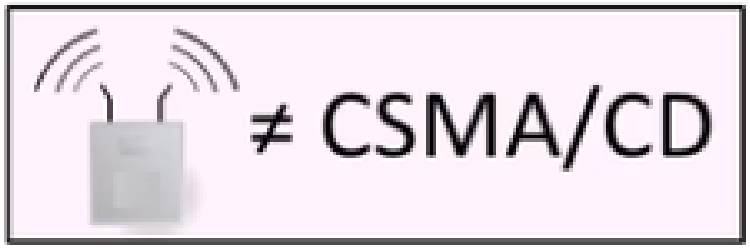
Wireless Multiple Access

Overview

- How do wireless nodes share a common link? (Yes, this is WiFi!)
- Build on our simple, wired model

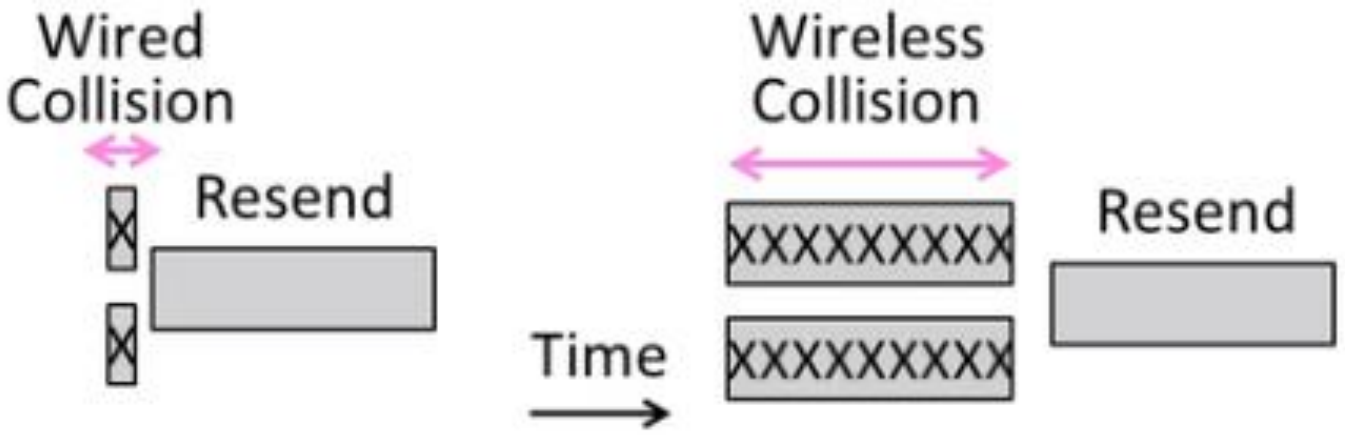
Wireless Complications

- Wireless is more complicated than the wired case
  1. Nodes may have different areas of coverage-doesn't fit Carrier Sense
  2. Nodes can't hear while sending- can't Collision Detect



Nodes Can't Hear While Sending

- With wires, detecting collisions (and aborting) lowers their cost
- More wasted time with wireless



Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- In a wired network, the received signal has almost the same energy as the sent signal because either the length of the cable is short or there are repeaters that amplify the energy between the sender and the receiver.
- This means that in a collision, the detected energy almost doubles.

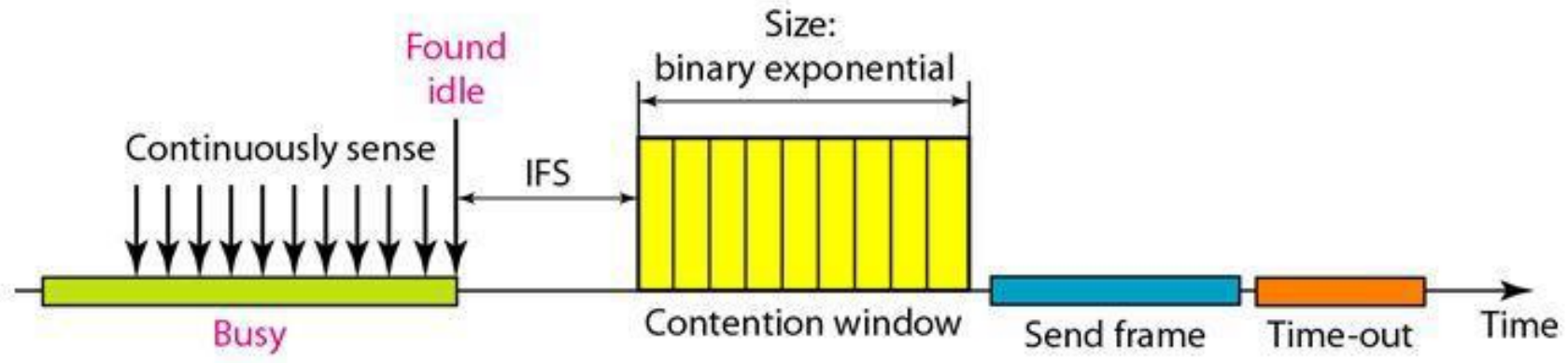
Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- In a wireless network, much of the sent energy is lost in transmission.
- The received signal has very little energy.
- Therefore, a collision may add only 5 to 10 percent additional energy.
- This is not useful for effective collision detection.

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- CSMA/CA protocol is used in wireless networks because they cannot detect the collision so the only solution is collision avoidance.
- CSMA/CA avoids the collisions using three basic techniques.
  i. Interframe space
  ii. Contention window
  iii. Acknowledgements

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)



Interframe Space (IFS)

- Whenever the channel is found idle, the station does not transmit immediately. It waits for a period of time called interframe space (IFS).
- When channel is sensed to be idle, it may be possible that same distant station may have already started transmitting and the signal of that distant station has not yet reached other stations.

Interframe Space (IFS)

- Therefore the purpose of IFS time is to allow this transmitted signal to reach other stations.
- If after this IFS time, the channel is still idle, the station can send, but it still needs to wait a time equal to contention time.
- IFS variable can also be used to define the priority of a station or a frame.

Contention Window

- Contention window is an amount of time divided into slots.
- A station that is ready to send chooses a random number of slots as its wait time.
- The number of slots in the window changes according to the binary exponential back-off strategy. It means that it is set of one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time.

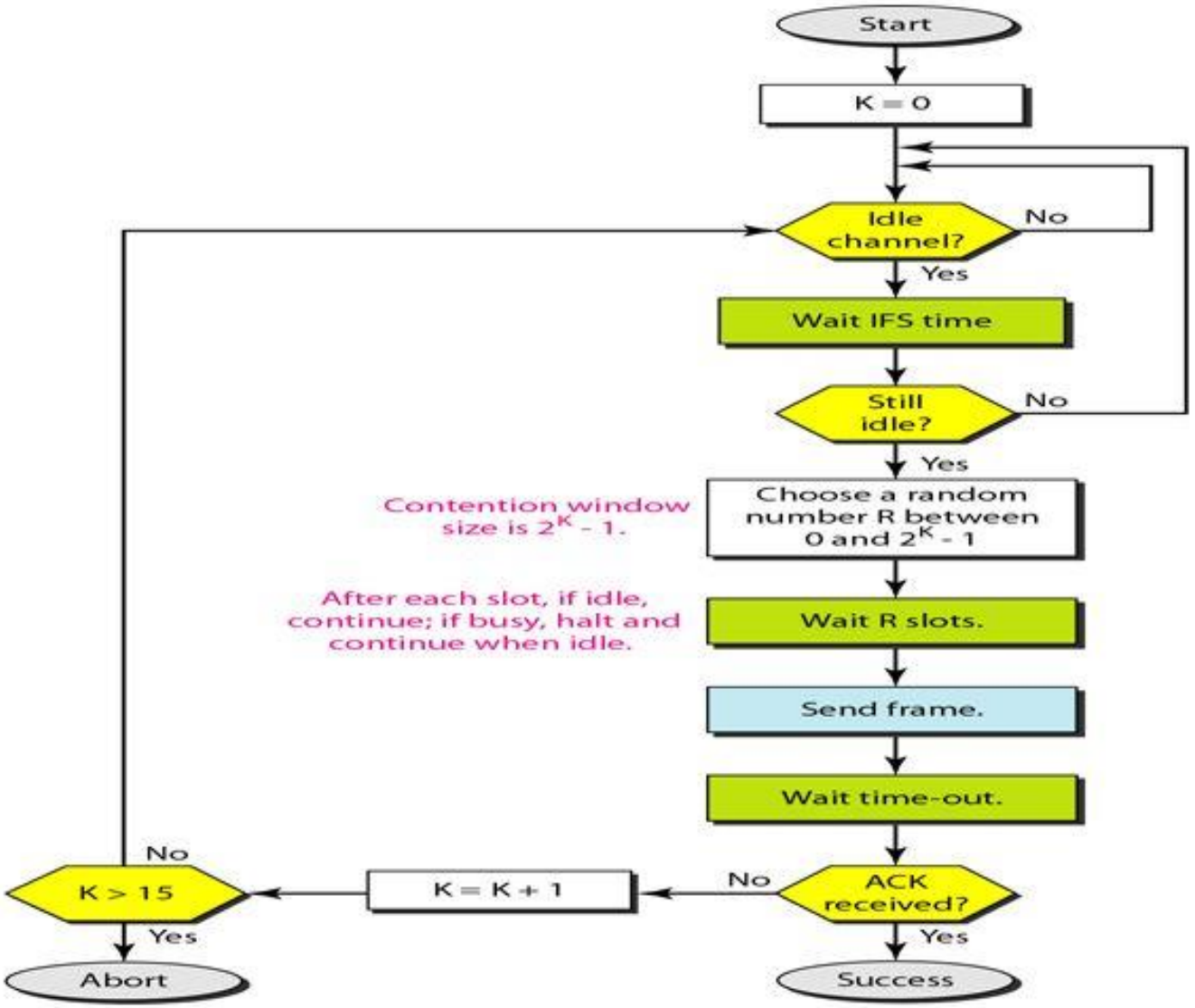
Contention Window

- This is very similar to the p-persistent method except that a random outcome defines the number of slots taken by the waiting station.
- In contention window the station needs to sense the channel after each time slot.
- If the station finds the channel busy, it does not restart the process. It just stops the timer & restarts it when the channel is sensed as idle.

Acknowledgement

- Despite all the precautions, collisions may occur and destroy the data.
- The positive acknowledgment and the time-out timer can help guarantee that receiver has received the frame.

CSMA/CA Procedure



CSMA/CA Procedure

- This is the CSMA protocol with collision avoidance.
- The station ready to transmit, senses the line by using one of the persistent strategies.
- As soon as it find the line to be idle, the station waits for an IFG (Interframe gap) amount of time.
- If then waits for some random time and sends the frame.

CSMA/CA Procedure

- After sending the frame, it sets a timer and waits for the acknowledgement from the receiver.
- If the acknowledgement is received before expiry of the timer, then the transmission is successful.
- But if the transmitting station does not receive the expected acknowledgement before the timer expiry then it increments the back off parameter, waits for the back off time and resenses the line.

CSMA/CA and Wireless Networks

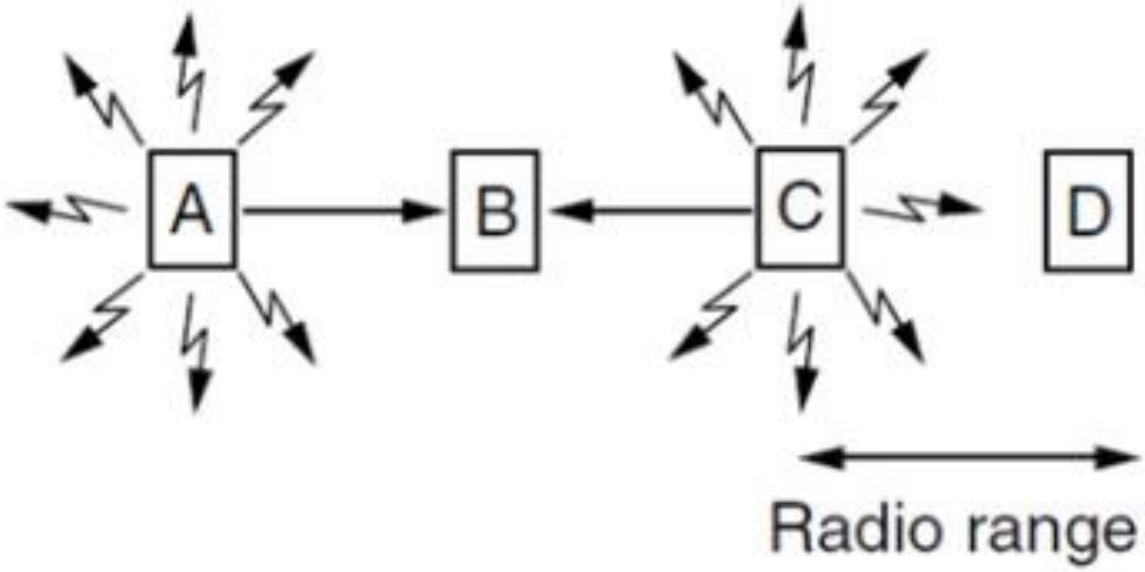
- CSMA/CA was mostly intended for use in wireless networks.
- However, it is not sophisticated enough to handle some particular issues related to wireless networks, such as hidden terminals or exposed terminals.

Difference between CSMA/CA and CSMA/CD

S.NO	CSMA/CD	CSMA/CA
1.	CSMA / CD is effective after a collision.	Whereas CSMA / CA is effective before a collision.
2.	CSMA / CD is used in wired networks.	Whereas CSMA / CA is commonly used in wireless networks.
3.	It only reduces the recovery time.	Whereas CSMA/ CA minimizes the possibility of collision.
4.	CSMA / CD resends the data frame whenever a conflict occurs.	Whereas CSMA / CA will first transmit the intent to send for data transmission.
5.	CSMA / CD is used in 802.3 standard.	While CSMA / CA is used in 802.11 standard.
6.	It is more efficient than simple CSMA(Carrier Sense Multiple Access).	While it is similar to simple CSMA(Carrier Sense Multiple Access).

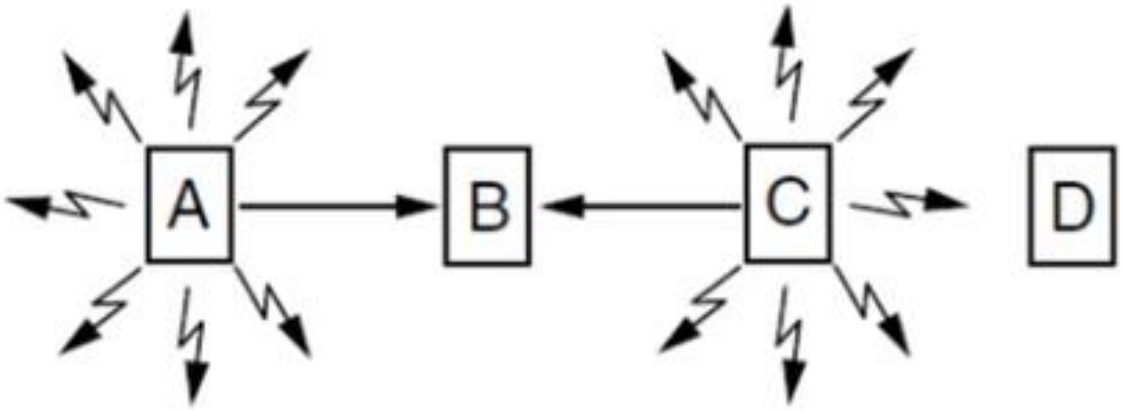
Different Coverage Areas

- Wireless signal is broadcast and received nearby, where there is sufficient SNR



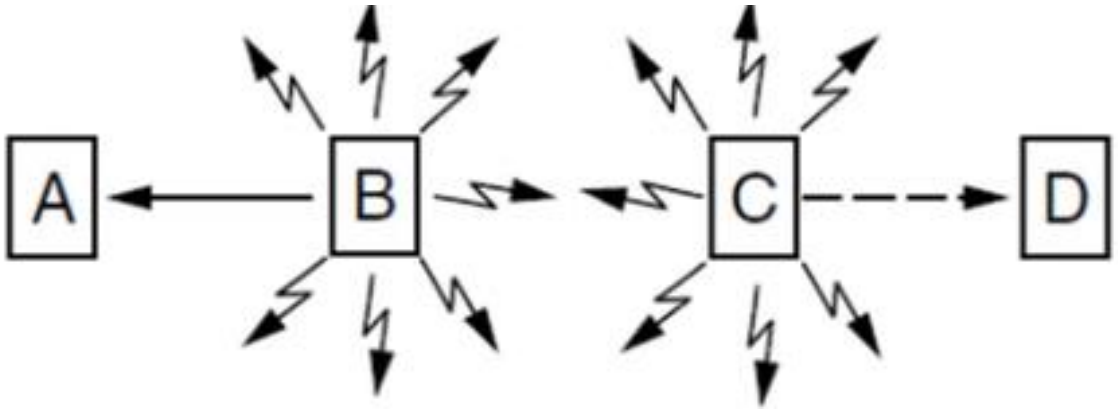
Hidden Terminals

- Nodes A and C are hidden terminals when sending to B
- Can't hear each other (to coordinate) yet collide at B
- We want to avoid the inefficiency of collisions



Exposed Terminals

- B and C are exposed terminals when sending to A and D
  - Can't hear each other yet don't collide at receivers A and D
  - We want to send concurrently to increase performance



Possible Solution: MACA

- MACA uses a short handshake instead of CSMA (Karn, 1990)
- Protocol rules:
  1. A sender node transmits a RTS (Request-To-Send, with frame length)
  2. The receiver replies with a CTS (Clear-To-Send, with frame length)
  3. Sender transmits the frame while nodes hearing the CTS stay silent

  —Collisions on the RTS/CTS are still possible, but less likely

MACA-Hidden Terminals

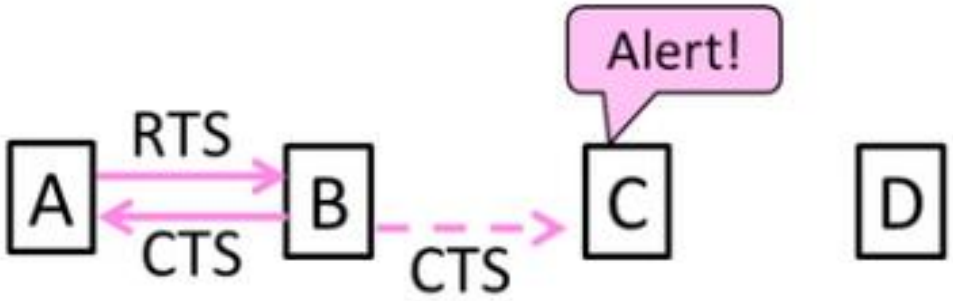
- A → B with hidden terminal C
  1. A sends RTS, to B

MACA-Hidden Terminals (2)

- A → B with hidden terminal C
  2. B sends CTS, to A, and C too

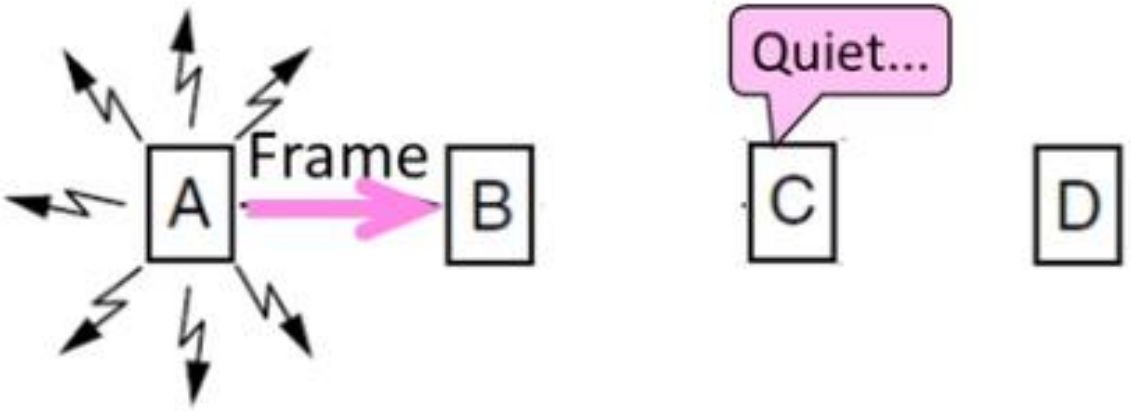
MACA-Hidden Terminals (3)

- A → B with hidden terminal C
  2. B sends CTS, to A, and C too



MACA-Hidden Terminals (4)

- A → B with hidden terminal C
  3. A sends frame while C defers



MACA-Exposed Terminals

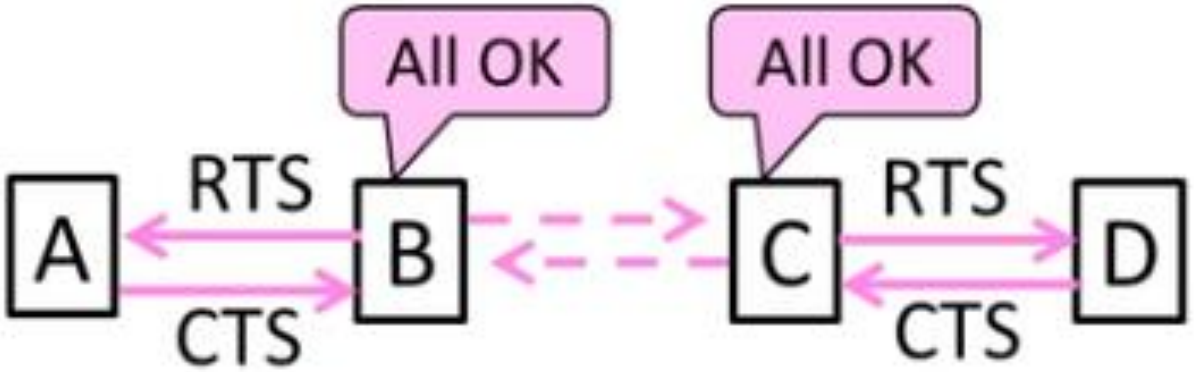
- B → A, C → D as exposed terminals
  - B and C sends RTS to A and D

MACA-Exposed Terminals (2)

- B → A, C → D as exposed terminals
  - A and D send CTS to B and C

MACA-Exposed Terminals (3)

- B → A, C → D as exposed terminals
  - A and D send CTS to B and C



MACA-Exposed Terminals (4)

- B → A, C → D as exposed terminals
  - A and D send CTS to B and C

Simplified CSMA/CA flowchart diagram with and without RTS/CTS