3: OFDMA modulation scheme introduced)
  - 802.11n: 600 Mbps (Wi-Fi 4: multiple antennas, from 2009)
  - 802.11ac: 1 Gbps (Wi-Fi 5: 2.4 GHz and 5 GHz multiuser MIMO, 2013)
  - 802.11ax: (Wi-Fi 6: (adds 6 GHz channel 2019)<br>802.11be: Wi-Fi 7: (46 Gbps 2024)

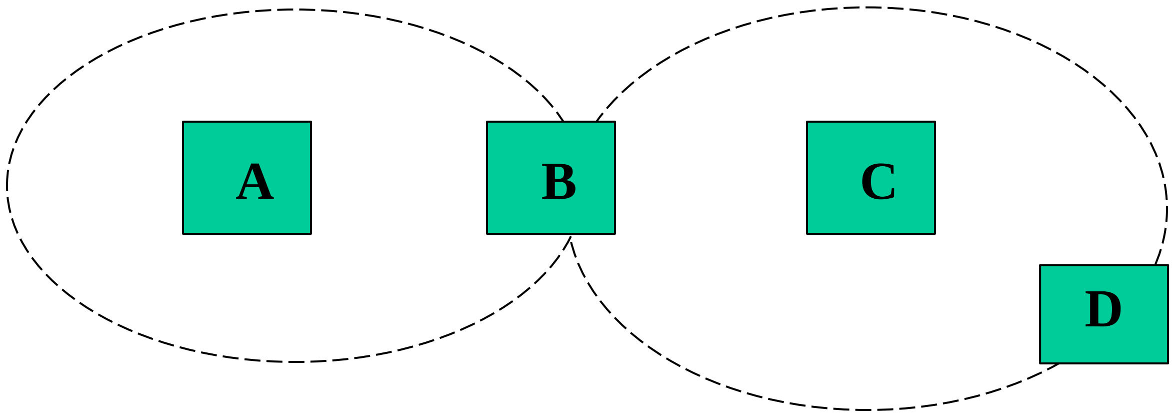
# Overview of Wireless LAN Protocols

- Differences from wired LANs
  - It is difficult to detect collisions while they are occurring
  - The received signal  is much fainter than the transmitted signal
  - Limited radio range of stations – hence all stations may not be able to receive  a signal from a transmitting station



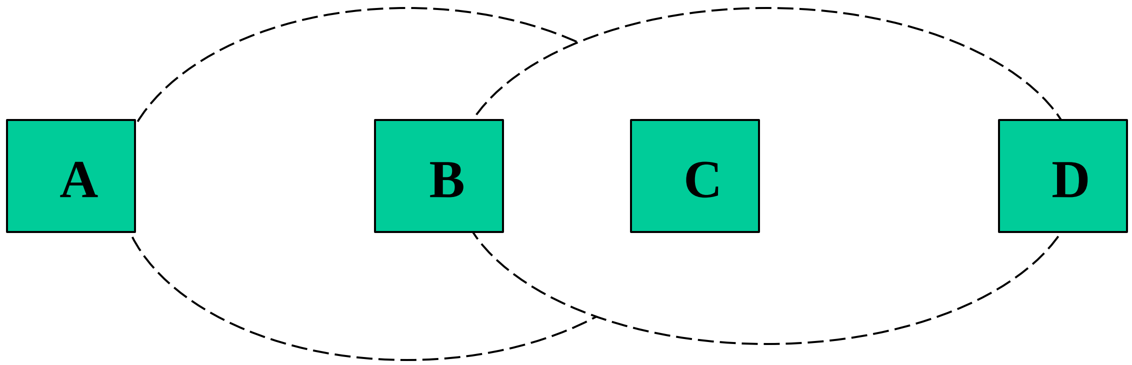

- Basic transmission / reception issues
  - When a station transmits it does not receive. So it does one or the other.
  - There are very few frequencies, and these are chosen ad-hoc. So if two different stations are transmitting to a third station within signal range, the transmissions collide, and the receiving station gets a garbled reception
  - Discussing actions using carrier sense only. Transmitter cannot detect collisions at another station

# Overview of Wireless LAN Protocols




The **hidden terminal problem**: when A transmits to B, A is hidden from C. Suppose C is transmitting to D. Since C is hidden from A, if A decides to transmit to B then A’s transmission would interfere with C’s transmission (to D) at B. But A has no way to know this. So any protocol for transmission must be cognizant of this.

# Overview Continued

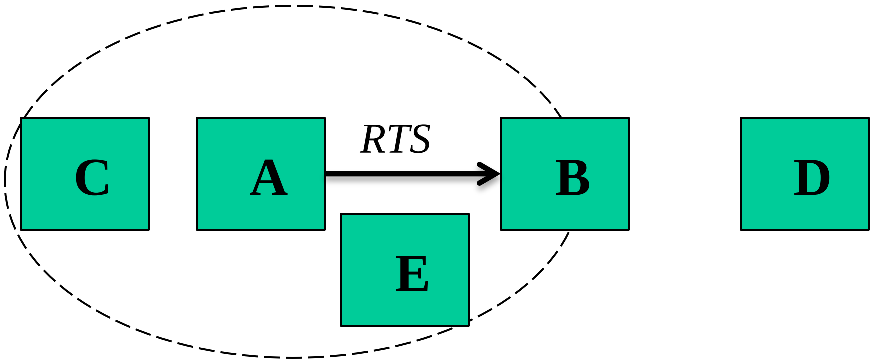



The **Exposed terminal problem**: assume B is transmitting to A. If C want to transmit to D there is actually no problem doing so. But C hears B’s transmission and believes it should wait until B is finished.  B is exposed to C. There is really no reason it should wait. So any protocol for transmission must be cognizant of this also.

# Multiple Access with Collision Avoidance

- Clearly CSMA (carrier sense multiple access) does not work, where we sense if someone else is transmitting and defer and if not we go ahead and transmit.
- MACA was developed by Karn in 1990. It solves some of these problems. Suppose station A wants to transmit to station B.
  - A first sends a Request to Send (RTS) packet to B. This is a short packet of 30 bytes. It also contains the length of the data frame that will follow. Any station within range of A waits enough time for the response from B to get back to A.
  - B replies to the RTS by sending a Clear to Send (CTS) if B does not hear anyone else that is transmitting in B's range. The reply also contains the length of the data frame that A wishes to send. Any station within range of B's CTS must remain silent through the upcoming data transmission from A to B.

# Actions of various stations under MACA

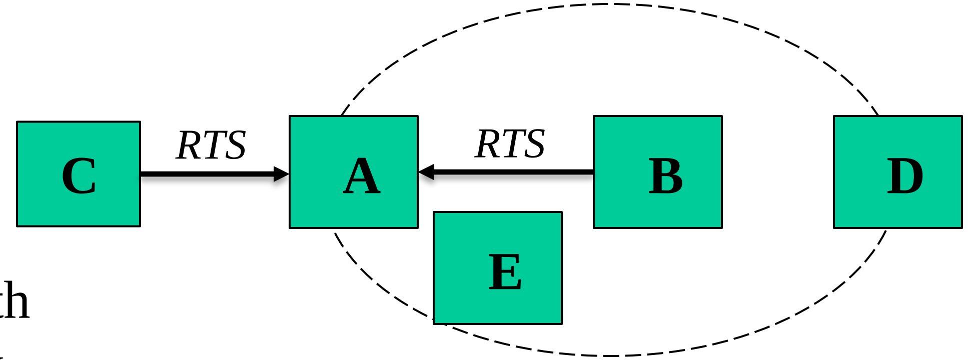



C hears A's RTS and waits until A should get the CTS. After that it is free to transmit since C did not hear B's reply and hence will not interfere with B's reception.

E does the same but also hears the reply and waits until A's data frame is sent.

D does not hear the RTS but as soon as it hears the CTS it waits until A's data frame is sent.

Note however that if both C and B send an RTS to A these will collide. Since neither hears a CTS back both wait a random time and retry.



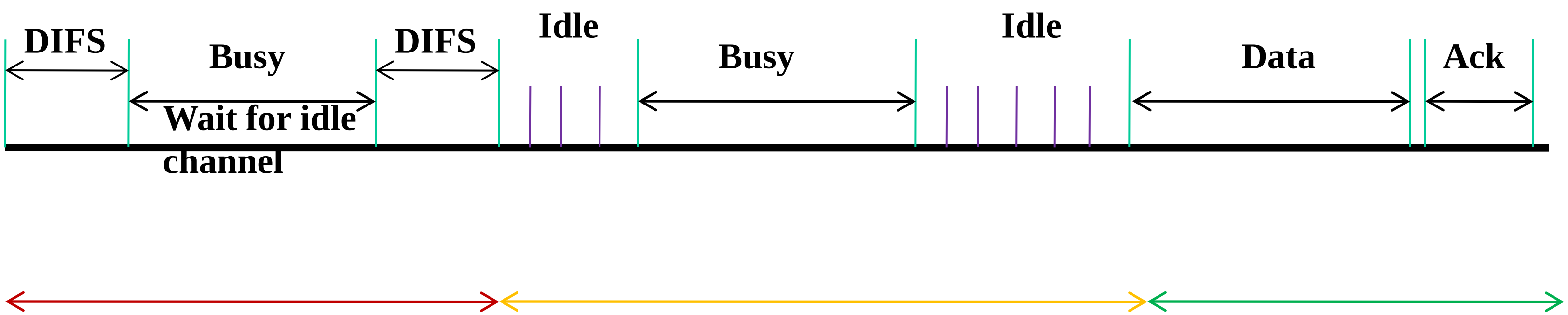
# 802.11 MAC Layer Protocol: CSMA/CA

- The protocol that is actually used is call CSMA with Collision Avoidance
- Two modes:
  - DCF (distributed coordination function) where each station behaves independently.
  - PCF (point coordination function) where the access point (AP) behaves like a cellular base station and controls all activity. Not often used in practice.
- The protocol is fairly complex
- An RTS/CTS mechanism (similar but not exactly the same as MACA) is optional

# Basic Ideas of the CSMA/CA

- Once a data frame  starts being sent, since there is no collision detection, the entire frame is sent.
- The destination sends an immediate ACK if the data frame arrive ok.
- Lack of acknowledgement means there was a problem and the backoff period is doubled etc. as in the Ethernet truncated exponential backoff algorithm.
- Backoff is viewed as determining a number of slots to wait before trying to transmit. Different from CSMA/CD, this means once the backoff value is determined (say 10 slots) the channel is monitored as being either busy or idle. The slot times to wait are counted down to 0 only during the idle periods. If the channel is free and slot times to wait is 0, the data frame is transmitted.
- After another station finishes, a station desiring to transmit first waits for the interframe spacing time. This is typically the value DIFS (DCF Interframe Spacing).

# Sending a frame with CSMA/CA – Basic Channel Acquisition



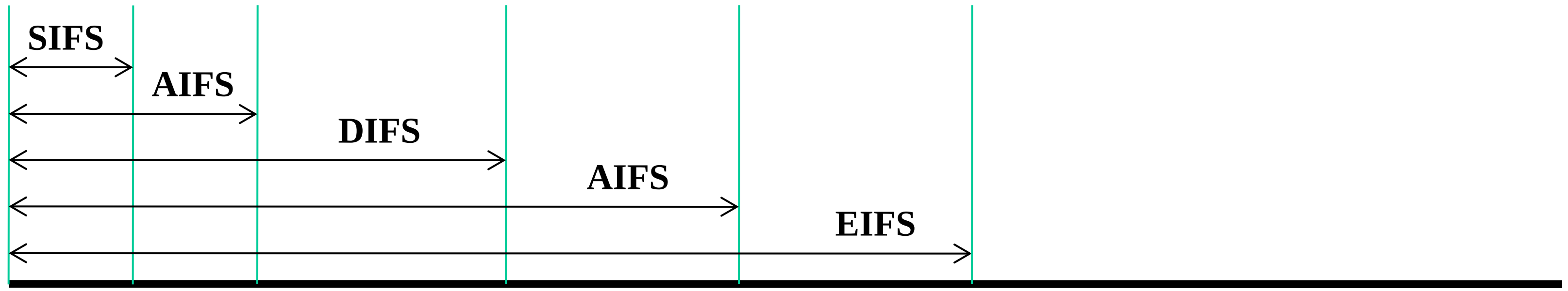

**Defer until contention period which is after idle channel and interframe spacing**

**Contention period. Choose backoff number of slots. Count down slots (say 10) when channel is idle.**

**Transmit data frame and get ack**

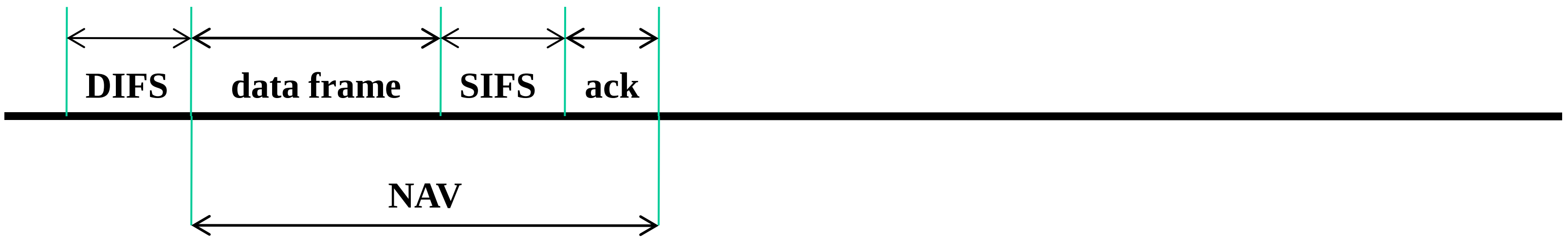
# Interframe Spacing in 802.11

- Five different Interframe interval are defined each longer than the previous.
  - SIFS (short interframe spacing): this is when you need the channel as soon as possible and need to send a control frame or the next fragment of a frame. Also used for ACKs, CTS, etc.
  - $AIFS_1$ (arbitration interframe space 1): for sending a high-priority frame
  - DIFS (DCF interframe spacing): normal spacing for any station to contend for the next frame to be sent.
  - $AIFS_4$ (arbitration interframe space 4): for sending a low-priority frame
  - EIFS (extended interframe spacing): bad frame or reporting a problem
- Exact values depend on the version of 802.11. Also, other names are sometimes used for the AIFs.



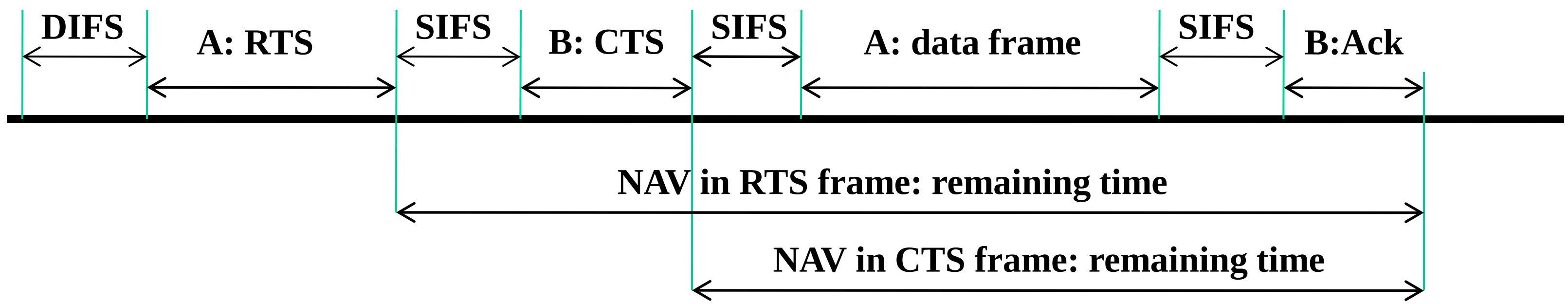
# Virtual Sensing

- So far, we have essentially sensed if the channel is idle / busy by actually seeing if anyone is transmitting. This is called physical sensing.
- We can also sense if the channel is idle / busy by virtual sensing which simply means we know by information in some previous frame sent how long the channel is to be busy by the next frame.
- This is by tracking the NAV (Network Allocation Vector). Each frame sent has a NAV field that says how long is the exchange sequence associated which this frame. For example, a data frame NAV would include the time needed to send an ack.



# Using RTS & CTS

- RTS & CTS can be used to solve the hidden terminal problem. For many reasons, this is generally not used.
- Sender A: sends RTS to receiver B after DIFS.
- Receiver B sends CTS back to A after SIFS.
- Sender A sends data frame to B after SIFS.
- Receiver B acks to A after SIFS if frame was received ok.
- NAV used by other stations to decide how long to wait. Note that all stations wait for the full time as compared to the MACA protocol. Thus hidden terminal problem ok but not exposed terminal problem.



# Other Issues

- Fragmenting frames for better reliability
- Power saving mode using periodic signals from the AP called beacon frames
- Format of the data and control frames
- Rate anomalies – fairness between fast and slow senders
  - 1 frame each
  - Equal air time
- How do mobile stations connect to APs
  - Association
  - Reassociation
- Authentication
  - WPA, WPA2, WPA3