Medium Access Control

Now we are here

- OSI 7 layer model

Components of data link layer

What is multiple access?

- Techniques to share a common link
- point-to-point vs. shared
- In shared link, only one user should transmit at a time → medium access control necessary

point-to-point

케이블을 많이 안써서 좋음

shared

Types of multiple access protocols

동시에 말하는 충돌을 생기지 않게 하는 것

한번에 한 사람이 말할 때

운으로 말할 사람 결정

누가 말하는 순서를 줌

아예 말하는 게 다른 사람에게 안들리기

Random Access Protocols

Random access protocols

- Collisions are avoided using “randomness”

보낼 게 있으면 바로 보냄
충돌이 나면 좀 기다리고 보냄
기다리는 시간은 쟤하고 나하고 달라야함
랜덤이기 때문에 또 같이 말할 수 있음 -> 다시 기다림

- Similar to a group of people chatting in a room
  - No one knows when a person will start talking
  - If two person starts to talk at the same time, they both stop, wait for a moment, and start to talk again

ALOHA

- Developed in 1970s at University of Hawaii
- A basic protocol in the line of random access protocols
- Summary
  - If there is anything to send, send it.
  - If collision occurs, wait for a certain duration of time and retransmit

Pure ALOHA

- If sender has a frame to send, send it immediately.
- When receiver receives the frame, send ACK.
- If ACK does not arrive during timeout period, sender retransmits the frame
- However, the sender does not retransmit immediately; <u>the sender waits for a random delay and retransmits</u>

Pure ALOHA

- If two frames are on the link at the same time, a collision occurs

1이 1.1을 보내는 동안 아무도 안보냄 -> 성공

Pure ALOHA

- What does the sender do when a frame is lost?
  - Wait for some random time and retransmit

- How long should it wait?
  - Choose a random number R between 0 and 2^K-1

    같은 수 또는 인접한 수를 둘 다 고르면 다시 고름

  - Wait for T x R seconds (T: time slot)

  - K is incremented after each timeout

    충돌할 때마다 K 값을 증가 -> 기다리는 시간이 증가하지만 충돌 날 확률 줄어듬

    → <u>binary exponential backoff</u>

Pure ALOHA

16번이나 충돌하면 뭔가 다른 문제가 있다 판단해서 중지

Pure ALOHA

- Time slot T

time slot을 얼마로 정할 것인가
0,1처럼 가까운걸 골랐을 때 충돌 안되게 하려면
timeslot이 한 프레임 보내는 시간 이상으로 해야함

  - Fixed (T_s)

bandwidth / frame개수

  - Transmission time (T_{fr}): when $T_{fr} >> T_p$

  - Maximum propagation time (T_p): when $T_p >> T_{fr}$

거리가 길면 propagation time > transmission time
짧으면 반대

Example

- A network of nodes uses pure ALOHA as medium access protocol

- Backoff time T_B is calculated as follows:
  - $T_B = T \times R$
  - T (time slot) = 20ms
  - $R = rand(0, 2^K-1)$

- If K = 1, what are the possible backoff durations?

Vulnerable time

- Suppose a frame is transmitted on the channel. The <u>vulnerable time</u> is the duration of time in which a collision occurs, if another frame is transmitted.

Vulnerable time

- Vulnerable time of pure ALOHA
  - Assume T_p is minimal compared to T_{fr}
  - Suppose A's transmission starts at t and finishes at $t + T_{fr}$
  - No one should transmit between $t - T_{fr}$ and $t + T_{fr}$
  - Vulnerable time: $2 \times T_{fr}$

프레임 전송 시간이 다 동일하다 생각했을 때,
내가 보내기 한 프레임 전에 누가 보내면 안되고
내가 보낼 때 누가 보내면 안됨

Example

- A pure ALOHA network transmits 200-bit frames on a shared channel of 200kbps. What is the requirement to make this frame collision-free? (neglect propagation delay)

200kbit : 1s = 200bit : 1ms

200kbps / 200 = 1ms

2ms 동안 나의 vulnerable time

Throughput

- To calculate throughput of pure ALOHA, we compute S as follows

p(k) = g^k*e^-g / k!
푸아송

p(0) - 아무것도 전송안할 확률
p(0) - e^-g
근데 우리는 두 프레임 타임을 보니까
e^-2g

s = g * p(0)

$$S = G \times e^{-2G}$$

p(k) - 프레임 타임동안 k개 프레임이 전송될 확률

g - number of transmission attempts per frame time

- S: ratio of frames that are successfully transmitted

- G: average number of frames generated in the system during T_{fr}

Throughput

- If G = 1, 프레임 타임 동안에 평균적으로 transmission attempt가 1번 일어남
- $S = 1 \times e^{-2} = 0.135$ 13.5% 프레임이 전송됨
- If 1000 frames are generated per second in the system, 135 frames can be successfully sent
- $e^{-1} = 0.367$
- $e^{-2} = 0.135$
- $e^{-0.5} = 0.606$

Example

200kbps / 200 = 1ms for 1 frame

- A pure ALOHA network transmits 200-bit frames on a shared channel of 200kbps. What is the number of frames that are successfully sent if the system (all stations together) produces

1ms에 1프레임 생성

- a. 1000 frames per second G = 1

2ms에 1프레임 생성

- b. 500 frames per second G = 1/2

4ms에 1프레임 생성

- c. 250 frames per second G = 1/4

Solution for b.

- T_{fr} = 200 bits / 200 kbps = 1ms

  한 프레임을 전송하는 시간

- If the system creates 500 frames per second, it is 0.5 frames per T_{fr} → G = 0.5

  1ms 동안 0.5 프레임이 생성됨

- $S = G \times e^{-2G} = 0.184$

- So 500 x 0.184 = 92 frames will survive

Slotted ALOHA

말을 시작하는 타임을 정해줌

Slotted ALOHA

- Throughput of slotted ALOHA

아까는 두 프레임 타임에 겹치면 안됐는데
이젠 한 프레임 타임만 신경쓰면 됨

slotted aloha하려면 동기화가 되어야함
아주 정밀하게 시간을 맞추지 않으면 안되는 것들과 적당히 맞추면 되는 게 있음
조금 앞뒤 당겨지면 시스템 마진으로 해결할 수 있음

- $S = G \times e^{-G}$

- Suppose we send 200-bit frames at 200kbps. How many frames can be successfully sent if

  – 1000 frames are generated per second
  – 500 frames are generated per second 500 * 0.5*e^-0.5 = 500 * 0.303
  – 250 frames are generated per second

CSMA (Carrier Sense Multiple Access)

- In ALOHA (pure/slotted), a node starts transmission even if other nodes are transmitting.
  - If the node is lucky, no other node transmits in the vulnerable time period
  - If the node is unlucky, a collision occurs
- If we use hardware that can sense the medium (channel), we can further reduce collision
- CSMA: “Listen before you talk”

CSMA

- When a node has a frame to transmit
- It first senses the channel
- If there is no signal on the channel (idle), go ahead and transmit
- If there is signal on the channel (busy), do not transmit

CSMA

- Human analogy
  - Suppose a group of people are chatting in a room
  - I want to talk
  - If no one is talking, I start to talk
  - If someone is talking, I wait

CSMA

- Can CSMA avoid collision completely?

no

CSMA

- Can CSMA avoid collision completely? No. 동시에 전송하면 충돌
  - If B transmits at time t_1, it reaches C at t_3
  - At t_2 ($t_1 < t_2 < t_3$), C sees channel as idle (no signal).
  - If C transmits at t_2 a collision occurs.

CSMA

- Vulnerable time: T_p (propagation delay)

CSMA

- If the channel is busy, what should the node do?

- Depending on the action, there are 3 types of CSMA.

- 1-persistent CSMA
- non-persistent CSMA
- p-persistent CSMA

1-persistent CSMA

- Continuously sense the channel while the channel is busy.
- As soon as the channel becomes idle, transmit

중돌이 많이남
평균적으로 전송하는 애가 1명이면 충돌이 덜하지만
노드가 많으면 충돌

- Since multiple nodes can wait for the channel to become idle, collision probability is high.

Non-persistent CSMA

- If the channel is busy, the node waits for some time and sense the channel again.
- If the channel is idle, transmit

- Collision probability is low, but the node can waste time waiting even after the channel becomes idle

p-persistent CSMA

- While busy, the node continuously sense the channel until it becomes idle.
- If the channel becomes idle
  - The node transmits with probability p
  - The node defers transmission with probability 1-p

Three types of CSMA

a. 1-persistent

b. Nonpersistent

c. p-persistent

CSMA/CD (Collision Detection)

- Collision detection
  - If a collision occurs, the sender detects the collision and acts upon it

- Used in Ethernet

이더넷에 사용
csma에 collision detect를 더함

CSMA/CD

- Collision situation
  - Node A transmits a frame, but C transmits before the frame reaches C

CSMA/CD

- How a node detects collision
  - <u>While transmitting my frame, another frame arrives</u> → detect collision

a에서 c로 보냈는데
c는 a 프레임이 오기 직전에 a로 보냄
c의 프레임이 a로 가고 있는 도중 a의 프레임이 도착 -> 내 프레임이 가고 있는 중간에 나에게 프레임이 옴 즉 충돌

CSMA/CD

- Successful collision detection
  - If a collision occurs, the node should be able to detect it
  - <u>When the collided frame arrives, I should still be transmitting my frame</u>
  - If I finish my transmission too early, I might not be able to detect collision

CSMA/CD

- Condition for collision detection failure
  - Node A transmits at t_1
  - Node C transmits at t_2 → collides with A's frame
  - The collided frame reaches node A at t_4
  - If node A finishes transmission before t_4, A fails to detect collision

CSMA/CD

최소 프레임 사이즈가 존재

- Then how long should node A transmit?
  - If A starts transmission at t_1, the first bit reaches C at t_1+t_p
  - If C transmits before t_1+t_p, it is a collision
    - C cannot transmit after t_1+t_p because of carrier sensing.

CSMA/CD

- Then how long should node A transmit?
  - Suppose C transmits just before t_1+t_p.
  - The collided frame reaches node A at t_1+2t_p
  - At t_1+2t_p, node A should still be transmitting
  - Thus, $t_{fr} \geq 2t_p$

프레임 전송시간 2tp 동안은 전송해야함 거의 다 오기전에 c에서 전송함

CSMA/CD

- A network using CSMA/CD has a bandwidth of 10Mbps. If the maximum propagation time is 25.6µs, what is the minimum size of the frame?

tp = 25.6us

tfr = framesize / 10mbps
>= 2 * 25.6 * 10^-6
>= 512bits

CSMA/CD

- When collision is detected
  - Binary exponential backoff similar to ALOHA

기다렸다 다시전송
서

CSMA/CA (Collision Avoidance)

- Used in systems that cannot detect collision

- Used in Wi-Fi

자기 간섭을 막기 위해
전송 모드 / 수신 모드 두개를 만들어 하나만 하게

- The Wi-Fi interface is half-duplex, which means it cannot transmit and receive at the same time → cannot detect collision as CSMA/CD

CSMA/CA

- Do random backoff before a collision occurs
- Wait for ACK
- If ACK does not come, regard it as collision → increase backoff exponentially

프레임 전송이 실패하면 충돌로 봄 -> binary expansion backoff

CSMA/CA

- IFS (inter-frame spacing)
  - Even if channel is sensed idle, another node might just started transmission. IFS is inserted in order to avoid collision.

신호가 올 수 있으니 고정된 IFS 만큼 쉼

계속 sensing

idle 상태가 됨

아직도 idle이면 카운드 다운을 함
중간에 busy로 바뀌면 카운트를 멈춤
다시 idle이 되면 IFS만큼 쉬고 멈춘 곳에서 다시 카운트

CSMA/CA

- Contention window
  - A range from where a node selects a random number (e.g. 0-31)
  - A node selects a number from the contention window
  - For each time slot, if the channel is idle, count down the number (e.g. time slot = 20us)
  - If the counter becomes zero, start transmitting.

CSMA/CA

- Contention window
  - Suppose a node selects 15 as the random number (time slot = 20us)
  - If the channel is idle for 15x20=300us, the node transmits
  - If the channel becomes busy while counting, the node pauses until the channel becomes idle again
  - If the channel becomes idle, the node waits for IFS and then restart counting

CSMA/CA

- Binary exponential backoff
  - If a collision occurs, the contention window size doubles (e.g. 0-31 → 0-63)
  - If a frame is successfully transmitted, the contention window reduces to initial range

Example

- Assumptions
  - time slot = 20us
  - IFS = 50us inter frame spacing
  - initial contention window: [0-31]
  - transmission time of a frame = 100us
  - propagation time = 0 (neglected)
  - no ACK

- Nodes A, B, C are in the same wireless LAN
  - If node A transmits, node B and C both can see the signal

Example

- 0us: A, B, and C has frames to send, they all wait for IFS (50us).

- 50us: A, B, and C selects a random number. A chooses 10, B chooses 15, and C chooses 20. They start counting down.

- 250us: A's counter becomes 0. A starts transmission. B and C see A's signal and <u>pause</u> count down.

Example

- 350us: A's transmission finishes. B and C sense channel as idle. B and C wait for IFS.
- 400us: B and C resume counting. B's counter = 5, C's counter = 10
- 500us: B's counter becomes 0. B starts transmission. C pauses counting.

Example

- 600us: B’s transmission finishes. C waits for IFS.
- 650us: C resumes count down. C’s counter = 5
- 750us: C starts transmission
- 850us: C finishes transmission

Controlled Access

Controlled Access

- Does not depend on random number to avoid collision

- Nodes agree on a rule to take turns and transmit

- Reservation
- Polling
- Token Passing

Reservation

- Before transmission, a node “reserves” a time slot.
- There is a “reservation slot” before data slots, where nodes reserve their transmission.
- If there are N nodes, the reservation slot has N mini slots, where a node can send signal to announce that it will be transmitting.

Reservation

- In the first period, nodes 1, 3, and 4 reserve channel
- In the second period, only node 1 reserves channel

1 - 보낼게 있다
0 - 보낼게 없다

1,3,4이 보내고 싶으니 얘네를 다 보낸 다음
다음 reservation을 함

Polling

- One of the nodes become the “primary node”
- Other nodes become secondary nodes
- The primary node controls data transmissions
  - It decides who sends when
- The data should always go through the primary node
  - Two secondary nodes cannot communicate

Polling

- Select (SEL): a primary node sends data to a secondary node
- Poll: a secondary node sends data to a primary node

primary가 주관해서 통신

Token Passing

- The goal of medium access control is to have only a single node transmit at a time
- This mechanism uses a unique “token” to determine who should transmit.
  - There is only one token in the system
- A node can transmit only if it possesses the token. After data transmission, the node passes the token to the next node in the predefined sequence.

Token Passing

- Various types of token passing sequence (topology)

a. Physical ring

b. Dual ring

c. Bus ring

d. Star ring

Token Passing

- Physical ring
  - Simple, token does not need to have address of next node
  - If a link breaks, the system is stalled
- Dual ring
  - When a link breaks, the system can continue using the second link
- Bus
  - The token should indicate the next node
  - Tolerant to link break
- Star
  - The token is always passed through the hub
  - Easy to add/remove node from the system

Channelization

Channelization

- A mechanism to divide the medium into multiple channels using time, frequency, and code division techniques
- Frequency-Division Multiple Access (FDMA)
- Time-Division Multiple Access (TDMA)
- Code-Division Multiple Access (CDMA)

FDMA

- Divide the bandwidth into multiple frequency sub-channels, and allocate sub-channels to different users

FDMA

- Guard band needed between sub-channels

- Suppose we divide a 20MHz channel into 1MHz sub-channels. If we need 100kHz of guard band, how many sub-channels can we make?

1mhz 서브 채널 사이간에 100khz 간격이 있어야 함

20mhz / 1mhz 하면 20. 20개의 서브 채널을 만들 수 있을 거라 생각하지만
guard band 때문에 그렇게 못함

TDMA

- Divide the channel into time slots and allocate time slots to different users

TDMA

- TDMA is more flexible than FDMA, since we can change the allocation rate dynamically.

- However, TDMA requires time synchronization among nodes, which is a difficult problem.

신호가 간섭돼도
원래 어떻게 생겼는지 구분할 수 있게 인코딩

CDMA

- In CDMA, multiple nodes transmits at the same time, using the same frequency

- The signals will be mixed, but they can be separated using code division technique

CDMA

- Suppose 4 nodes transmit data

CDMA

- Codes are allocated to nodes. Suppose c_i is the code allocated to node i. The codes have the following characteristics:
  - $c_i \times c_j = 0$, if $i \neq j$
  - $c_i \times c_i = 4$ (number of nodes)

CDMA

- When a node transmits data, it multiplies its data and its code. (d_i x c_i)
- If all four nodes transmit, the mixed data will be:

$$d_1 \cdot c_1 + d_2 \cdot c_2 + d_3 \cdot c_3 + d_4 \cdot c_4$$

CDMA

- The receiver multiplies <u>the sender's code</u> to the received data
- If a node wants to receive node 1's data, it multiplies c_1 and the received data
- Then, divide the result by 4 (number of nodes)

CDMA chips

- The following code set can be used in CDMA

C_1	C_2	C_3	C_4
[+1 +1 +1 +1]	[+1 -1 +1 -1]	[+1 +1 -1 - 1]	[+1 -1 -1 +1]

각자 고유한 코드 벡터를 배정받음

코드들의 조건

- The code set has the required characteristics
  - If multiplied by itself (inner product), the result is 4.
  - If multiplied by other code, the result is 0.

Transmission using CDMA

- To transmit data bits using CDMA, we make the following rules

Transmission using CDMA

- At an instance, node 1 sends bit 0, node 2 sends bit 0, node 4 sends bit 1, and node 3 does not transmit.

Transmission using CDMA

- The transmitted signals

Transmission using CDMA

- Suppose node 3 wants to receive node 2's data
- Then, node 3 multiplies (inner product) received data with node 2's code

Transmission using CDMA

- Finally, add all numbers and divide by 4.
- The result is -1, which means bit 0.

Generating CDMA chips

- To generate CDMA chips, we can use Walsh table.
- To make Walsh table, we create a 2N x 2N matrix from N x N matrix, starting from a 1x1 matrix.
- Invert only the last (bottom-right) component in the generated matrix.

2명일 때

user1 user2

$$W_1 = \begin{bmatrix} +1 \end{bmatrix} \qquad W_{2N} = \begin{bmatrix} W_N & W_N \\ W_N & \overline{W_N} \end{bmatrix}$$

마지막 부호를 반대

Generating CDMA chips

- Create 4x4 matrix from 2x2 matrix

4명일 때

2x2 행렬을 4번 쓰고
마지막은 뒤집어서

$$W_1 = \begin{bmatrix} +1 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}$$

5 - 8 개가 있으면
4x4 행렬을 4개 한 8x8 행렬

- The <u>columns</u> of the matrix become the CDMA codes.

End of Chapter