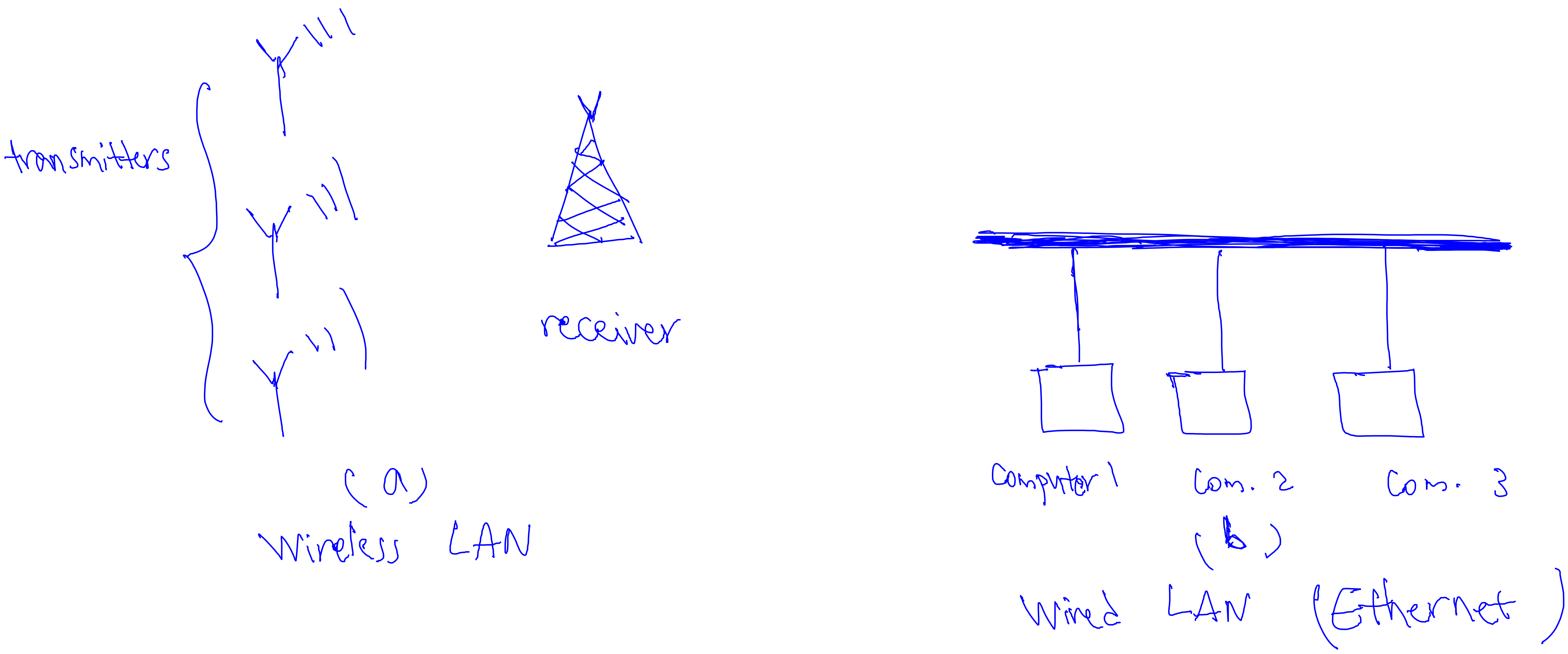
Medium Access Algorithms

Consider a set of m stations that wish to access the Internet by transmitting over a shared medium; examples include Local Area Networks (LANs), Satellite networks, Radio networks.



FDM or TDM can be used but potential difficulty with these approaches is: increased delay and underutilization of medium when stations are not active all the time. Also we want the medium access control (MAC) algorithm to be "distributed."

We assume packets have unit size. The time duration for transmission of a packet is called "time slot".

Basic Constraint: at each time, only one of the stations can transmit successfully.

If no stations transmits a packet, the channel (medium) is called "idle". If packet transmission of two or more stations overlap in time, a "collision" is said to happen.

Slotted ALOHA Protocol

Assume that all the stations have synchronized clock & time is divided into time slots. Stations attempt to transmit at the begining of time slots. If more than one stations transmit in the slot, collision happens and transmitters will be notified (feedback from the receiver).

Assume there n backlogged stations (i.e., have packets)

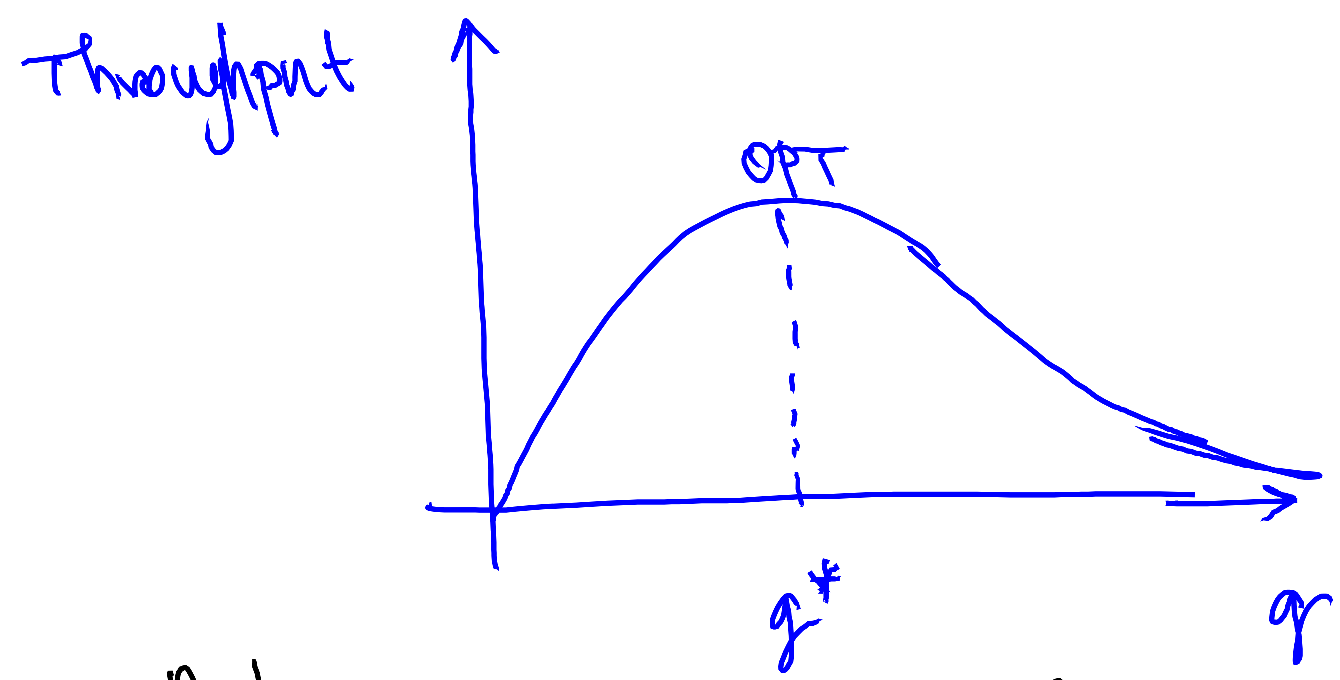
Each station attempts to transmit with probability q at a given time slot. Therefore,

$$E[\#\text{ of transmissions}] = nq$$

$$P(\text{successful transmission}) = \binom{n}{1} q (1-q)^{n-1} = r(q)$$

$$\mathbb{P}(\text{idle slot}) = (1-q)^n$$

$$\Rightarrow \quad \text{Throughput} = nq(1-q)^{n-1} = r(q)$$



$$\frac{dr(q)}{dq} = n(1-q)^{n-1} - nq(n-1)(1-q)^{n-2}$$

$$\frac{d}{dq} r(q) = 0 \Rightarrow 1-q-q(n-1) = 0$$

$$\Rightarrow \boxed{q^* = \frac{1}{n}}$$

$$r^*(n) = n \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}$$

$$= \left(1-\frac{1}{n}\right)^{n-1} > \boxed{\frac{1}{e} \approx 0.37}$$

$$\lim_{n\to\infty} r^*(n) = \frac{1}{e}$$

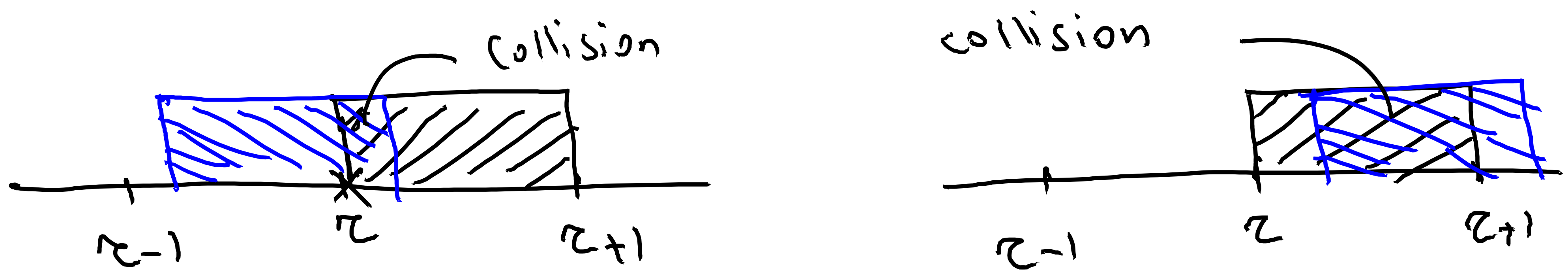
The algorithm is not completely distributed because every station needs to know the number of backlogged stations at every time slot. The feedback from idle slots, collisions, and successful transmissions can be used to estimate the number of backlogged stations.

Unslotted ALOHA Protocol

In this protocol, clocks of stations are not synchronized and stations can transmit at any time. For simplicity, assume that each node transmits according to a poisson process with rate q.

From the perspective of a station, its transmission at time τ is successful if no stations transmitted in $[\tau-1, \tau]$ and no stations will start transmitting during $[\tau, \tau+1]$

By the merging property of poisson processes, the overall transmission points of all n stations is a poisson process with rate nq.



$$\mathbb{P}(\text{success at } \tau) = \mathbb{P}(\text{no transmissions in } [\tau-1, \tau], \text{ no transmissions in } [\tau, \tau+1])$$

$$= \mathbb{P}(\text{no trans. in } [\tau-1, \tau]) \cdot \mathbb{P}(\text{no trans. in } [\tau, \tau+1])$$

By Independent Increment property of poisson process

$$= e^{-nq} \cdot e^{-nq}$$

$$= e^{-2nq}$$

The average # of transmissions in a unit time interval is qn,

Thus the average # of successful transmission per time unit is

$$\text{Throughput} = qn\, e^{-2qn} = r(q;n)$$

$$\frac{\partial r}{\partial q} r = 0 \implies \boxed{\begin{aligned} q^* &= \frac{1}{2n} \\ r^* &= \frac{1}{2} e^{-1} \approx 0.185 \end{aligned}}$$

Slotted ALOHA with Infinitely Many Stations

Assume that packets arrive according to a poisson process with rate λ and each new packet enters a new (unbacklogged) station. Let $N(t) :=$ # of backlogged stations at the beginning of time slot t

The infinite station model provides an upper bound on the delay achievable in the finite station model (with the same arrival rate), as a station with multiple packets in the latter case will not try to send more than one packet in each time slot, thus avoiding such sure collisions.

No Control : Algorithm uses a fixed q all the time, then exists a n_0 large enough, for all $n > n_0$,

$$\mathbb{E}\left[N(t+1) - N(t) \mid N(t) = n\right] = \lambda - nq(1-q)^{n-1} > \lambda/2$$

because $nq(1-q)^{n-1} \longrightarrow 0$ as $n \to \infty$

Thus Lyapunov drift outside the set $B = \{n : n \le n_0\}$ is strictly positive. This suggests that the system is unstable, i.e.,

$$\mathbb{P}\left[\lim_{t \to \infty} N(t) = \infty\right] = 1$$ for any $\lambda > 0$

This is in fact true. (a Lyapunov instability criterion that we do not discuss in this class)

Centralized (unrealistic) control:

As we saw earlier, the optimal $q(t)$ where $N(t)$ is known is $q^*(t) = \frac{1}{N(t)}$. If $N(t) \neq 0$, this yields a successful transmission with probability greater than e^{-1}. Thus the mean delay is no longer than that of a M\Geo\1 queue, where $a(t) \sim$ poisson random variable with mean λ, and

$S(t) \sim$ Bernoulli with mean $1/e$

$\Rightarrow$ the system is stable (mean delay finite) if $\lambda < 1/e$.

delay analysis for the M\Geo\1 queue:

waiting time of a packet before getting service:
$$W = R + \sum_{j=1}^{n} t_j + (t_{n+1} - 1)$$

n = # of packets in the system when the packet arrives (excluding the one in service).

R = residual time to the begining of next time slot.

t_j = time duration from $(j-1)$-th successful transmission to j-th successful transmission

Taking the average from both sides:

$$\overline{W} = \overline{R} + \overline{n}\, e + (e-1)$$

$$= \overline{R} + \lambda \overline{W}\, e + (e-1) \longrightarrow \text{Little's law}$$

But $\overline{R} < \frac{1}{2}$ (of time slot) (Infact $\overline{R} = \frac{\lambda}{2}$; Recall PK formula)

$\Rightarrow$ $$\overline{W} = \frac{e + \frac{1}{2} - 1}{1 - \lambda e}$$ average waiting time to get service

$$\overline{D} = \frac{e + \frac{1}{2} - 1}{1 - \lambda e} + 1$$ average delay including the service time

This is an upperbound on the delay performance, i.e

delay (unlimited station ALOHA) $<$

delay (M\Geo\1) $= \overline{D}$

because the success probability of ALOHA $= \left(1 - \frac{1}{N(t)}\right)^{N(t)-1} > \frac{1}{e}$

for example if $N(t) = 1$ (only one packet in the system)

$q^*(N(t)) = 1$ while under M\Geo\1 it is $\frac{1}{e}$.

Decentralized Control:

We need a variant of ALOHA where stations estimate the number of backlogged stations (i.e., those with packets).

This can be done by using the knowledge of feedback $Z(t)$, where

$$Z(t) = \begin{cases} 0 & \text{if slot "}t\text{" was idle} \\ 1 & \text{if successful trans. in slot "}t\text{"} \\ e & \text{if there was a collision in slot "}t\text{"} \end{cases}$$

$N(t)$ = # of packets in the system at time slot "t"

$\hat{N}(t)$ = estimation of $N(t)$

$$N(t+1) = N(t) - \underbrace{\mathbb{1}(Z(t)=1)}_{\text{indicator function}} + \underbrace{a(t)}_{\text{\# of arrivals in slot "}t\text{"}}$$

Assume we know $E[a(t)] = \lambda$ (this can be done by simply calculating the average over time), then a reasonable way to update $\hat{N}(t)$ is through the following update equation:

$$\hat{N}(t+1) = \left(\hat{N}(t) - \mathbb{1}(Z(t)=1) + c(Z(t))\right)_+ + \lambda$$

where $c = (c(0), c(1), c(e))$ is chosen properly so that the "estimation error" $D(t) = \hat{N}(t) - N(t)$ remains small.

Then each backlogged station attempts to transmit with probability $q(\hat{N}(t)) = \min\left(1, \frac{1}{\hat{N}(t)}\right)$.

This is called Rivest's Pseudo-Bayesian ALOHA. [Riv '85].

Q: How to choose "c" ?

A: Suppose $\hat{N}(t) > 1 - \min\{c(0), c(1), c(e)\}$, thus

$$D(t+1) = \hat{N}(t+1) - N(t+1)$$

$$= \hat{N}(t) - \cancel{\mathbb{1}(Z(t)=1)} + c(Z(t)) + \lambda - N(t) + \cancel{\mathbb{1}(Z(t)=1)} - a(t)$$

$$= \left(\hat{N}(t) - N(t)\right) - (a(t) - \lambda) + c(Z(t))$$

$$= D(t) - (a(t) - \lambda) + c(Z(t))$$

Let us see how $D(t)$ behaves by looking at the drift of $D(t)$:

$$\mathbb{E}\left[D(t+1) - D(t) \,\middle|\, N(t), \hat{N}(t)\right] = c(0)\, \mathbb{P}(Z=0 \mid N(t), \hat{N}(t)) + c(1)\, \mathbb{P}(Z=1 \mid N(t), \hat{N}(t)) + c(e)\, \mathbb{P}(Z=e \mid N(t), \hat{N}(t))$$

$$\mathbb{P}(Z=0 \mid N(t), \hat{N}(t)) = \left(1 - q(\hat{N}(t))\right)^{N(t)}$$

$$\mathbb{P}(Z=1 \mid N(t), \hat{N}(t)) = N(t)\, q(\hat{N}(t)) \left(1 - q(\hat{N}(t))\right)^{N(t)-1}$$

$$\mathbb{P}(Z=e \mid N(t), \hat{N}(t)) = 1 - \mathbb{P}(Z=0 \mid N(t), \hat{N}(t)) - \mathbb{P}(Z=1 \mid N(t), \hat{N}(t))$$

whenever $N(t)$ or $\hat{N}(t)$ is large enough.

$$\mathbb{P}(Z=0 \mid N(t), \hat{N}(t)) \approx \exp(-G(t))$$

$$\mathbb{P}(Z=1 \mid N(t), \hat{N}(t)) \approx G(t) \exp(-G(t))$$

$$\mathbb{P}(Z=e \mid N(t), \hat{N}(t)) \approx 1 - (1 + G(t)) \exp(-G(t))$$

where $G(t) = \begin{cases} \dfrac{N(t)}{\hat{N}(t)} & \text{if } \hat{N}(t) \neq 0 \\ 0 & \text{if } \hat{N}(t) = 0 \end{cases}$

(using the fact that $\left(1 - \frac{1}{x}\right)^x \approx e^{-1}$, x large enough)

Therefore drift of $D(t)$ satisfies

$$\mathbb{E}[D(t+1) - D(t) \mid N(t), \hat{N}(t)] \approx d(G(t))$$

$$d(G) := c(0)\, e^{-G} + c(1)\, G\, e^{-G} + c(e)\left(1 - (1+G)\, e^{-G}\right)$$

We would like to design the drift such that sign of drift is oposite of the sign of $D(t)$ so that error process is stable (i.e., error remains bounded)

(this is similar to the dynamics of $\dot{x} = -x$ where $x(t) \longrightarrow 0$ as $t \rightarrow \infty$)

If $D(t) > 0$: then $G(t) < 1$, so we want

$$d(G) < 0 \quad \text{when} \quad G < 1 \qquad \text{I}$$

If $D(t) < 0$: then $G(t) > 1$. So we want

$$d(G) > 0 \quad \text{when} \quad G > 1 \qquad \text{II}$$

By continuity of $d(G)$, this forces $d(1) = 1$ III

One choice of c to ensure I, II, III hold is

$$c(0) = -\frac{e-2}{e-1} \approx -0.418 \ , \quad c(1) = 0 \ , \quad c(e) = \frac{1}{e-1} \approx 0.582$$

This system performs almost as well as the system with centralized control. (the system is stable for any $\lambda < e^{-1}$). We ignore the rigorous stability proof.

Intuitively, the choice of "c" as above, reduces the attempt probability if there is a collision and increases it if there is idle slots.

This suggests the use of backoff mechanisms:

1- each packet upon its arrival is assigned a counter c . $K=0$

2- at each time slot, do the following:

- if $c > 0$, set $c = c-1$

- if $c = 0$, transmit

  * if successful, stop

  * If collision, $k = k+1$, $c = f(c_0, k)$

- REPEAT

- exponential back off : $f(c_0, k) = c_0 2^k$

- polynomial backoff : $f(c_0, k) = c_0 k^\alpha$, $\alpha > 1$

(Intuitively c must be proportional to $\frac{1}{N(t)}$)