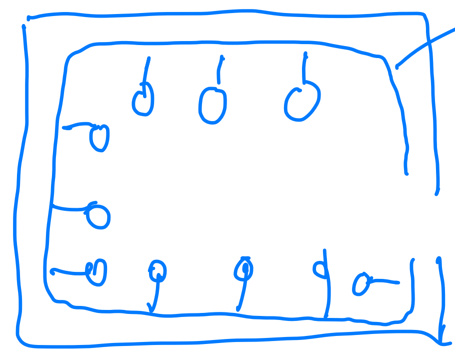
LANS: Ethernet



channel is the wire.

Outline

1. Contention
   (a) Unlimited
   (b) Limited
2. MAC Protocols
   (a) CSMA-CD, In Wireless we use a diff paradigm: CA
   (b) Aloha
   (c) Persistence
3. Limited Contention
4. Reservation
   (a) Bit-Map
   (b) Binary Countdown
   (c) Splitting Algoritgms

Contention

MAC and Contention

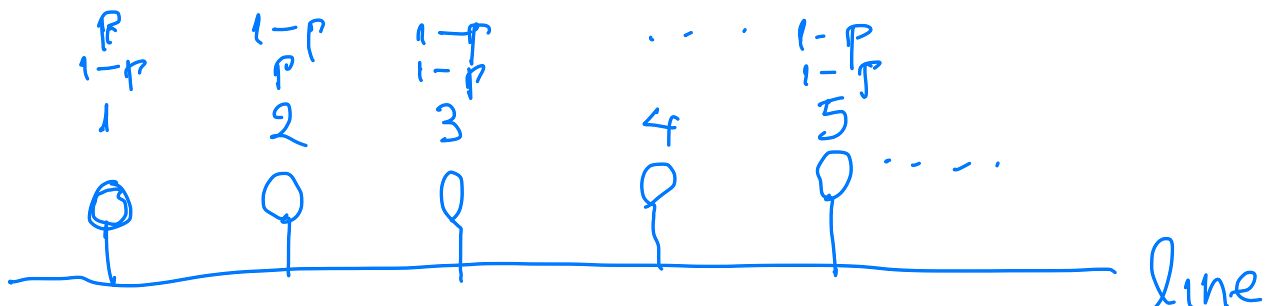
- MAC (Multiple Access Control) requires contention resolution,
  - i.e., prioritize the use of the medium so as to ensure good performance.
- There are two types of contention.
  - Uncoordinated
  - Coordinated
- In the former, no scheduling is required, while in the latter a coordination algorithm must be performed by all users.

Uncoordinated Contention (1/3)

- Suppose n nodes are contending for a channel.
- A node transmits during a contention with probability p.
- Probability of a successful transmission is equal to the probability that exactly one node transmits:

$$\Pr[\text{Success}] = np(1-p)^{n-1}.$$

- Pr[Success] is maximized (as a function of p) when $p = 1/n$.
- Why?



$p(1-p)^{n-1}$
$+ p(1-p)^{n-1}$
$\vdots$

p = the prob. that a given host will talk

$1-p$ prob will not talk

Uncoordinated Contention (2/3)

- Take the derivative of the function

$$f(p) := np(1-p)^{n-1}$$

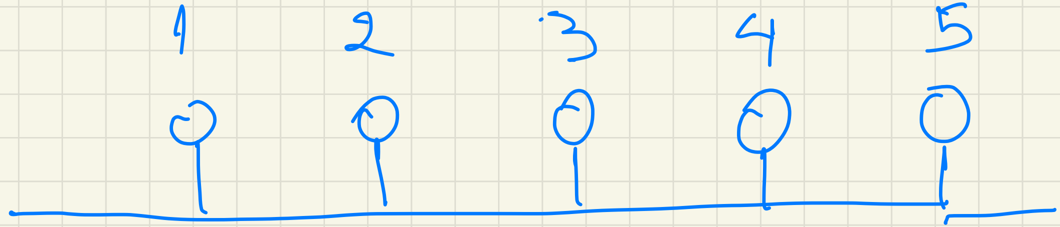
with respect to p.

$$\begin{aligned}\frac{d}{dp}f(p) &= n(1-p)^{n-1} + np(n-1)(-1)(1-p)^{n-2} \\ &= n(1-p)^{n-2}\left((1-p)-(n-1)p\right)\end{aligned}$$

- If you set $\frac{d}{dp}f(p) = 0$ we see that

$$1 - p - (n-1)p = 0.$$

- Hence, $p = 1/n$.



$$\Pr[\text{Success}] = 5 \frac{1}{5}(1-p)^4$$

$$= 5 \frac{1}{5}\left(1 - \frac{1}{5}\right)^4$$

167

$$\Pr[\text{Success}] = 167 \frac{1}{167}\left(1 - \frac{1}{167}\right)^{166}$$

Uncoordinated Contention (3/3)

- Hence, $f(p)$ obtains its maximum value when $p = 1/n$.
- In which case

$$
\begin{aligned}
P_{\max} &:= \Pr[\text{Success}] \\
&= n\frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1} \\
&= \left(1-\frac{1}{n}\right)^{n-1} \\
&\approx 1/e, \text{ as } n \to \infty.
\end{aligned}
$$

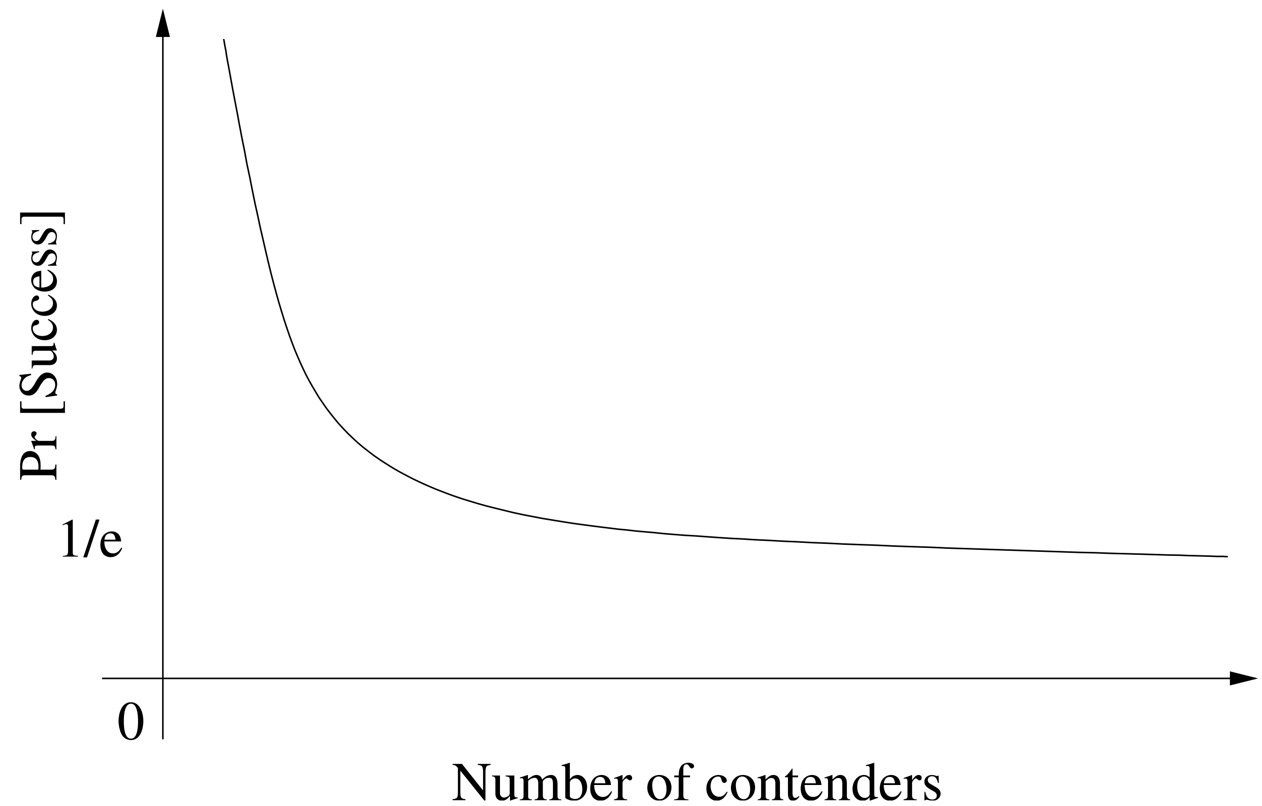
e=2.781

- The average number of contentions until success occurs is measured as the mean of a geometric distribution and is equal to

$$\frac{1}{1/e} = e.$$

Limits of Contention

The only way to improve the performance of contention is by limiting the number of users.



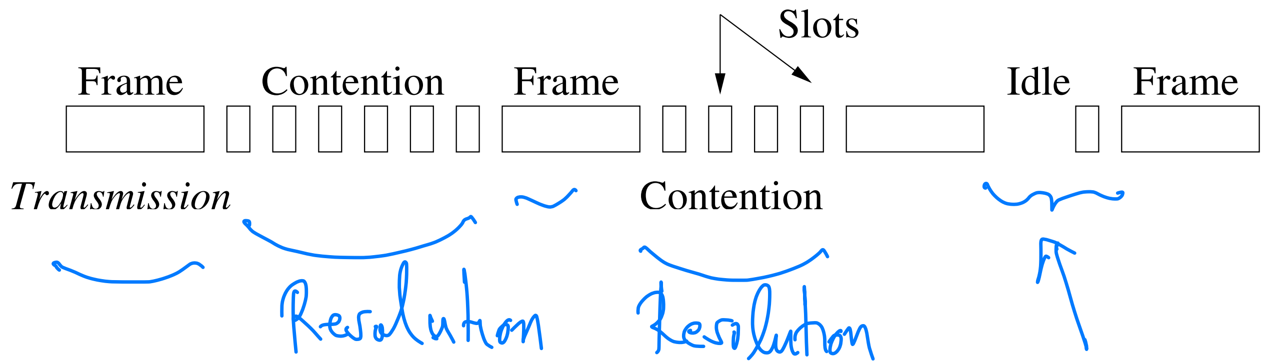
Limited Contention

Access Method: Limited Contention Algorithms

- The problem with the previous approach is that it assumes that each of the contenders wants to talk all the time.
- In practice this is not the case.

Access Method: Limited Contention Algorithms

- After a collision if some way could be found of resolving the collision quickly it may be possible to increase the throughput
- Algorithms for resolving collisions after they occur are sometimes called limited contention algorithms
- Periods of normal operation (with no collisions) are punctuated by periods where a different algorithm is used to resolve a collision



Limited Contention Algorithms

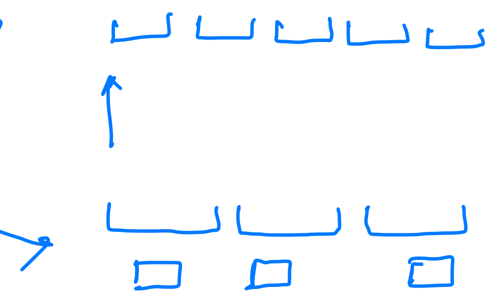
- **Limited Contention** protocols combine best properties of
  1. contention (contention at low load provides low delay) and
  2. collision-free (good channel efficiency at high loads).

Limited Contention are algorithms that limit the number of users contending in a given round.

Types of MAC Protocols

Two basic classes of MAC (Multiple Access Control) protocols.

1. Random Access
   (a) Slotted Aloha
   (b) Unslotted (or Pure) Aloha
   (c) CSMA
   (d) CSMA-CD



2. Scheduling
   (a) Reservation
   (b) Polling
   (c) Token-Passing

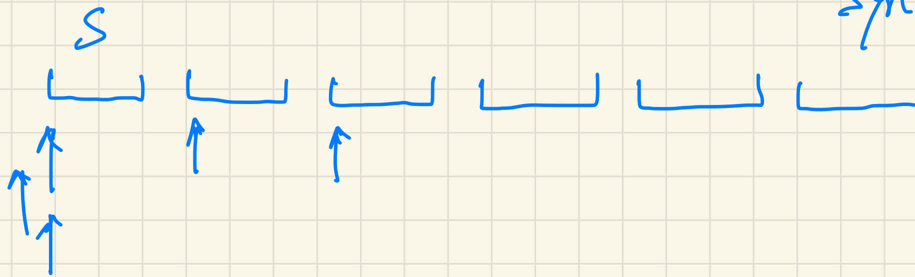
Aloha Ethernet: In two Varieties

- Devised in 60's by Abramson at U of Hawaii to interconnect terminals located at campuses of different islands. Created for a wireless net.
- A radio transmitter was attached to the terminals; a message is transmitted as soon as it becomes available.
- If packets collide they are retransmitted.
- Adapted by Boggs and Metcalf at Xerox in 1973 to get Ethernet
  $\underset{l}{\lambda}, \underset{m}{\mu}, \underset{r}{\rho}, \ldots$
- Aloha Ethernet come in two varieties: Slotted and Uslotted.
  - **Slotted:** time divided into slots that handle fixed length packets with transmissions only at slot boundary
  - **Unslotted:** no restrictions on packet size or time of transmission

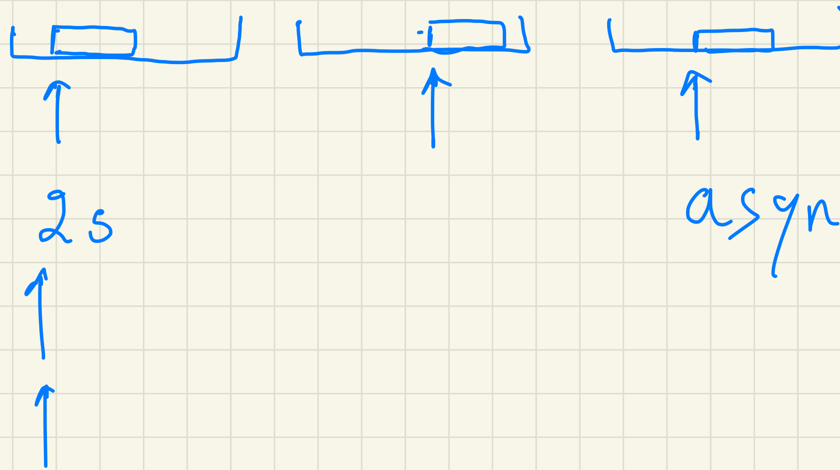
synchronous

Slotted.

multiple user



Unslotted



asynchronous

Backoff Protocols

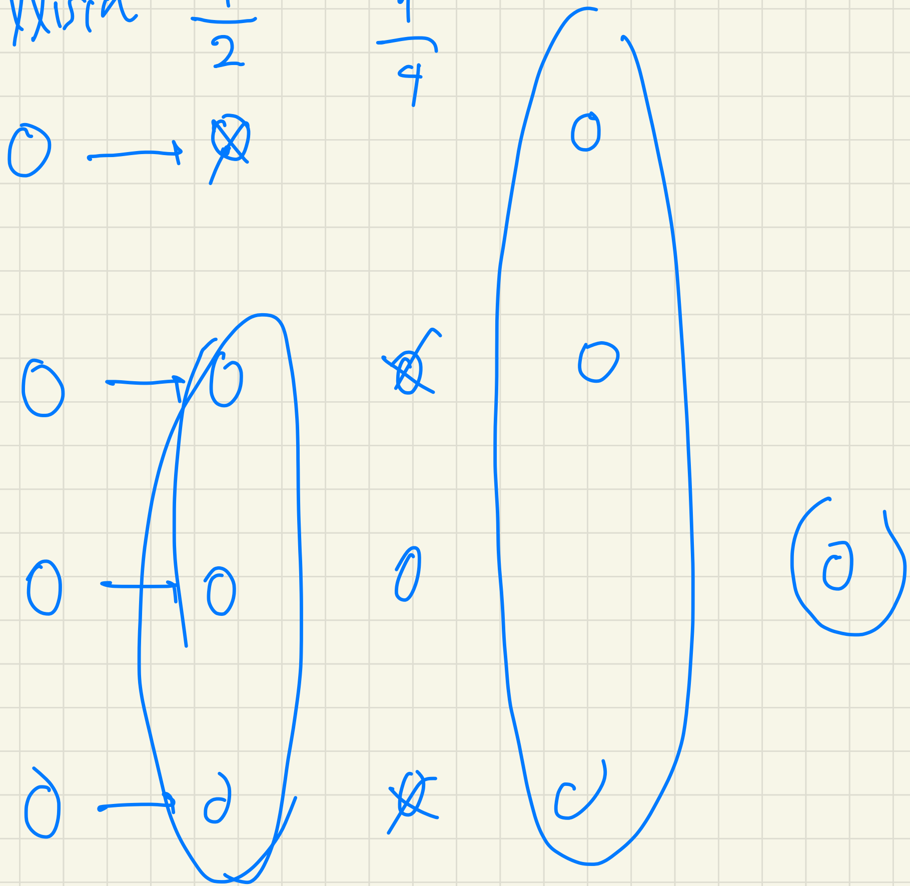
- Based on a suitable probability distribution and a queue.
- There is a queue of stations (nodes) waiting to transmit.
- Each station (node) keeps track of the number of attempts to transmit
- A function $p(x)$ (known to all hosts) is defined in advance:
  - $p(x)$ is the probability you transmit in the x-th attempt
  - same function is used by all nodes
  - $p(x)$ decreasing in x (x = # of attempts)

$p(1)$, $p(2)$, $p(3)$, ...

Backoff Algorithms

- The main backoff algorithm is as follows.
  - Station i has a variable bck_i:
    1. Initialize $bck_i \leftarrow 0$;
    2. Station attempts transmission if queue is not empty with probability $p(bck_i)$;
       (a) if it fails to transmit due to collision it increments the variable bck_i (e.g., $bck_i \leftarrow bck_i + 1$);
       (b) Else it assigns 0 to the variable bck_i;
    3. If queue is empty and no transmission is made the variable bck_i remains unchanged.
- Major concern:
  - What functions $p(x)$ should we use and are what criteria?
  - How stable is traffic under various functions $p(x)$?

Collision $\frac{1}{2}$ $\frac{1}{4}$



Aggressiveness: Binary Exponential Backoff

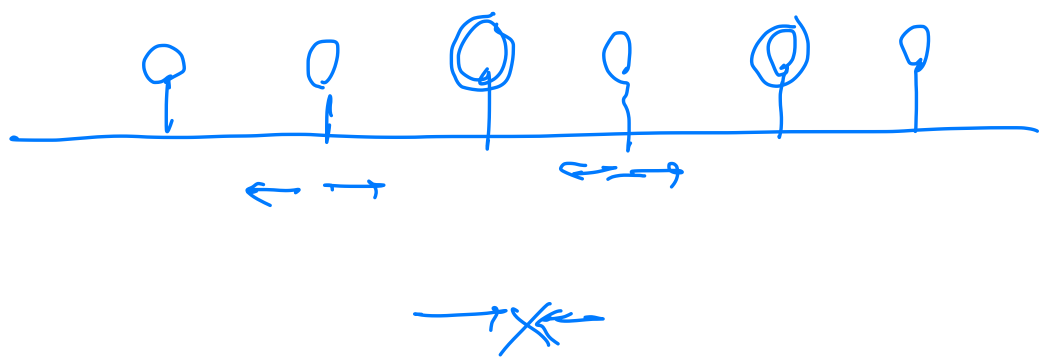
- The function $p(x)$ is important: x = # of round you're in
  - it tells you how aggressive you should be in your x-th attempt to transmit.
- Backlogged packet attempts with probability $p(x)$ where x is the # of unsuccessful attempts.
  1. Exponential backoff: $p(x) = 2^{-x}$. IEEE
  2. Polynomial backoff (k is constant): $p(x) = \frac{1}{(x+1)^k}$.
     - $k = 0$: constant backoff
     - $k = 1$: linear backoff
     - $k = 2$: quadratic backoff
- Polynomial backoff has been shown to be stable for $k > 1$.
- Proving these claims requires sophisticated research!

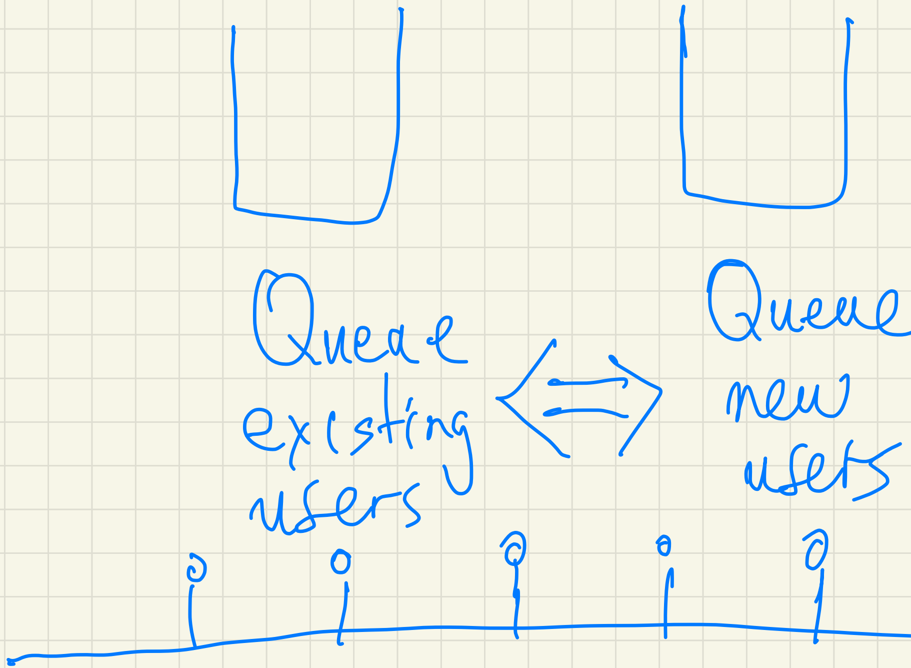
not to discuss

Carrier Sense Multiple Access/Collision Detection

There are two important aspects to Ethernet:

- **CSMA** you need a NC (network card)
  - whereby the carrier is sensed for the presence of signals, and
- **CD**
  - whereby the hosts attempt detection of collisions.
  - This is used to cut down on unnecessary collisions!




Queue existing users
Queue new users

Round:

$$\left\{ \begin{array}{l} 1.\ \text{choose value } x \\ 2.\ \text{compute } 2^{-x} \\ 3.\ \text{w.p. } \frac{1}{2^x} \text{ you talk} \\ 4.\ \text{listen} \end{array} \right\}$$

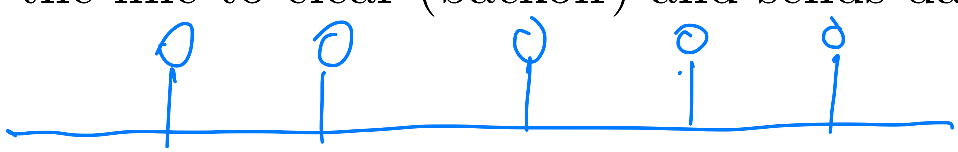
Round: [(Receive inform.) (Computes some) (Send infor)]

Ethernet CSMA/CD

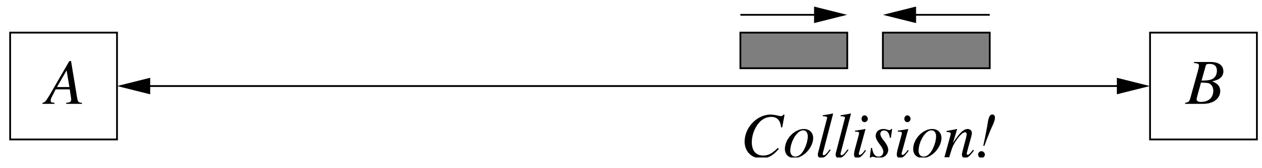
- **CSMA (Carrier Sense Multiple Access)**
  1. A station wishing to transmit must first listen for existing traffic on the line. tap line with NC card.
  2. If no voltage detected, the line is considered to be idle and a transmission is initiated.

     noise packet

- **CD (Collision Detection)**
  1. If no voltage detected, a transmission is initiated. During transmission, station checks line for extremely high voltages that indicate collisions.
  2. If collision detected station waits a predetermined amount of time for the line to clear (backoff) and sends data again.



What Do I Do after a Collision? Exponential Backoff!



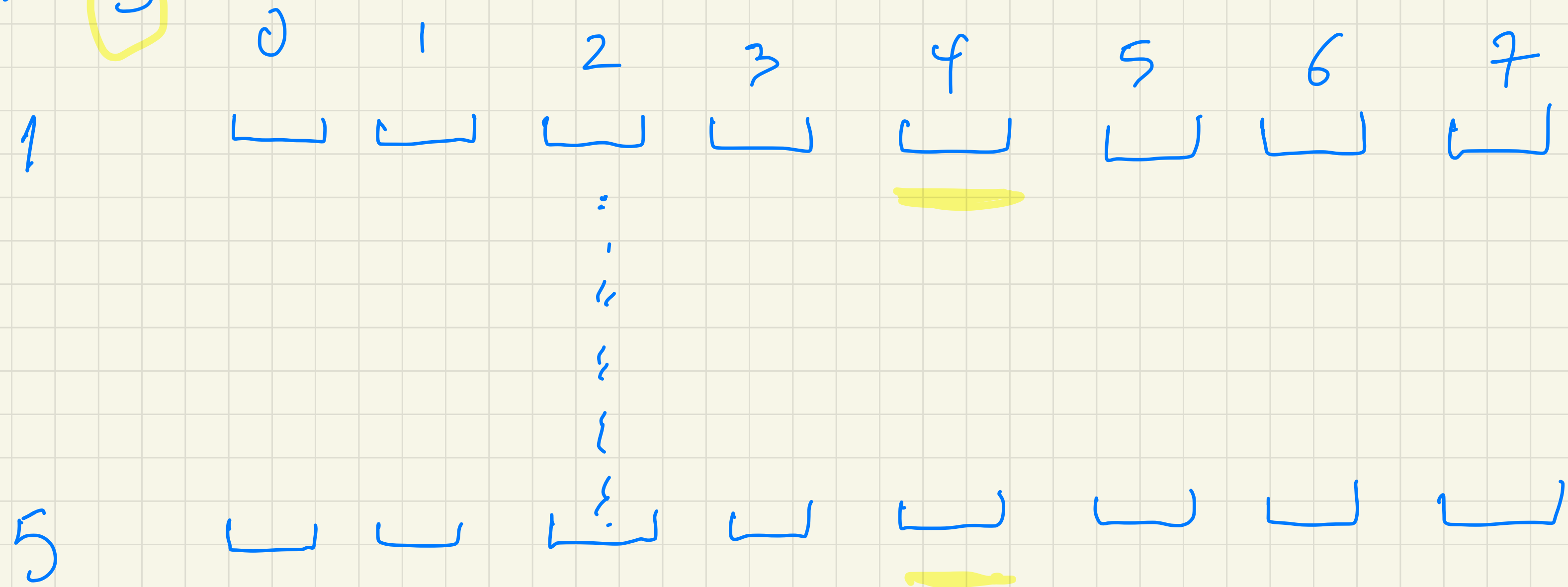
1. After 1st collision, each station "waits at random" either 0 or 1 time slots and tries again.

2. If they pick same random value they collide again. After 2nd collision, each station "waits at random" either 0 or 1 or 2 or 3 time slots and tries again.

3. If they pick same random value they collide again. After 3rd collision, each station "waits at random" either 0 or 1 or 2 or 3 or 4 or 5 or 6 or 7 time slots and tries again.

4. And so on...

$x = 3$



$x = t$

How Long Does it Take before a Successful Transmission?

- Remember, **collisions** are random events!
- It makes no sense to talk about deterministic time!
- It is more precise to talk about expected time!
- The expected time depends on the backoff protocol being used!

Expected Time

- Suppose that two synchronous ethernet stations A, B are at contention period t:
- This means, they have failed in previous contention periods $0, 1, \ldots, t-1$, and are now in contention period t.
- They each pick a random number in the interval $0..2^t - 1$.
  - A picks random value a in $0..2^t - 1$. 0 ... 7
  - B picks random value b in $0..2^t - 1$ 0 ... 7
- The expected number of steps to reach contention period t is cumulative,
  - i.e., you must add the "waiting times", for all the trials before t.

How Persistent Should I Be?

- Successful transmission (i.e., no collision) can occur only when the two stations select different values at random, i.e., $a_t \neq b_t$.

$2^t = m$

- Hence, if that happens then

$$\Pr[\text{no collision}] = \frac{2^t(2^t-1)}{(2^t)^2} = 1 - \frac{1}{2^t}$$

$\frac{m(m-1)}{2}$ $\binom{m}{2}$

- The waiting time for a success in t-th attempt will be

$$\min\{a_t, b_t\},$$

where a_t, b_t are the values chosen above!

- **NB:** The bigger the t is the higher (closer to 1) is the probability that you will succeed next time $(t+1)$.
- **NB:** Don't forget that there are many stations (not just two) contending for the medium!

100 people try to talk

$\rightarrow$ 50 remain

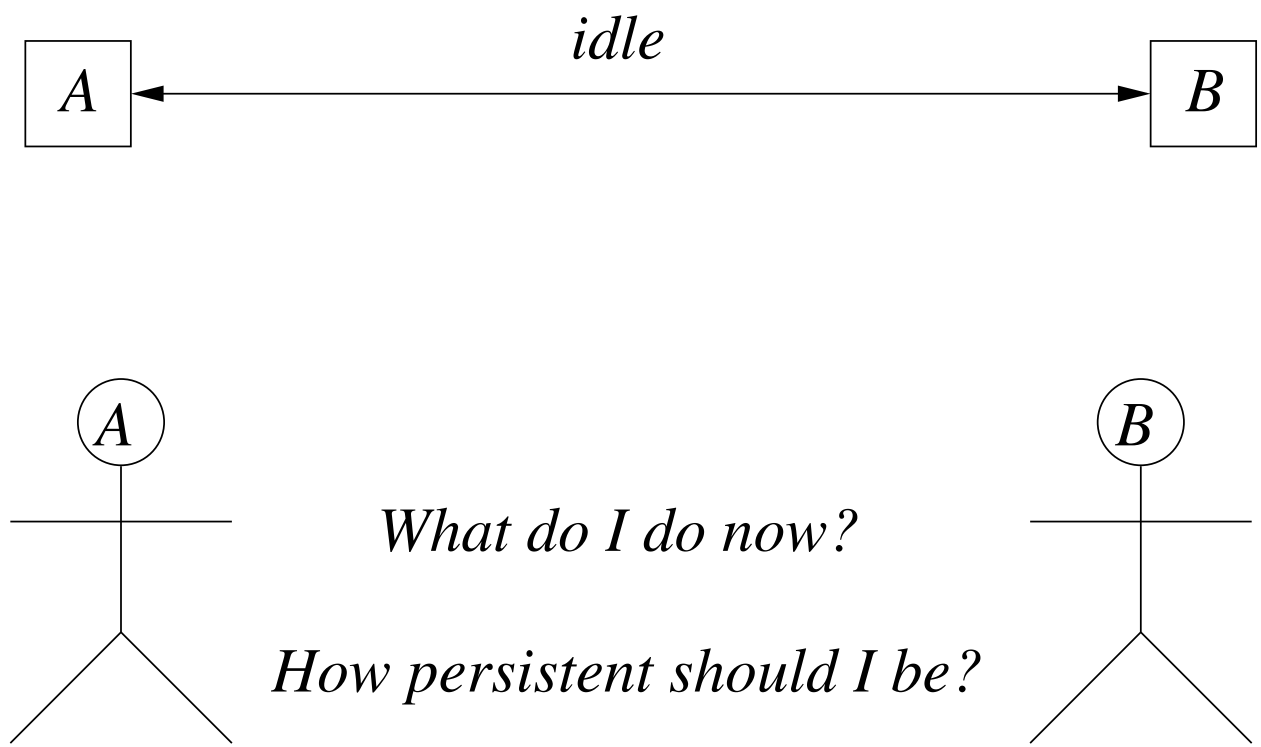
$\rightarrow$ 25 remain

$\rightarrow$ 12 "

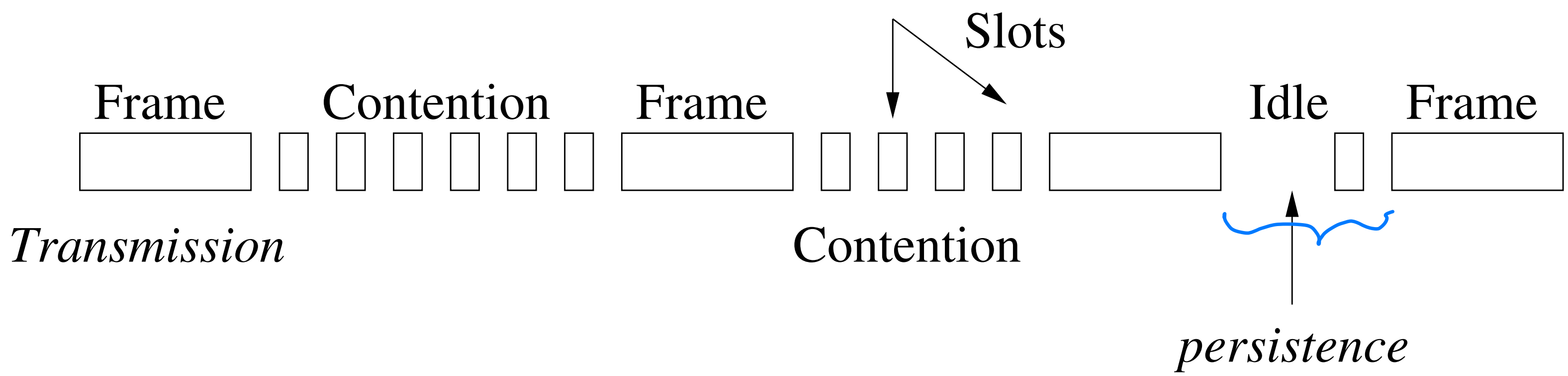
$\vdots$

$$2^t \geq 100 \qquad t = 7$$

How Persistent Should I Be?

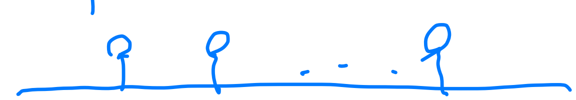


Variants of CSMA: Smart Transmitters



- When a station has data to send and line is busy it waits for the line to go idle.
- How persistent is a transmitter in capturing an idle channel?

Proactive Mechanism to take advantage of the idle line.



Avoid transmissions that are certain to cause collisions!

- In most LAN systems, a node can sense if another node is using the channel after some delay for the signal to reach it
- CSMA senses the medium for the presence of carrier signal.
- In this case, a node can wait until the channel becomes idle before transmitting
- Schemes that take advantage of this are called Carrier Sense Multiple Access (CSMA)

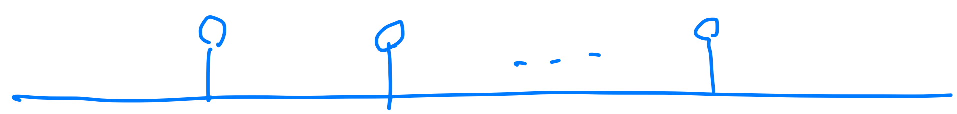
1. 1-Persistent CSMA
2. Non-persistent CSMA
3. p-persistent CSMA

$0 \leq p \leq 1$

with carefully chosen prob. p will have the best performance

Slotted CSMA: 1-Persistence

- Remember, there are many stations competing!
- When node has data to send.
  1. First listens to channel to see if anybody else is transmitting.
  2. If channel busy the node waits until channel becomes idle.
  3. When node detects idle channel it transmits frame.
  4. If collision occurs node waits a random amount of time and starts over again.
- 1-persistent means that a transmitter with a packet to send transmits with probability 1 whenever a busy line goes idle.



Slotted CSMA: Non-Persistence

- **Nonpersistent:** Do not persist! Act like a collision and wait a random number of slots before retrying.
- When node has data to send.
  1. First listens to channel to see if anybody else is transmitting.
  2. When node detects idle channel it transmits frame.
  3. If channel is busy the node does not continually listen in order to seize it for transmission when idle. Instead it waits a random period of time and then repeats the algorithm.
- Non-Persistenxe behaves better than 1-persistence.

Slotted CSMA: p-Persistence

- **p-Persistent:** send with probability p in next available slot
- When node has data to send.
  1. First listens to channel to see if anybody else is transmitting.
  2. If channel busy wait until next slot and try again.
  3. If channel idle then **Repeat** until either sucess or another node begins transmitting.
     (a) it transmits with probability p.
     (b) it defers until next slot with probability $1 - p$.
  4. if another node begins transmitting node acts as if there is collision: waits a random amount of time and starts again.

$p = \frac{1}{3}$. 3 users

(Example) Slotted CSMA: p-Persistence

- A p-persistent protocol transmits with probability p, where $0 \leq p \leq 1$, after a line becomes idle.
- E.g., Assume $p = 1/4$ and 100 stations are waiting for the busy line to become idle.
- Each of these stations transmits with probability $1/4$.
- Hence only ≈ 25 stations will attempt to transmit and the remaining ≈ 75 stations defer until next time slot.

CSMA Performance

p = 0.01

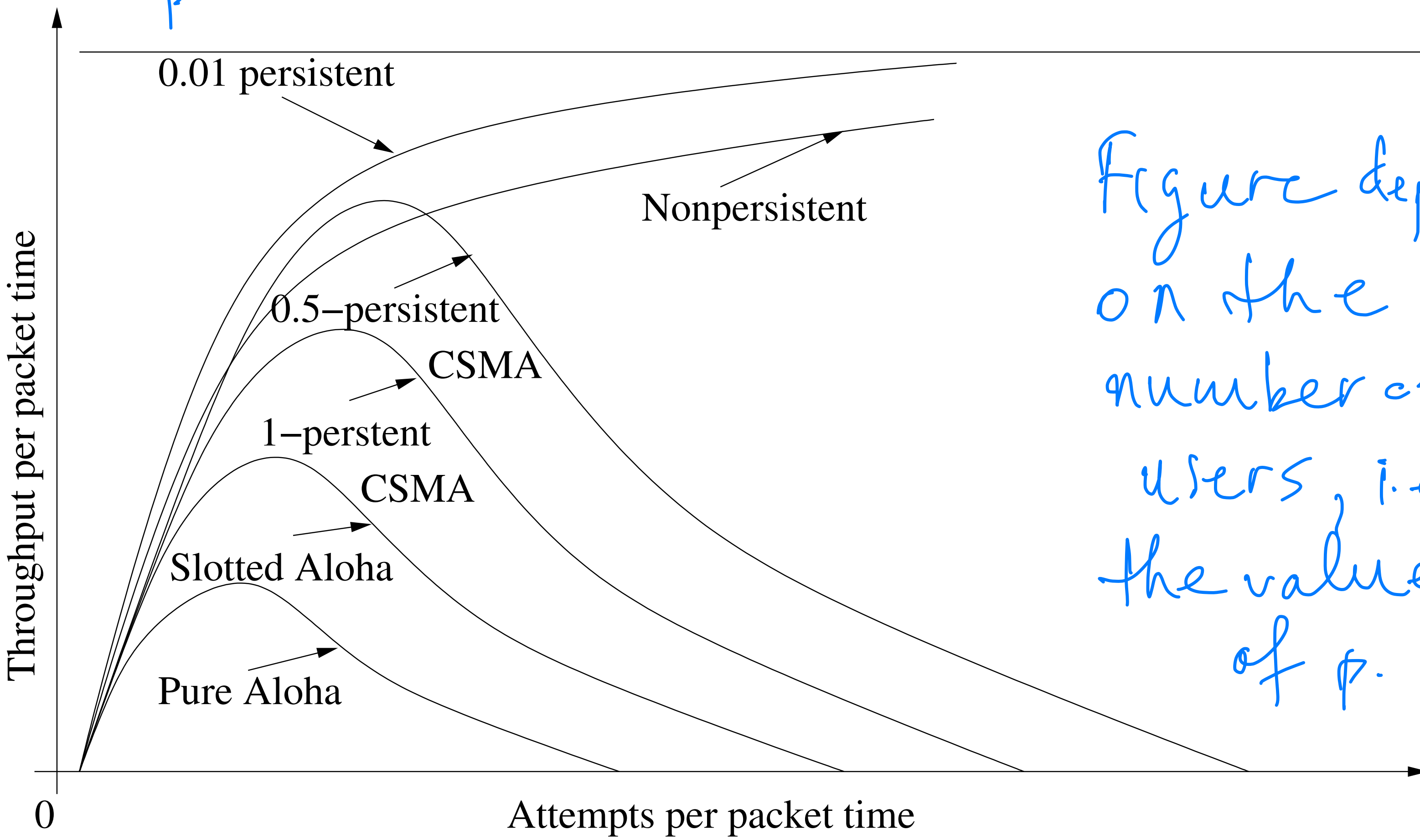


Figure depends on the number of users, i.e. the value of p.

Ethernet CSMA-CD

- CSMA improves over Aloha because it senses the carrier.
- Still, however, collisions involve entire packets!
- An obvious improvement is to abort transmission when collision is detected!

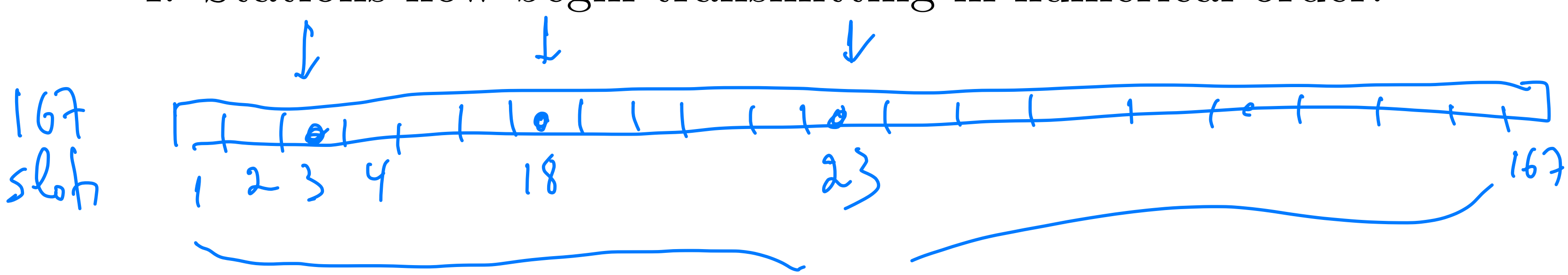
1. Station senses channel
2.   **if** channel idle **then** transmit
3.   **else** use a CSMA strategy (p-, non-persistent)
4.   **if** collision detected during transmision
5.   **then** transmit jamming signal **and** abort transmission
6.   schedule future transmission with backoff.

Reservation

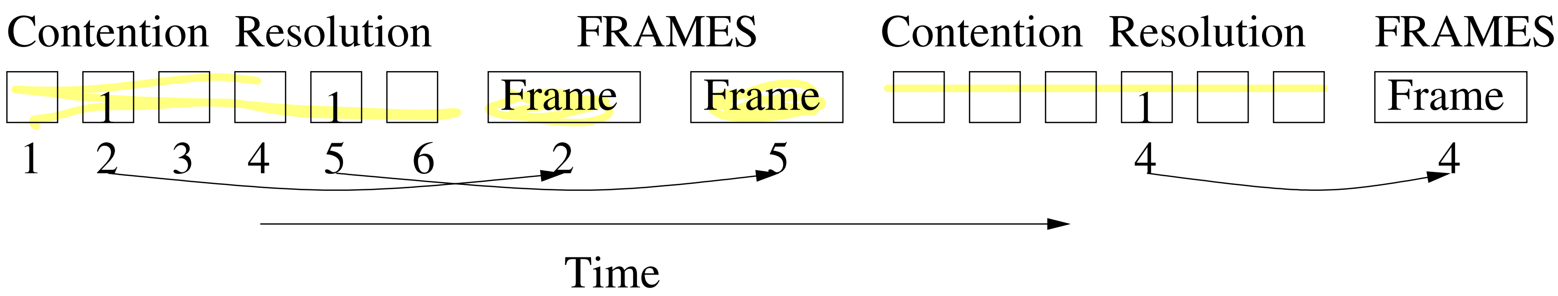
You reserve the channel, and you talk when your turn comes up.

Collision-Free Reservation Protocols: Basic Bit-Map

- Assume there are exactly N stations labeled 0 to $N-1$. Let the propagation delay be negligible.
- **Basic Bit-Map Method:**
  1. Each contention period is exactly N slots.
  2. If station i has a packet to send then it inserts a 1 bit in the i-th slot. No other station is allowed to transmit during this slot.
  3. After N slots have passed by, each station has complete knowledge of which stations wish to transmit.
  4. Stations now begin transmitting in numerical order.



Analysis of Basic Bit-Map

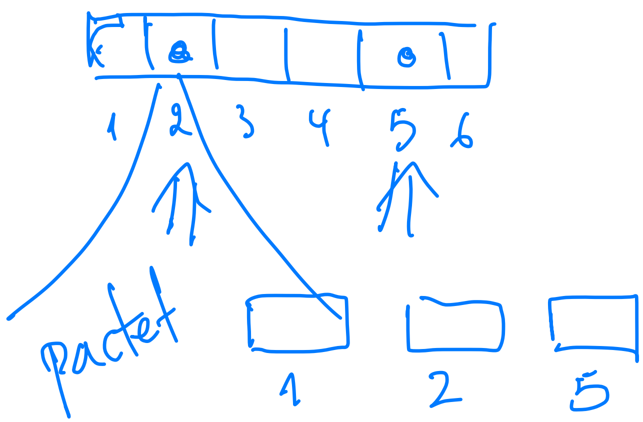


A station is out of luck if it becomes ready just after its bit slot passes by.

6 users

Tape has length n

$n + n$



Collision-Free Reservation Protocols: Binary Countdown

Here we assume all addresses are in binary and of the same length.

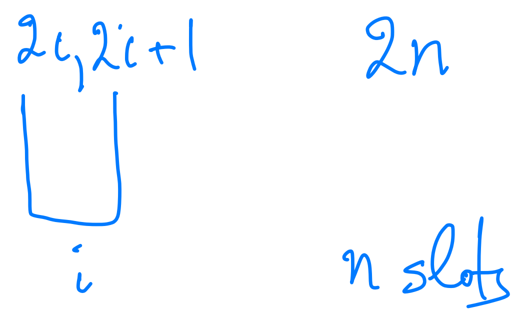
Binary Address	Bidding Cycle	
0 0 1 0	0 _ _ _	
0 1 0 0	0 _ _ _	
1 0 0 1	1 0 0 _	
1 0 1 0	1 0 1 0	← Winning Node
	$\boxed{1\ 0\ 1\ 0}$	← Result

Assume that the "losing" stations will eventually be scheduled

Nodes write their bit starting from highest to lowest order to a register. They drop out as long as another node has written a 1 in current position, but continue otherwise.

Splitting Algorithms

- Idea is to partition set of nodes into "groups".
- Each group is assigned to a slot.
- Thus a collision can only occur within a group.
- **Example 1:** Assume the nodes have unique (fixed length) identifiers (e.g., $1, 2, \ldots, m$)
  - Divide nodes into pairs.
  - Node $2i$ or $2i + 1$ goes in slot i
  - If collision occurs they go in order

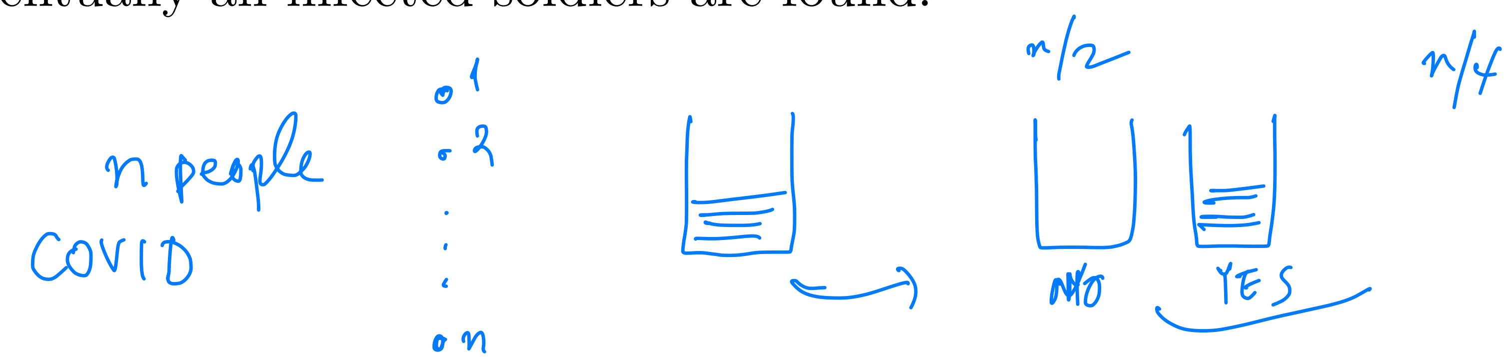


167 student

A B C ... Z

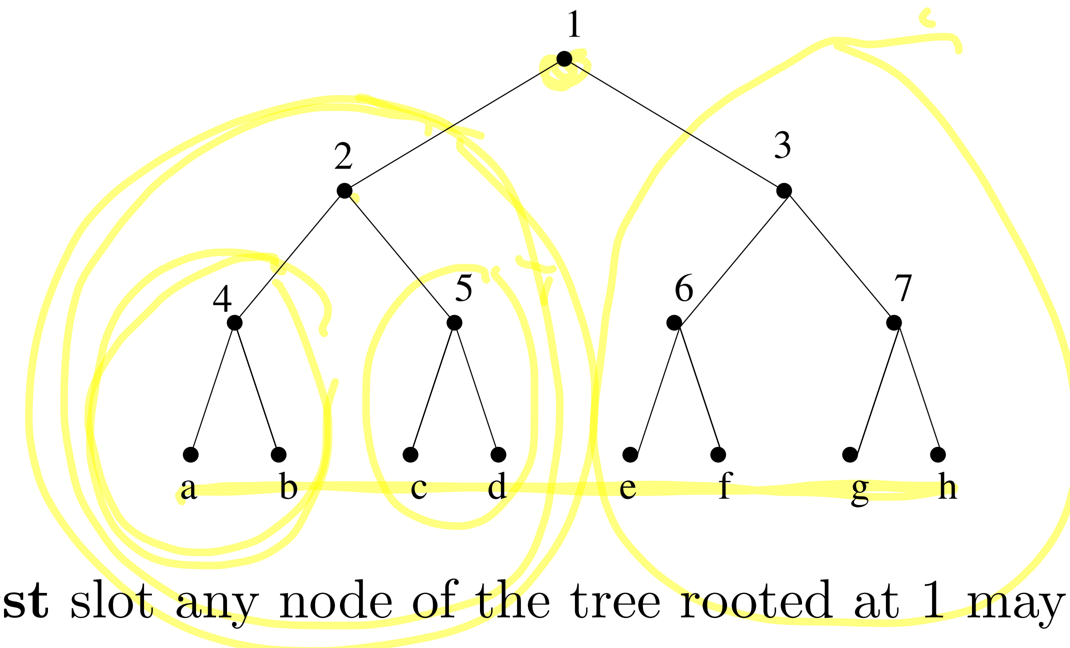
Paradigm

- Similar to US Army testing for syphilis. Take blood test from n soldiers.
- A portion of each sample is poured into a single test tube. This is tested for antibodies.
- If none found all soldiers in group declared healthy.
- If antibodies were found then split soldiers into two groups: $1..n/2$ and $n/2+1..n$ and iterate.
- Eventually all infected soldiers are found!



Tree Splitting Algorithm

Nodes are leaves of a tree.



During **first** slot any node of the tree rooted at 1 may contend.

If collision occurs, in **next** slot only nodes in left subtree (rooted at 2) may contend.

Again, if collision occurs, in **next** slot only nodes in left subtree (rooted at 4) may contend.

Tree Algorithm: Example

Eight stations $a, b, \ldots, h$ contend for a shared channel. Stations a, c, d, f, g suddenly become ready at once!

Slot 1: a, c, d, f, g; (collision)

Slot 2: a, c, d; (collision)

Slot 3: a; (success)

Slot 4: c, d; (collision)

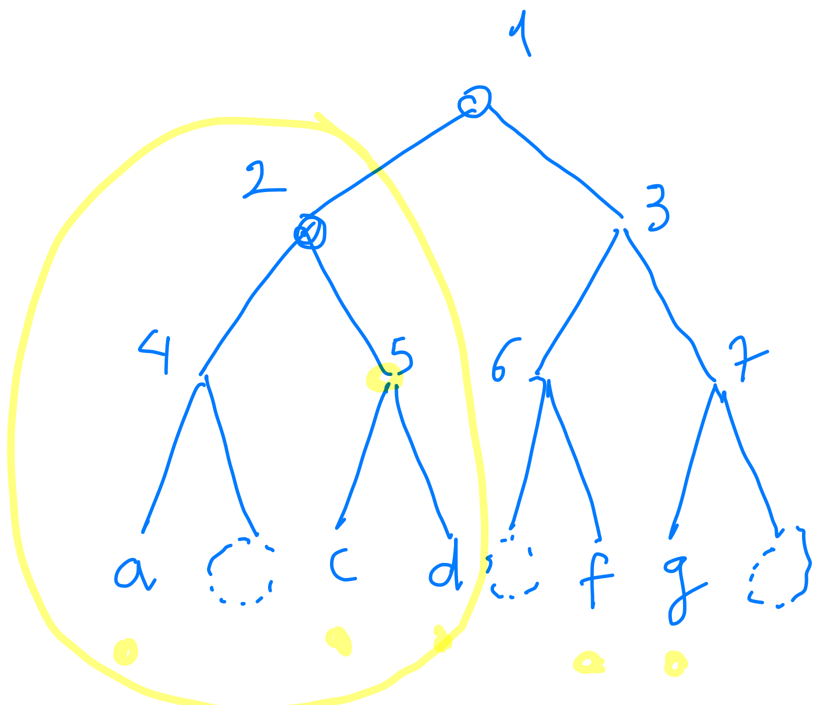
Slot 5: c; (success)

Slot 6: d; (success)

Slot 7: f, g; (collision)

Slot 8: f; (success)

Slot 9: g; (success)



Exercises[a]

1. What is the difference between coordinated and uncoordinated access?
2. Suppose that n nodes in uncoordinated contention select $p = \frac{1}{kn}$, where $k \geq 1$ is a given integer. What is the $\Pr[\text{Success}]$ as a function of k? Derive the formula.
3. Suppose that in uncoordinated contention at any given time at most $\sqrt{n}$ of the hosts contend for the channel. What is the $\Pr[\text{Success}]$?
4. What makes unslotted Aloha less efficient than slotted Aloha? What is the tradeoff?
5. In what ways do exponential and polynomial backoff algorithms differ?

[a] Do not submit.

6. Eight stations $a, b, \ldots, h$ contend for a shared channel. Stations c, d, f, g suddenly become ready at once!. Describe how the splitting algorithm allocates slots.
7. Eight stations $a, b, \ldots, h$ contend for a shared channel. Stations a, d, f, g suddenly become ready at once!. Describe how the splitting algorithm allocates slots.
8. Use binary countdown to resolve contention between hosts with the following addresses:

$$01001011, 01001010, 00000011$$

9. ($\star$) A medical lab tests the blood of x individuals for a blood-borne pathogen that affects a random individual w.p. p. The x individual blood samples are sent to the lab. If each sample is individually tested, x tests are required. Another option would be to pool the x blood samples forming one composite. If the composite tests negative then all x individuals

are assumed to be pathogen free and no additional testing is required. On the other hand, if the pathogen is detected and the composite tests positive, then each individual sample would have to be tested to determine exactly which of the samples are effected. In such a case a total of $x+1$ tests would be performed, the original composite test plus x individual tests.[b]

(a) What is the expected number of tests performed?

(b) What is the expected number of tests performed per individual?

(c) Composite testing would be preferred to individual testing if the expected number of tests per individual is < 1. What is the maximum value of p for which this can happen? (**Hint** Set the "expected number of tests per individual to 1" and optimize.)

[b] A more sophisticated but similar technique (tree splitting) is used in Ethernet to find collisions.