Lecture 6

Binary Exponential Backoff Algorithm

How should nodes that collide schedule their reattempts?

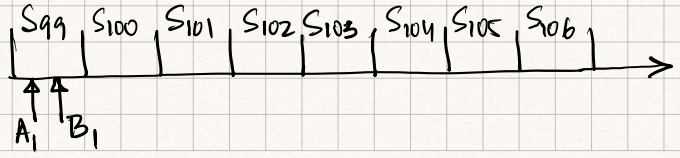
- If they simply use CSMA, then they will likely collide again
- They could independently choose a random delay to wait before reattempting
- The more the collisions, the random delay should be chosen over a larger range to reduce the chance of collision

n : number of reattempts for the pkt being transmitted
n = 0 when a new pkt is transmitted (initial value)

We consider a slotted system for simplicity

Algorithm: On detecting collision, the number of slots to delay to reattempt, r is chosen randomly in the range $(0, 2^k)$ where $k = \min(n, 10)$

Example: We consider 2 nodes (A & B) & a slotted system



- A_1 and B_1 refers to pkt arrivals in nodes A & B respectively. $n=0$ for both these pkts
- Both nodes transmit in S_{100} and they collide

  Node A chooses randomly from (0,1)

  Say it choose 0 n is increment to 1

  Node B chooses randomly from (0,1)

  Say it choose 0 n is incremented to 1

- Both nodes transmit in S_{101} and they collide

  Node A chooses randomly from (0,2)

  Say it chooses 1 n is now 2

  Node B chooses randomly from (0,2)

  Say it chooses 1 n is now 2

- Both nodes transmit in S_{103} and they collide

  Node A chooses randomly from (0,4)

  Say it chooses 0 n is now 3

  Node B chooses randomly from (0,4)

  Say it chooses 2 n is now 3

- Node A successfully transmits in S_{104}

  Node B successfully transmits in S_{106}

Notes:

- The range over which r (the number slots to delay) is randomly picked expands exponentially with collisions

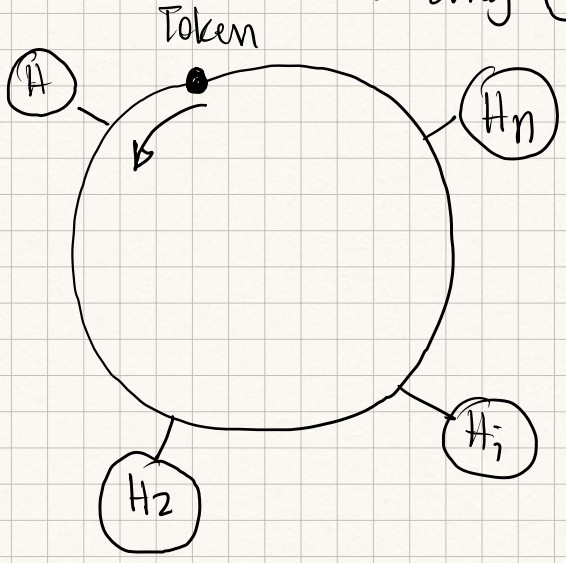
- n is the current number of reattempts and is incremented after determining k
- n is reset to 0 after successful transmission
- n is capped at 10 through the function
  $k = \min(n, 10)$ 2^{10} is 1024 which is large
- In the real algorithm, the time delay until reattempt is k * (transmission time of 512 bytes) or is it bits?

Project 1: We will see that this algorithm works really well.

Taking Turns Medium Access Protocol

- Nodes share a broadcast channel and they take turns in accessing the channels and so only 1 node is transmitting at a time
- Many ways to implement this. The most well known method is called the Token Ring Protocol
  - → Known as the FDDI (Fiber Distributed Data Interface)
  - → Nodes are connected in a ring network

→ A token (a special bit pattern) goes around the ring (in one direction)



When a node has a pkt. to transmit

→ Wait for the token to arrive

→ Hold the token

→ Transmit pkt(s)

→ Releases token

Service Policies:

Limited: only 1 pkt is transmitted per token

Gated: All pkts in the queue when the token arrives are transmitted. New pkts that arrive during service are buffered

Exhaustive: The token is released when the queue becomes empty

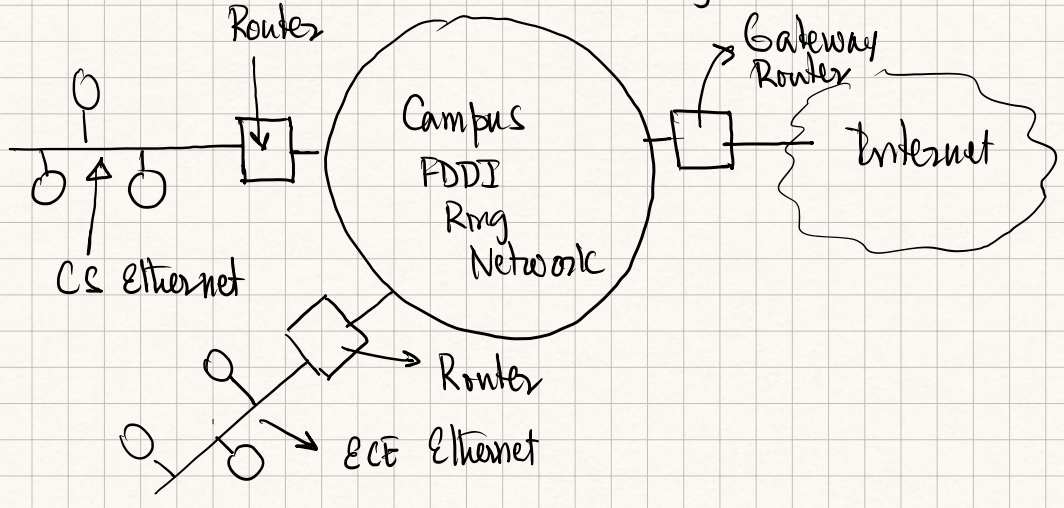
Token Rotation Time (TRT): Time between two consecutive arrivals of the token at a node

Minimum TRT : Propagation delay + bit delays at the repeaters

Maximum TRT : Depends on the service policy

For the limited service policy
Min TRT + a transmission time at each node

Example: UC Davis campus backbone was a FDDI token ring network



Slotted Ring Network

- Instead of a token, slots go around the ring
- Slots are marked Free or Busy
- A node wishing to transmit waits for a free slot, marks the slot as Busy, and writes data into the data part of the slot
- The destination node reads the data and marks the slot as Free (Destination Removal)