

Wireless Networks

Youssef Ghatas



Acknowledgement

- Slides are adapted from the lectures of
 - Dr. Sandra Wahid (S023)
 - Dr. Hoda Baraka (Before S2022)



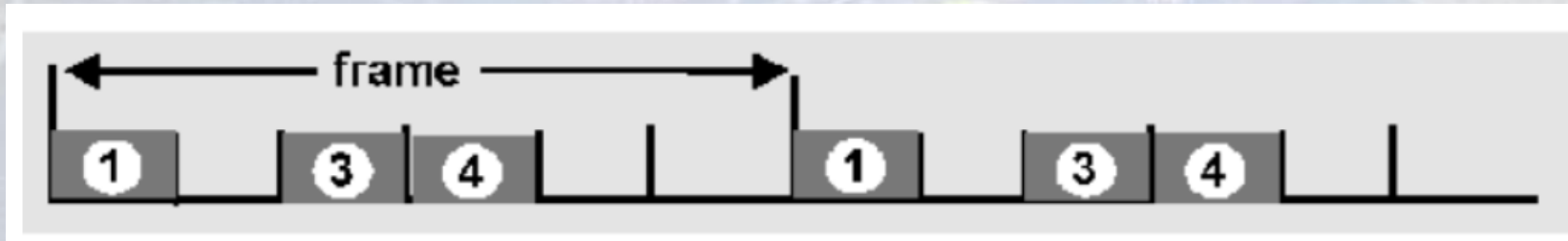
Medium Access Control

- Required for any shared medium
- Goals:
 - Maximize channel utilization
 - Minimize delay
- Static (also called Channelization Protocols)
 - FDMA
 - TDMA
 - CDMA
- Dynamic
 - Centralized vs. Distributed
 - Contention-based vs. Contention-free



Static MAC Protocols: TDMA

- TDMA: time division multiple access
- Access to channel in "rounds"
- Each station gets fixed length slot (length = $x \times \text{packet transmission time}$) in each round (e.g.: $x=1$, one packet each round)
- Unused slots go idle



Static MAC Protocols: FDMA

- FDMA: frequency division multiple access
- Channel spectrum divided into frequency bands
- Each station assigned fixed frequency band
- Unused transmission time in frequency bands go idle



Dynamic: Contention-Based vs. Contention-Free

- Contention-Based (Random Access)

- stations compete, sometimes leading to collisions
- “**recover**” from collisions
- typically a **distributed** scheme
- Examples: Wi-Fi - Bluetooth

Unslotted
ALOHA

Slotted
ALOHA

CSMA

CSMA/CD

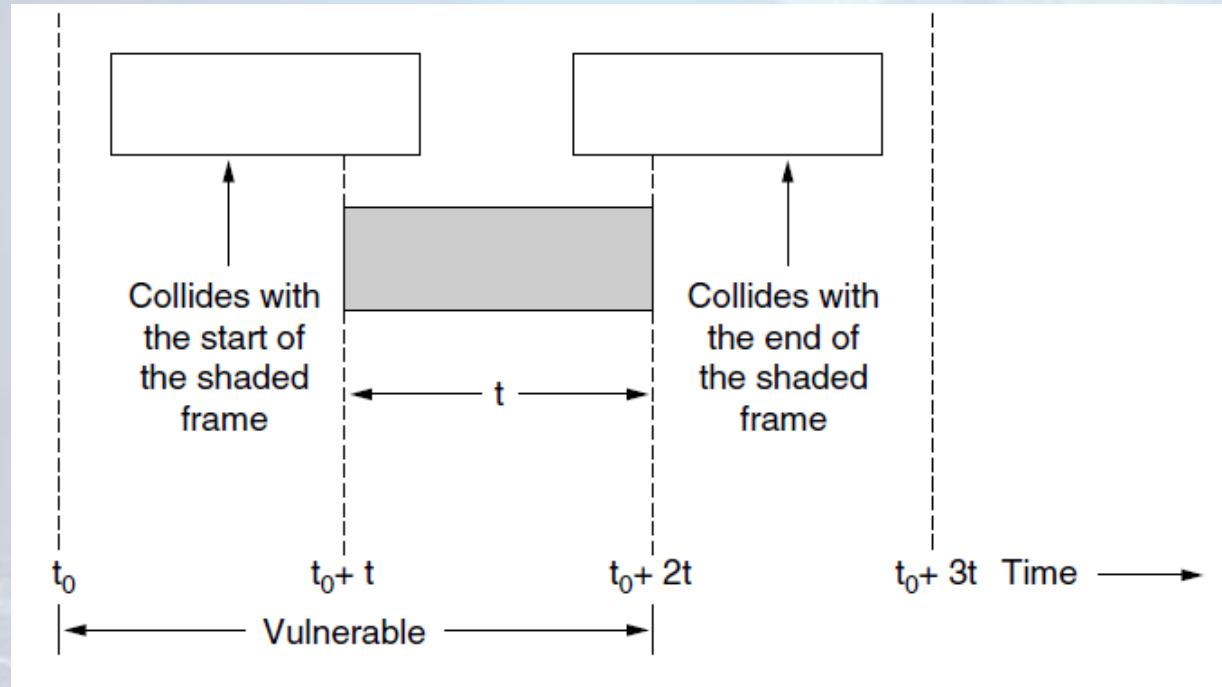
- Contention-Free

- tightly coordinate shared access to **avoid** collisions
- each node is granted temporary exclusive access to the medium
- usually a **centralized** scheme
- Example: Wireless Sensor networks (WSN)



Unslotted ALOHA

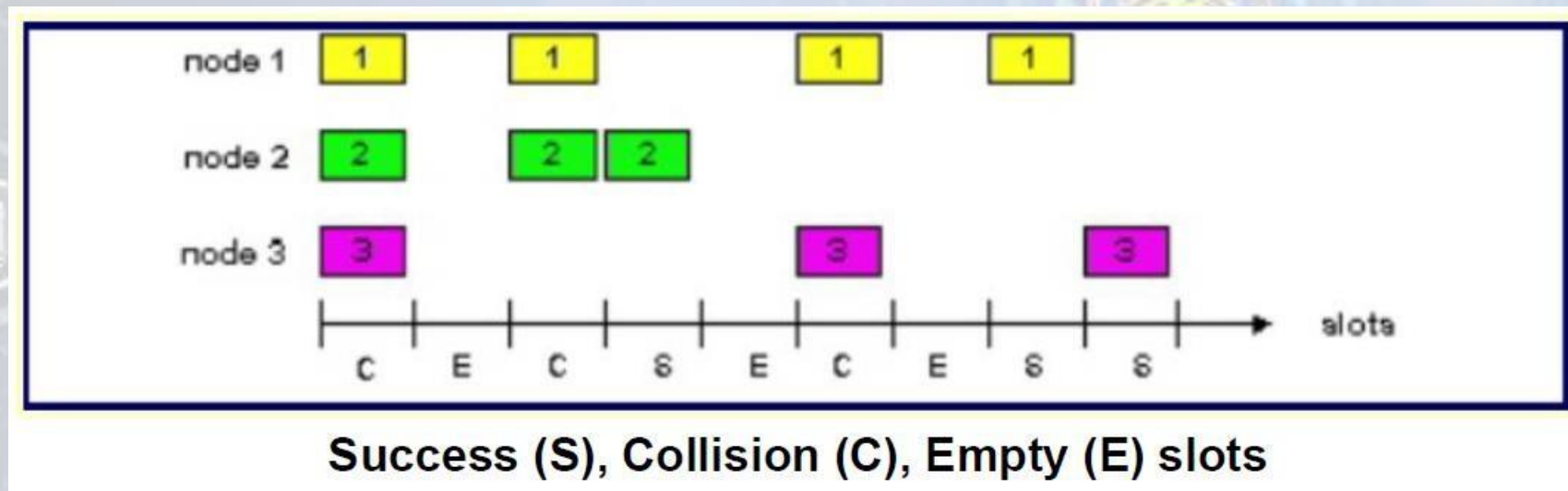
- No synchronization
- Packet needs transmission:
 - Send without awaiting for beginning of slot



- Vulnerability period (time where collision can happen) = $2t$

Slotted ALOHA

- Time is divided into equal size slots
- Transmits only at the **beginning of the slot**
 - Still does not check if other stations are transmitting before it transmits
- Vulnerability period (time where collision can happen)= t



Carrier Sense Multiple Access (CSMA)

- Listen to the channel before transmitting
 - If the channel is sensed busy, then the user waits until the channel becomes idle
- This avoids collision with ongoing transmission.
- Collisions may still occur



Why??

If two stations become ready in the middle of a third station's transmission, both will wait until the transmission ends, and then both will begin transmitting exactly **simultaneously** (when the channel becomes free), resulting in a collision.



CSMA with Collision Detection (CSMA/CD)

- Listen to the channel before transmitting
 - If idle, start transmitting
- Listen to the channel while transmitting
 - If collision is detected:
 - Abort immediately
 - Back off for some time then retry
 - Randomized back off interval to avoid repeated collisions
- Maximum time to detect collision (Contention slot) = twice maximum propagation delay

Why??

Consider the following worst-case scenario. Let the time for a signal to propagate between the two farthest stations be τ . At t_0 , one station begins transmitting. At $t_0 + \tau - \epsilon$, an instant before the signal arrives at the most distant station, that station also begins transmitting. Of course, it detects the collision almost instantly and stops, but the little noise burst caused by the collision does not get back to the original station until time $2\tau - \epsilon$. In other words, in the worst case a station cannot be sure that it has seized the channel until it has transmitted for 2τ without hearing a collision.



Back-off Algorithm

- After undergoing the collision:
 - Transmitting station chooses a **random number** in the range **$[0, 2^n - 1]$** if the packet is undergoing collision for the n^{th} time.
 - If station chooses a number k , then **Back off time = $k \times \text{Time slot}$** (Time slot = $2t$)

CW is the contention window

• Consider the following scenario: (for simplicity set Time slot = 1)

1] 1st data packet of both stations A and D:

- Both A and D start transmitting their 1st data packet simultaneously.

→ This leads to a collision.

- The collision on both the packets is occurring for the 1st time.

→ n for the 1st data packet of both stations = 1.

- At station A: after detecting the collision:

- Station A randomly chooses a number in the range $[0, 2^1 - 1] = [0, 1]$.
- If station A chooses the number K_A , then back off time = K_A units.

- At station D: after detecting the collision:

- Station D randomly chooses a number in the range $[0, 2^1 - 1] = [0, 1]$.
- If station D chooses the number K_D , then back off time = K_D units.

Back-off Algorithm

■ 4 possible cases:

K_A	K_D	Remarks
0	0	<ul style="list-style-type: none">• In this case, both the stations start retransmitting their data immediately.• This case leads to a collision again.
0	1	<ul style="list-style-type: none">• In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time.• This case leads to A successfully retransmitting its data after the 1st collision.
1	0	<ul style="list-style-type: none">• In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately.• This case leads to D successfully retransmitting its data after the 1st collision.
1	1	<ul style="list-style-type: none">• In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously.• This case leads to a collision again.

- Probability of A to successfully retransmit its data after the 1st collision = $1/4$
- Probability of D to successfully retransmit its data after the 1st collision = $1/4$
- Probability of occurrence of collision again after the 1st collision = $2/4 = 1/2$



Back-off Algorithm

2] Consider case2 (A successfully retransmits its 1st packet after the 1st collision):

Now we have 2nd data packet of A and 1st data packet of D:

- A starts transmitting its 2nd packet and D starts transmitting its 1st packet simultaneously.

→ This leads to a collision.

- For A collision of the 2nd packet for the 1st time

→ n for the 2nd data packet = 1.

- Station A randomly chooses a number in the range $[0, 2^1 - 1] = [0, 1]$.
- If station A chooses the number K_A , then back off time = K_A units.

- For D collision of the 1st packet for the 2nd time

→ n for the 1st data packet = 2.

- Station D randomly chooses a number in the range $[0, 2^2 - 1] = [0, 3]$.
- If station D chooses the number K_D , then back off time = K_D units.



Back-off Algorithm

8 possible cases.

K_A	K_D	Remarks
0	0	<ul style="list-style-type: none"> In this case, both the stations start retransmitting their data immediately. This case leads to a collision again.
0	1	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. This case leads to A successfully retransmitting its data after the 2nd collision.
0	2	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. This case leads to A successfully retransmitting its data after the 2nd collision.
0	3	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. This case leads to A successfully retransmitting its data after the 2nd collision.

1	0	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. This case leads to D successfully retransmitting its data after the 2nd collision.
1	1	<ul style="list-style-type: none"> In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. This case leads to a collision again.
1	2	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. This case leads to A successfully retransmitting its data after the 2nd collision.
1	3	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. This case leads to A successfully retransmitting its data after the 2nd collision.



- Probability of A to successfully retransmit its data after the 2nd collision = $5/8$
- Probability of D to successfully retransmit its data after the 2nd collision = $1/8$
- Probability of occurrence of collision again after the 2nd collision = $2/8 = 1/4$

Back-off Algorithm

3] Consider case3 (A successfully transmits its 2nd packet after the 2nd collision):

Now we have 3rd data packet of A and 1st data packet of D:

- A starts transmitting its 3rd packet and D starts transmitting its 1st packet simultaneously.

→ This leads to a collision.

- For A collision of the 3rd packet for the 1st time

→ n for the 3rd data packet = 1.

- Station A randomly chooses a number in the range $[0, 2^1 - 1] = [0, 1]$.
- If station A chooses the number K_A , then back off time = K_A units.

- For D collision of the 1st packet for the 3rd time

→ n for the 1st data packet = 3.

- Station D randomly chooses a number in the range $[0, 2^3 - 1] = [0, 7]$.
- If station D chooses the number K_D , then back off time = K_D units.



Back-off Algorithm

16 possible cases.

K_A	K_D	Remarks						
0	0	<ul style="list-style-type: none"> In this case, both the stations start retransmitting their data immediately. This case leads to a collision again. 	0	6	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 6 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	4	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 4 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision.
0	1	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	0	7	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 7 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 			
0	2	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	0	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. This case leads to D successfully retransmitting its data after the 3rd collision. 	1	5	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 5 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision.
0	3	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	1	<ul style="list-style-type: none"> In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. This case leads to a collision again. 	1	6	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 6 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision.
0	4	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 4 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	2	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 			
0	5	<ul style="list-style-type: none"> In this case, station A starts retransmitting its data immediately while station D waits for 5 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	3	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision. 	1	7	<ul style="list-style-type: none"> In this case, station A waits for 1 unit of time while station D waits for 7 unit of time. This case leads to A successfully retransmitting its data after the 3rd collision.



- Probability of A to successfully retransmit its data after the 3rd collision = $13/16$
- Probability of D to successfully retransmit its data after the 3rd collision = $1/16$
- Probability of occurrence of collision again after the 3rd collision = $2/16 = 1/8$

Back-off Algorithm

- Known as **Binary Exponential Back Off Algorithm:**

1. It works for only **two** stations.

With each successive collision:

2. The back off time **increases exponentially.**
3. Collision probability **decreases exponentially.**

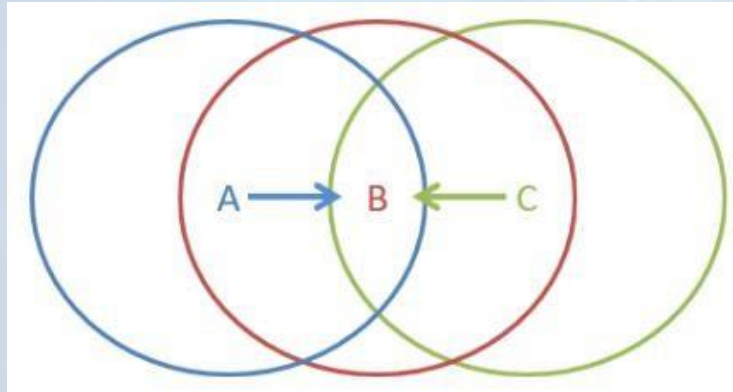
- One disadvantage of Back Off Algorithm:

If a particular station wins the collision one time, then its probability of winning the successive collisions increases exponentially.



A Problem with CSMA: the Hidden Terminal Problem

- It is a transmission problem that arises when two or more stations who are **out of range of each other transmit simultaneously to a common recipient**.
 - E.g.: This occurs when a station is visible from a wireless access point (AP), but is hidden from other stations that communicate with the AP.

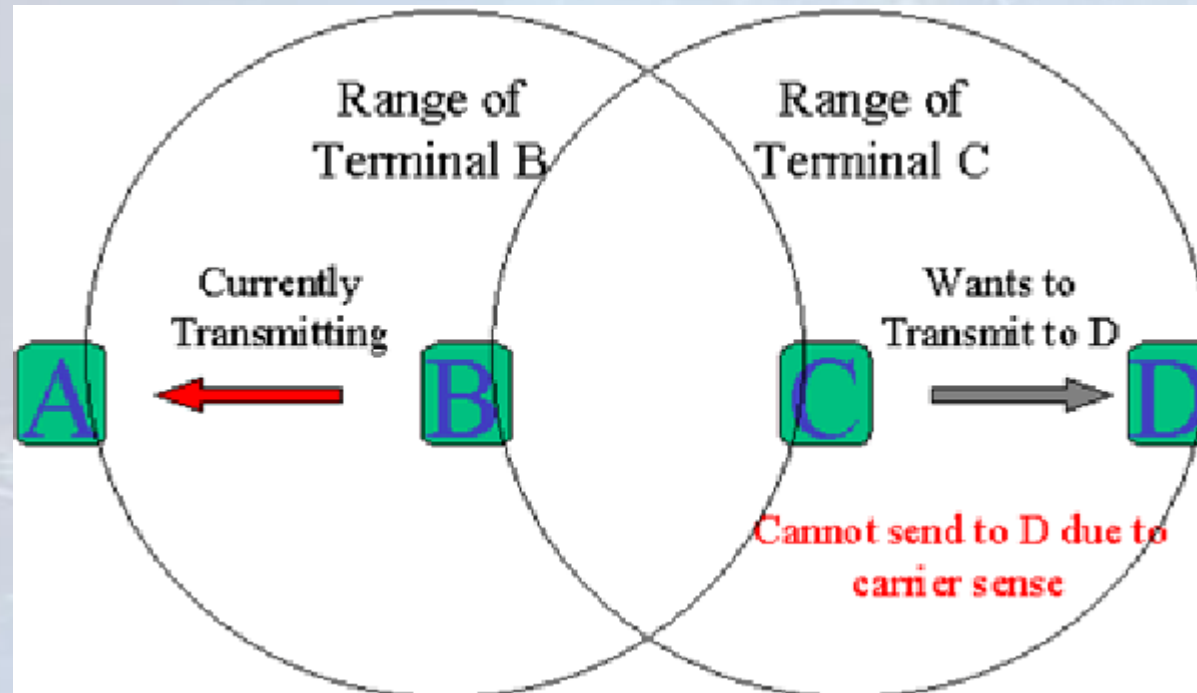


- Nodes A and C cannot hear each other.
 - Out of the sensing range.
- Transmissions by nodes A and C can collide at node B.
- Nodes A and C are hidden from each other.



The Exposed Terminal Problem

- A data transmission which **do not interfere may be blocked**.
- B is sending to A and C wants to send to D.
 - Both transmissions can happen without interference.
 - But C remains silent since it is in the sensing range of B.



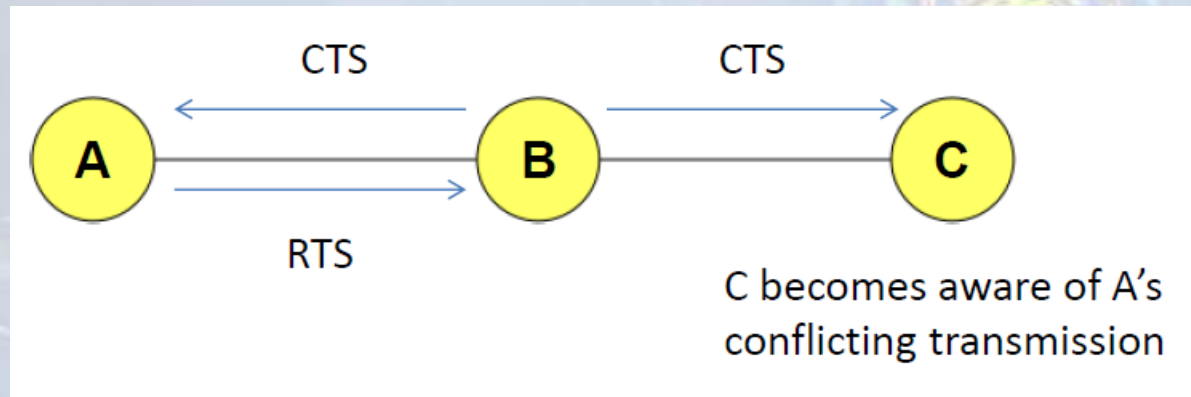
CSMA/CD vs CSMA/CA

- CSMA/CD does not work in wireless channel
 - difficult to detect collisions in a radio environment
 - Power level varies due to other factors such as fading
 - difficult to control the wireless channel
 - hidden and exposed terminal problems
- CSMA/CA is used in wireless environments
 - Sense before transmit
 - Contend for channel usage
 - “Collision detection” via the acknowledgment
 - Time-spacing to set priority
 - Avoid collisions via carrier sensing and virtual carrier sensing



The Hidden Terminal Solution

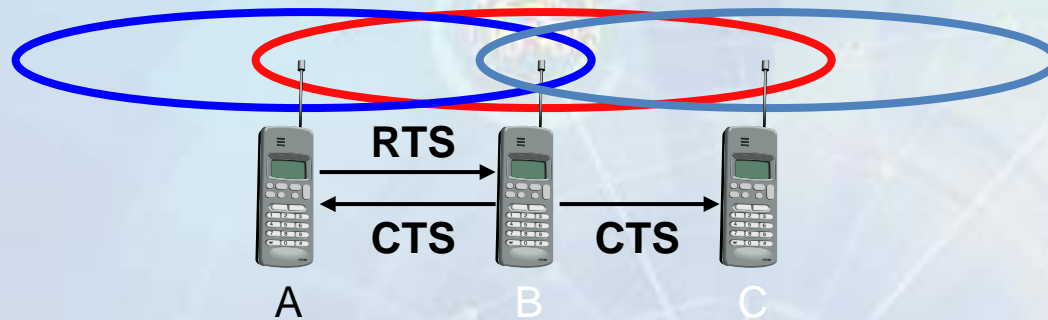
- **Signaling Packets (RTS/CTS)** is the mechanism used to resolve the hidden terminal problem.
- Sender first sends a **Request-to-Send (RTS)**
- If ready, receiver responds with **Clear-to-Send (CTS)**
- All nodes hearing RTS or CTS **keep quiet** for the duration of the transfer
 - Transfer duration included in both RTS and CTS.
- Signaling packets contain
 - sender address
 - receiver address
 - packet size



MAC Collision Avoidance (MACA) examples

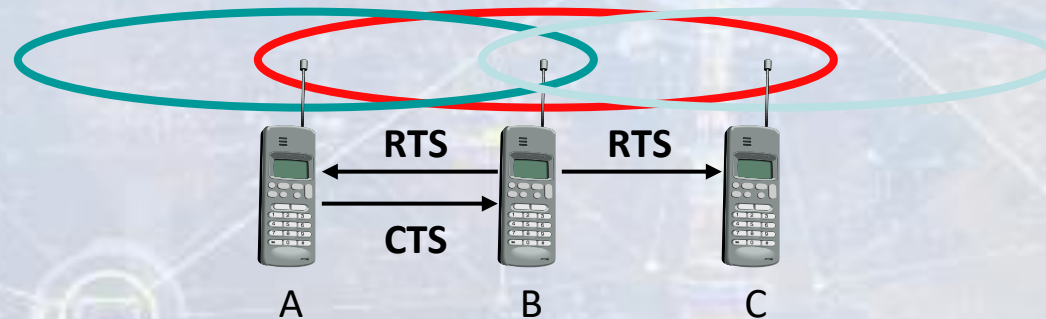
- MACA avoids the problem of hidden terminals

- A and C want to send to B
- A sends RTS first
- C waits after receiving CTS from B



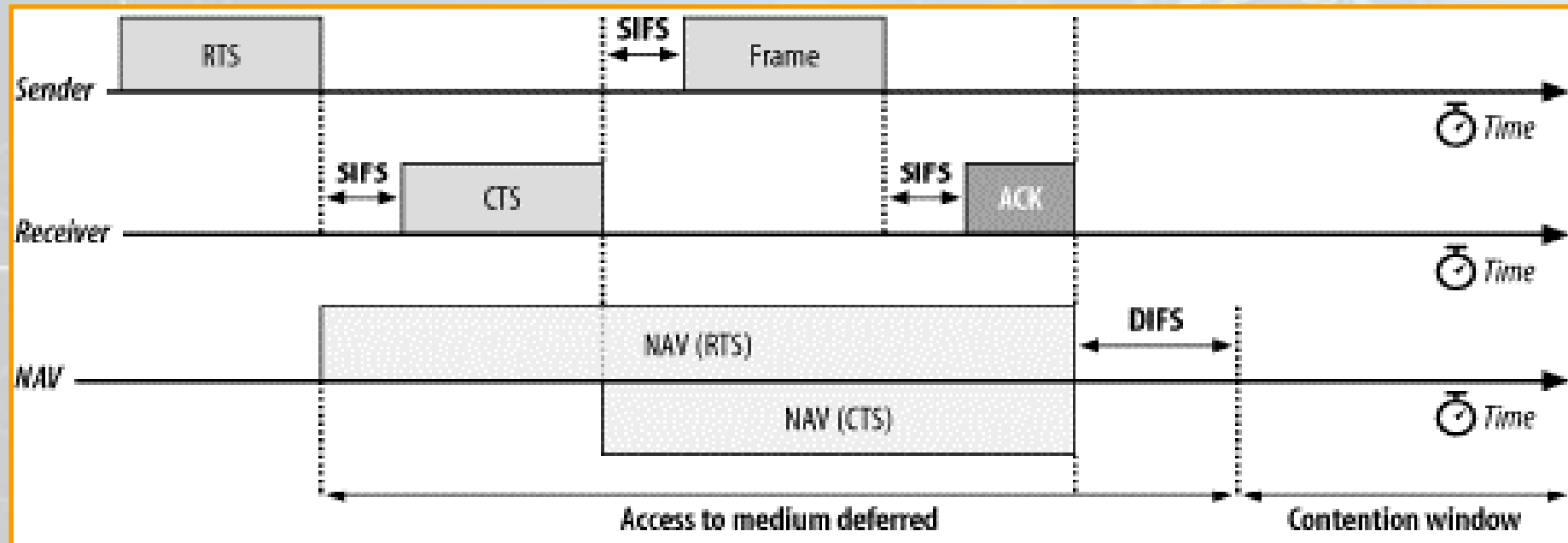
- MACA avoids the problem of exposed terminals?

- B wants to send to A, C to another terminal
- now C does not have to wait for it cannot receive CTS from A



NAV: Network Allocation Vector

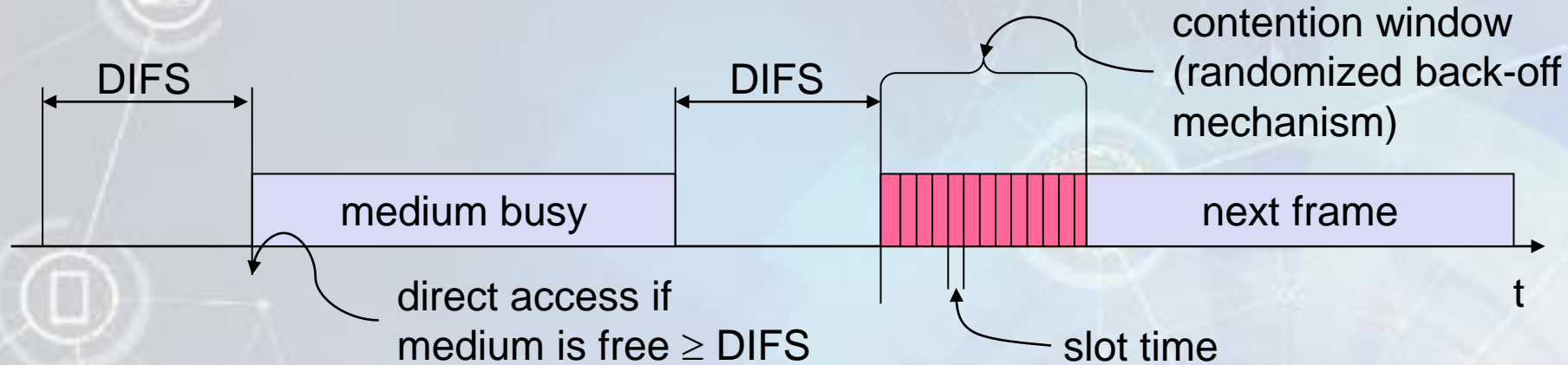
- **Idea:** Most 802.11 frames carry a *duration* field, used to reserve the medium for a fixed time period
- Station sets NAV field to the duration it will reserve the medium
 - Including RTS/CTS and inter-frame spacing
- The NAV is carried in the frame headers on the RTS and CTS
- Every station must remember the NAV value in the last frame



Virtual Carrier Sense

- **Network Allocation Vector (NAV)** is a virtual carrier-sensing mechanism used with wireless network protocols such as IEEE 802.11 (Wi-Fi).
- The virtual carrier-sensing is a logical abstraction which **limits the need for physical carrier-sensing** to save power.
- The MAC layer frame headers contain a **duration field** that specifies the transmission time required for the frame: time the medium will be busy.
- The stations listening on the wireless medium read the duration field and set their NAV, which is an indicator for a station on how long it must defer from accessing the medium

CSMA/CA with Contention Window

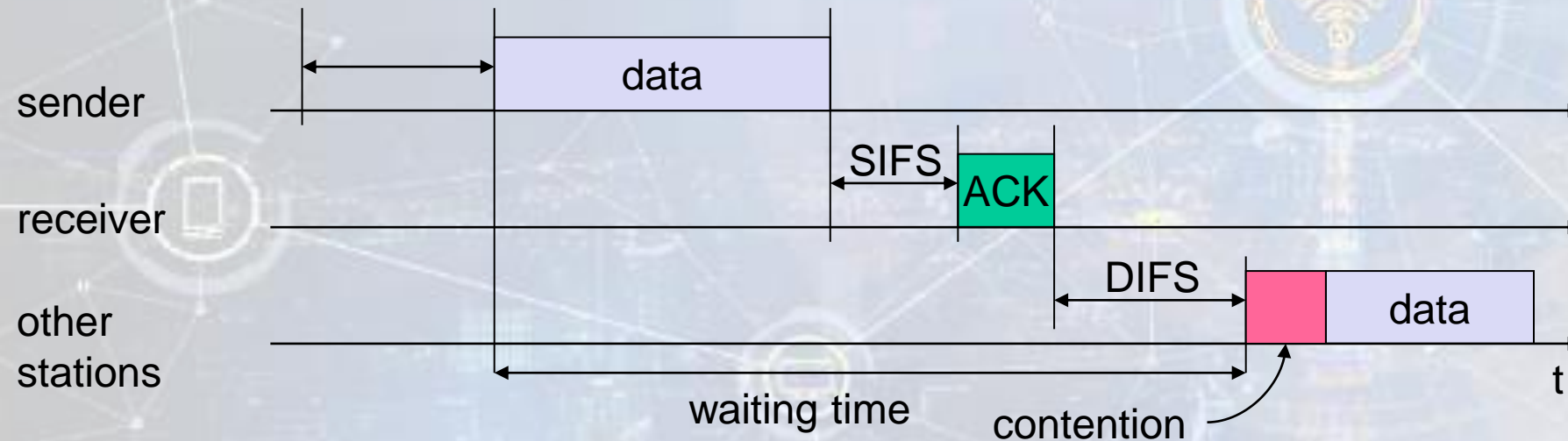


- If the medium is busy, the station has to wait, then the station must additionally wait a random back-off time (collision avoidance, multiple of slot-time)
- If another station occupies the medium during the back-off time of the station, the back-off timer pauses (fairness)



CSMA CA

- Sending unicast packets
 - station has to wait for DIFS before sending data
 - receivers acknowledge at once (after waiting for SIFS) if the packet was received correctly (CRC)
 - automatic retransmission of data packets in case of transmission errors





Thank You

