

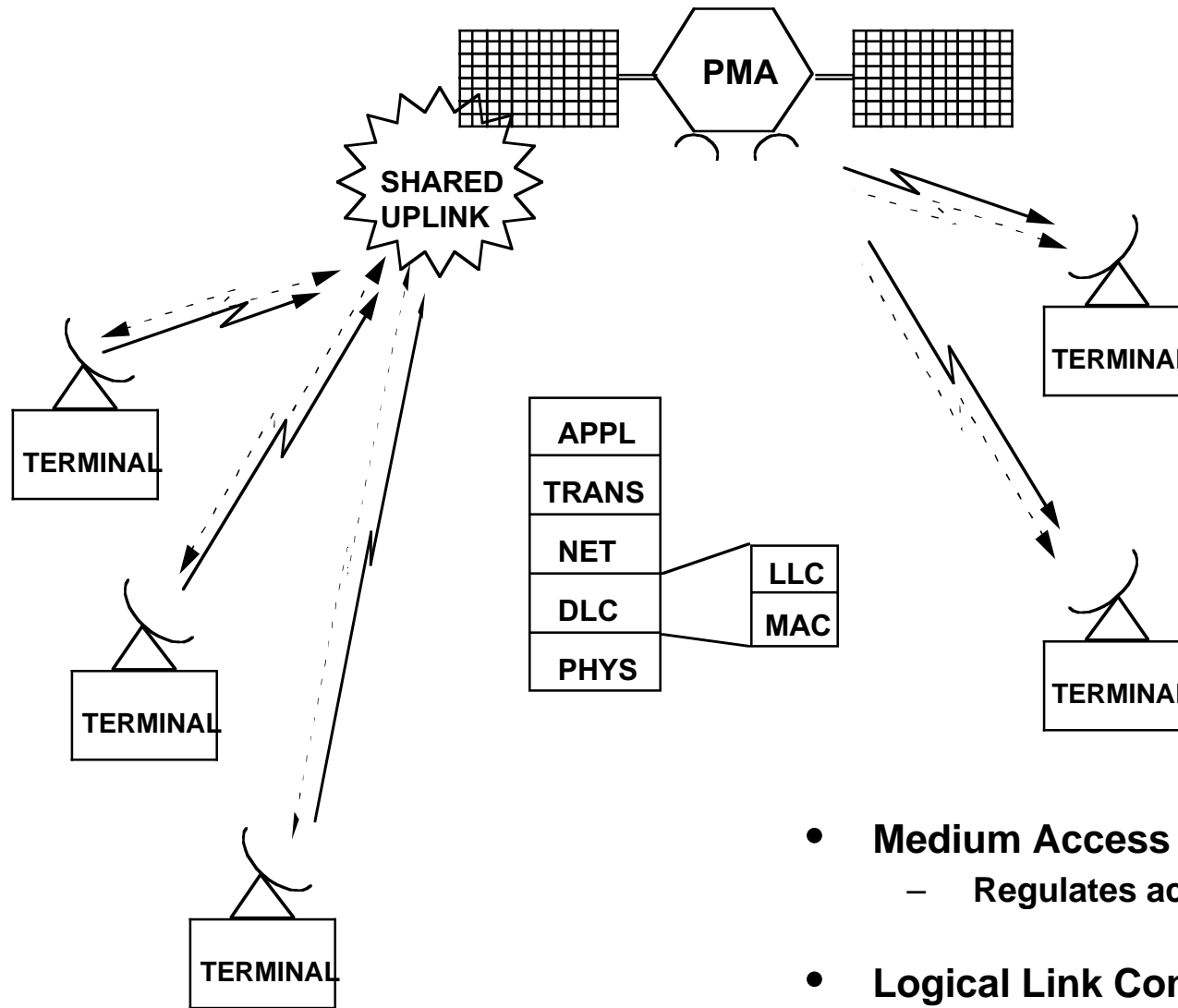


Packet multiple access and the Aloha protocol

Eytan Modiano
Massachusetts Institute of Technology
Department of Aeronautics and Astronautics



Packet Multiple Access



- **Medium Access Control (MAC)**
 - Regulates access to channel
- **Logical Link Control (LLC)**
 - All other DLC functions



Examples of Multiple Access Channels



- **Local area networks (LANs)**
- **Satellite channels**
- **Wireless radio**

- **Characteristics of Multiple Access Channel**
 - **Shared Transmission Medium**
 - A receiver can hear multiple transmitters
 - A transmitter can be heard by multiple receivers
 - **The major problem with multiple access is allocating the channel between the users**
 - Nodes do not know when other nodes have data to send
 - Need to coordinate transmissions



Approaches to Multiple Access



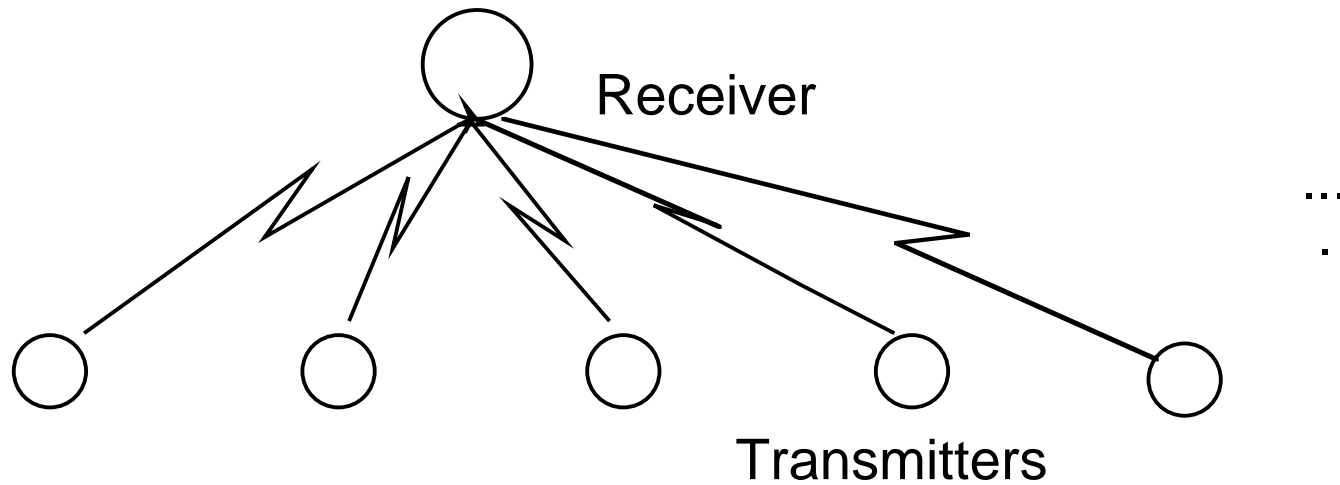
- **Fixed Assignment (TDMA, FDMA, CDMA)**
 - Each node is allocated a fixed fraction of bandwidth
 - Equivalent to circuit switching
 - very inefficient for low duty factor traffic
- **Packet multiple access**
 - Polling
 - Reservations and Scheduling
 - Random Access



Aloha



Single receiver, many transmitters

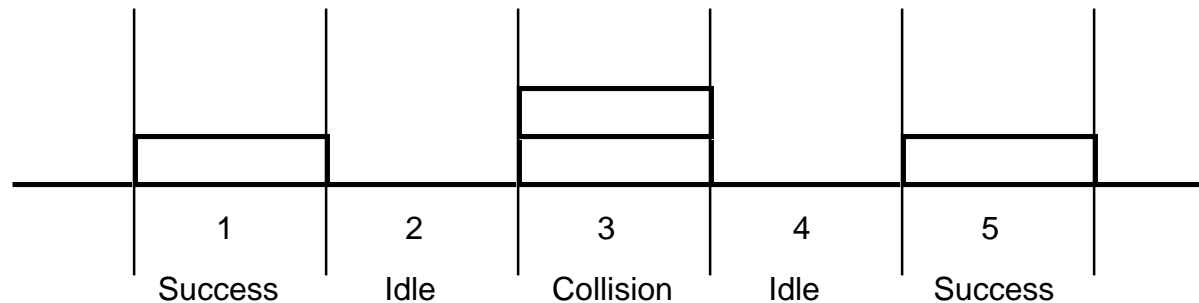


E.g., Satellite system, wireless



Slotted Aloha

- Time is divided into “slots” of one packet duration
 - E.g., fixed size packets
- When a node has a packet to send, it waits until the start of the next slot to send it
 - Requires synchronization
- If no other nodes attempt transmission during that slot, the transmission is successful
 - Otherwise “collision”
 - Collided packet are retransmitted after a random delay





Slotted Aloha Assumptions



- **Poisson external arrivals**
- **No capture**
 - Packets involved in a collision are lost
 - Capture models are also possible
- **Immediate feedback**
 - Idle (0) , Success (1), Collision (e)
- **If a new packet arrives during a slot, transmit in next slot**
- **If a transmission has a collision, it becomes backlogged and retransmitted after a random delay**
 - Let n be the number of backlogged nodes



slotted aloha

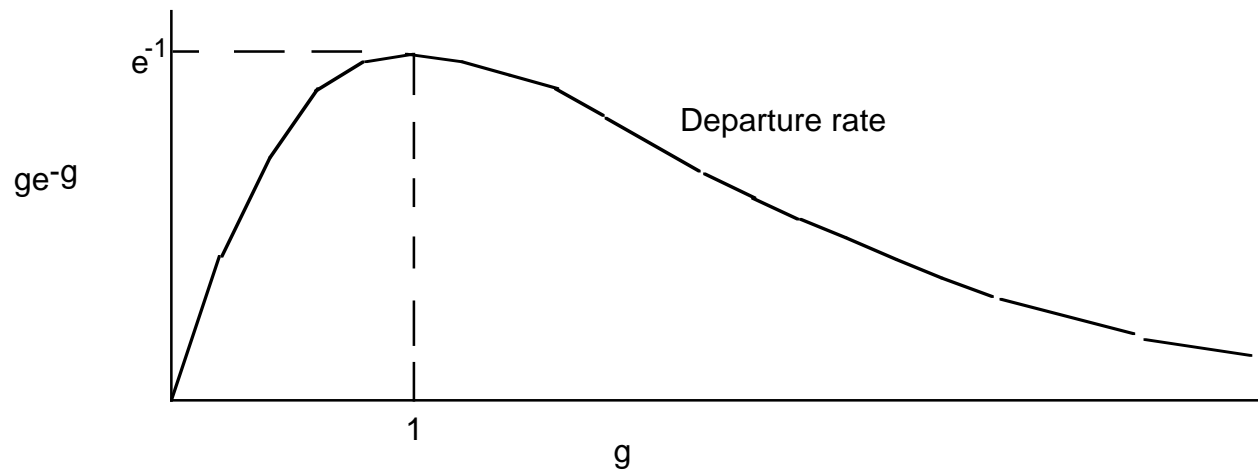


- Let g be the attempt rate (the expected number of packets transmitted in a slot)
 - The number of attempted packets per slot is approximately a Poisson random variable of mean $g = \lambda + n \cdot q_r$
 - q_r = probability that a backlogged packet is retransmitted in a slot
 - n = number of backlogged packets
 - $P(m \text{ attempts}) = g^m e^{-g} / m!$
 - $P(\text{idle}) = \text{probability of no attempts in a slot} = e^{-g}$
 - $p(\text{success}) = \text{probability of one attempt in a slot} = g e^{-g}$
 - $P(\text{collision}) = P(\text{two or more attempts}) = 1 - P(\text{idle}) - P(\text{success})$



Throughput of Slotted Aloha

- The throughput is the fraction of slots that contain a successful transmission = $P(\text{success}) = ge^{-g}$
 - When system is stable throughput must also equal the external arrival rate (λ)



- What value of g maximizes throughput?

$$\frac{d}{dg(n)} ge^{-g} = e^{-g} - ge^{-g} = 0$$

$$\Rightarrow g = 1$$

- $g < 1 \Rightarrow$ too many idle slots
- $g > 1 \Rightarrow$ too many collisions

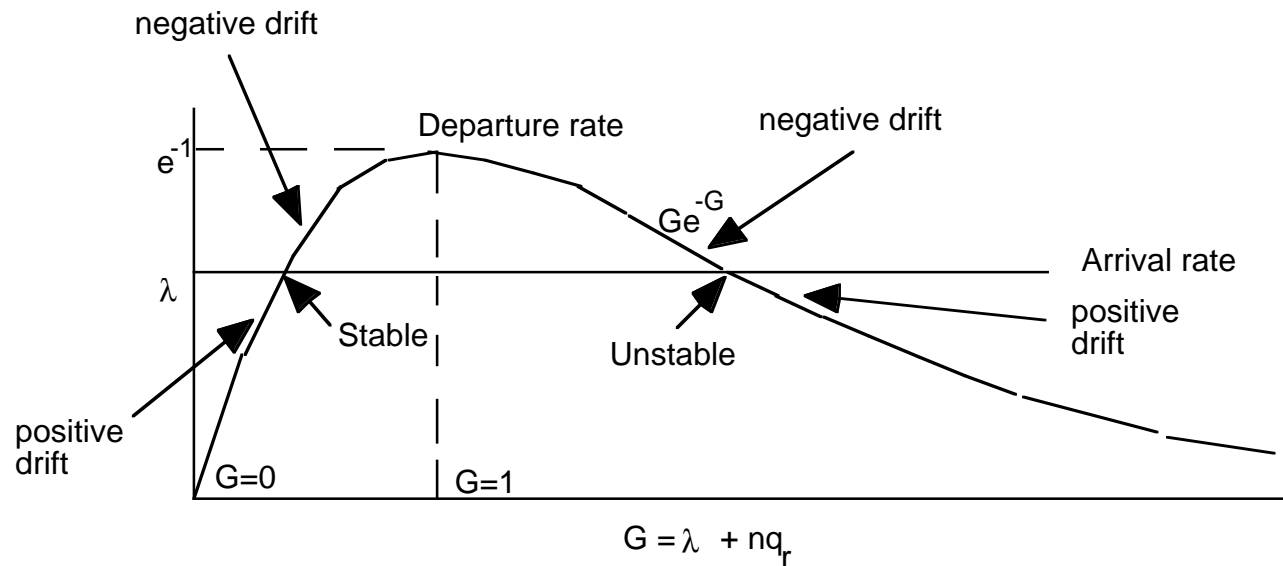
$$\Rightarrow P(\text{success}) = ge^{-g} = 1/e \approx 0.36$$

- If g can be kept close to 1, an external arrival rate of $1/e$ packets per slot can be sustained



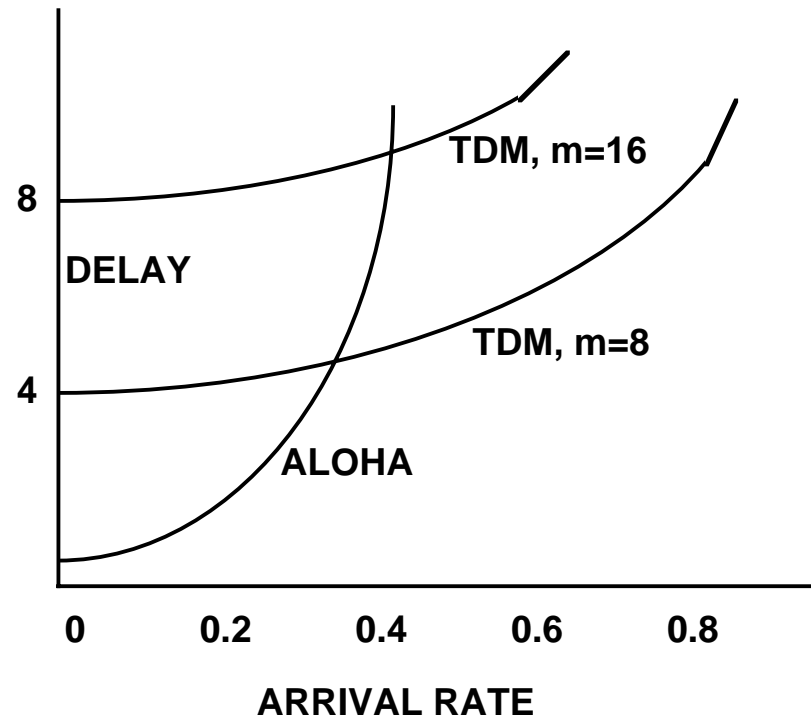
Instability of slotted aloha

- if backlog increases beyond unstable point (bad luck) then it tends to increase without limit and the departure rate drops to 0
 - Aloha is inherently unstable and needs algorithm to keep it stable
- Drift in state n , $D(n)$ is the expected change in backlog over one time slot
 - $D(n) = \lambda - P(\text{success}) = \lambda - g(n)e^{-g(n)}$





TDM vs. slotted aloha

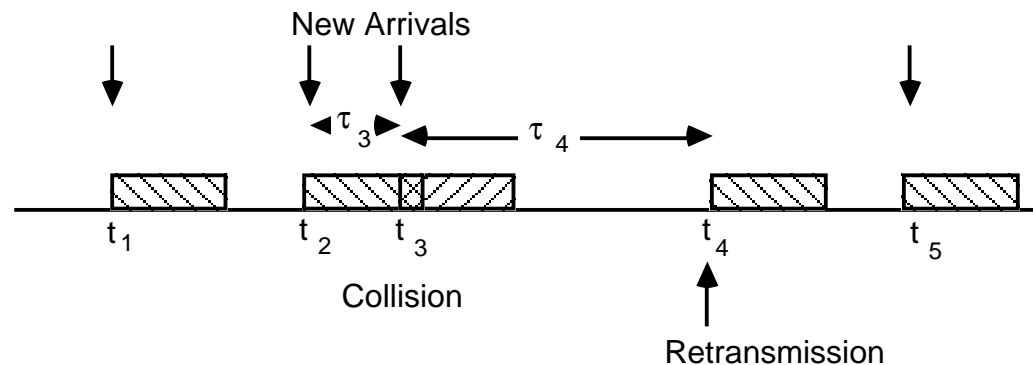


- Aloha achieves lower delays when arrival rates are low
- TDM results in very large delays with large number of users, while Aloha is independent of the number of users



Pure (unslotted) Aloha

- **New arrivals are transmitted immediately (no slots)**
 - No need for synchronization
 - No need for fixed length packets
- **A backlogged packet is retried after an exponentially distributed random delay with some mean $1/x$**
- **The total arrival process is a time varying Poisson process of rate $g(n) = \lambda + nx$ ($n = \text{backlog}$, $1/x = \text{ave. time between retransmissions}$)**
- **Note that an attempt suffers a collision if the previous attempt is not yet finished ($t_i - t_{i-1} < 1$) or the next attempt starts too soon ($t_{i+1} - t_i < 1$)**





Throughput of Unslotted Aloha



- **An attempt is successful if the inter-attempt intervals on both sides exceed 1 (for unit duration packets)**
 - **$P(\text{success}) = e^{-g} \times e^{-g} = e^{-2g}$**
 - **Throughput (success rate) = ge^{-2g}**
 - **Max throughput at $g = 1/2$, Throughput = $1/2e \sim 0.18$**
 - **Stabilization issues are similar to slotted aloha**
 - **Advantages of unslotted aloha are simplicity and possibility of unequal length packets**



Splitting Algorithms

- **More efficient approach to resolving collisions**

- **Simple feedback (0,1,e)**
- **Basic idea: assume only two packets are involved in a collision**

Suppose all other nodes remain quiet until collision is resolved, and nodes in the collision each transmit with probability $1/2$ until one is successful

On the next slot after this success, the other node transmits

The expected number of slots for the first success is 2, so the expected number of slots to transmit 2 packets is 3 slots

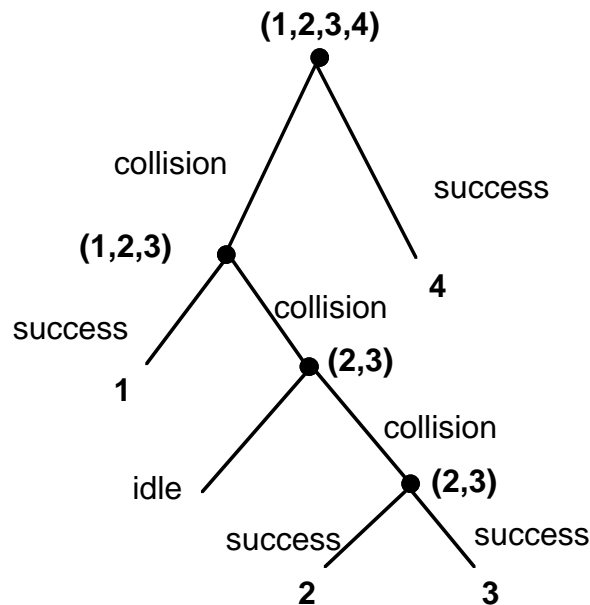
Throughput over the 3 slots = $2/3$

- **In practice above algorithm cannot really work**
 - Cannot assume only two users involved in collision
 - Practical algorithm must allow for collisions involving unknown number of users



Tree algorithms

- After a collision, all new arrivals and all backlogged packets not involved in the collision wait
- Each colliding packet randomly joins either one of two groups (Left and Right groups)
 - Toss of a fair coin
 - Left group transmits during next slot while Right group waits
 - If collision occurs Left group splits again (stack algorithm)
 - Right group waits until Left collision is resolved
 - When Left group is done, right group transmits



Notice that after the idle slot, collision between (2,3) was sure to happen and could have been avoided

Many variations and improvements on the original tree splitting algorithm



Throughput comparison



- **Stabilized pure aloha $T = 0.184 = (1/(2e))$**
- **Stabilized slotted aloha $T = 0.368 = (1/e)$**
- **Basic tree algorithm $T = 0.434$**
- **Best known variation on tree algorithm $T = 0.4878$**
- **Upper bound on any collision resolution algorithm with $(0,1,e)$ feedback $T \leq 0.568$**
- **TDM achieves throughputs up to 1 packet per slot, but the delay increases linearly with the number of nodes**