Random Access Protocols

ALOHA

ALOHA

- Invented by N. Abramson in 1970-Pure ALOHA
- Uncontrolled users (no coordination among users)
- Same packet (frame) size
- Instant feedback
- Large (~ infinite) population

Protocol

- Simple Protocol
  - Users transmit packets as and when these are ready
  - Packets are broadcast by a hub to *all* users.
  - Any overlap of packets $\Rightarrow$ Collision
  - All packets involved in a collision are back-logged
  - Back logged packets are retransmitted after a *random* delay.
  - Throughput $\Rightarrow$ packets suffering no collision (fresh or retransmitted)

The Pure ALOHA Algorithm

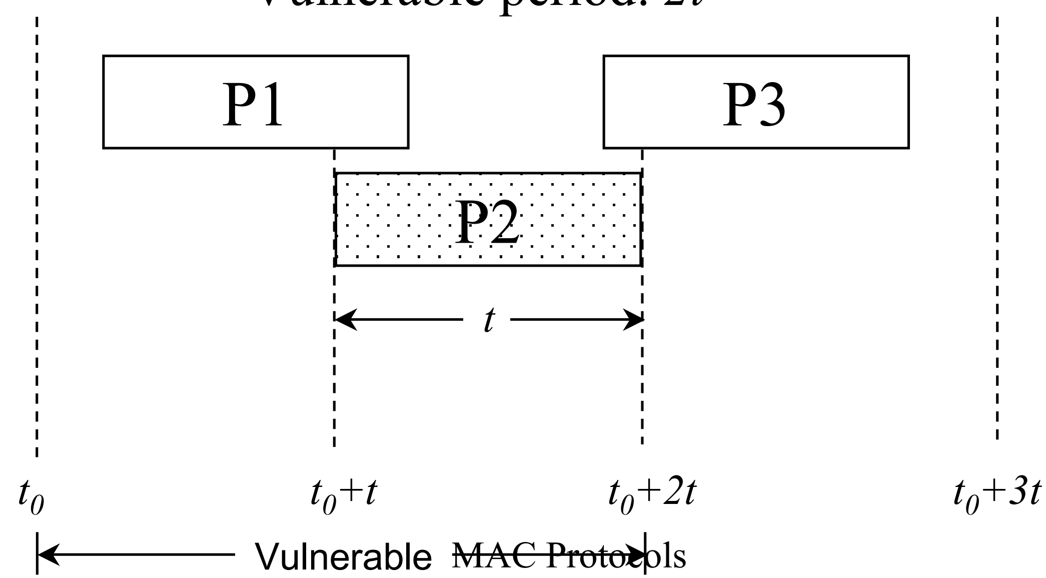
1. Transmit whenever you have data to send
2. Listen to the broadcast
   – Because broadcast is fed back, the sending host can always find out if its packet was destroyed just by listening to the downward broadcast one round-trip time after sending the packet
3. If the packet was destroyed, wait a random amount of time and send it again
   – The waiting time must be random to prevent the same packets from colliding over and over again

Pure ALOHA *(cont'd)*

- Note that if the first bit of a new packet overlaps with the last bit of a packet almost finished, both packets are totally destroyed.

t : one packet transmission time

Vulnerable period: *2t*



Pure ALOHA

Channel load = freshly generated packets +Retransmitted packets

G = Channel load *per* packet time

S = Throughput *per* packet time

P(k arrivals) = $\dfrac{G^{K}e^{-G}}{G!}, k = 0,1,2\ldots$

Psuc= Pr(Packet is successfully transmitted)

= Pr(No collision in vulnerability interval)

Pure ALOHA…(contd.)

- Vulnerability interval = 2 x packet time
- Average channel load in the vulnerability interval = 2G
- P_{succ} = Pr (channel is silent in the vulnerability interval)
  $= e^{-2G} = P_{suc}$
- Normalized Throughput = S = Average Channel load x $P_{suc} = G\, e^{-2G}$

  $S_{max} = 1 / 2e = 0.184$ at $G = 1 / 2$

- Maximum throughput 18% *but* protocol is simple.

Slotted ALOHA (L.G. Roberts)

- Users can transmit packets in slots only. Slot boundaries cannot be transgressed.
- Packets P2 and P3 collide
- Vulnerability interval is one slot (packet) time
- $P_{suc}=e^{-G}$
- $S_{max} = 1/e = 0.368$ at $G=1$

Slotted ALOHA

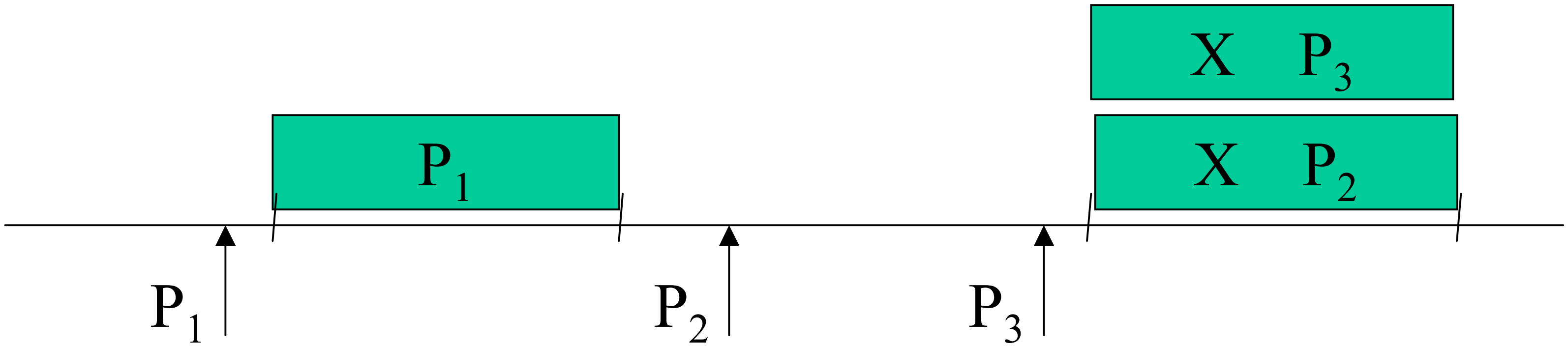
Assumptions

- all frames same size
- time is divided into equal size slots, time to transmit 1 frame
- nodes start to transmit frames only at beginning of slots
- nodes are synchronized
- if 2 or more nodes transmit in slot, all nodes detect collision

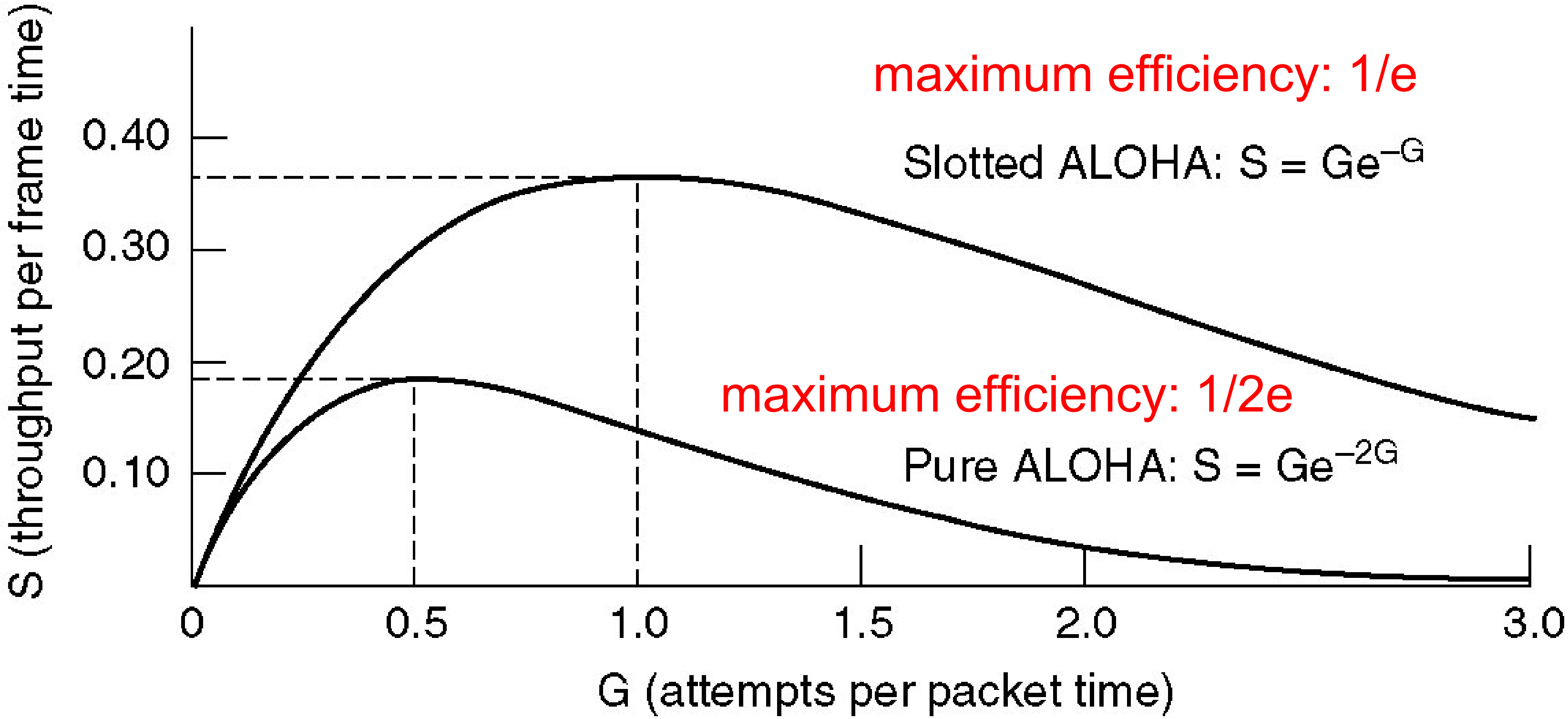
Operation

- when node obtains fresh frame, it transmits in next slot
- no collision, node can send new frame in next slot
- if collision, node retransmits frame in each subsequent slot with prob. p until success

Slotted ALOHA…(contd.)



Slotted Aloha



Throughput versus offered traffic for ALOHA network

Average Number of transmission per packet

P(k) = P_r (exactly k attempts are required for a successful transmission)

= P_r [(k-1) collisions and one success]

$P(k) = (1-e^{-G})^{k-1}\, e^{-G}$ (Geometric Distribution)

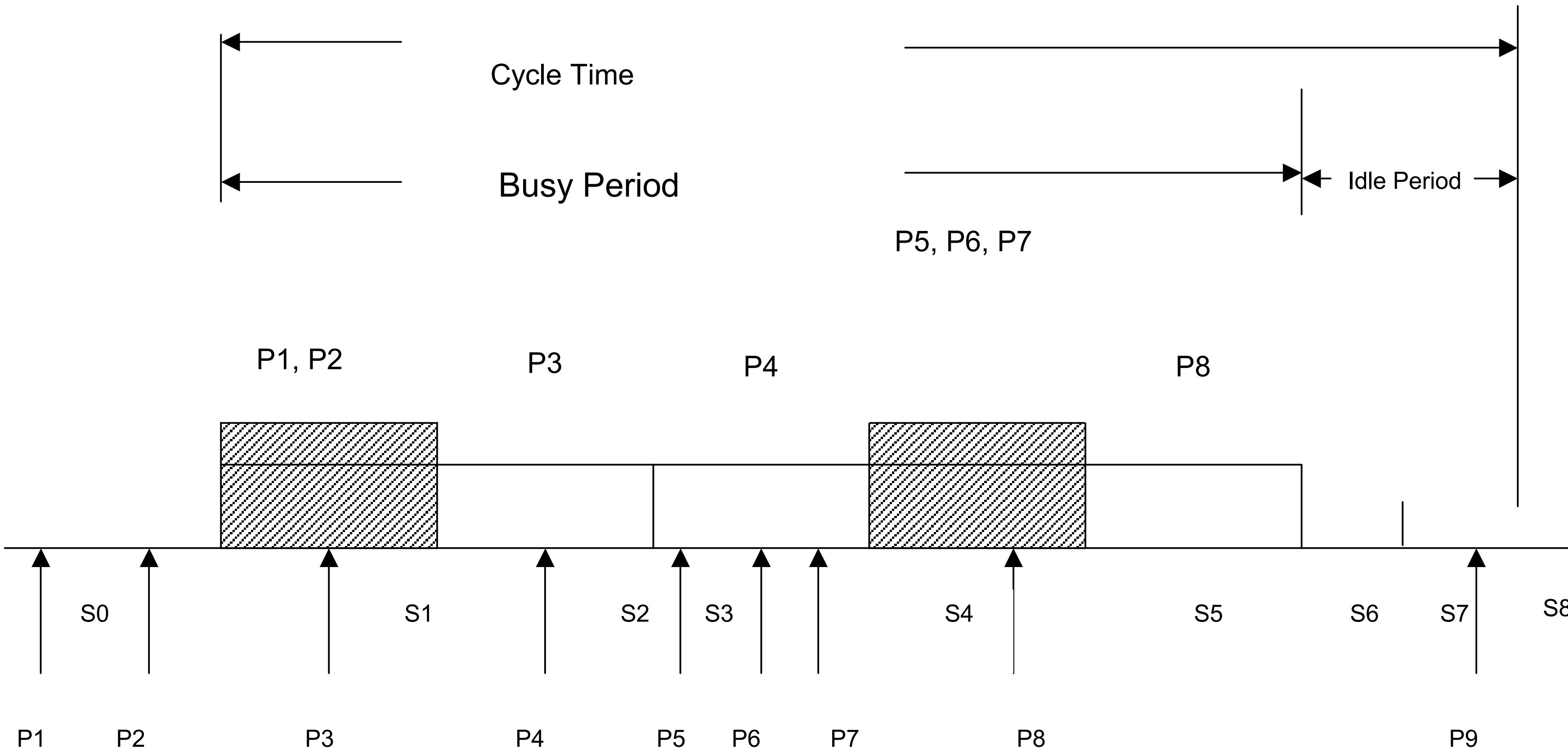
Hence,

$$E[K] = \sum_{K=1}^{\infty} KP(k) = e^{G}$$

Average number of transmissions increase *exponentially* with channel load

Slotted ALOHA- Cycle Time Analysis

- Cycle = Busy Period + Idle Period
- C = B + I
- Busy Period = Collision Intervals + Successful transmission intervals
- For a cycle to complete : at least one slot be present in the idle period and at least one transmission is scheduled in the slot *following* Idle Period.


Cycle Time
Busy Period
Idle Period
P5, P6, P7
P1, P2
P3
P4
P8
S0
S1
S2
S3
S4
S5
S6
S7
S8
P1
P2
P3
P4
P5
P6
P7
P8
P9

Slotted ALOHA- Cycle Time Analysis

•	P1, P2	scheduled in S1 : both packets collide
•	P3	scheduled in S2 : successful
•	P4	scheduled in S3 : successful
•	P5, P6, P7	scheduled in S4 : all the packets collide
•	P8	scheduled in S5 : successful

•	No arrivals in S5 and S6	:	S6 and S7 are idle
•	P9 arrives in S7	:	it is scheduled in the first slot of the next cycle.
•	Busy period	:	S1 – S5
•	Collision periods	:	S1 + S4
•	Successful periods	:	S2 + S3 + S5
•	Idle period	:	S6 + S7

Slotted ALOHA- Cycle Time Analysis

- Busy period, B, and idle period, I, are RVs.
- P (I=1) = P (one or more packets scheduled in the first slot)
- = 1-P (no packet is scheduled in the first slot)
- = $1 - e^{-G}$
- P(I=2) = P (no packet is scheduled in the 'first' slot and at least one packet is scheduled in the following slot)

  = $e^{-G} (1-e^{-G})$

- In General
- $P(I = k) = \quad (e^{-G})^{k-1} (1-e^{-G})$ (Geometric Distribution)

  Average Idle Period = $\quad E[I] = 1 /(1- e^{-G})$
- $P(B=k) =$ P(one or more arrivals in k-1 slots and no arrival in kth slot)

  $= \quad (1-e^{-G})^{k-1} e^{-G}$ (Geometric Distribution)
- Average Busy Period = $\quad E[B] = 1 / e^{-G}$

- U = Utilization = Number of successful slots in a busy period
- P (Successful Transmission) = P (one arrival in the previous slot given some arrivals have occurred)
- $P_{suc} = Ge^{-G} / (1- e^{-G})$
- P(U=k/B) = P (k slots are successful/ busy period is B slots)

$$= \binom{B}{k} (P_{suc})^k (1 - P_{suc})^{B-k}$$

Binomial Distribution

- Hence

$$E[U/B] = B.\ P_{suc}$$

$$E[U/B] = \quad E[E[U/B]] = E[B]\ P_{suc}$$

$$\text{Throughput} = E[U] / E[C]$$

$$= E[U] / (\ E[B] + E[I]\ )$$

$$= E[B]\ .\ P_{suc}\ /\ (\ E[B] + E[I]\ )$$

Using E[B], E[I] and P_{suc} expressions

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

Finite Population ALOHA

- Define,
- G_i = P (ith user will transmit in some slot)
- S_i = P (ith users packet does not suffer a collision)
- Normalized Channel Load for n users

$$G = \sum_{i=1}^{n} G_i$$

- Normalized Channel throughput for n users $= S = \sum_{i=1}^{n} S_i$

Finite Population ALOHA

- Hence,
- $S_i \;=\; G_i \quad \prod_{\substack{j=1 \\ j \neq i}}^{n} \left(1 - G_j\right)$

  = (ith users packet is in the system and remaining users are not attempting)
- Case 1: Identical users, $n \to \infty$

  $G_i = G / n$ and $S_i = S/n$

  $S/n = G/n\ (1-G/n)^{n-1}$

  using $\lim_{n\to\infty} \left(1 + \frac{x}{n}\right)^{n} = e^{x}$

Finite Population ALOHA

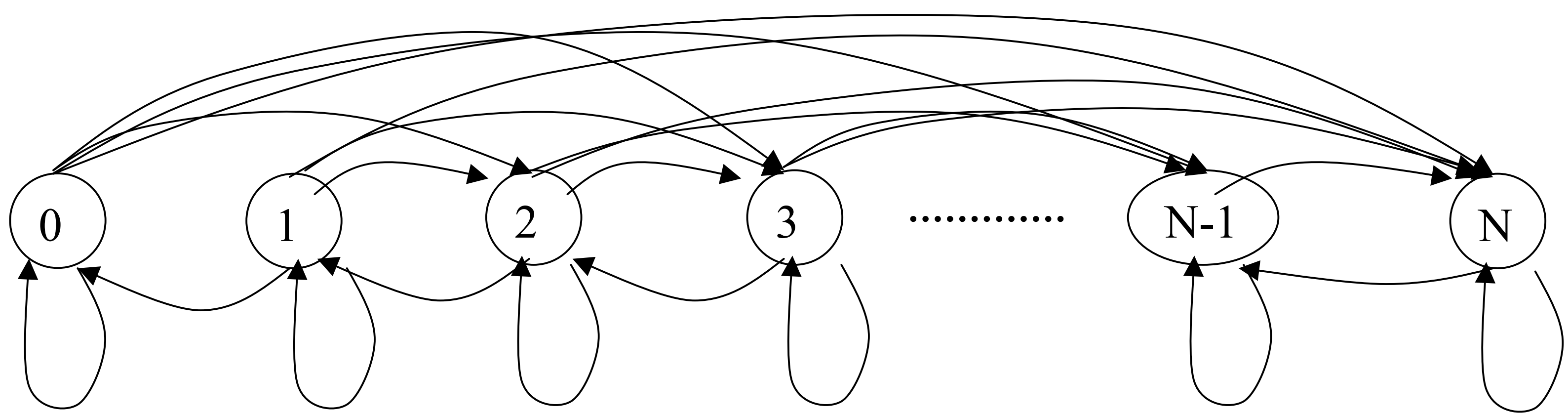
- Case 2: Two classes of users
  - Class 1: n_1 users, S_1, G_1
  - Class 2: n_2 users, S_2, G_2
  - $n = n1 + n2$
  - $S_1 = G_1(1-G_1)^{n1-1} \quad (1-G2)^{n2}$
  - $S2 = G1(1-G2)^{n2-1} \quad (1-G1)^{n1}$
  - Condition for maximum throughput
  - $n_1 G_1 + n_2 G_2 = 1$
  - Throughput in Class 1 : $n_1 S_1$
  - Throughput in Class 2 : $n_1 S_1$

Delay and throughput in S-ALOHA

- Assume finite number of users (=N)
- Thinking State : P (Generation of a fresh packet) = p
- Backlogged State: P (retransmission of a collided packet in the next slot = α)
- P (exactly R slots is the retransmission delay)
- $P(R) = (1-\alpha)^{R-1} \alpha$ [Geometric Distribution]
- Average retransmission delay = $E[k] = 1/\alpha$

Markov Chain (S-ALOHA)

- State of the system : Number of backlogged users
- State = K => (N-K) unblocked users (thinking)
- Markov Chain



Delay and throughput in S-ALOHA

- All S_i to S_j (j>i) forward transitions are possible except S_0 to S_1
- Only S_j to S_{j-1} backward transitions are possible.
- All S_i to S_i transitions are possible

• P (system is in state i) = P(i)

$$\sum_{i=1}^{N} P(i) = 1$$

• P (transition from state S_i to S_j) = p_{ij}

Delay and throughput in S-ALOHA

- When the system is in S_i, some of (N-i) stations can inject fresh packets (=f)
- Some of i stations can attempt retransmissions (=r)

$$0 \le f \le (N - i)$$

$$0 \le r \le i$$

Delay and throughput in S-ALOHA

(a) Backward Transitions

$p_{i,\,j} = 0$ for $f \leq i - 2$

$p_{i,\,i\text{-}1} =$ (no fresh packet generated and exactly one retransmission occurs)

$p_{i,\,i\text{-}1} = P(f{=}0)\;.\;P(r{=}1)$

Same state transitions

E1 : No fresh packet generated and no successful retransmission($r \neq 1$)

E2 : One fresh packet generated (and through) and no retransmission attempted ($r=0$). (same backlogged stations)

$$p_{ii} = \quad P(E_1) + P(E_2)$$

$$p_{ii} = \quad P(f=0)\ P(r \neq 1) + P(f=1)\ P(r=0)$$

Forward Transitions

$p_{0,1}$ = 0

$p_{i,\ i+1}$ = P (one fresh packet generated *and* at least one retransmission attempt)

$p_{i,\ i+1}$ = $P(f=1)\ P(r\geq 1)$

$p_{i,j}$ = $P(f=j-i)$ for $j \geq i+2$

$$P(f=0) = \binom{N-i}{0}(1-p)^{N-i}\,p^{0} = (1-p)^{N-i}$$

Forward Transitions…(contd.)

$$P(f=1) = \binom{N-i}{1} p(1-p)^{N-i-1} = (N-i)p(1-p)^{N-i-1}$$

$$P(f=j-i) = \binom{N-i}{j-i} p^{j-i}(1-p)^{N-j}$$

$$P(r=0) = \binom{i}{0} \alpha^{0}(1-\alpha)^{i} = (1-\alpha)^{i}$$

$$P(r=1) = \binom{i}{1} \alpha(1-\alpha)^{i-1} = i\alpha(1-\alpha)^{i-1}$$

$$\mathrm{P}(\mathrm{r}\neq 1) = 1 - P(r = 1) = 1 - i\alpha(1-\alpha)^{i-1}$$

$$\mathrm{P}(\mathrm{r}\geq 1) = 1 - P(r = 0) = 1 - (1-\alpha)^{i}$$

$$P(k) = \sum_{i=0}^{N} P(i)\, p_{ik}$$

Throughput

Successful transmission when state is S_i :
All backlog users are silent and one fresh packet is transmitted *or* one backlogged user transmits and no fresh packet is generated

Throughput and Backlog

P_{suc} (i) = P(f=1) P (r=0) + P(f=0) P(r=1)

P_{suc} (i)= $$(N-i)p(1-p)^{N-i-1}(1-\alpha)^{i}+(1-p)^{N-i}i\alpha(1-\alpha)^{i-1}=S(i$$

Mean throughput = S = P_{suc} = E $\left[P_{suc}(i)\right]=\sum_{i=0}^{N}P_{suc}(i)P(i)$

Mean backlog $=E[D]=MeanDelay=\frac{E[k]}{S}=\frac{\bar{k}}{S}=\bar{D}[[Little'sFormu$

Overall average packet arrival rate = p$\left(N-\bar{k}\right)=S$

Delay

Total number of packets in the system

= Average back logged packets + Average fresh arrivals

$$W = \overline{K} + p\left(N - \overline{k}\right)$$

Using Little's Formula

$$\overline{D}.p\left(N - \overline{k}\right) = \overline{K} + p\left(N - \overline{k}\right)$$

$$\overline{D} = 1 + \frac{\overline{K}}{p\left(N - \overline{K}\right)}$$

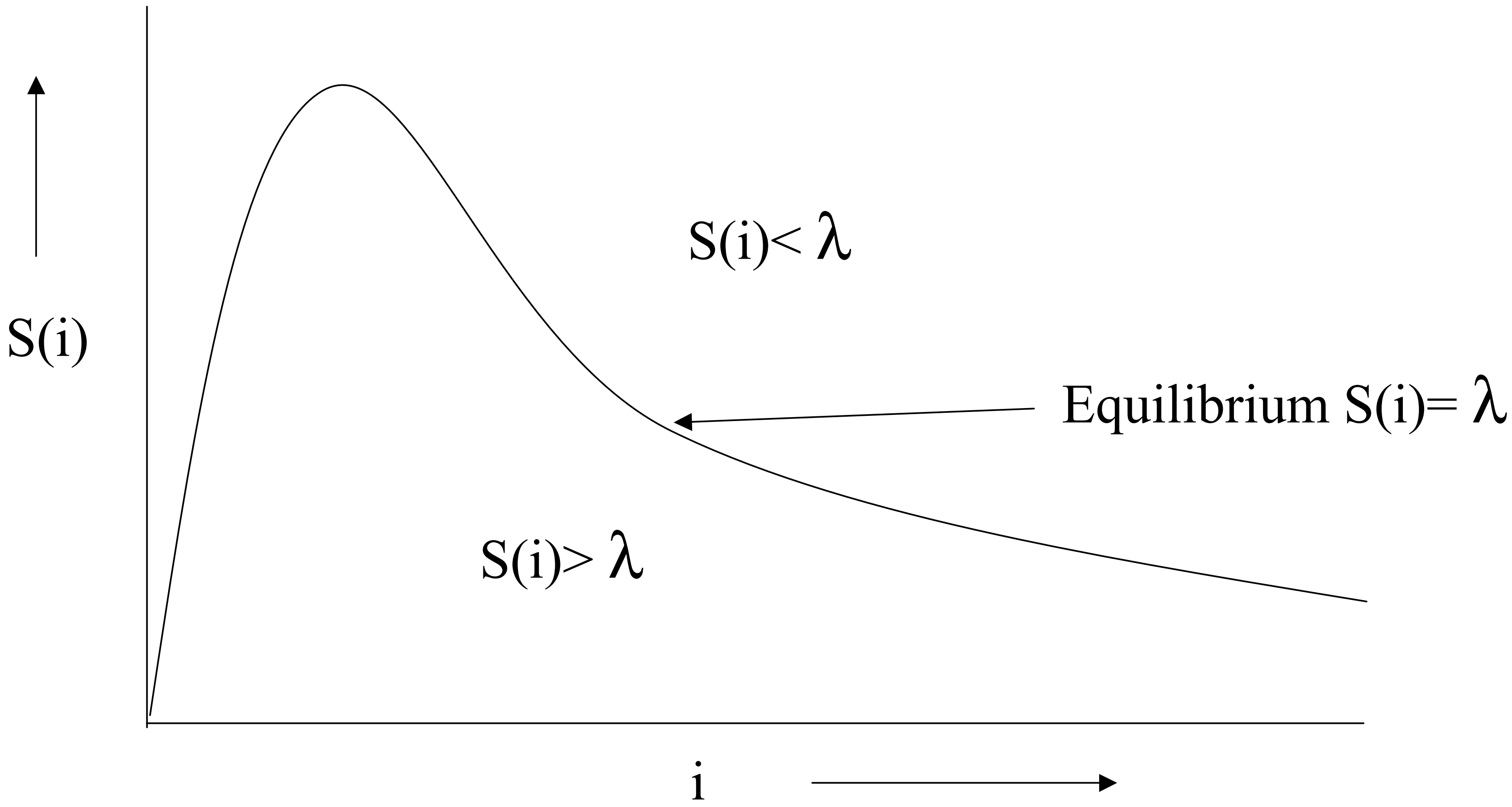
Stability in S-ALOHA

ALHOA Systems are potentially unstable When

$$N \rightarrow \infty, Np \longrightarrow \lambda = S$$

$$S(i) = (1-\alpha)^{i} Se^{-s} + i\alpha(1-\alpha)^{i-1} e^{-s}$$

S(i) vs i is throughput – backlog plot for fixed α


S(i)
S(i)< λ
Equilibrium S(i)= λ
S(i)> λ
i

Stability in S-ALOHA

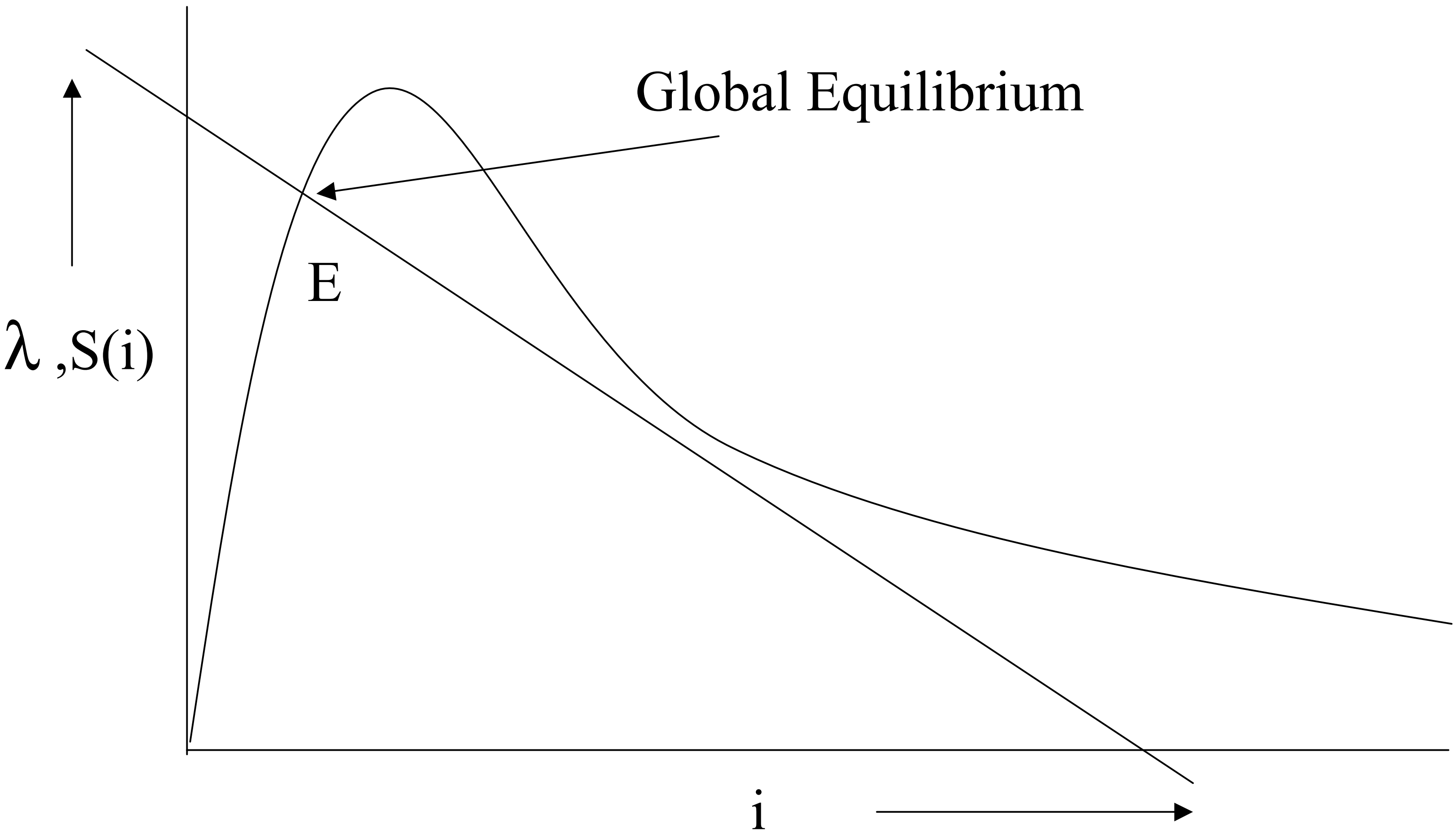
Packets leave system at S(i) rate

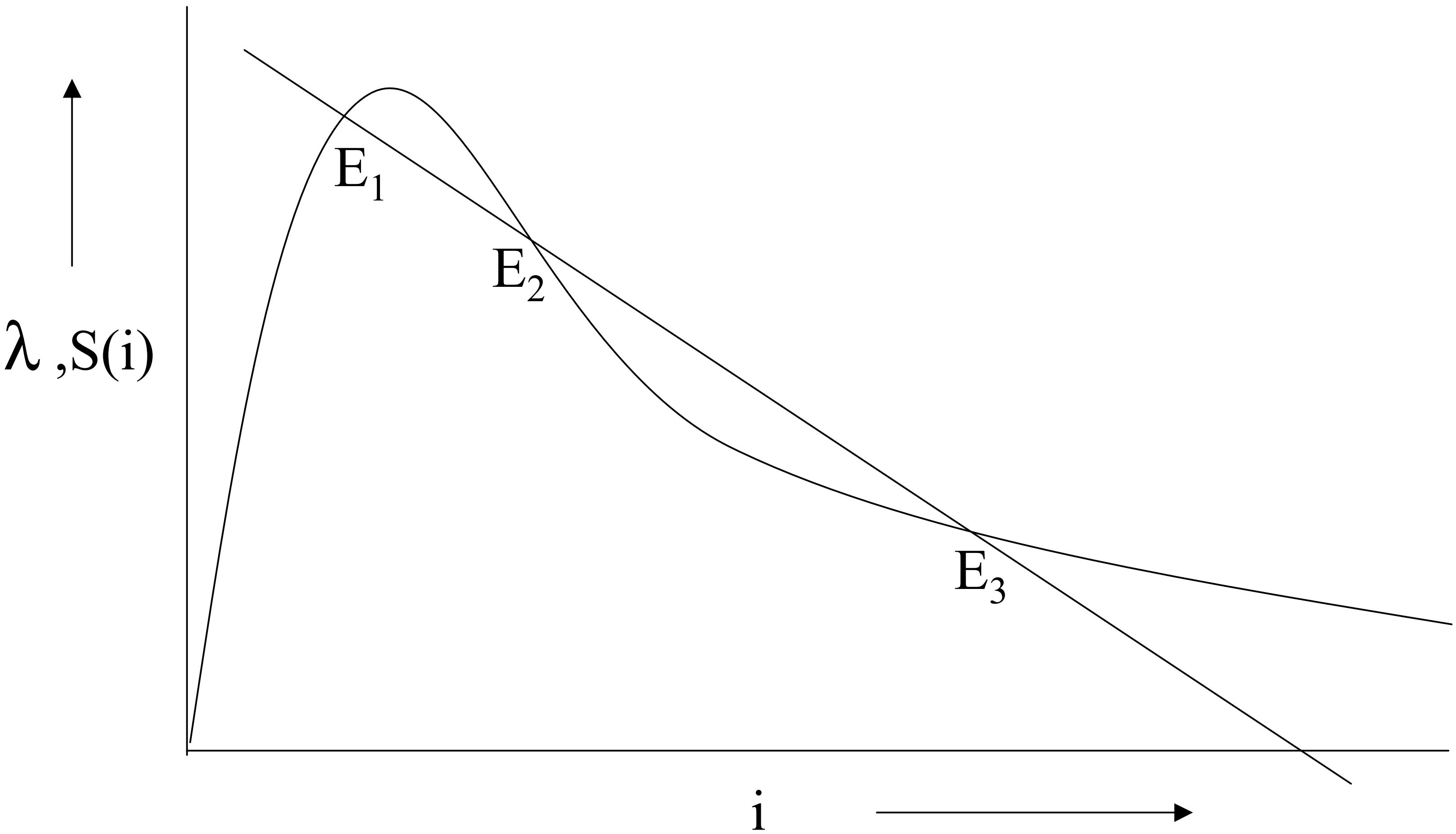
Packets enter the system at S=(N-i)p=λ rate

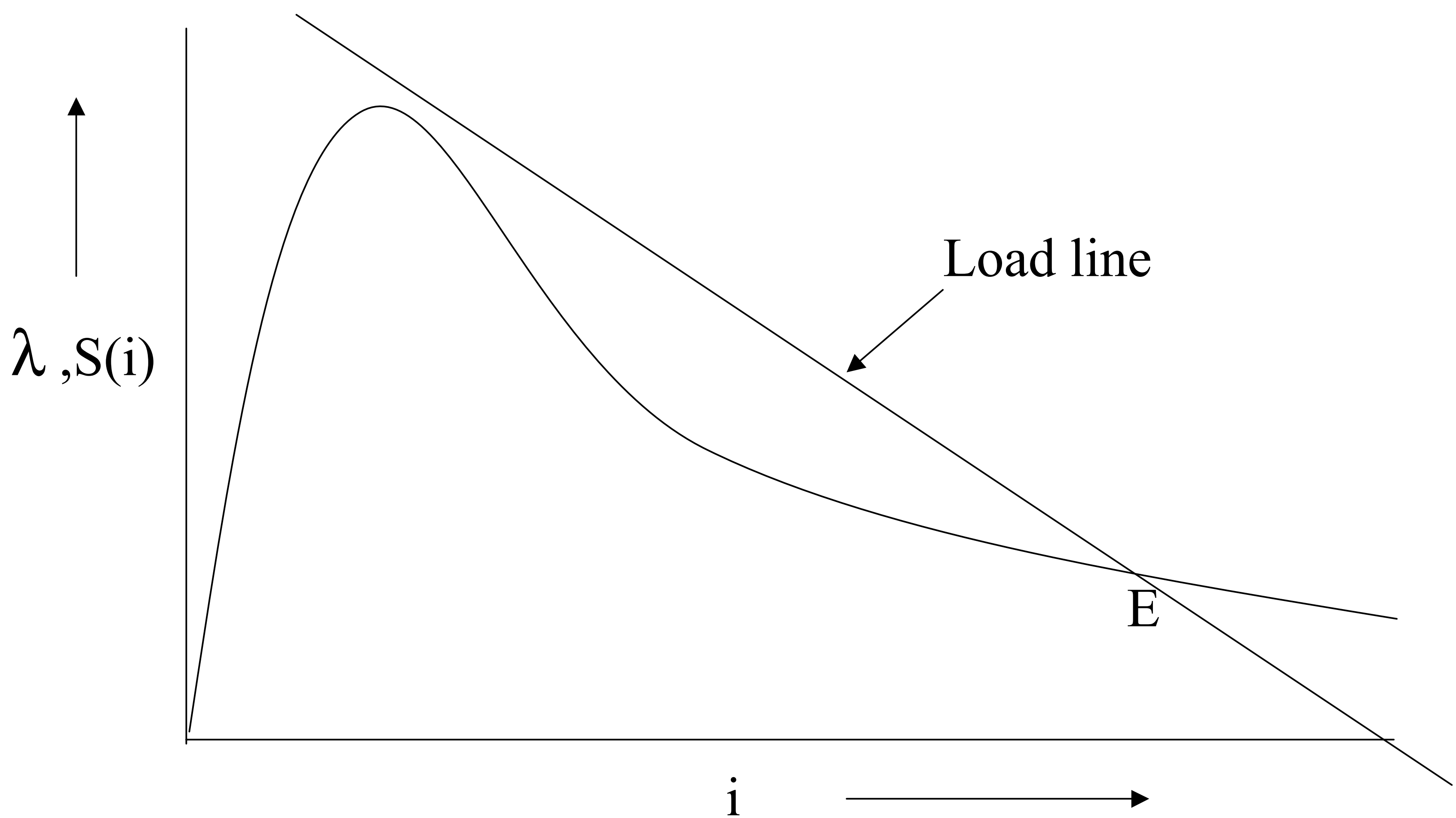
S(i) > λ ⇒ Packets are leaving at faster rate than the input

$$\lambda = (N - i)p$$

is the load line


Global Equilibrium
E
λ ,S(i)
i


λ ,S(i)
E1
E2
E3
i

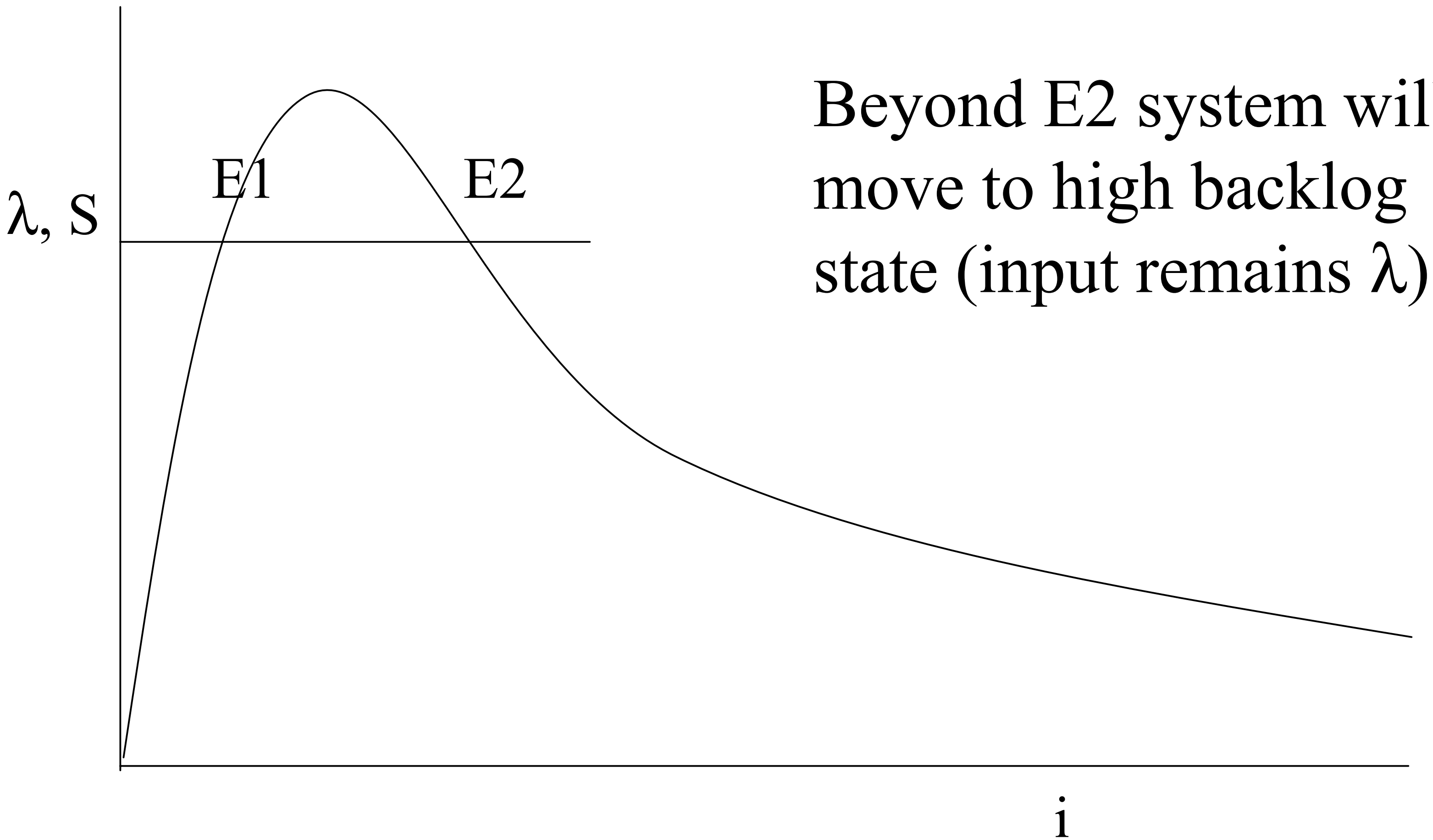

λ ,S(i)
Load line
E
i

- E_2 is unstable equilibrium
- E_1 and E_3 are stable equilibrium
- E_3 is high backlog equilibrium

High backlog equilibrium

Backlog-throughput curve is lifted up by decreasing α

For $n \to \infty$ input $= \lambda$



Beyond E2 system will move to high backlog state (input remains λ)

Looking at G-S Curve (E_1=G_1, E_2=G_2, i=G)

Around G_1 (Small backlog)

(a) λ increases ($\lambda+\Delta \lambda$) : Small variation

$\downarrow$

Packets leave faster than they arrive

$\downarrow$

G tends to reduce (Slower arrivals)

$\downarrow$

Stabilizes at G_1

(b) λ decreases (λ-$\Delta\lambda$) : Small variation

$\downarrow$

Throughput decreases

$\downarrow$

Packets leave slower than they arrive

$\downarrow$

G tends to increase (faster arrivals)

$\downarrow$

Stabilizes at G_1

Around G_2 (Large Backlog)

G exceeds G_2 (Large variation in λ)

↓

Throughput decreases

↓

Packets leave slower than they arrive

↓

Further increase in G and backlog

↓

Unstable System

CAPTURE IN ALOHA

- Packets arriving at receiver with highest energy get detected
  - Low energy packets are not deemed to be through
- Power variations can occur due to
  - Varying distance of receivers from hub
  - Fading in radio environment
- Overall system throughput increases

Demonstration of Capture (Metzner)

- Two classes of users
  - High Power Class (H), [G_H, S_H]
  - Low Power Class (L), [G_L, S_L]
- Low power packets cannot affect high power packets

$$S_H = G_H \, . \, e^{-G_H}$$

Low power packet transfer requires no transmission from high power class

$$S_L = G_L \cdot e^{-G_L} \cdot \quad e^{-G_H}$$

$$S_L(\max) = \frac{e^{-G_H}}{e}$$

at $G_L = 1$

$$= \quad \frac{1}{e}\frac{S_H}{G_H}$$

- Plot Total throughput S_T [S_L (max) + S_H] vs S_H
  - Fix G_H
  - Obtain $S_H = G_H \cdot e^{-G_H}$
  - Compute S_T

For $$G_H = 1, S_H = \frac{1}{e}, S_T = 0.503$$
$$G_H = 0.8, S_H = 0.359, S_T = 0.524$$

It shows, capture increases throughput (maximum throughput 0.53)
Throughput can be increased by increasing power based classes (Hellman): 18 classes required for $S_T = 0.9$

ANOTHER CAPTURE PHENOMENON IN FINITE POPULATION ALOHA

- Small α, $N\alpha << 1$
- Non-negligible p
- Most users backlogged (state is N or (N-1))

Then,

$$S \approx \frac{Np}{N+(N-1)p}$$

$$\overline{D} = N + \frac{(N-1)}{p}$$

- Throughput *increases* with *increasing* load (p) !!
- Delay *decreases* with *increasing* load (p)!!
- $S, \overline{D}$ independent of α

Why??

- Eventually some station tried retransmission and is through (small α).
- This station generates a fresh packet (probability p)
  - sends and is through
  - generates next packet and is through (small α), and so on until back-logged
- This station has "*captured*" the system for a while.
  - Larger p increases throughput and decreases delay

Collision Resolution

Tree (splitting) Algorithm(s) [Capatanakis, Sybakov & Mikhailov, Hayes]

- Resolve collisions of back-logged population by splitting into sub-sets.
- Splitting continues until all back-logged packets are transmitted.
- Splitting can be done on
  - "Flipping an unbiased coin" (Probabilistic).
  - Binary string identifier.
  - Time of arrival of the collided packet
- No new users

Basic Algorithm

- Collision detected in kth slot. Stations not involved in collision go to waiting mode.
- Colliding nodes split in *two* sub-sets, L and R.
- All member of L transmit in (k+1)th slot.
- If (k+1)th slot is *idle* or *successful* then members of R transmit in (k+2)th slot.
- If *collision* occurs in (k+1)th slot then L is split in LL and LR sub-sets.
- Splitting continues until collisions resolved.

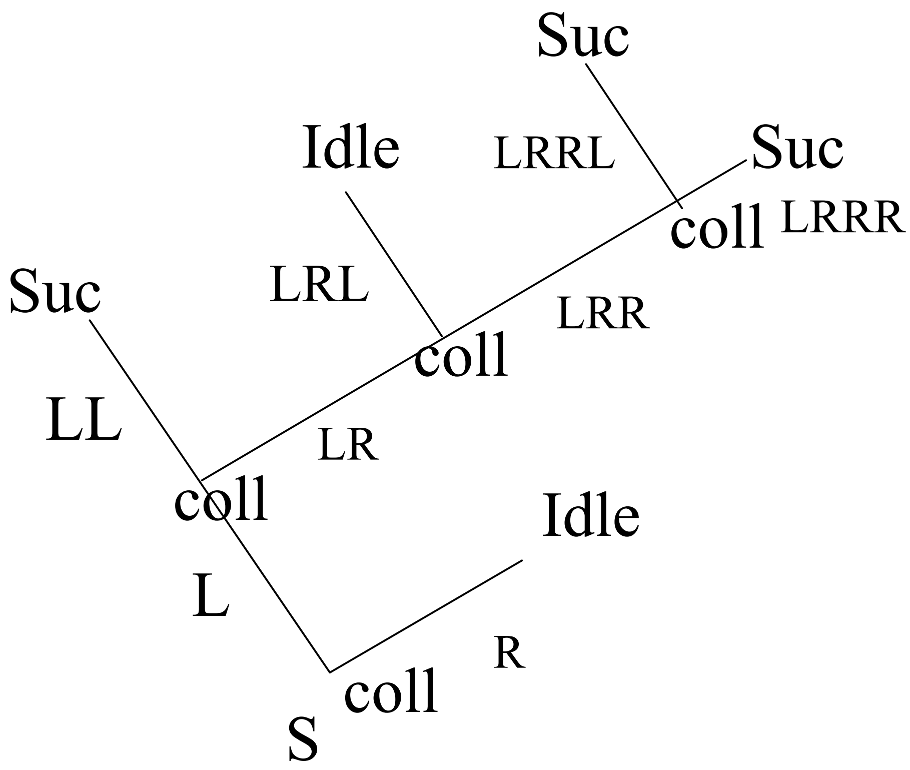
Basic Algorithm…example

Three back-logged nodes A, B and C

Collision (error)	:	e
Success	:	1
Idle	:	0
Null Set	:	*

Slot	Transmit Set	Waiting Set	Feedback
1	S(ABC)	-	e
2	L(ABC)	R(*)	e
3	LL(A)	LR(BC), R(*)	1
4	LR(BC)	R(*)	e
5	LRL(*)	LRR(BC), R(*)	0
6	LRR(BC)	R(*)	e
7	LRRL(B)	LRRR©, R(*)	1
8	LRRR©	R(*)	1
9	R(*)	-	0

Resultant Tree



Equivalent Stack Algorithm

- All nodes in transmit set are at top of the stack
- In case of collision waiting set is pushed down.
- After *Idle* or *Success* contents of stack are pushed up.

S_1 (e)	ABC -	S_2 (e)	ABC *	S_3 (1)	A BC *
S_4 (e)	BC *	S_5 (0)	* BC *	S_6	BC *
S_7 (1)	B C *	S_8 (1)	C *	S_9 (0)	*

Improvement (Massey)

In the previous algorithm

- When ‘0’ is followed by ‘e’, R set is null.
- L set transmission in the next slot will certainly generate ‘e’.
- Hence, split L set and then transmit.

In the example it would be

Slot	Transmit Set	Waiting Set	Feedback
	:		
4	LR	R	e
5	LRL(*)	LRR, R	O
6	LRRL	LRRR, R	.
	:		

- Split has taken place in slot 6.
- Split is dependent on *correct* detection of ‘o’. If incorrect, splitting will continue indefinitely.

CONTROLLED ALOHA

ALOHA can be stabilized by controlling α

- When $G > G_{Th}$, Choose $\alpha = \alpha_1$
  $G <= G_{Th}$, Choose $\alpha = \alpha_2$
  $\alpha_1 < \alpha_2$
- G_{Th} can be estimated by counting empty slots (probability is e^{-G}).
- More than one thresholds can be chosen.
- It is possible to choose optimal state dependent $\alpha(i)$.

TIME – FREQUENCY RANDOM MULTIPLE ACCESS (RMA)

- RMA sequences proposed by Wu
- Symbol is given a time-frequency code

Example

T-F matrix (3x3)

$$\begin{array}{cc} \begin{bmatrix} 0 & 1 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \end{bmatrix} & \begin{array}{c} f_2 \\ f_1 \\ f_0 \end{array} \\ \begin{array}{ccc} t_0 & t_1 & t_2 \end{array} & \end{array}$$

Binary 1 : $(3,1,2) = f_1 t_0 + f_2 t_1 + f_2 t_2$

Binary 0 : $(6,4,8) = f_0 t_0 + f_1 t_1 + f_0 t_2$

RMA

- RMA sequences are generated using finite Euclidean geometry difference sets
  - Number of time divisions = n
  - Number of frequency divisions = n^2
  - Number of symbols = n^3
  - Number of sequences = $n^2 (n^2+n+1)$
  - Minimum distance in sequences = $n-1$

RMA

- Distance in sequences = Number of symbols disagreements between a pair of sequences
  - Symbols are considered in *agreement* if they appear in *any* position of the pair of sequences.
  - Any two code sequences can have at most one symbol in common.

RMA

Collision of two packets

$\Downarrow$

Collision (overlap) of bits

$\Downarrow$

- More than one symbol match. **One symbol match is *not* a collision.**
- RMA Protocol can operate in Pure [Satija & Gupta] and slotted [Gupta & Sharma] mode

RMA

- Clash scenario in Pure RMA (n=3)

			8	e	f	(Packet 1)
		c	8	d		(Packet 2)
	6	a	b			(Packet 3)
3	6	8				(Test Packet)

Symbols 6 and 8 match

Here the clashing symbols (8) can occur at first, second or the last position

RMA

Clash scenario in slotted RMA

3	6	8	(Test Packet)
a	6	b	(Packet 1)
c	d	8	(Packet 2)

Matching symbols need to be at the same position.

RMA

- **Simulations show significant improvement over slotted ALOHA at higher loads.**
  - Unity slope upto G=2 in G-S curve   (n=3).
  - At G=10, slotted RMA (n=3) utilization is 48% against slotted ALOHA utilization of 0.045%.
  - Bandwidth utilization of Slotted RMA is (Throughput/n) (16% at G=10, n=3)
  - Higher sequence length increases the throughput.
- Complex coding and synchronization procedures.