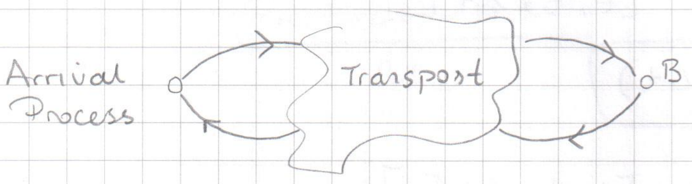


{ May be wired (including OPTICS)
{ " " wireless

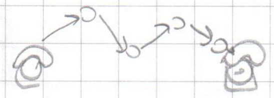
"Marconi" (Italian inventor ⇒ long distance radio transmission)
(guglielmo)
1909 ⇝ Nobel price in Physics with Karl Ferdinand Braun
in recognition of their contributions to development of wireless telegraphy"

PALME expert on telephone

ethernet / MAC ⇝ transport is shared over small # of channels →
collusion → re-transmission

Point to point → multipath + sequential ⇝ over capacity + scheduling

Hierarchical



IP Protocol | S | D | - - - |

Reliability }
Survivability } ⇒ use multipath

Question: How many ladies should be working in telephone switching inorder to ensure people waiting too much. ⇝ capacity planing

$0 \quad a_1 \; a_2 \quad a_3 \quad a_4 \longrightarrow$ time

$$\underline{\underline{A(t)}} = \sum_{i=1}^{\infty} 1 \cdot [a_i < t] \qquad 1 \cdot [X] = \begin{cases} 1 & X = \text{True} \\ 0 & X = \text{False} \end{cases}$$

\# of arrivals by time t

Probability Model

$t \qquad\qquad t+\Delta t$

Assumption $P[1 \text{ arrival in } [t, t+\Delta t[\,]$

$$= \boxed{\lambda \cdot \Delta t + o(\Delta t)}$$

$P[0 \text{ arrivals in } [t, t+\Delta t[\,]$

$$= \boxed{1 - \lambda \Delta t + o(\Delta t)}$$

* $P[n>0 \text{ arrivals in } [t, t+\Delta t[\,] = \boxed{o(\Delta t)}$

$o(\Delta t)$ is any function that goes to 0 faster than Δt

$$\lim_{\Delta t \to 0} \frac{o(\Delta t)}{\Delta t} = 0$$

λ is a positive constant.

* Assumption ⇝ Between a very little gap, probability of having more than 1 arrival is very small $(o(\Delta t))$ forget about it

$$P_n(t) \equiv \text{Prob}[A(t)=n]$$

$$P_n(t+\Delta t) = P_n(t)\,[\overbrace{1-\lambda\Delta t + o(\Delta t)}^{P_0}] + P_{n-1}(t) \cdot \overbrace{\lambda\Delta t + o(\Delta t)}^{P_1}$$

$$+ \sum_{k=0}^{n-2} P_{n-k}(t)\, o(\Delta t)$$

How P_n changes over time ?

$$\frac{P_n(t+\Delta t) - P_n(t)}{\Delta t} = \frac{\cancel{P_n(t)} - \lambda\Delta t\, P_n(t) + o(\Delta t) P_n(t) + P_{n-1}(t)\lambda\Delta t + P_{n-1}(t)\, o(\Delta t) + \sum_{k=0}^{n-2} P_{n-k}(t)\, o(\Delta t) \cancel{- P_n(t)}}{\Delta t}$$

$$= -\lambda P_n(t) + \frac{P_n(t)\, o(\Delta t)}{\Delta t} + \lambda P_{n-1}(t) + \frac{P_{n-1}(t)\, o(\Delta t)}{\Delta t} + \sum_{k=0}^{n-2} P_{n-k}(t)\, o(\Delta t)$$

$$= -\lambda P_n(t) + \frac{o(\Delta t)}{\Delta t} \sum_{k=0}^{n} P_{n-k}(t) + \lambda P_{n-1}(t)$$

FABER-CASTELL

$$\lim_{\Delta t \to 0} \frac{P_n(t+\Delta t) - P_n(t)}{\Delta t} \rightsquigarrow \frac{d}{dt} P_n(t)$$

$$n>0 \Rightarrow \frac{d}{dt} P_n(t) = -\lambda P_n(t) + \lambda P_{n-1}(t) + \cancel{\frac{o(\Delta t)}{\Delta t} \sum_{k=0}^{n} P_{n-k}(t)}$$

$$\left(\text{remember: } \lim_{\Delta t \to 0} \frac{o(\Delta t)}{\Delta t} = 0 \right)$$

$n=0$

$$P_0(t+\Delta t) = P_0(t)\left[1 - \lambda \Delta t + o(\Delta t)\right]$$

$$\lim_{\Delta t \to 0} \frac{P_0(t+\Delta t) - P_0(t)}{\Delta t} \rightsquigarrow \frac{d}{dt} P_0$$

$$\lim_{\Delta t \to 0} \frac{\cancel{P_0(t)} - \lambda \Delta t P_0(t) + P_0\, o(\Delta t) - \cancel{P_0(t)}}{\Delta t}$$

$$\lim_{\Delta t \to 0} -\lambda P_0(t) + P_0 \cdot \frac{o(\Delta t)}{\Delta t}$$

$$= -\lambda P_0(t)$$

$n=0$ $\quad \frac{d}{dt} P_0(t) = -\lambda P_0(t) \Rightarrow$ solution to this eq $\boxed{P_0(t) = e^{-\lambda t}}$

$n>0$ $\quad \frac{d}{dt} P_n(t) = -\lambda P_n(t) + \lambda P_{n-1}(t)$

$\forall n \rightsquigarrow$ countably infinite equations & hard to write!

Generating Function (Laplace bulmuş, 1779 da)

napoleon's student

$$\boxed{G(x,t) = \sum_{n=0}^{\infty} P_n(t)\, x^n} \quad \boxed{|x| \leq 1}$$

how to apply

$n > 0 \Rightarrow \quad x^n \frac{d}{dt} P_n(t) = \left[-\lambda P_n(t) + \lambda P_{n-1}(t)\right] x^n$

$n = n \qquad\qquad = -\lambda x^n P_n(t) + \lambda x^n P_{n-1}(t)$

$n = n-1 \quad x^{n-1} \frac{d}{dt} P_{n-1}(t) = -\lambda x^{n-1} \cancel{P_{n-1}(t)} + \cancel{\lambda x^{n-1} P_{n-2}(t)} \quad \lambda x^{n-2}(x-1)\; P_{n-2}(t)$

$n = 2 \qquad\qquad + \lambda x^2 P_1(t) \Rightarrow \lambda x (x-1) P_1(t)$

$n = 1 \quad x \frac{d}{dt} P_1(t) = -\lambda x \, P_1(t) + \cancel{\lambda x P_0(t)}$

$\Downarrow \; \lambda (x-1) P_0(t)$

$n = 0 \quad 1 \; \frac{d}{dt} P_0(t) = -\lambda \; \cancel{P_0(t)}$

$+$

$$\frac{d}{dt} \underbrace{\sum_{n=0}^{\infty} P_n(t) x^n}_{G(x,t)} = -\lambda x^n P_n(t) + \sum_{n=0}^{\infty} \lambda x^{n-1}(x-1) P_n(t)$$

$$-\lambda x^{n+1} P_{n+1}(t) + \sum_{n=0}^{\infty} \lambda x^n (x-1) P_n(t)$$

$$\frac{d}{dt} G(x,t) = \sum_{n=0}^{\infty} \lambda x P_n(t) x^n - \sum_{n=0}^{\infty} \lambda P_n(t) x^n$$

$$\lambda x \sum_{n=0}^{\infty} P_n(t) x^n$$

$$\boxed{\frac{d}{dt} G(x,t) = -\lambda G(x,t) + \lambda x \, G(x,t)}$$

$\Downarrow$

Solution of this eq $\Rightarrow$

$$\boxed{G(x,t) = e^{-\lambda(1-x)t}}$$

$$\sum_{n=0}^{\infty} P_n(t) x^n = e^{-\lambda(1-x)t} = e^{-\lambda t} \cdot e^{\lambda x t}$$

remember $e^a = \sum_{n=0}^{\infty} \frac{a^n}{n!}$

$$\sum_{n=0}^{\infty} P_n(t)\, x^n = e^{\lambda x t} \cdot e^{-\lambda t}$$

$$= \sum_{n=0}^{\infty} \frac{(\lambda x t)^n}{n!} \cdot e^{-\lambda t}$$

$$\sum_{n=0}^{\infty} P_n(t) = \sum_{n=0}^{\infty} \frac{(\lambda t)^n}{n!} \cdot e^{-\lambda t}$$

$$\left\{ P_n(t) = \frac{(\lambda t)^n e^{-\lambda t}}{n!} \right\}$$

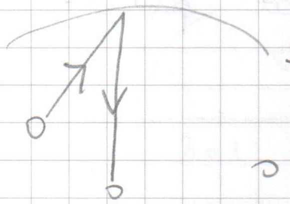
Poisson Distribution

MR. FISH :)

remember $E[N] = \sum_{n=0}^{\infty} n \cdot P_n = \alpha = \int^{t} \lambda t\, dt = \lambda$

variance $= \alpha^2$

ethernet ~1973 Hawaian Islands.



ionosphere   problem ⇝ reliability

ionosphere is not always reliable...

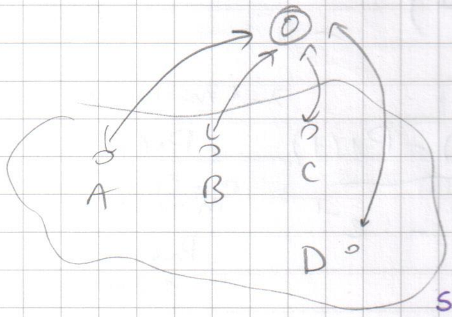
⇓

use artificial ionosphere ⇒ satellite

Whole ethernet principle comes from this technique.

(kinds:
- geo-stationary
- observation satellites.)

Satellite Cost: ↑↑↑   ALOHA SYSTEM:



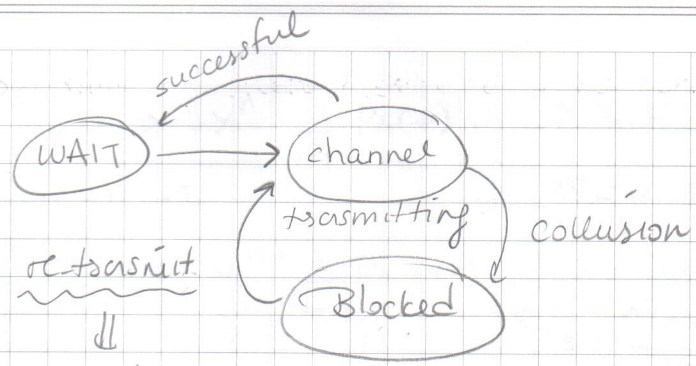
why random access?

⇓

synchronize...

shadow area

→

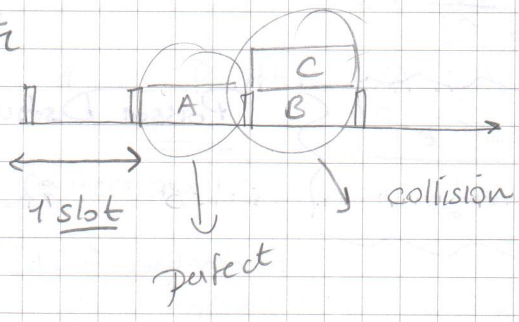


re-transmit

$\Downarrow$

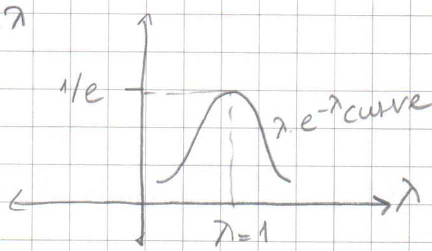
you must randomized your re-transmit time.

satellites clock



I don't know who is transmitting in the channel but I assume Poisson arrival process of transmissions with λ. what's prob. of a successful transmission?

$$P_1(1) = \frac{(\lambda 1)^1 \, e^{-\lambda . 1}}{1!} = \lambda . e^{-\lambda}$$



$$\frac{d}{d\lambda} \lambda . e^{-\lambda} = \lambda(-e^{-\lambda}) + 1 . e^{-\lambda}$$

$$\left(\frac{d}{dx} uv = \frac{d}{dx}u . v + \frac{d}{dx} v . u \right)$$

$$\left(\frac{d}{dx} e^x = e^x \right)$$

$$\left(\frac{d}{dx} e^{f(x)} = f'(x) e^{f(x)} \right)$$

$$\frac{d}{d\lambda} e^{-\lambda} = -e^{-\lambda}$$

$$e^{-\lambda}(1-\lambda) = 0$$

$$(\lambda = 1 \Rightarrow \text{makes } f \text{ maximal})$$

Prob. of collusion $\quad 1 - P_0(1) - P_1(t)$

$\lambda = 1$ maximizes $P_1(t)$ minimizes Prob of collusion.

$$(\text{if } \lambda = 1) \quad = 1 - e^{-1} - e^{-1}$$

$$= 1 - \frac{2}{e}$$

to keep system performace good (collision may occur), my traffic should have $\boxed{\lambda^* = 1}$

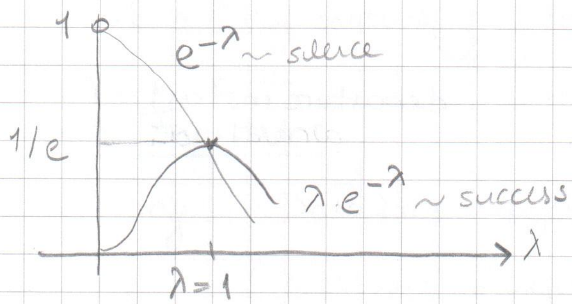
If you can estimate λ, you can change your tx rate to get near to λ^*

* why trivial? ⇒ Because of re-transmissions.

$P[\text{silence}] = e^{-\lambda}$ $\left(P_0(1) = \frac{e^{-\lambda \cdot 1} (\lambda \cdot 1)^0}{0!} = e^{-\lambda} \right)$

If I count silence slots, I may estimate λ $\lambda \approx \ln P[\text{silence}]$

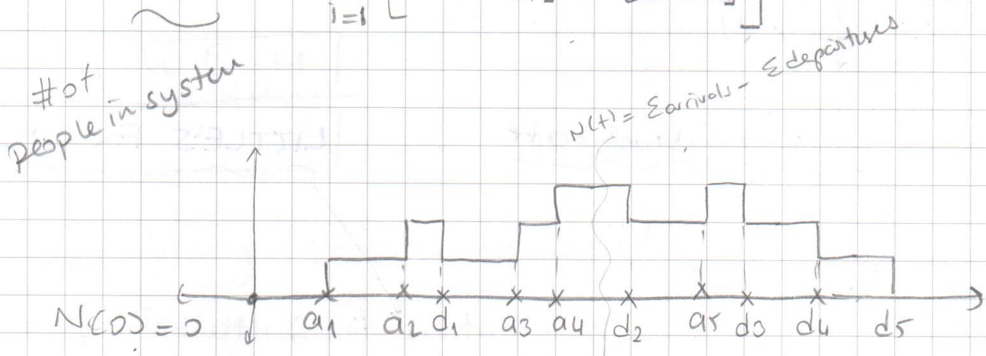
$P[\text{success}] \approx \lambda e^{-\lambda}$ ~ can we estimate from P success?
$(P_1(1))$



If $\lambda < 1$ use silence
If $\lambda > 1$ use success
} use large # of samples for good accuracy

GSM, Fransa'da IBM tarafından bulunmuş, speech recognition yapan grupta!

$$N(t) = \sum_{i=1}^{\infty} \left[1[a_i < t] - 1[d_i < t] \right]$$



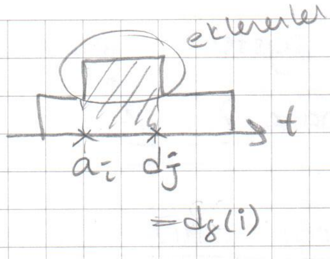
-- d 8(k)
N(t) = 0

On average how many people?

$$\boxed{\frac{\text{area}}{T} = \frac{1}{T} \int_0^t N(t)\,dt}$$

Beginning of each rectagle ⇒ arrival

End of each " ⇒ departure



$$\frac{1}{t} \sum_{i=1}^{k} d_{\delta(i)} - a_i$$

$$d_{\delta(i)} = \underbrace{\inf}_{\text{smallest}} \{ d_i : \underbrace{N(d_{\delta(i)}^{+})}_{\text{state AFTER departure}} = \underbrace{N(a_i^{-})}_{\text{state BEFORE arrival}} \}$$

question

Is $\{ d_{\delta(1)}, \ldots d_{\delta(k)} \}$ identical to
$\{ d_1, \ldots d_k \}$ AND
$\{ d_{D(1)}, \ldots d_{D(k)} \}$ } YES

$d_{D(1)}$: departure instant of arrival a_i

$$N(a_i^{-}) = \frac{1}{t} \sum_{i=1}^{k} (d_i - a_i)$$

$$\boxed{N(t) = \frac{1}{t} \sum_{i=1}^{k} (d_{D(i)} - a_i)}$$

k. average response time

$$\bar{N} = \frac{k}{t} \cdot W$$

$\bar{N}$: avg # of people

$\frac{k}{t}$: average arrival rate

$\boxed{N = \lambda W}$

LITTLE'S FORMULA

2 assumptions for Little's Formula

1) Arrivals / Departures appear at DISTINCT times

2) Start from empty state
End at an " "

(2nd week)

22.02.2013

Components Of a Network

Links (Facilities)

- cost
- capacity
- reliability

1- MTTR
MTBF

mean time to repair
mean time between failures

- availability

related to capacity & traffic
equal → fully consuming, more traffic → wait in queue...

Nodes (Devices)

- terminals
- host, switch, routers, mux...
- cost
- capacity
- reliability → * usually reliability is considered of links because usually nodes have back-ups & on any problem, you switch to back-up.
- compatibility

Network Functions

- Switching (Data Link Layer (L2))
- Routing (Network Layer (L3))
- Flow Control (ex// TCP)
- Speed & Code Conversion
- Security
- Back-up
- Failure Monitoring
- Traffic Monitoring
- Accountability (related to costs!)
- Internetworking
- Network Management

Switching Nodes

(remember)

circuit switching bağlantı boyunca kurulan yol sabit (zamana duyarlı gerçek zaman uyg. için ideal)

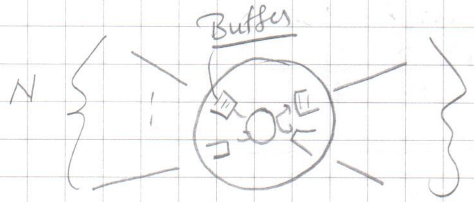
message switching "mesaj" store & forward şeklinde ilerler

packet switching smaller sized "packets" (aynı bağlantıya ait paketler başka yollarda gidebilir)

virtual circuit (circuit switch + pack) ⇝ bağlantı gelince yol belirler yoldaki kapasitenin bir kısmı ayrılır. ⇝ veri paketler halinde taşınır (tüm paketler aynı yoldan gider) ⇝ hat kapasitesi elverdiğince, pek çok sanal devre tarafından paylaşılır.

eg ATM : you reserve slots.

Delay Models in Data Networks

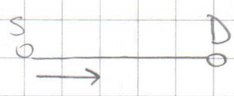


Processing delay ⇝ time spent between taking a packet from inlet & assigning it to an outlet

queuing delay ⇝ waiting for packets infront

transmission delay ⇝ time between you start transmitting 1st & last bit

Propogation delay ⇝

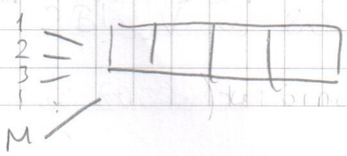


(çok hızlı bağlantılarda gözardı edilebilir)

* re-transmissions are NOT considered up to now.

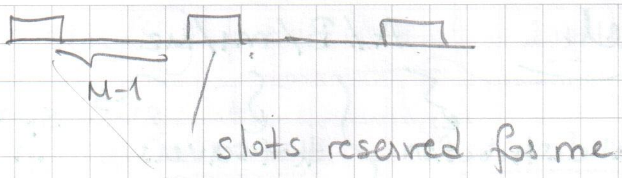
Multiplexing Techniques:

⇒ Time Division Multiplexing (TDM)



$\frac{C}{M}$

$L \cdot \frac{C}{M}$ seconds to transmit a message of size L



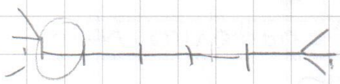
⇓

may loose some spaces but easy to implement (receiver knows where to look)

(FDM ex/ genelde 12 adet 4 KHz lik ses kanalı 60–108 KHz bandına koyulurlar.)

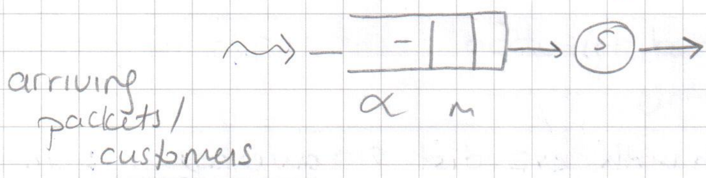
↓

Statistical Multiplexing



In every slot time, all inlets are checked in order and the one who is having data is selected BUT you need to add header to say which inlet is coming from & going to ⇒ * Causes overhead on both data & Processing but you utilize capacity better than TDM.

Modelling a Queuing System



1) Interarrival time. how frequent packets are coming
2) service time. (sadece adrese bakacaksa kısa, paket boyuna bağlı ise değişken vs olabilir)
3) # of servers
4) queuing method ex FIFO
5) buffer space usually considered as ∞ (infinitely big)

* when we say size = M it includes people being served + being waiting

not: Buffer size'ı arttırarak, paket kayıplarını azaltabilirsin ama bazen paketi atmak, çok geç göndermekle aynı şey (örn Real-Time)

G/M/1/D
M/M/1

Queing Model

A/B/m1/m2

inter-arrival distribution

service time distribution

# of servers

space in system

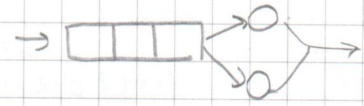
* Buffer space + # of servers

→ If buffer space is ∞, you do not specify m2

A/B/M1 means: A/B/M1/∞

A/B/M1/M2 ⇝ loss system (Because we have limited buffer space, we'll loose some if buffer is full)

→ m2 ≥ m1   A/B/2/5 ⇒



M: markov process
D: deterministic
G: general ⇝ no info

M: Markov Process: (we assume Poisson arrivals in markovian process)

→ If arrivals are represented by Poisson process, interarrival times are distributed according to exponential distribution (neg exp)

λ packets/sec in Poisson distribution

⇓

Interarrivals match with exp. dist & average interarrival duration is 1/λ seconds.

D: Deterministic: always x packets/sec

→ λ arrival > λ service ⇒ queue size will increase (increase infinitely by queue.

λ arrival < λ service ⇒ we can NOT say there will not be any queue, processes are NOT deterministic (we compare only means)

FABER-CASTELL

remember $\boxed{N = \lambda . T}$ Little's theorem (no relation with service time)

N: avg # of packets

λ: arrival rate

T: avg waiting time

---

intensity ρ of traffic on a line, between 0 – 1

0 ~ NO TRAFFIC

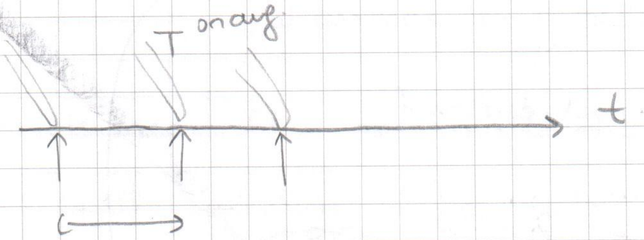
1 ~ full-capacity is used

genelde $\rho = 0.8 E$ ~ erlang

λ: mean frequency of arrivals (1/sn)

T: mean duration of transmission (sn) ~ can be considered as service time but usually size of packets are not matching with neg. exp. dist. (wrt packet size)

instead very small packets or 10K.



average duration $1/\lambda$ sec.

Consider for m packets

$(m \to \infty)$ $\rho = \dfrac{m . T}{\underbrace{m . \frac{1}{\lambda}}_{\text{total duration}}}$ ~ time period line will be busy $= \boxed{T \lambda}$ (unitless)

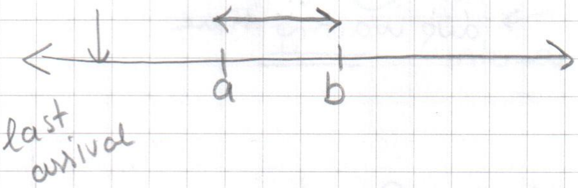
$\rho > 1 \Rightarrow$ you have transmit more than available time, not possible

---

"Markov Systems are Memoryless" $\Rightarrow$ whenever something occurs, I behave like it is happening right now, not dependent on past

Memoryless Property

~> ex// I waited up to a & saw no arrival, may I say that I have more probability to see an arrival up to b?



$$P(Y>b \mid Y>a) = \frac{P(Y>b)}{P(Y>a)} = \frac{e^{-\lambda b}}{e^{-\lambda a}} = \underbrace{e^{-\lambda(b-a)}}_{\text{related only } \underline{b-a} \text{ interval}}$$

no arrival up to b.

no arrival up to a

$$P_0(b) = \frac{(\lambda b)^0 \cdot e^{-\lambda b}}{0!} = e^{-\lambda b}$$

$$P_0(a) = \frac{(\lambda t)^0 \cdot e^{-\lambda 0}}{0!} = e^{-\lambda a}$$

n.t

$$P(Y \leq y) = 1 - e^{-\lambda y}$$

cumulative prob. density f

$$\frac{d}{dy} F_Y(y) = \frac{d\,(1-e^{-\lambda y})}{dy} = \lambda e^{-\lambda y} \quad y>0$$

pdf of neg. exp. dist

$E[y] = 1/\lambda$

variance $= (1/\lambda)^2$

Markov Models

- continious (index values are continious) continious statespace !
- discrete (CHAINs)

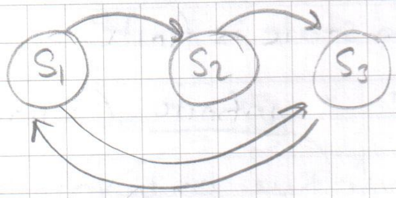
S_i: denotes state.

$$P[S_{n+1} = S_k \mid S_n = S_i, S_{n-1} = S_j \ldots S_1 = S_i]$$

$$= P[S_{n+1} = S_k \mid S_n = S_i]$$

(states need not be unique)

FABER-CASTELL



By knowing all the history, what is the probability of being at state$_k$?

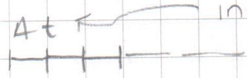
∝ intuitively, the random process is a Markov Process if its future is independent of the past when the present is known.

$$P\left[S_{n+1} = s_k \mid S_n = s_i\right]$$

(we will model # of people in system as states)

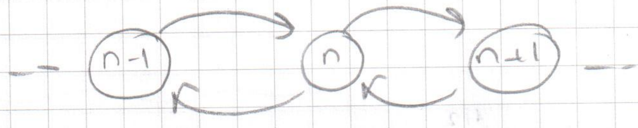
Birth – Death Processes (special markov chain systems)

memorylessness ⇝ further the next state probabilities to be non-zero only to nearest neighbor state.

(

in this small Δt duration, we assume we have at most 1 arrival & 1 departure at most.

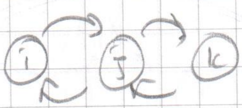
2 arrivals mean Δt is not small enough :))



i.e: $$|k-j| > 1 \Rightarrow P\left[S_j \mid S_k\right] = 0$$

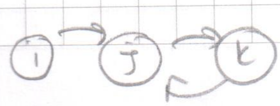
State Classification

•) Recurrent state: If the system once in that state, will return to that state through series of transitions with prob. 1.

ex


state$_i$: recurrent

•) Transient not recurrent



state$_i$: recurrent.

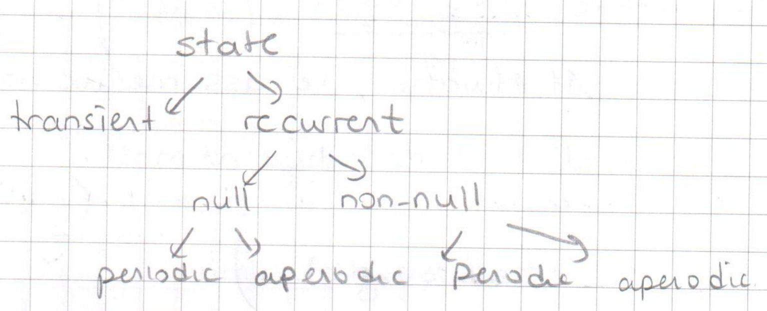
FABER-CASTELL

- **Recurrent Non-null** mean time to return to state is finite.
- **Recurrent Null** " " " " " infinite (infinite sized waiting room ⇒ infinite # of states ⇒ return time can be infinite)
- **Aperiodic** A recurrent state is aperiodic if for some number k, there is a way to return to state at k, k+1, k+2, ... ∞ transitions.

SELF LOOPS always make a state aperiodic

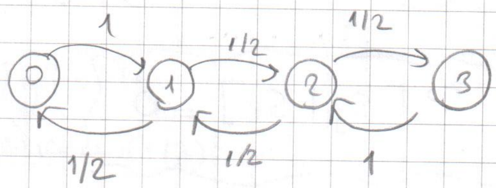
k=1 k=2,3,4,...

- **Periodic** a recurrent state is periodic if it is NOT aperiodic. It must have a period p>1 p•k passes return to state k=1,2,...



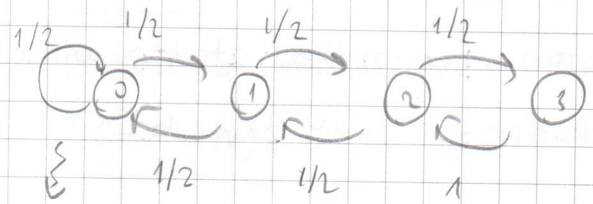
* Tum stateler aynı özellikteise, Markov Chain de o özellikte olur. ex// recurrent MC.

ex//



all states
- recurrent
- non-null
- periodic (p=2)

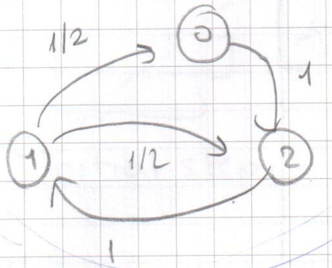
ex//



made all states aperiodic

* Having no self-loops is not enough to be periodic

ex



state0 periodic

state1 aperiodic 1-0-2-1 $p=2$

1-2-1 $p=1$

for $k=1$ $k+1, k+2, k+3, \ldots$

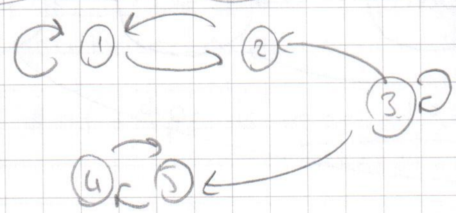
state2 2-1-0-2 (same) aperiodic

2-1-2

•) IRREDUCABLE Markov Chain all states are reachable from all other states.

---

ex



state1: recurrent, non-null, aperiodic

state2: recurrent, non-null, periodic $p=2$

state3: transient: once leave 3, you may not come back

state4,5 recurrent, non-null, periodic

not an irreducable MC

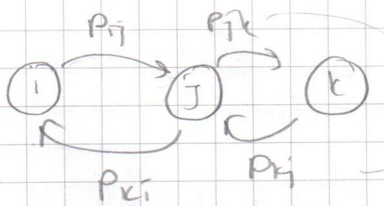
•) ERGODIC Markov Chain

⇒ irreducable
⇒ recurrent non-null
⇒ aperiodic

} ERGODIC

Theorem

If MC is ergodic, there exists a unique distribution for being in state k denoted as π_k independent of initial state. These probabilities are called steady state / equilibrium probabilities.

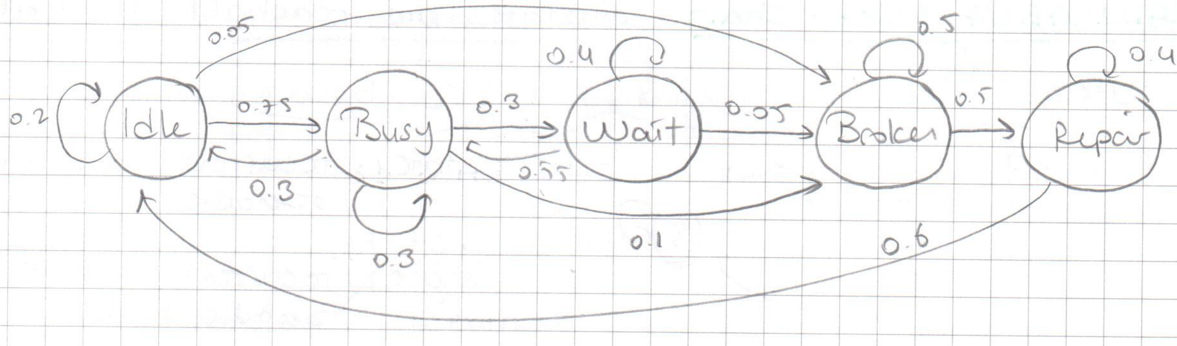


Bu olasılıklar da önemli ama göstermeyi için asıl önemli soru π_i

ex

States

	Idle	Busy	Wait	Broken	Repair
Idle	0.2	0.75	—	0.05	—
Busy	0.3	0.3	0.3	0.1	—
Wait	—	0.55	0.4	0.05	—
Broken	—	—	—	0.5	0.5
Repair	0.6	—	—	—	0.4



- irreducible (all states reachable from each other)
- recurrent non-null
- aperiodic

Idle : k=1 Idle k=2 Idle Busy k=3 ...
Busy k= 2, 3, ...
Wait k= 4, 5, ...
Broken k=1
Repair k=1 ...

Single step transition probability matrix $P = [P_{jk}]$

$$P = \begin{array}{c} \\ \text{idle} \\ \text{busy} \\ \text{wait} \\ \text{broken} \\ \text{repair} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{idle} & \text{busy} & \text{wait} & \text{broken} & \text{repair} \end{array} \\ \begin{bmatrix} 0.2 & 0.75 & 0 & 0.05 & 0 \\ 0.3 & 0.3 & 0.3 & 0.1 & 0 \\ 0 & 0.55 & 0.4 & 0.05 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 \\ 0.6 & 0 & 0 & 0 & 0.4 \end{bmatrix} \end{array}$$

$$\underbrace{[0\ 1\ 0\ 0\ 0]}_{\text{You are in BUSY state}} \cdot P = \underbrace{[-\ -\ -\ -\ -]}_{\text{prob of your next state}}$$

What happens after infinite steps later?

$\underbrace{[\,-\,-\,-\,-\,-\,]}_{\text{initial step}} \cdot P^{\infty}$ $\qquad P^{\infty} = \left[\begin{array}{ccccc} - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \\ - & - & - & - & - \end{array}\right]$

all same

(steady state probs)

$\underbrace{[\pi_1 \; \pi_2 \; \ldots \; \pi_m]}_{\text{steady state probs.}} \cdot P = [\pi_1 \; \pi_2 \; \ldots \; \pi_M]$

$$\left(\lim_{n \to \infty} p(n)\, P = \lim_{n \to \infty} p(n+1) \right)$$

$$\boxed{\pi \cdot P = \pi}$$

" If you have an ergodic chain, being at a specific state after large # of steps is irrespective of initial state "

$$\pi\, P = \pi$$

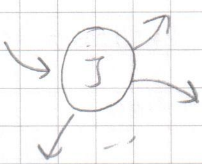
$$[\pi_1 \; \pi_2 \; \ldots \; \pi_n] \begin{bmatrix} \\ n \times n \\ \\ \end{bmatrix} = [\pi_1 \; \pi_2 \; \ldots \; \pi_n]$$

n unknowns, n equations (dependent)

how to solve? $\sum_{i=1}^{n} \pi_i = 1$ 1 independent eq...

→ $\boxed{\sum_{i=1}^{n} P_i \cdot i}$ (π_i) ⇝ average # of people for queuing system!

Flow Conservation Equality



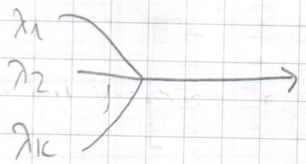
$$\boxed{\sum_{k=0}^{n} P_j \cdot P_{jk} = \sum_{k=0}^{n} P_k \cdot P_{kj}}$$

(If not equal chain may be transient)

notlar

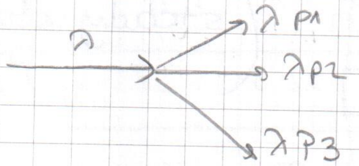
Poisson Process

a)



$\lambda = \sum_{i=1}^{k} \lambda_k$ resulting flow is another Poisson process with λ

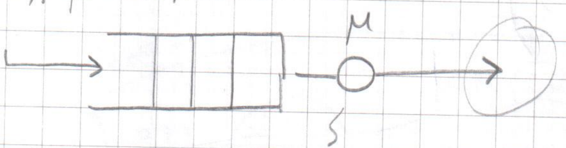
b)



You distribute randomly $P_1 + P_2 + P_3 = 1$

all lines new Poisson with $\lambda' = \lambda . P_i$

c) λ packets/sec

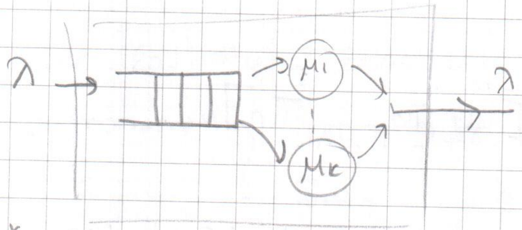


Buradaki proses poisson with λ

service rate: μ packets/sec

(Buffer surekli dolmasin diye $\lambda < \mu$ yaparsin)

d)

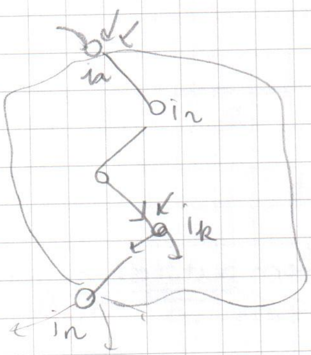


Departure Poisson with λ

$\left(\lambda < \sum_{i=1}^{k} \mu_i\right)$

week #3
01.03.2013

(S, D)



$\pi(S,D) = \{ i_1, \ldots i_n \}$

$i_1 \Downarrow S$, $i_n \Downarrow D$

During communication, path may change

$\lambda(S,D)$ rate packets / unit time

λ_{i_k} total input traffic rate to node i_k

$$\lambda_{i_k} = \sum_{(S,D)} \lambda(S,D) \cdot 1\left[i_k \in \pi(S,D) \right]$$

Node Delay
Packet Loss } impact of buffer size & processing speed

path change $\rightsquigarrow$ order of packets problem

packet ℓ $\pi(S,D)$
packet $\ell+1$ $\pi'(S,D)$ } If delay is shorter on 2nd path, older packet may arrive later than newer one

Desequencing (disorder of packets) is one of main reasons for packet loss. Receivers has finite-sized re-sequencing buffer (that keeps packet until predecessor comes.

$\Rightarrow$ Desequencing $\to$ because of path change
$\searrow$ another reason?? $\rightsquigarrow$ Routers have interrupt driven OS, that may mix the order of packets

every router work with interrupt driven handling

Fraction of packets that leave node i to other node j

$$\phi_{ij} = \frac{\sum_{\ell=1}^{n-1} \sum_{S,D} \lambda(S,D)\, 1\left[i_\ell = i \in \pi(S,D),\ i_{\ell+1} = j \in \pi(S,D) \right]}{\lambda_i}$$

(n: path length)

FABER-CASTELL

(we are looking at all paths length = 1, 2, .. n-1

in position = i

out position = $\bar{j} = i+1$ $\left(\begin{array}{l} i_\ell = \bar{i} \\ i_{\ell+1} = \bar{j} \end{array} \right)$

for all positions of ℓ

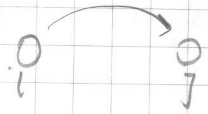
For any node i, let γ_i (gamma) be traffic arrival rate from outside network

$$\boxed{\lambda_i = \gamma_i + \sum_{\text{all } j} \lambda_j \phi_{ji}}$$

(nodes do not loose traffic)
no loss

$$\lambda_i = \gamma_i + \sum_{\bar{j}} (1 - L_{\bar{j}}) \lambda_{\bar{j}} \phi_{\bar{j}i}$$

$L_{\bar{j}}$: fraction of traffic that enters $\bar{j}$ and is lost



speed at which node forwards packet to another one? ⇒ CPU of the node determines speed.

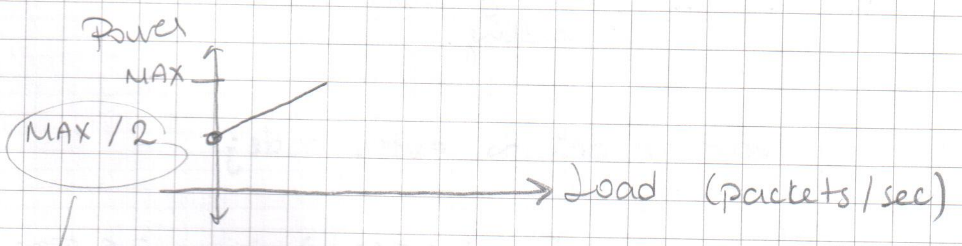
Routing Table

S D Nexthop

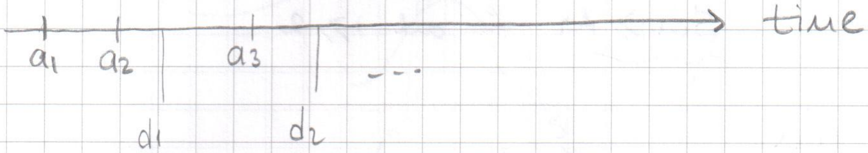
we keep routing table in software (because can be updated) Most recently used are in cache.

μ_i: rate at which node processes packets

cost & power consumption makes it impractical to use very high end routers everywhere.



even router is not doing anything, it consumes energy



$A_{D(i)}$ ~ Departure instace of a packet arriving at a_i

– Assume that $A_{D(i)} = d_i$ (FCFS a.k.a FIFO)

$d_i - a_i \rightarrow$ **response time** for i^{th} arrival

$$d_i - a_i = \underbrace{W_i}_{\text{waiting-time}} + \underbrace{S_i}_{\text{service-time}}$$

$$W_i = \begin{cases} 0 & \text{if } d_{i-1} < a_i \\ d_{i-1} - a_i & \text{otherwise} \end{cases}$$

$$d_{i-1} - a_i = \underbrace{a_{i-1} + W_{i-1} + S_{i-1}}_{d_{i-1}} - a_i$$

$$W_i = \begin{cases} 0 & \text{if } d_{i-1} < a_i \\ a_{i-1} + W_{i-1} + S_{i-1} - a_i & \text{otherwise} \end{cases}$$

$$W_{i-1} + S_{i-1} - \underbrace{(a_i - a_{i-1})}_{\text{interarrival time}}$$

$$\boxed{W_i = [W_{i-1} + S_{i-1} - A_i]^+}$$ LINDLEY's equation

(FIFO holds for Lindley's Eq)

$$[X]^+ = \begin{cases} 0 & \text{if } X < 0 \\ X & \text{if } X \geq 0 \end{cases}$$

$W_0 \to W_1 \to - W_2$

$W_1 = 0$ Assume that $S_{i-1} \geq A_i$ for all $i \geq 2$

$$\left(\begin{array}{l} A_1 = a_1 \\ A_2 = a_2 - a_1 \\ A_3 = a_3 - a_2 \\ \quad \vdots \end{array} \right)$$

Then $W_i = \sum_{l=2}^{i} (S_{l-1} - A_l)$

$w_1 = 0$

$w_2 = w_1 + S_1 - A_2 = S_1 - A_2$

$w_3 = w_2 + S_2 - A_3 = S_1 - A_2 + S_2 - A_3$

$\vdots$

$w_i = (S_1 + S_2 + \dots S_{i-1}) - (A_2 + A_3 + \dots A_i) = \sum_{l=2}^{i} S_{l-1} - A_l$

$S_{i-1} \geq A_i \;\; \forall \; i \geq 2 \Rightarrow$

$w_i \to +\infty$

Let $\xi_i = (S_{i-1} - A_i)$ Assume ξ_i are independent & identically distributed random variables.

Using central limit theorem

(as n gets larger difference between S_n (sample average) and its limit μ $\sqrt{n}(S_n - \mu)$ approximates normal dist with mean=0, variance σ^2

For large n, dist of S_n is normal dist with μ & variance $\frac{\sigma^2}{n}$)

$w_i = \sum_{l=2}^{i} \xi_l$

sample average of ξ

$\hookrightarrow \frac{w_i}{i}$

$\Downarrow$ is a normal distributed random variable.

$w_i \geq w_{i-1} + S_{i-1} - A_i$

$w_i \geq w_{i-1} + \xi_i$

if you know asymphotics of this, you know lower bound on w_i

$w_{i-1} \geq w_{i-2} + \xi_i$

$w_2 \geq \overset{=0}{w_1} + \xi_2$

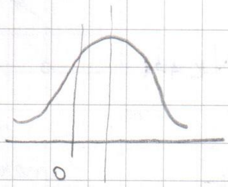
$\to$ $w_2 \geq \xi_2$

$w_3 \geq w_2 + \xi_3$

$\xi_2 + \xi_3$

FABER-CASTELL

$W_i \geq \sum_{l=2}^{i} \xi_i$ will have a Gaussian Distribution



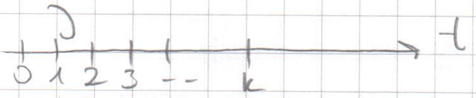
remember if $\xi_{i-1} > A_i \Rightarrow W_i \to \infty$

Stable System

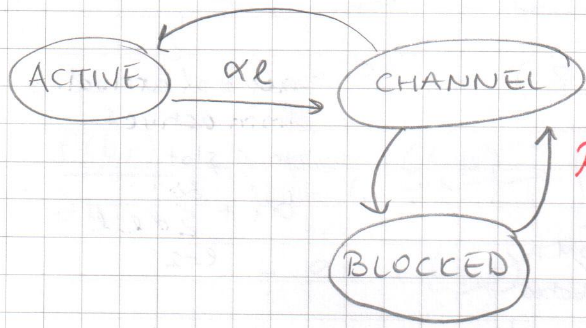
$$W = [W + \xi_{i-1} + A_i]^+$$

$$\left(W = \lim_{i\to\infty} W_i\right)$$

Random Access ALOHA

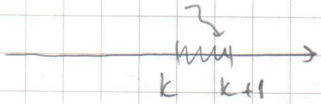


α_l: probability that l active users transmit at one slot.



β_l ~ prob that l blocked users transmit at 1 slot

N_k: # of blocked users just after k^{th} slot



previously blocked

increases blocked # by 1

$N_k = N_{k-1} + 1$ wp $\beta_1 \alpha_0$ +

$N_{k+1} + 0$ wp $\underbrace{\beta_0 \alpha_0}_{\text{no blocked, no active transmits}} + \underbrace{(1-\beta_0-\beta_1)}_{\text{2 or more blocked are transmitting \& they collide}}\alpha_0$

$+ \underbrace{\alpha_1 \beta_0}_{\text{1 active transmits}}$

wp $\Rightarrow \alpha_1 (1-\beta_0)$

exactly 1 active transmits

1 or more blocked transmits

1 active comes, 1^+ blocked comes

⇓

N_{k-1} becomes N_{k+1} ?

0) $N_k = N_{k+1}$ (no blocking at slot k) or some of blocked gets transmitting but gets blocked again

$$\alpha_0\beta_0 + \underbrace{(1-\beta_0\beta_1)}_{?}\alpha_0 + \alpha_1\beta_0$$

$$\cancel{\alpha_0\beta_0} + \alpha_0 - \cancel{\alpha_0\beta_0} - \alpha_0\beta_1 + \alpha_1\beta_0$$

$$\alpha_1\beta_0 + (1-\beta_1)\alpha_0$$

0) $N_k = N_{k+l}$ $l \geq 2$

$$E[N_k - N_{k-1} \mid N_{k-1} = x]$$

$$= \underbrace{-1\,\alpha_0\beta_1}_{\text{1 blocked gets active \& successful}} + \underbrace{\alpha_1 - \alpha_1\beta_0}_{\alpha_1(1-\beta_0)} + \sum_{l=2}^{\infty}\alpha_l \cdot l - \underbrace{\bar{A}}_{\text{avg \# of arrivals from actives in 1 slot}} - \underbrace{[\alpha_0\beta_1 + \alpha_1\beta_0]}_{\text{success prob.}}$$

α_1: 1 active tries to transmit; $(1-\beta_0)$: ≥ 1 blocked transmits → gets blocked

$\bar{A}$: $\alpha_1 + \sum_{l=2}^{\infty}\alpha_l \cdot l$

must have $\bar{A} < \underbrace{\alpha_0\beta_1 + \alpha_1\beta_0}_{\text{part of arrival rate}}$?

βs are dependent on m

ex $\begin{pmatrix} \text{head prob.} & b \\ \text{tail prob} & 1-b \end{pmatrix}$ $\quad B_\ell = \binom{m}{\ell} b^\ell (1-b)^{m-\ell}$

$$\bar{A} < \alpha_0 \underbrace{m\, b(1-b)^{m-1}}_{B_1} + \alpha_1 \underbrace{(1-b)^m}_{B_0}$$

$$a_1 + \sum_{\ell=2}^{\infty} \ell a_\ell < \alpha_0\, m\, b(1-b)^{m-1} + \alpha_1 (1-b)^m$$

how should I choose b? (Suppose we cannot do anything about arrivals, they just appear)

$$\bar{A} < \alpha_0 B_1 + \alpha_1 B_0$$

find b for given n that maximizes $\alpha_0 B_1 + \alpha_1 B_0$

$$\bar{A} < \underbrace{\alpha_0\, n\, b(1-b)^{n-1} + \alpha_1 (1-b)^n}_{f(b,n)}$$

$$\frac{\partial f(b,n)}{\partial b} = \alpha_0 n \underbrace{(1-b)^{n-1}}_{(1-b)} - \alpha_0 n (n-1) b \underbrace{(1-b)^{n-2}} - \alpha_1 n \underbrace{(1-b)^{n-1}}$$

$$= 0$$

$$\alpha_0 n (1-b) = \alpha_0 n (n-1) b + \alpha_1 n (1-b)$$

$$\alpha_0 n - \alpha_0 n b = \alpha_0 n^2 b - \alpha_0 n b + \alpha_1 n - \alpha_1 n b$$

$$\cancel{n}(\alpha_1 + \alpha_0) = \cancel{n} b (\alpha_0 + \alpha_0 n + \alpha_0 - \alpha_1)$$

$$b^* = \frac{\alpha_1 + \alpha_0}{\alpha_0 (n+1) + \alpha_0 - \alpha_1} = \frac{\alpha_1 + \alpha_0}{1 + \frac{\alpha_0}{\alpha_0 - \alpha_1}(n-1)}$$

optimum b
maximize f

(2^{nd} derivative should be positive)

4th week Back...

is $\alpha_0 - \alpha_1$ positive?

$\bar{A} \leq 1$ because we have 1sbt

$\alpha_1 + 2\alpha_2 + \dots \leq 1$

$\alpha_1 + 2\alpha_2 + \dots \leq \alpha_0 + \alpha_1 \dots$

$$\alpha_0 \geq \sum_{\ell=2}^{\infty} (\ell-1)\alpha_\ell \geq \sum_{\ell=2}^{\infty} \alpha_\ell$$

$$\geq (1 - \alpha_0 - \alpha_1)$$

?

How about α's? T terminals: n are blocked T-n are active

$$\alpha_\ell = \binom{T-n}{\ell} a^\ell (\ell - a)^{T-n-\ell}$$

$$f(a,b,n) = \alpha_0 \, m \, b(1-b)^{n-1} + \alpha_1 (1-b)^M$$

$$\Downarrow$$

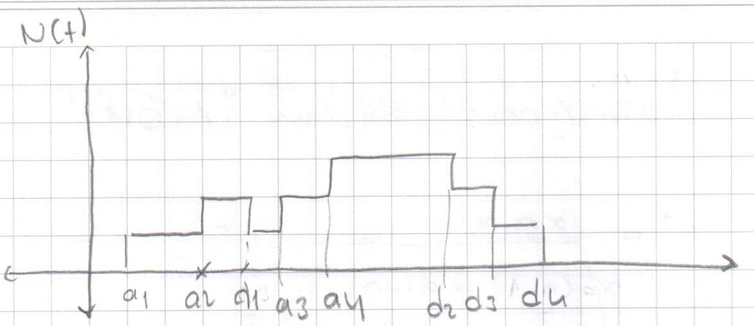
$$(1-a)^{T-n} \cdot m b (1-b)^{n-1} + (T-n) a (1-a)^{T-n-1} (1-b)^M$$

$$f(a,b,n) = \left(\frac{1-b}{1-a}\right)^n \frac{nb(1-a)^T}{(1-b)} + \left(\frac{1-b}{1-a}\right)^n (T-m) a (1-a)^{T-1}$$

how to choose a & b to optimize?

a^* b^*

?



$\pi(m)$ proportion of time $N(t)=m$

$T(m)$ total time $N(t)=m$

$A(m)$ # of arrivals to level m

$D(m)$ " " departures from level m

$\boxed{A(n) = D(n+1)}$ because system starts / ends at $N(t)=0$

T: total time

$$\pi(n) = \frac{T(n)}{T}$$

$$\frac{A(n)}{T(n)} = \frac{A(n)}{\pi(n) . T}$$

$$\frac{\pi(n+1)}{\pi(n)} = \frac{\lambda_{n+1}}{\mu_{n+1}} = \frac{A(n)}{T(n)} \cdot \frac{T(n+1)}{D(n+1)}$$

what we want to proove

$$\frac{\pi(n+1)}{\pi(n)} = \underbrace{\frac{A(n)}{T(n)}}_{\lambda_n} \Big/ \underbrace{\frac{D(n+1)}{T(n+1)}}_{\mu_{n+1}}$$

ratio of prob of time spent in levels

arrival rate to level n

departure rate from level n+1

$$\frac{\frac{T(n+1)}{\cancel{T}}}{\frac{T(n)}{\cancel{T}}} = \frac{A(n)\ T(n+1)}{D(n+1) \cdot T(n)}$$

$= 1$

4th week

08.03.2013

remember

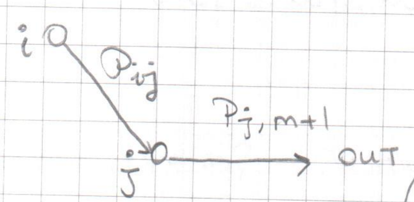
$\bar{A} < \alpha_0 B_1 + \alpha_1 B_0$ Random Access System "ALOHA"

<u>Stability Condition</u> $b^* = \dfrac{\alpha_0 - \alpha_1}{\alpha_0(n-1) + \alpha_1 - \alpha_2}$

is $\alpha_0 - \alpha_1 > 0$

$\alpha_1 \leq \underbrace{\alpha_1 + 2\alpha_2 + 3\alpha_3 + \dots}_{\bar{A}} \leq \alpha_0 B_1 + \alpha_1 B_0 < \alpha_0 B_1 + \alpha_1 (1 - B_1)$

$\alpha_1 B_1 \leq \alpha_0 B_1 \Rightarrow \alpha_0 \geq \alpha_1$



packets/time

λ_i : total traffic entering network through node i from <u>outside the network</u>

Λ_i : <u>total traffic entering node</u>

Assuming no loss $\hookrightarrow$ $\boxed{\Lambda_i = \lambda_i + \sum_{j=1}^{n} \Lambda_j P_{ji}}$

μ_i: the rate at which a node processes packets assuming FIFO $1/\mu_i$ average processing time at node i.

nodes: $1 \dots m$ $m+1^{th}$ node $\Rightarrow$ <u>outside</u>

$\{\lambda_i\}$ are rates of m independent Poisson processes

$\{\mu_i\}$ are rates of exponential random variables

At queue i, there is a sequence of service times $S_1^i > S_2^i > S_3^i \dots$

$\boxed{P[S_\ell^i > x] = e^{-\mu_i x}}$ exponential distribution

$P[S > y+x \mid S > x]$ for $y \geq 0$

$$= \frac{P[S > y+x \ \& \ S > x]}{P[S > x]}$$

MEMORYLESS property

$$= \frac{P[S > y+x]}{P[S > x]}$$

$K(t) = (k_1(t), \dots k_n(t))$

$k_i(t) \Rightarrow$ # of packets at node i at time t

$k_i(t) \Rightarrow \{0, 1, 2, \dots\}$

$p(K, t) = \text{Prob}[K(t) = (k_1, k_2 \dots k_M)]$

Define

$$K_i^+ = \{k_1, \dots k_i + 1, k_{i+1}, \dots k_n)$$

$$K_i^- = \{k_1, \dots k_i - 1, k_{i+1}, \dots k_n) \quad k_i > 0$$

$$K_{ij}^{+-} = \{k_1, \dots k_i + 1, \dots, k_j - 1, \dots k_M) \quad k_j > 0$$

Chapman- Kolmogorov Equations

(SISMIC waves)

$$P(K, t + \Delta t) = \sum_{i=1}^{n} P(K_i^-, t)[\lambda_i \Delta t + o(\Delta t)] 1[k_i > 0]$$

arrival rate at i^{th} poisson process

$(k_1, \dots k_n)$

$(k_1 \dots k_i - 1, \dots k_n)$

Before a single arrival you have 1 less at i^{th} position

1 arrival at Δt

$e^{-\lambda_i \Delta t}$ does not occur

$\approx 1 - \lambda_i \Delta t + o(\Delta t)$

does occur $\lambda_i \Delta t$

$$+ \sum_{i=1}^{n} P(K_i^+, t)[\mu_i \Delta t + o(\Delta t)$$

$P_{i,r}$

outside

from one queue to another.

$$+ \sum_{i,j=1}^{n} P(k_{ij}^{+-}, t)\,[\mu_i \Delta t + o(\Delta t)]\,P_{ij}\; 1[k_j>0] \;+$$

$$+ \; p(k,t) \prod_{i=1}^{n} [1 - \lambda_i \Delta t + o(\Delta t)]\,[1 - \mu_i \Delta t + o(\Delta t)]\; 1[k_i>0].$$

case nothing (no arrival / no departure) happens in Δt

$$\Big(\prod_{i=1}^{n} [1 - \lambda_i \Delta t + o(\Delta t)]\,[1 - \mu_i \Delta t + o(\Delta t)]$$

$$= 1 - \sum_{i=1}^{n} (\lambda_i \Delta t + \mu_i \Delta t) + o(\Delta t)$$

* Departures can ONLY occur if queue is NOT empty

$$= 1 - \sum_{i=1}^{n} \lambda_i \Delta t + \mu_i \Delta t \; 1[k_i>0] + o(\Delta t)$$

$$p(k, t+\Delta t) - p(k,t) = \sum_{i=1}^{n} P(k_i^{-}, t)\,\lambda_i \cancel{\Delta t}\; 1[k_i>0]$$

$$+ \sum_{i=1}^{n} P(k_i^{+}, t)\,\mu_i \cancel{\Delta t}$$

$$+ \sum_{\substack{i,j \\ (i\neq j)}}^{n} P(k_{ij}^{+-}, t)\,\mu_i \cancel{\Delta t}\, P_{ij}\; 1[k_j>0]$$

$$- \sum_{i=1}^{n} p(k,t)\,[\lambda_i \cancel{\Delta t} + \mu_i \Delta t\; 1[k_i>0]]$$

$$+ \frac{o(\Delta t)}{\Delta t}$$

$(p_{ii} = 0)$

*

If we divide by Δt —

$$\Big(\text{remember } \lim_{\Delta t \to 0} \frac{o(\Delta t)}{\Delta t} = 0 \Big)$$

$$\frac{d}{dt} p(k,t) \Big|_{\lim \Delta t \to 0} = (*)$$

$$= \sum_{i=1}^{n} P(k_i^-, t) \lambda_i \, 1[k_i > 0]$$

$$+ \sum_{i=1}^{n} P(k_i^+, t) \mu_i \, p_{i,n+1} \quad ?$$

$$+ \sum_{\substack{i,j \\ i \neq j}}^{n} P(k_{ij}^{+-}, t) \mu_i \, p_{ij} \, 1[k_j > 0]$$

$$- \sum_{i=1}^{n} p(k,t) \left[\lambda_i + \mu_i \, 1[k_i > 0] \right]$$

infinitely many equations to solve $\frac{d}{dt}$ ...

Let η be total numbers of packets the system may contain.

| 1 | 1 | 1 | .. | 1 | 0 | 1 | .. | 1 | .. | Binary vector.

$\underbrace{\qquad}_{k_1}$ $\underbrace{\qquad}_{k_2}$

0's to separate

$$k_1 + k_2 + \ldots k_n = M$$

$$\text{Binary vector's length} = k_1 + k_2 + \ldots + k_n + \underbrace{(n-1)}_{\text{0s.}}$$

$$= M + n - 1$$

- for every $k_1 k_2 .. k_M$, a unique vector

$$\binom{M+(n-1)}{(n-1)} \rightsquigarrow \frac{(M+n-1)!}{(n-1)! \, M!}$$

n-1 tane 0'ın yeri değişebilir

$$\sum_{M=0}^{M\_max}$$

0'ların yerlerini seçiyoruz. M+(n-1) tane'den (n-1) tane seçiyoruz! (Comb)

FABER-CASTELL

If we know upper limit, (M_{max})

If you try to solve numerically, we'll have LOTS of spaces.

Jackson's Theorem

Let $p(k) = \lim_{t \to \infty} p(k,t)$ if it exists

\# of packets at nodes 1..n at time t

and let Λ_i $i=1,\dots,n$ be the solution of the linear system of equations.

$$\Lambda_i = \underbrace{\lambda_i}_{\text{outside traffic entering to node } i} + \sum_j \Lambda_j p_{ji}$$

total traffic entering node j

(we can solve because # of equations is n)

Then, if all $\Lambda_i < \mu_i$ $i=1,\dots,n$, we have

1967 $$\boxed{p(k) = \prod_{i=1}^{n} \left(1 - \frac{\Lambda_i}{\mu_i}\right)\left(\frac{\Lambda_i}{\mu_i}\right)^{k_i}}$$

(in steady state, they became independent)

product form

Proof The stationary solution must satisfy the CK equations

with $\frac{d}{dt} p(k,t) = 0$

$$p(k) \sum_{i=1}^{n} \left[\lambda_i + \mu_i \, 1[k_i > 0]\right]$$

$$\frac{p(k)}{p(k)} \sum_{i=1}^{n} [\lambda_i + \mu_i 1[k_i>0]] = \sum_{i=1}^{n} \frac{p(k_i^-)}{p(k)} \lambda_i 1[k_i>0] +$$

$$\sum_{i=1}^{n} \frac{p(k_i^+)}{p(k)} \mu_i p_{i,n+1} +$$

$$\sum_{\substack{i,j \\ i\neq j}}^{n} \frac{p(k_{ij}^{+-})}{p(k)} \mu_i p_{ij} 1[k_j>0]$$

(∞ # of equations BUT they are linear.)

Divide all by $p(k)$ (note: $p(k)$s are positive because $\Lambda_i < \mu_i$ $\forall i \in 1,\dots,n$)

$$\sum_{i=1}^{n} [\lambda_i + \mu_i 1[k_i>0]] = \sum_{i=1}^{n} \frac{p(k_i^-)}{p(k)} = \frac{\mu_i}{\Lambda_i} \lambda_i 1[k_i>0] +$$

$$\sum_{i=1}^{n} \frac{\Lambda_i}{\cancel{\mu_i}} \cdot \cancel{\mu_i} p_{i,n+1} +$$

$$\sum_{\substack{i,j \\ i\neq j}}^{n} \frac{\Lambda_i}{\mu_i} \frac{\mu_j}{\Lambda_j} \mu_i p_{ij} 1[k_j>0]$$

(look at $p(k) = \pi$)

$$\left(p_{i,n+1} = 1 - \sum_{j=1}^{n} p_{ij} \right)$$

$$\underline{\underline{\sum_{i=1}^{n} [\lambda_i + \mu_i 1[k_i>0]]}} = \sum_{i=1}^{n} \left[\frac{\mu_i}{\Lambda_i} \lambda_i 1[k_i>0] + \Lambda_i \left[1 - \sum_{j=1}^{n} p_{ij} \right] \right] +$$

$$\sum_{\substack{i,j \\ i\neq j}}^{n} \frac{\Lambda_i}{\Lambda_j} \mu_j p_{ij} 1[k_j>0]$$

$$\left(\sum_i \Lambda_i p_{ij} = \Lambda_j - \lambda_j \right) \longrightarrow \sum_{j=1}^{m} \frac{\mu_j}{\underline{\underline{\Lambda_j}}} [\Lambda_j - \lambda_j] 1[k_j>0].$$

= } cancel each other

?

$$\sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{n} \Lambda_i = \sum_{i=1}^{n} \sum_{j=1}^{n} \Lambda_i p_{ij}$$

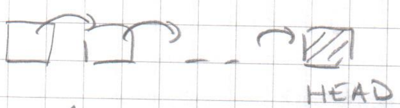
$$\cancel{\sum_{i=1}^{n} \lambda_i} = \cancel{\sum_{i=1}^{n} \Lambda_i} - \sum_{j=1}^{m} \Lambda_j - \lambda_j$$

$\square$

Generalizations for Jackson's Theorem

Let μ_i be a function of k_i

ex



linked list

adding $O(1)$

removing $O(1)$

some s/d ? find one us fastest: $O(\log n)$

μ can be $f(k_i)$ not always $O(1)$

$$\boxed{p(k) = \left(\prod_{i=1}^{n} \Lambda_i^{k_i} \prod_{l=1}^{k_i} \left[\frac{1}{\mu(l)} \right] \right) \cdot p(0)}$$

if $\sum_k p(k)$ exists.

($\mu_i(k_i)$: decreasing function $\Rightarrow$ you would expect no sol exist)

$$\frac{d}{dt} p(K,t) = \sum_{i=1}^{n} p(K_i^-, t) \lambda_i \, 1[k_i > 0]$$ does not change

$$+ \sum_{i=1}^{n} p(K_i^+, t) \, \mu_i(k+1) \, p_{i,M+1}$$

$$+ \sum_{\substack{i,j \\ i \neq j}}^{n} p(K_{ij}^{+-}, t) \, \mu_i(k+1) \, p_{i,\cancel{M+1}\, j} \, [k_j > 0]$$

$$- \sum_{i=1}^{n} p(k,t) \left[\lambda_i + \mu_i(k_i) \, 1[k_i > 0] \right]$$

$\Rightarrow$

$$\sum_{i=1}^{n}\left[\lambda_i + \mu_i(k_i)\, 1[k_i > 0]\right] = \sum_{i=1}^{n} \lambda_i \frac{\mu_i(k_i)}{\Lambda_i} 1[k_i > 0]$$

$$+ \sum_{i=1}^{n} \frac{\Lambda_i}{\cancel{\mu_i(k_i+1)}} \cancel{\mu_i(k_i+1)}\, p_{i,n+1}$$

$$+ \sum_{\substack{i,j \\ i \neq j}} \frac{\Lambda_i}{\cancel{\mu_i(k_i+1)}} \frac{\mu_j(k_j)}{\Lambda_j} \cancel{\mu_i(k_i+1)}\, p_i\, 1[k_j > 0]$$

Another Generalization

90's

negative customers (negative packets)

$P_{ij} \rightarrow P_{ij}^{+}, P_{ij}^{-}$

$\rightarrow$ I go to j as normal packet

$\rightarrow$ " " " negative packet

$P_{ij} = (P_{ij}^{+} + P_{ij}^{-})$

$\lambda_i = (\lambda_i^{+} + \lambda_i^{-})$

$\Lambda_i = (\Lambda_i^{+} + \Lambda_i^{-})$

ρ: utilization factor. ratio of the rate at which work enters system to the maximum rate (cap) at which system can perform this work
work: arriving customer brings to system: # of seconds of service he requires.

$$\rho = \frac{\lambda}{\mu}$$

$\frac{1}{\mu}$: service time

$$\Lambda_i = \lambda_i + \sum_j \Lambda_j P_{ji}$$

$\left(\text{Define m. } \rho_j = \frac{\Lambda_j}{\mu_i} < 1\right)$

shows how BUSY server is.
multiple servers $\rho = \frac{\lambda}{m.\mu}$
erlang

$$\Lambda_i = \lambda_i + \sum_j \rho_j \mu_j P_{ji}$$

a) $\Lambda_j^{+} = \lambda_j^{+} + \sum_i \rho_i \mu_i P_{ij}^{+}$

b) $\Lambda_j^{-} = \lambda_j^{-} + \sum_i \rho_i \mu_i P_{ij}^{-}$

Let $\rho_i = \dfrac{\Lambda_i^{+}}{\mu_i + \Lambda_i^{-}}$

If solution to a) & b) exists and $\rho_i < 1$ then $\rightarrow$

$$\boxed{p(k) = \prod_{i=1}^{n} \rho_i^{k_i} (1-\rho_i)}$$

$$\rho_i = \frac{\lambda_i^+ + \sum \rho_j \mu_j p_{ji}^+}{\mu_i + \lambda_i^- + \sum \rho_j \mu_j p_{ji}^-}$$ $\rightsquigarrow$ NON-LINEAR

you may have non-existing sollutions

(OLD $\rho_i = \dfrac{\lambda_i + \sum \rho_j \mu_j p_{ji}}{\mu_i}$ (LINEAR))

$$\frac{d}{dt} p(k,t) = \sum_{i=1}^{n} p(k_i^-, t)\, \lambda_i^+ \, 1[k_i > 0]$$

$$+ \sum_{i=1}^{n} p(k_i^+, t)\, \mu_i \, p_{i,m+1}$$

$$+ \sum_{\substack{i,j \\ i \neq j}} p(k_{ij}^{+-}, t)\, \mu_i \, p_{ij}^+ \, 1[k_j > 0]$$

$$- \sum_{i=1}^{n} p(k,t) [\lambda_i^+ + \lambda_i^- + \mu_i \, 1[k_i > 0]]$$

no exit

$$+ \sum_{i=1}^{n} p(k_i^+, t)\, \lambda_i^- \rightsquigarrow \text{arrival of bad guy}$$

$$+ \sum_{\substack{i,j \\ i \neq j}} p(k_i^{++}, t)\, \mu_i \, p_{ij}^-$$

$$\Downarrow$$

divide by $p(k)$

$$\frac{p(k)}{p(k)} \sum_{i=1}^{n} \left(\lambda_i^+ + [\lambda_i^- + \mu_i] \, 1[k_i>0] \right) = \sum_{i=1}^{n} \frac{p(k_i^+)}{p(k)} \left[\mu_i p_{i,m+1} + \lambda_i \right]$$

$= \rho_i$

$$+ \sum_{\substack{i \\ i\neq j}} \sum_{j} \left[\frac{p(k^{++})}{p(k)} p_{ij}^- \mu_i + \frac{p(k_{ij}^{+-})}{p(k)} p_{ij}^+ \mu_i \, 1[k_j>0] \right]$$

$\rho_i \rho_j$ $\qquad \frac{\rho_i}{\rho_j}$

$$\sum_{i=1}^{n} \rho_i \cancel{\mu_i} - \sum_{j=1}^{m} \mu_i \rho_i p_{ij} + \lambda_i^- \rho_i$$

$\rho_i \mu_i p_{ij}^-$

$$\sum_j \left[\rho_j \sum_j \Lambda_j^- - \lambda_j^- + \frac{\Lambda_j^+ - \lambda_j^-}{\rho_j} 1[k_j>0] \right]$$

$$\Lambda_j^+ - \cancel{\rho_j \mu_j} - \rho_j \lambda_j^- + \frac{\mu_j + \Lambda_j^- \rho_j}{\cancel{\rho_j}}$$

$$\sum_{i=1}^{m} \lambda_i^+ + \lambda_i^- 1[k_i>0] = \sum_{\substack{i\neq j \\ 1,n}} \mu_i \rho_i (p_{ij}^+ + p_{ij}^-) + \underbrace{\sum_i \cancel{\lambda} \rho_i}_{\text{gonderme}}$$

$$+ \sum_j \Lambda_j^+ - \rho_j \cancel{\lambda_j^-} + \left(\Lambda_j^- - \frac{\lambda_j^+}{\rho_j} \right) 1[k_j>0]$$

$$= \sum_j \left[\rho_i \mu_i - \sum_{j=1}^{n} \rho_i \mu_i (p_{ij}^+ + p_{ij}^-) + \cancel{\lambda_i^-} \rho_i \right]$$

$$+ \sum_j \rho_j \left[\Lambda_j^- - \cancel{\lambda_j^-} \right] + \frac{\Lambda_j^+ - \lambda_j^+}{\rho_j} 1[k_j>0].$$

$$\left(\frac{\Lambda_j^+}{\rho_j} = \mu_j + \Lambda_j^- \right)$$

$$= \sum_j \left[\Lambda_j^+ - \rho_j \mu_j + \frac{\Lambda_j^+ - \lambda_j^+}{\rho_j} 1[k_j>0] \right]$$

(note about utilization (ρ) **trade-off** server-ı efficient kullanmak için ($\rho = \frac{\lambda}{\mu}$) λ'yı arttırırız ama bu durumda queue'deki customer sayısı ↑, delay ↑ , bazı uygulamalarda bu istemez örn. RT app)

(5th week)

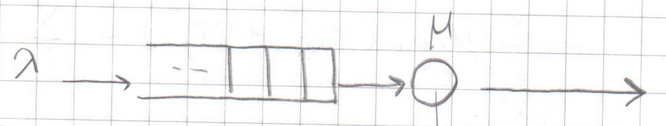
15/03/2013

$\rho = \left(\text{average rate of customers arriving to system}\right) \times \left(\text{average service time}\right)$

$= \lambda \cdot \frac{1}{\mu}$

$\boxed{\rho = \frac{\lambda}{m \cdot \mu}}$ → # of servers.

M/M/1



Poisson Distributed

exponential service time

(related to amount of data (size of packet) in network. Paket boyu sabitse, sabit service time olabilir: M/D/1)

State$|_t$ = N_t → total # of customers in the system (including one in server)

$P_r(N_t = n) = ?$

•) $P_r\left(N_{t+\Delta t} = n+1 \mid N_t = n\right) = \underbrace{\lambda . \Delta t \left(1 - \mu \Delta t\right)}_{\text{single arrival + no departure}} + \underbrace{o(\Delta t)}_{\text{other possibilities}}$

(2 arrival 1 dep
3 " 2 dep
⋮)

FABER-CASTELL

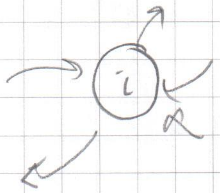
•) $Pr(N_{t+\Delta t} = 1 \mid N_t = 0) = \lambda \Delta t + o(\Delta t)$

•) $Pr(N_{t+\Delta t} = n-1 \mid N_t = n) = \underbrace{\mu \Delta t}_{\text{1 departure}} \underbrace{(1 - \lambda \Delta t)}_{\text{no arrival}} + \underbrace{o(\Delta t)}_{\text{others}}$

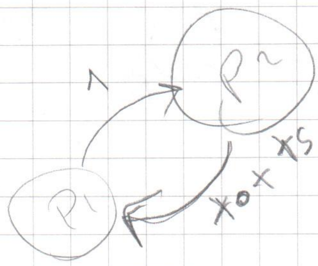
•) $Pr(N_{t+\Delta t} = n \mid N_t = n) =$

$$\underbrace{(1-\mu\Delta t)(1-\lambda\Delta t)}_{\substack{\text{no-arrival}\\ \text{no-dep.}}} + \underbrace{\mu\Delta t\,\lambda\Delta t}_{\substack{\text{single arrival/}\\ \text{departure.}}} + o(\Delta t)$$

Flow Conservation Equality



$$\sum_{i \neq j} \underbrace{P_i P_{ij}}_{\text{outflow}} = \sum_{j \neq i} \underbrace{P_j P_{ji}}_{\text{in flow}}$$



$P_{ij} = ?$

Transition Probability

$$\boxed{P_{ij} = \lim_{\Delta t \to 0} \frac{P(N_{t+\Delta t} = j \mid N_t = i)}{\Delta t}}$$

$$P_{i,i+1} = \lim_{\Delta t \to 0} \frac{P(N_{t+\Delta t} = i+1 \mid N_t = i)}{\Delta t}$$

single arrival
no departure

$$= \lim_{\Delta t \to 0} \frac{\lambda \Delta t (1 - \mu \Delta t) + o(\Delta t)}{\Delta t}$$

$$= \lim_{\Delta t \to 0} \lambda (1 - \mu \Delta t) + \frac{o(\Delta t)}{\Delta t} = 0 \text{ by definition}$$

$$\boxed{P_{i,i+1} = \lambda}$$

$$P_{i+1,i} = \lim_{\Delta t \to 0} \frac{P(N_{t+\Delta t} = i \mid N_t = i+1)}{\Delta t}$$

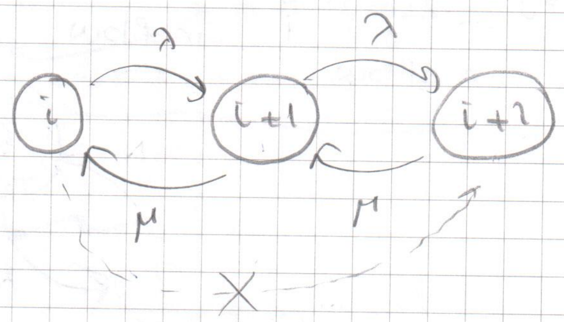
single dep. + no arrival

$$= \lim_{\Delta t \to 0} \frac{\overbrace{\mu \cdot \Delta t \, (1-\lambda \cdot \Delta t)}^{} + o(\Delta t)}{\Delta t}$$

$$= \lim_{\Delta t \to 0} \mu(1-\cancel{\lambda \Delta t}) + \cancel{\frac{o(\Delta t)}{\Delta t}}$$

$$\boxed{P_{i+1,i} = \mu}$$

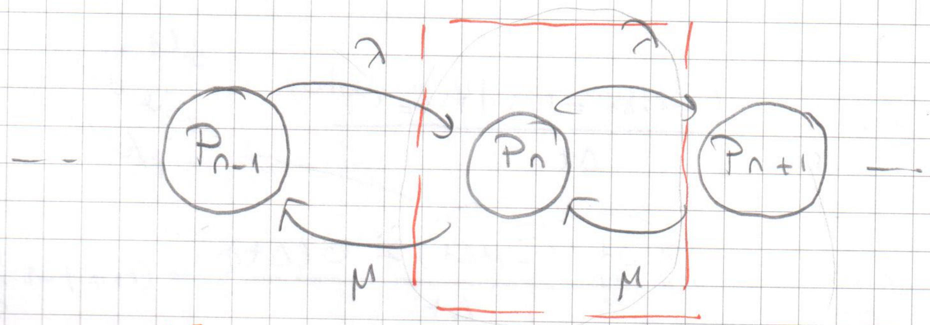
Markov Chain'i oluşturalım;



Δt'yi çok küçük tuttuğumuz için bu geçiş olmaz.

BIRTH-DEATH CHAIN

$(P_{i,i+k} = 0 \ \forall \ k = 2, 3, \ldots)$



flow conservation

•) $\underbrace{P_{n-1} \cdot \lambda + P_{n+1} \cdot \mu}_{\text{incoming}} = P_n(\lambda + \mu)$

For $n=0$

$$\begin{tikzcd} P_0 \arrow[r, bend left, "\lambda"] & P_1 \arrow[l, bend left, "\mu"] \arrow[r, bend left, "\lambda"] & P_2 \arrow[l, bend left, "\mu"] \end{tikzcd}$$

•) $\lambda . P_0 = \mu . P_1$

•) $\sum_{i=0}^{n} P_i = 1$

P_n'i, P_0 cinsinden nasıl yazarız?

$P_0 = P_0$

$P_1 = \frac{\lambda}{\mu} . P_0$

$P_2 = \ ? \ . P_0$

$= \left(\frac{\lambda}{\mu}\right)^2 . P_0$

$$\begin{tikzcd} P_0 \arrow[r, bend left, "\lambda"] & P_1 \arrow[l, bend left, "\mu"] \arrow[r, bend left, "\lambda"] & P_2 \arrow[l, bend left, "\mu"] \end{tikzcd}$$

$P_0 . \lambda + P_2 . \mu = P_1 (\lambda + \mu)$

$P_2 = \frac{P_1 (\lambda + \mu) - P_0 \lambda}{\mu}$

$P_2 = \frac{\frac{\lambda}{\mu} P_0 (\lambda + \mu) - P_0 \lambda}{\mu}$

$P_2 = \frac{P_0 \left(\frac{\lambda^2}{\mu} + \cancel{\lambda} - \cancel{\lambda}\right)}{\mu}$

$P_2 = \left(\frac{\lambda}{\mu}\right)^2 . P_0$

$P_3 = \ ? \ . P_0$

$$\boxed{P_n = \left(\frac{\lambda}{\mu}\right)^n . P_0}$$

Genelde λ ve μ verilir, P_0 sorulur.

$\sum_{n} P_n = 1$ den buluruz

$$1 = \sum_{i=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^i \cdot P_0$$

$$1 = P_0 \sum_{i=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^i$$

(remember $\frac{\lambda}{\mu} < 1$

if $\lambda > \mu \Rightarrow$ unstable system

(increasing amount of queue size

$\Downarrow$

$$1 = P_0 . \frac{1}{1-\frac{\lambda}{\mu}}$$

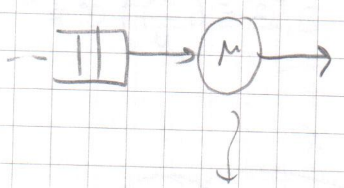
(Geometrik seri : $a + a.r + a.r^2 + a.r^3 + \ldots$)

$$\boxed{P_0 = 1 - \frac{\lambda}{\mu}}$$

$$\boxed{P_n = \left(\frac{\lambda}{\mu}\right)^n \left(1 - \frac{\lambda}{\mu}\right)}$$

$$\left(\sum_{k=0}^{\infty} a.r^k = \frac{a}{1-r} \Leftrightarrow |r| < 1\right)$$

Utilization: $\rho = ?$



Server direk hatta bağlı, hattın yoğunluğu = serverın yoğunluğu

Busy time of server ? Nasıl ifade ederiz ?

expected # of people at server = 1 ⇒ BUSY ⇒ fully utilised
= 0 ⇒ idle

$$\boxed{\rho = \frac{E[N_s]}{m}}$$ → # of servers

FABER-CASTELL

# of customers in system.

$$E[N_s] = \cancel{Pr(N_s = 0) \cdot 0} + \underbrace{Pr(N_c \geq 1)}_{1 - P_0} \cdot 1$$

$$\rho = \frac{E[N_s]}{1} = (1 - P_0) = 1 - \left(1 - \frac{\lambda}{\mu}\right) = \boxed{\frac{\lambda}{\mu}}$$

$$\boxed{P_n = \rho^n (1 - \rho)}$$

$E[N]$ ✓ expected # of customers

$$\boxed{E[N] = \sum_{n=0}^{\infty} P_n \cdot n}$$

$$= \sum_{n=0}^{\infty} \rho^n (1-\rho) \cdot n$$

$$= (1-\rho)\rho \boxed{\sum_{n=0}^{\infty} n \cdot \rho^{n-1}}$$

$$= (1-\rho)\rho \sum_{n=0}^{\infty} \frac{d\rho^n}{d\rho}$$

$$= (1-\rho)\rho \frac{d}{d\rho}\left[\sum_{n=0}^{\infty} \rho^n\right]$$

since $\rho < 0$ $\sum_{n=0}^{\infty} \rho^n = \frac{1}{1-\rho}$

$$= (1-\rho)\rho \frac{d}{d\rho}\left(\frac{1}{1-\rho}\right)$$

$(ab)' = a'b + b'a$

$\left(1 \cdot \frac{1}{1-\rho}\right)' = \cancel{0} + \left(\frac{1}{1-\rho}\right)' \cdot 1$

$= ((1-\rho)^{-1})' = -1 \frac{1}{(1-\rho)^2} \cdot -1$

$$= \cancel{(1-\rho)}\rho \frac{1}{(1-\rho)^2}$$

$$\boxed{E[N] = \frac{\rho}{(1-\rho)}} = \frac{\lambda/\mu}{1 - \lambda/\mu} = \frac{\lambda}{\cancel{\mu}} \cdot \frac{\cancel{\mu}}{\mu - \lambda} = \boxed{\frac{\lambda}{\mu - \lambda}}$$

Little's Rule $\Rightarrow \boxed{N = \lambda . T}$

$$\frac{\lambda}{\mu-\lambda} = \lambda \cdot T$$

$$T = \frac{1}{\mu-\lambda}$$

expected duration for a packet / customer in queueing system. average delay

service duration

$$T_q = T - \frac{1}{\mu}$$

expected duration on queue only

$$\boxed{N_q = \lambda . T_q}$$

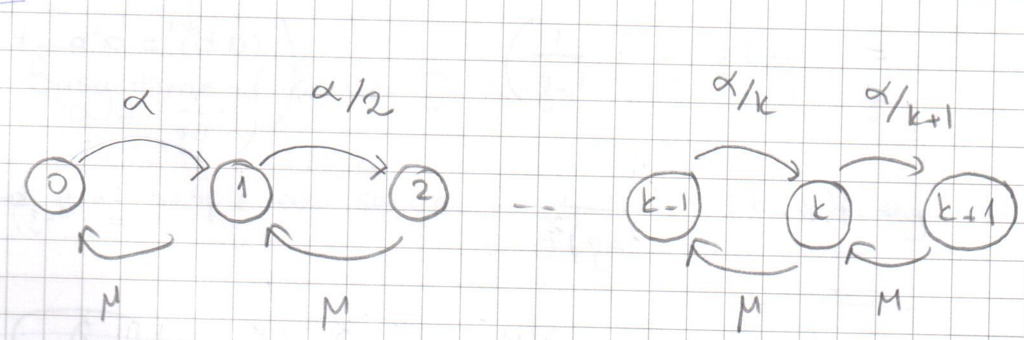
M/M/1 - Discouraged Arrivals λ changes according to people waiting in the queue.

constant

$$\lambda_k = \frac{\alpha}{k+1} \qquad k = 0, 1, 2, \ldots$$

# of people in the system

$$\mu_k = \mu$$



$$P_0 \cdot \alpha = P_1 \cdot \mu$$

$$P_1 = \frac{\alpha}{\mu} P_0$$

$P_n = ?$

$$P_0 \cdot \alpha + P_2 \mu = P_1 \left(\frac{\alpha}{2} + \mu \right)$$

$$P_2 = \frac{\frac{\alpha}{\mu} P_0 \left(\frac{\alpha}{2} + \mu \right) - P_0 \alpha}{\mu}$$

$$P_2 = \frac{P_0 \left(\frac{\alpha^2}{2\mu} + \cancel{\alpha - \alpha} \right)}{\mu}$$

$$P_2 = P_0 \frac{\alpha^2}{\mu^2} \cdot \frac{1}{2}$$

$$\boxed{P_2 = P_0 \cdot \frac{1}{2} \left(\frac{\alpha}{\mu} \right)^2}$$

!

$$\boxed{P_n = P_0 \cdot \frac{1}{n!} \left(\frac{\alpha}{\mu} \right)^n} \qquad \boxed{\sum_{i=0}^{\infty} P_i = 1}$$

$$1 = \sum_{n=0}^{\infty} P_n$$

$$1 = P_0 \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{\alpha}{\mu} \right)^n$$

$$1 = P_0 \underbrace{\left(1 + \frac{\alpha}{\mu} + \frac{1}{2} \left(\frac{\alpha}{\mu} \right)^2 + \frac{1}{3!} \left(\frac{\alpha}{\mu} \right)^3 + \dots \right)}_{\text{TAYLOR exponsion of } e^{\alpha/\mu}}$$

$f(x) = e^x$

$f^{(n)}(x) = e^x$

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)\,(x-a)^n}{n!} \quad \text{Taylor series.}$$

n. türevi

$$e^x = \sum_{n=0}^{\infty} \frac{1 \cdot x^n}{n!} \qquad (e^0 = 1) \qquad a=0, \; x = \frac{\alpha}{\mu}$$

$$e^{\alpha/\mu} = \sum_{n=0}^{\infty} \frac{1}{n!}\left(\frac{\alpha}{\mu}\right)^n$$

$$1 = P_0 \cdot e^{\alpha/\mu}$$

$$\boxed{P_0 = e^{-\alpha/\mu}}$$

$$P_n = e^{-\alpha/\mu}\left(\frac{\alpha}{\mu}\right)^n \cdot \frac{1}{n!}$$ Poisson dist.

LITTLE's rule $\quad N = \lambda . T$

sabit λ $\quad \bar{\lambda} = \sum_k \lambda_k P_k$

$$E[N] = \sum_{n=0}^{\infty} n . P_n$$

M/M/1 Summary

Utilization $\rightarrow \rho = \lambda / \mu$

probOf n cust. in system $\rightarrow P_n = \rho^n(1-\rho)$

Avg # of cust $\rightarrow \dfrac{\rho}{1-\rho}$

$T \rightarrow \dfrac{1}{\mu - \lambda}$

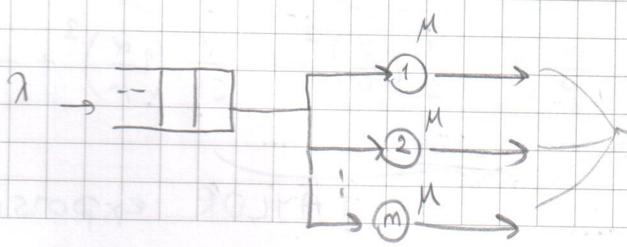
$N_Q = \dfrac{\rho^2}{1-\rho}$

$W = \dfrac{\rho}{\mu - \lambda}$

avg waiting time in queue

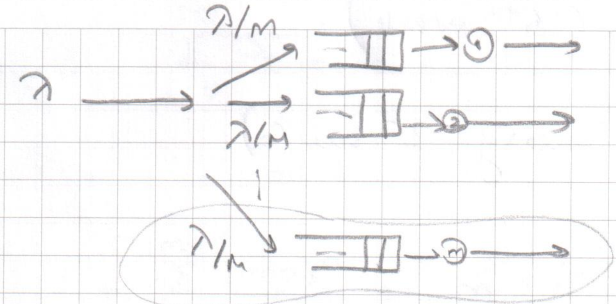
M/M/m

(The m server case)



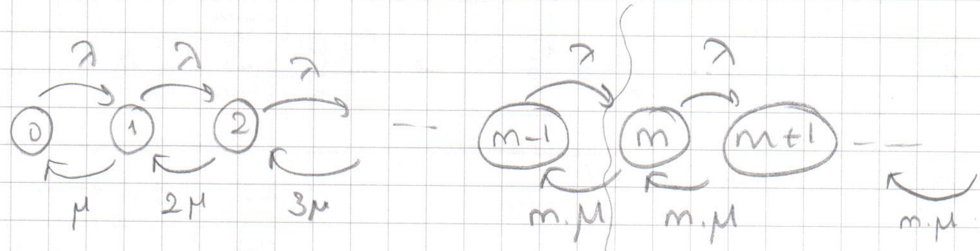
$\rho^n(1-\rho)$

OR



each M/M/1

how we can model MARKOV chain



$$\lambda P_{n-1} = n\mu P_n \qquad n \le m$$
$$\lambda P_{n-1} = m \cdot \mu P_n \qquad n > m$$

$P_n = ?$

$$P_0 \cdot \lambda = P_1 \mu$$
$$P_1 = \frac{\lambda}{\mu} P_0$$

$P_2 = ?$ $\quad P_1(\lambda + \mu) = P_0 \lambda + P_2 \, 2\mu$

$$\frac{\lambda}{\mu} P_0 (\lambda + \mu) = P_0 \lambda + P_2 \, 2\mu$$

$$P_2 = \frac{P_0 \left(\frac{\lambda^2}{\mu} \cancel{+ \lambda - \lambda} \right)}{2\mu}$$

$$P_2 = \frac{1}{2} \left(\frac{\lambda}{\mu} \right)^2 P_0$$

$$P_n = \frac{1}{n!} \left(\frac{\lambda}{\mu} \right)^n P_0 \qquad n \le m$$

$$= \frac{1}{m!} \frac{1}{m^{(n-m)}} \left(\frac{\lambda}{\mu} \right)^n P_0 \qquad n > m$$

Bunu cıkarmadı

22/03/2013 Cuma (6th week).

(Fundamentals of Queuing Theory 19. sayfa)

$$P_n = \begin{cases} \frac{1}{n!}\left(\frac{\lambda}{\mu}\right)^n P_0 & n \leq m \\ \frac{1}{m!}\,\frac{1}{m^{(n-m)}}\left(\frac{\lambda}{\mu}\right)^n P_0 & n > m \end{cases}$$

utilization ?

ρ : Keeping system BUSY

$E[N_s] \rightarrow$ expected # of people at server

$\boxed{\rho = \frac{E[N_s]}{m}}$ demiştik.

$\boxed{\rho = \frac{\lambda}{\mu \cdot m}}$ olur demiştik.

$\rho_1 + \rho_2$

~~$f(n) + f(s)$~~ $f(n) + f(n)$

$\rho_1 \neq \rho_2$

$E[N_s] = \frac{\lambda}{\mu}$ olmalı.

$\Downarrow$

* $E[N_s] = 0 \cdot P_0 + 1 \cdot P_1 + 2 \cdot P_2 + \dots (m-1) P_m + m\,[P_m + P_{m+1} + \dots]$

Pn in terms of ρ

$$P_n = \begin{cases} \frac{1}{n!}(m \cdot \rho)^n \; P_0 & n \leq m \\ \underbrace{\frac{1}{m!}\,\frac{1}{m^{(n-m)}} \cdot \rho^n m^n}_{\frac{1}{m!} m^m \rho^n} \; P_0 & n > m \end{cases}$$

$\frac{1}{m!} m^m \rho^n P_0$

To find P_0 $\quad \sum_{i=0}^{\infty} P_n = 1$

$$P_0 + P_1 + \ldots = 1$$

$$P_0 \left[1 + \sum_{n=1}^{m-1} \frac{(m\rho)^n}{n!} + \sum_{n=m}^{\infty} \frac{(m\rho)^n}{m!} \frac{1}{m^{(n-m)}} \right] = 1$$

$$P_0 = 1 + \sum_{n=1}^{m-1} \frac{(m\rho)^n}{n!} + m\rho^n \sum_{n=0}^{\infty} \frac{(m\rho)^n}{m!\, m^n}$$

$P\{\text{Queuing}\} = \sum_{n=m}^{\infty} P_n$ ← all servers are BUSY, incoming customers will wait.

$$= \sum_{n=m}^{\infty} \frac{m^m \rho^n}{m!} P_0$$

$$= P_0 \sum_{n=m}^{\infty} \frac{m^m \rho^n}{m!}$$

$$= P_0 \frac{(m\rho)^m}{m!} \underbrace{\sum_{n=m}^{\infty} \rho^{n-m}}$$

$$\sum_{k=0}^{\infty} \rho = \frac{1}{1-\rho}$$

since $\rho < 1$

$$\boxed{P\{\text{Queuing}\} = P_0 \cdot \frac{(m\rho)^m}{m!\,(1-\rho)}}$$

Erlang-C Formula

$N_Q = ?$ expected # of customers waiting at Queue.

$$N_Q = \sum_{n=0}^{\infty} n \cdot P_{m+n}$$

FABER-CASTELL

$$N_Q = \sum_{n=0}^{\infty} n \cdot P_0 \cdot \frac{m^m \rho^{n+m}}{m!}$$

$$N_Q = \frac{P_0 \cdot (m\rho)^m}{m!} \sum_{n=0}^{\infty} n \cdot \rho^n$$

$$= \frac{P_0 (m\rho)^m}{m!} \rho \sum_{n=0}^{\infty} n \cdot \rho^{n-1}$$

$$= \frac{P_0 (m\rho)^m}{m!} \rho \sum_{n=0}^{\infty} \frac{d}{d\rho}(\rho^n)$$

$$= \frac{P_0 (m\rho)^m}{m!} \rho \frac{d}{d\rho} \underbrace{\sum_{n=0}^{\infty} \rho^n}_{= \frac{1}{1-\rho}}$$

$$= \frac{P_0 (m\rho)^m}{m!} \rho \frac{d}{d\rho} \left(\frac{1}{1-\rho}\right)$$

$$\boxed{N_Q = \frac{P_0 (m\rho)^m}{m!} \rho \frac{1}{(1-\rho)^2}}$$

using Erlang-C Formula

$$N_Q = P_Q \frac{\rho}{(1-\rho)}$$

Little's Rule $\quad N = \lambda . T$

waiting time n queue

$$N_Q = \lambda \, T_Q \Rightarrow \boxed{T_Q = \frac{P_Q \, \rho}{(1-\rho) \lambda}}$$

$$\boxed{T = \frac{1}{\mu} + T_Q}$$

avg time spent on all system

$$N = \lambda T$$

$$= \frac{\lambda}{\mu} + \frac{\cancel{\lambda} P_Q \, \rho}{(1-\rho) \cancel{\lambda}} = \frac{\lambda}{\mu} + \frac{P_Q \; \lambda / \cancel{\mu} . m}{\frac{m\mu - \lambda}{\cancel{\mu} m}}$$

$\rho = \lambda / m\mu$

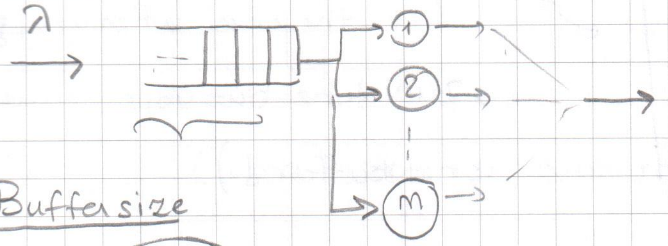
# of users in system M/M/m

$$\boxed{N = \frac{\lambda}{\mu} + \frac{P_Q \, \lambda}{m\mu - \lambda}}$$

(If you put $m = 1$ in all equations, you get M/M/1 formulas)

M/M/m/k

of servers — total # of customers in system $(k > m)$

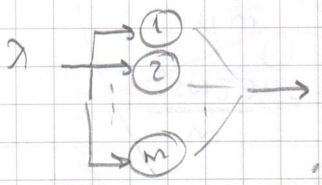


Buffer size

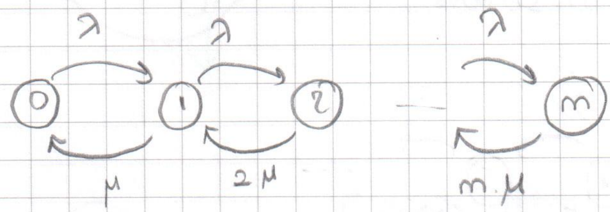
$\boxed{k - M}$

Summation in M/M/m formula will be up to k. Formulas change slightly.

M/M/m/m System ⇒ NO BUFFER



Loss Systems: why? An incoming customer will return if all servers are BUSY



$$P_1 = \frac{\lambda}{\mu} P_0$$

$$P_0 \lambda = P_1 \mu$$

$$P_1 \lambda = P_2 \, 2\mu$$

$$P_2 \lambda = P_3 \, 3\mu$$

$$\boxed{P_n = P_0 \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}}$$

$$P_0 = \frac{1}{\sum_{n=0}^{m} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}}$$

Probability of an incoming packet to be lost? $\underline{\underline{P_M}}$

$$P_m = P_0 \left(\frac{\lambda}{\mu}\right)^m \frac{1}{m!}$$

$$\boxed{P_m = \frac{\left(\frac{\lambda}{\mu}\right)^m \cdot \frac{1}{m!}}{\sum_{n=0}^{m} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}}}$$

Erlang-B Formula

(i.e. the case when you get a BUSY tone but your request is not buffered)

N, T, λ ?

$$E[N_s] = P_0 \cdot 0 + P_1 \cdot 1 + \dots = \underline{\underline{\frac{\lambda}{\mu}}}$$ çıkmalı

$T = \frac{1}{\mu}$ (hattı alabilrsen)

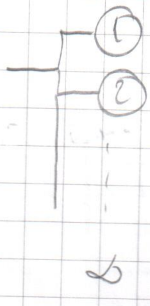
$N = \lambda \cdot T = \frac{\lambda}{\mu}$

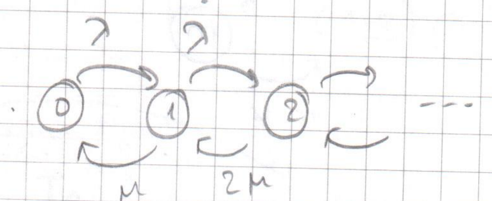
neden?

$\rho = \frac{\lambda}{m\mu}$

$\rho = \frac{E[N_s]}{m}$

M/M/∞ infinite # of servers.





$$P_n = P_0 \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}$$

$$\sum_{n=0}^{\infty} P_n = 1$$

$$P_0 = \left[1 + \sum_{n=1}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} \right]^{-1}$$

$$\boxed{P_0 = e^{-\lambda/\mu}}$$

$$\boxed{P_n = e^{-\lambda/\mu} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}}$$

✓ <u>POISSON</u> with $\frac{\lambda}{\mu}$

remember Taylor series

$$\sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} = e^{\lambda/\mu}$$

why? $f(x) = e^x$, $x = \lambda/\mu$

$f^{(n)}(x) = e^x$, n. türevi

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x-a)^n$$

$a = 0 \Rightarrow$ <u>Maclaurin Series</u>

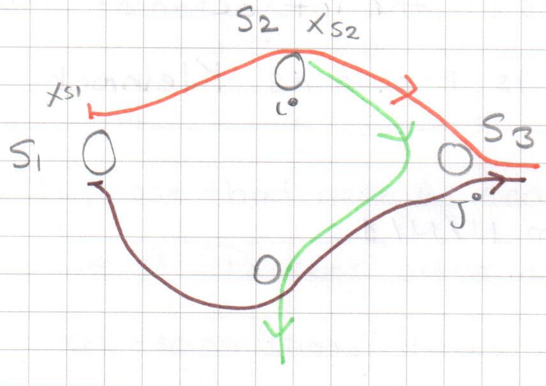
$$e^{\lambda/\mu} = \sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}$$

$T = \frac{1}{\mu}$ (ama sisten herzaman musait)

$N = \lambda \cdot T$

$N = \frac{\lambda}{\mu}$

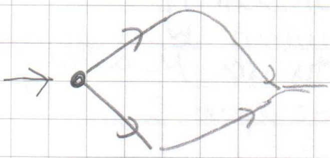
The Kleinrock Independence Approximation



$\lambda_{ij} = X_{s1} + X_{s2}$

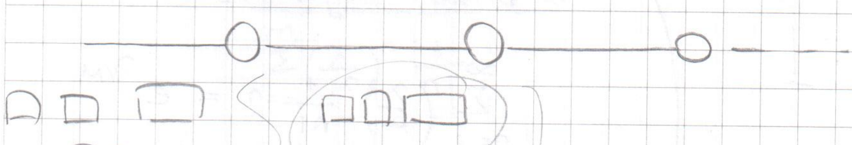
$\lambda_{ij} = \sum X_s$

allpacket streams that cross lnk ij



$f_{ij}(s)$ fraction of packets of stream s that go through lnk ij

$$\lambda_{ij} = \sum f_{ij} X_s$$



variable packet sizes

slow truck effect (İlk serverdan çıkınca dubirlerine yaklaştılar) İlk paketin service time-ı uzun

canbe represented as queueing system

Başka bir flow eklenirse, slow truck effect bozulabilir

Poisson özellik bozulabilir.

→ Her nodeu queue olarak düşün (M/M/1)

"I can find mean delay on specific link for specific connection)

It is often appropriate to adopt M/M/1 queueing model for each communication link regardless of the interaction of traffic on this link with other links. This is known as Kleinrock Independence Approximation

$$N_{ij} = \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}$$

from M/M/1

# of packets waiting to be transmitted on link ij

avg packet transmission time on link ij (hattın kapasitesini arttırırsan μ artar)

∝ O—O linkini O⇄O şeklinde ifade etmen çok önemli (yönler önemli)

* Average # of packets summed up over all queues:

$$N = \sum_{ij} \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}$$

By Little's Theorem ⇒ $T = \frac{1}{\underbrace{\sum_s x_s}_{\gamma}} \sum_{ij} \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}$

($N = \lambda T$)

avg # of time spent in system.

$$T = \frac{1}{\gamma} \sum_{ij} \left(\frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}} + \underbrace{\lambda_{ij} \cdot d_{ij}}_{\text{propogation delay}} \right)$$

$$\underbrace{T_p}_{\text{T for path } p} = \sum_{\substack{\text{all } (i,j) \\ \text{in path} \\ p}} \left(\frac{\lambda_{ij}}{(\mu_{ij} - \lambda_{ij}) \cdot \lambda_{ij}} + \frac{1}{\mu_{ij}} + \underbrace{d_{ij}}_{\text{prop. delay}} \right)$$

doesn't include processing delay (encryption, --)

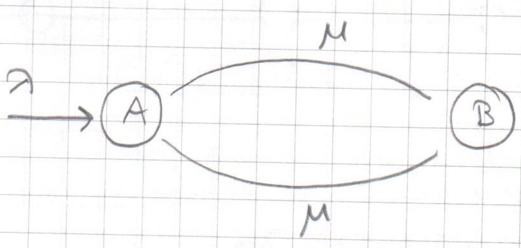
•) Suppose that node A sends traffic B along 2 links with service rate μ. Packets arrive at A acc to Poisson process with rate λ, packet trasmissions' times exponentially distributed & independent of arrival times.

Assume that arriving traffic is to be divided into equally among 2 links. 2 approaches → Randomization

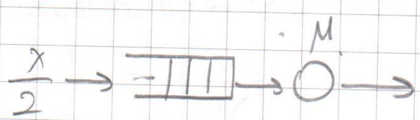
→ Metering (diyelim 2 linkte de islenen paket var, size-ı küçük olan linke gönder)

packet size

∝ How can we calculate transmission delay ?



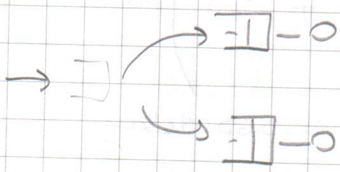
1) Randomization ⇝ each link may be modelled as



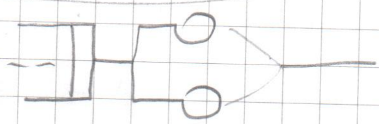
$$T = \frac{1}{\mu - \frac{\lambda}{2}}$$

2) Metering

how to model metering ?



⇒ M/M/2 system



2 server da meşgul diyelim,
3. paket zaten işi erken biten servera gidecek. (işlenen paketlerden boyu küçük olan erken bitecek)

Same philosophy ?

$$T_m \text{ (in M/M/2)} = \frac{2}{(2\mu-\lambda)(1+\rho)} \quad \left(\rho = \frac{\lambda}{2\mu} \right.$$

$$T_{random} = \frac{1}{\mu - \frac{\lambda}{2}}$$

which is better ?

$$T_{metering} = \frac{1}{\left(\mu - \frac{\lambda}{2}\right)\left(1 + \frac{\rho}{2}\right)}$$ additional term > 1

$$\left(T_{metering} < T_{random} \right)$$

BUT ! If you use metering, packets leaving does NOT match Poisson. Random keeps Poisson property.

7th week

29/03/2013

M/M/I Reversability

$\longrightarrow t$

$\longleftarrow$ Are we going to see same properties?

Markov Chain has the property

P[future | present, past] = P[future | present]

P[past | present, future] = P[last | present]

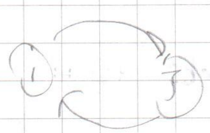
$$P[X_n = j \mid X_{n+1} = i_1, X_{n+2} = i_2, \ldots] =$$

$$P[X_n = j \mid X_{n+1} = i] = P_{ij}^*$$

future current

$i \; j$ from future to present

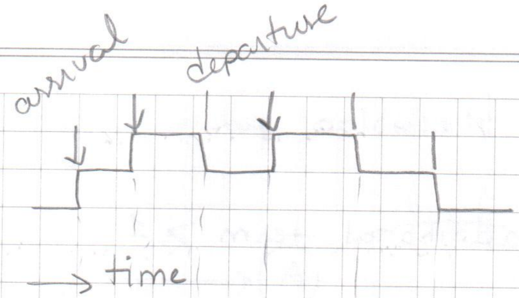
The state sequence run backward in time, in steady state, is a Markov Chain again and it can be shown that



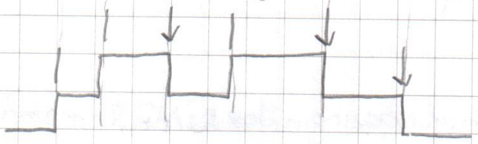
$$P_i \cdot P_{ij}^* = P_j P_{ji}$$

A Markov Chain is **reversible** if $\overset{P_i}{P_{ij}^*} = \overset{P_j}{P_{ji}}$

Birth-Death Process $\Leftrightarrow$ Reversible (Because balance equations should be satisfied)



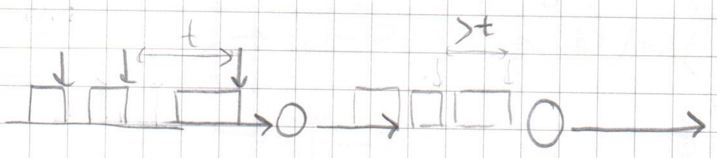
what if we go this direction ←



Remember service times are also exponentially distributed

Poisson λ → M/M/1 → (λ) M/M/1 → λ

What if 2^{nd} queue is not M/M/1 ?



(Remember, according to Kleinrock approximation)

$\lambda \to \lambda/2 - x_1$, $\lambda/2 - x_2$

$\lambda_{ij} = \sum_{l} x_p$ ← amount of data flowing path p

paths that traverse link ij

if randomly chosen!

* If you deterministically say send from 1 then 2, then 1 then 2...

it is NOT Poisson with $\lambda/2$

Network of Queues

Open networks: traffic can be received & sent outside the network

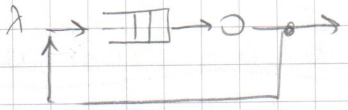
can be suited to model "store & forward" networks where different nodes (modelled by means of queues) exchange data traffic in form of variable-length messages.

FABER-CASTELL

Closed Networks traffic can NOT be exchanged with external nodes.

λ_i: node i receives traffic from outside network with λi & also receives traffic routed from other nodes.

Λ_i: total mean input rate

(**feedback loops**  (Feedbacks may change Markovian property))

Cyclic network: allows feedback loops
Acyclic network: does NOT allow " "

~ Poisson characteristic of input traffic can also be loosen in presence of queues with finite rooms that clear arrivals exceeding their capacity, in this case, the circulating traffic is **smoothed**.

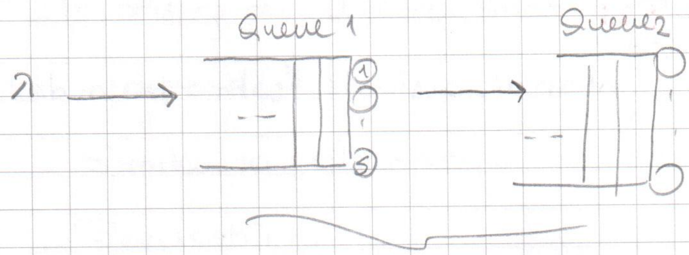
Avoiding feedback loops and blocking phenomena, the Poisson characteristics, of input processes is maintained within network due to <u>random split model</u>

The Burke's Theorem

M/M/S

Poisson arrivals / exp service times / S server / ∞ room.

<u>Tandem queues</u>



A product form to represent joint state probabilities.

n_1: # of messages in queue 1 (with related prob. P_{n_1})
n_2: # of " " " 2 (" " " P_{n_2})

Joint state probability (n_1, n_2) is characterized by $P_{n_1, n_2} = P_{n_1} \circ P_{n_2}$

$$P_{n_1 n_2} = P_{n_1} \cdot P_{n_2}$$

<u>Assumptions</u>

Sum of Poisson ind. processes at input of nodes
Random splitting at nodes
(i.e: stochastic routing at nodes)

No losses (∞ queue length)
No loops (acyclic network)

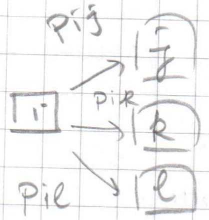
Jackson Network

We consider

⇒ independent, external Poisson arrivals

⇒ independent exponential service times
(same job has <u>independent service time</u> at different queues)

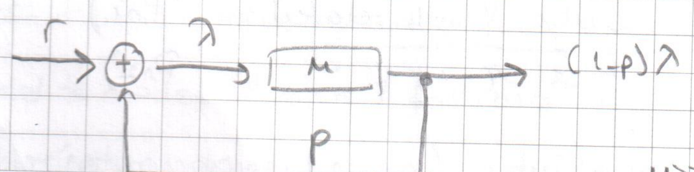
⇒ independent routing of packets (stochastic routing at each node)



$$\lambda_i = \underbrace{r_i}_{\text{external traffic}} + \underbrace{\Sigma \lambda_k p_{ki}}_{\text{internal arrivals from other node.}}$$

We know λ_is, μ_is

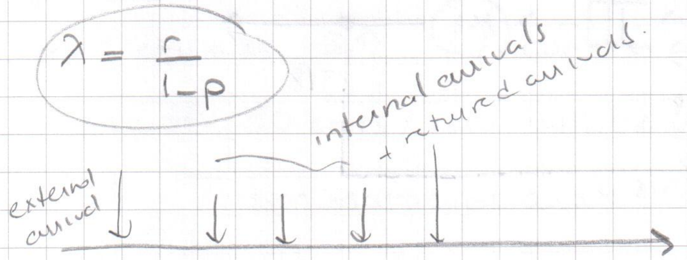
<u>Queues with feedback</u>



$\mu \gg \lambda$

$\lambda = r + p \cdot \lambda$

$$\lambda = \frac{r}{1-p}$$



When p is large, each arrival is followed by a burst of interval arrivals.

↳ Arrivals to queues are NOT Poisson.

We define state of system to be $n^v = (\underbrace{n_1, \dots n_k}_{\text{\# of people in each queue.}})$

$n_i \Rightarrow$ # of customers at node i

Jackson's Theorem Says That:

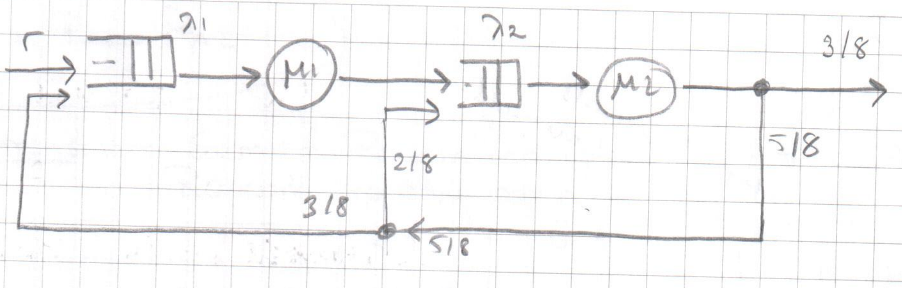
$$\boxed{P(n^v) = \prod_{i=1}^{k} P_i(n_i)} \qquad \rho_i = \frac{\lambda_i}{\mu_i}$$

$$\boxed{P(n^v) = \prod_{i=1}^{k} \rho^{n_i}(1-\rho_i)}$$

That is: In steady state of node i, n_i is independent of the states of all other nodes at given time:

- independent M/M/1 queues
- suprising result given that arrivals to each queue are neither Poisson nor independent
- Similar to Kleinrock's independence approximation
- Reversability
  - exogeneous arrivals are independent & Poisson
  - the state of the entire system is independent of past exogenous departures.

ex



$P(n_1, n_2) = ?$ (If n_1 & n_2 are not specified, you need to find average)

$P(1,5) = ?$

$\Rightarrow \lambda_1 + \frac{2}{8}\lambda_2 = \lambda_2$

$\Rightarrow \lambda_1 = r + 3/8\,\lambda_2$

$r + \frac{3}{8}\lambda_2 + \frac{2}{8}\lambda_2 = \lambda_2$

$r = \frac{3}{8}\lambda_2$

$\lambda_2 = \frac{8}{3} r$

$\lambda_1 = r + \frac{3}{8}\lambda_2 = r + \frac{3}{8}\cdot\frac{8}{3} r$

$\lambda_1 = 2r$

$N = \lambda . \underbrace{T}_{\frac{1}{\mu - \lambda}}$

$N_1 = 2r \cdot \frac{1}{\mu_1 - 2r}$

$N_2 = \frac{8}{3} r \cdot \frac{1}{\mu_2 - \frac{8}{3} r}$

Average # of customers.

$P(1,4) = ?$ $\underbrace{\left[\rho_1 (1-\rho_1)\right]}_{N_1 = 1} \cdot \underbrace{\left[\rho_2^{\,4} (1-\rho_2)\right]}_{N_2 = 4}$

$\left(\rho_1 = \frac{2r}{\mu_1} \quad \rho_2 = \frac{8r}{3\mu_2} \right)$

(remember M/M/1
$p_n = \rho^n (1-\rho)$)

Examples

Data Networks 3.9 A communication line capable of transmitting at rate of 50 kbits/sec. will be used to accommodate 10 sessions generating Poisson traffic at a rate 150 packets/min. Packet lengths exp. dist. with mean 1000 bits.

a) For each session, find avg # of packets in the queue, the avg # of packets in system, avg delay per packet when line is allocated to sessions.

1) 10 equal capacity TDM channels

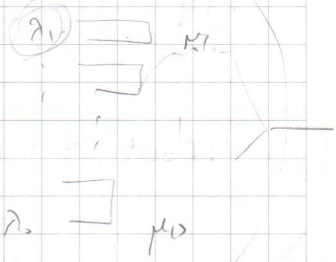
2) Statistical multiplexing.

$\lambda = \frac{150 \text{ packet}}{\text{min}} = 2{,}5$ packets/sec

channel capacity $= \frac{50 \text{ kbits}}{\text{sec}} = \frac{50 \text{kbits}}{\text{sec}} \cdot \frac{1 \text{ pac}}{1000 \text{bits}} = 50$ packet/sec

1) TDM ①②...⑩

$\underbrace{\text{①②...⑩}}$ each have 5 packets/sec capacity



Avg # of packets in queue N_q
" " " " in system N

Avg delay T

$N = \lambda . T = \lambda \frac{1}{\mu - \lambda} = 2.5$ packets/sec $\frac{1}{(5 - 2{,}5) \text{ pack/sec}} = \underline{\underline{1}}$

What about N_q? The # of people at server is determined by ρ!

$\rho = \frac{\lambda}{\mu} = \frac{2{,}5}{5} = 0.5$

$N_q = N - \rho = \frac{1}{2}$ // for each connection

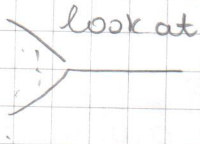
$T = \frac{1}{\mu - \lambda} = \frac{1}{5 - 2{,}5} = 0.4$ sec

System $N = 1 \cdot 10 = 10$

$N_q = \frac{1}{2} \cdot 10 = 5$

$(N = \lambda T)$

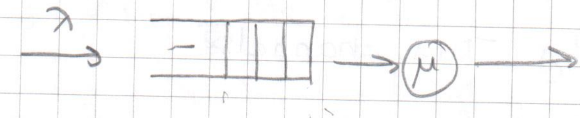
2) Statistical Multiplexing



look at 1st connection, if it has sth to send, it sends.

(1 idle, 2 idle ⇒ 3 gets)

$\lambda = 2{,}5 \times 10 = \underline{\underline{25 \text{ packets/sec.}}}$



$\mu = 50$ packets/sec

$T = \frac{1}{50-25} = 0.04 \text{ sec}$ (*) 10 times less delay

(why? in TDM, even if 1st don't have any data to send, 1st slot is reserved)

$N = \lambda . T = 25 \times 0.04 = \underline{\underline{1}}$

$N_q = N - \rho = 1 - \frac{1}{2} = \frac{1}{2}$ //

b) Repeat part a) for the case 5 sessions 250 pack/min
5 " 50 pack/min

Statistical Multiplexing (Results will NOT change)

$\frac{250 \text{ pack}}{\text{min}} = \frac{25}{6}$ pack/sec $\frac{50 \text{ pack}}{\text{min}} = \frac{5}{6}$ pack/sec.

$\lambda_{total} = \frac{25}{6} \cdot 5 + \frac{5}{6} \cdot 5 = \underline{\underline{25 \text{ pack/sec}}}$ (same λ)

TDM μ was 5 pack/sec

λ changes

$\lambda_1 = \frac{25}{6}$ pact/sec $\Rightarrow T_1 = \frac{1}{\mu_1 - \lambda_1} = \frac{1}{5 - \frac{25}{6}} = \frac{6}{5}$

$N_1 = \lambda_1 . T_1 = \frac{25}{6} \cdot \frac{6}{5} = \underline{5}$, $N_{q1} = 5 - \rho_1 = 5 - \frac{25/6}{5} = \frac{25}{6}$

$T_{q1} = \frac{6}{5} - \frac{1}{5} = 1$

$N_2 = \lambda_2 . T_2 = \frac{5}{6} \cdot \frac{1}{5 - \frac{5}{6}} = \frac{5}{6} \cdot \frac{6}{25} = \frac{1}{5}$

$N_{q2} = \frac{1}{5} - \rho_2 = \frac{1}{5} - \frac{5/6}{5} = 1/30$

$T_2 = \frac{1}{5 - \frac{5}{6}} = \frac{6}{25}$

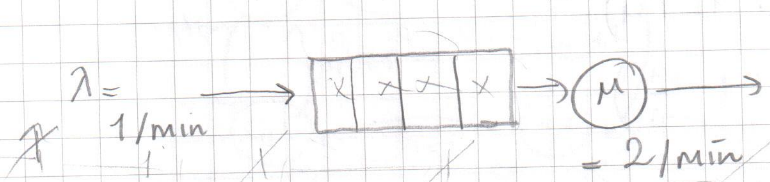
$T_{q2} = \frac{6}{25} - \frac{1}{5} = \frac{1}{25}$

$\underline{N_q \text{ for all system}} = N_{q1} . 5 + N_{q2} . 5 = 21$

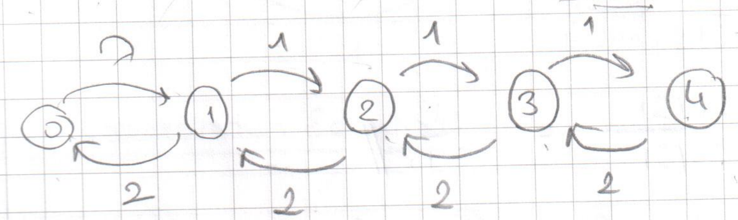
N for all system $= (N_1 + N_2) 5 = 26$

$T_{avg} = \frac{N}{\lambda_{all}} = \frac{26}{\frac{25}{6} \cdot 5 + \frac{5}{6} \cdot 5} = 1.04$

3.18 Empty taxi pass by a street corner at a Poisson rate of 2 per/min & pickup a pessanger if there is one. Passangers arrive at a Poisson rate of 1/min & wait for a taxi only if there are fewer than 4 persons waiting, otherwise they leave. Find avg waiting time of a passenger who enters queue.



(M/M/1/5 system)



$P_0 . 1 = P_1 . 2$, $P_2 . 2 = P_1 . 1 \Rightarrow P_2 = \frac{1}{2} P_1$

$$\left.\begin{aligned} P_1 &= \frac{1}{2} P_0 , \\ P_2 &= \left(\frac{1}{2}\right)\left(\frac{1}{2}\right) P_0 \\ P_3 &= \left(\frac{1}{2}\right)^3 P_0 \\ P_4 &= \left(\frac{1}{2}\right)^4 P_0 \end{aligned}\right\}$$

$\frac{16+8+4+2+1}{16} = \frac{31}{16}$

$$P_0 \left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} \right) = 1$$

$$P_0 = \frac{16}{31}$$

$$N = \sum_{n=1}^{4} n . P_n = 1 . \frac{1}{2} \frac{16}{31} + 2 . \frac{1}{4} \frac{16}{31} + 3 . \frac{1}{8} \frac{16}{31} + 4 \frac{1}{16} \frac{16}{31}$$

$$= \frac{16}{31} \left(\frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{1}{4} \right)$$

$\frac{4+4+3+2}{8} = \frac{13}{8}$

$$= \frac{\cancel{16}^{2}}{31} \frac{13}{\cancel{8}}$$

$$N = \frac{26}{31}$$

$N = \lambda . T \qquad T = \frac{26/31}{1} = \frac{26}{31}$

can we say?

! **NO** actual arrival rate is NOT λ

$\lambda' = \lambda (1 - P_4)$ If there are 4 people it leaves.

$P_0 + P_1 + P_2 + P_3$

$\lambda' = \lambda \left(1 - \frac{1}{31} \right)$

$\lambda' = 1 \frac{30}{31}$

$N = \lambda' T \qquad T = \frac{26/31}{30/31} = \frac{13}{15}$ //

Example Asymmetric M/M/2 Queue

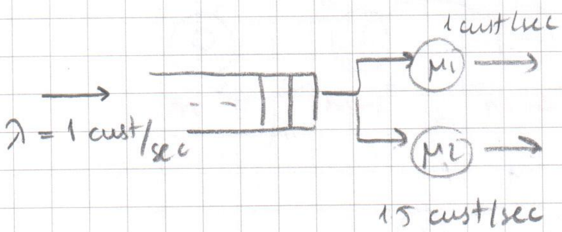
- Customers arrive at $\lambda = 1$ cust/sec.
- Servers $\mu_1 = 1$ cust/sec
  $\mu_2 = 1.5$ cust/sec

Draw corresponding Markov Chains & write down the necessary equations to obtain the steady state probabilities, assuming that

a) servers selected randomly

b) fastest free server is selected when both available

c) What is the utilizations of servers (ρ_1, ρ_2) for cases in a & b.



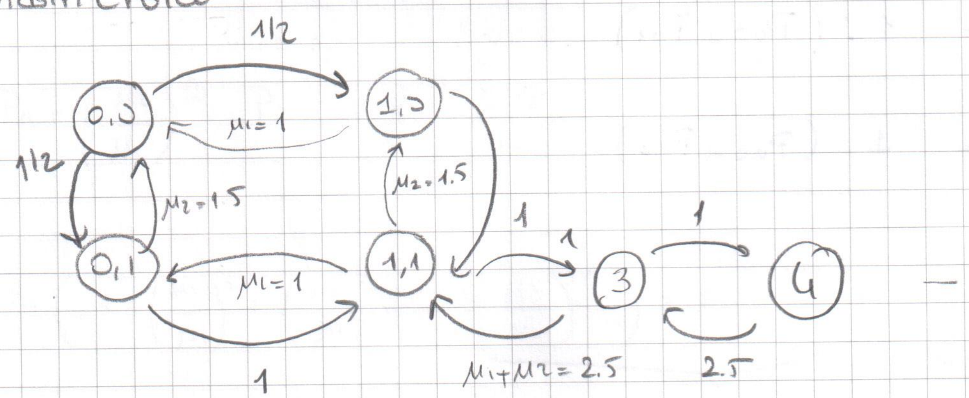
no person (0,0)

1 person (0,1)
(1,0)

2 people (1,1)

≥ 2 $\Rightarrow$ Both servers busy, coming customer will wait in the queue.

a) Random choice



Flow Conservation Equations

$$P_{00} \cdot \left(\frac{1}{2} + \frac{1}{2}\right) = P_{10} + P_{01} \, 1.5$$

for (1,0) ...

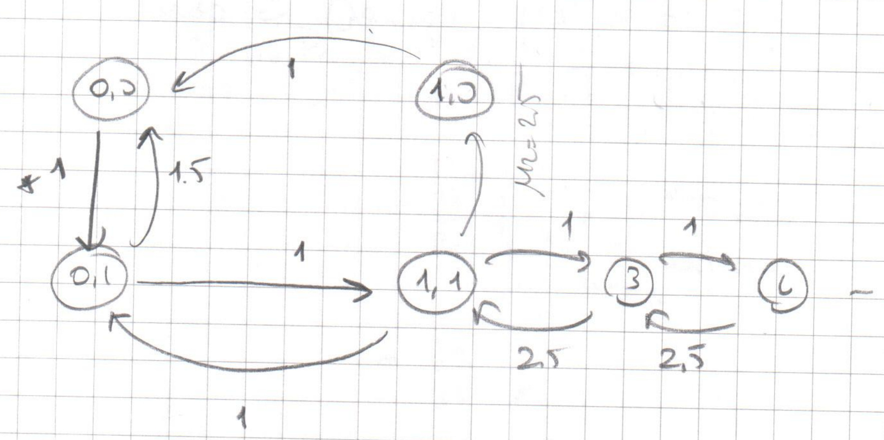
$P_{3} \cdot 1 = P_{11} \cdot 2.5$

$P_{3} \cdot 1 = P_{4} \cdot 2.5$

$P_{n-1} = P_n \cdot 2.5$ for $n \geq 4$

$$\sum_{\substack{i=0\\j=0}}^{i=j=1} P_{ij} + \sum_{i=3}^{\infty} P_i = 1$$

b) what if fastest server is selected?



c) Utilizations

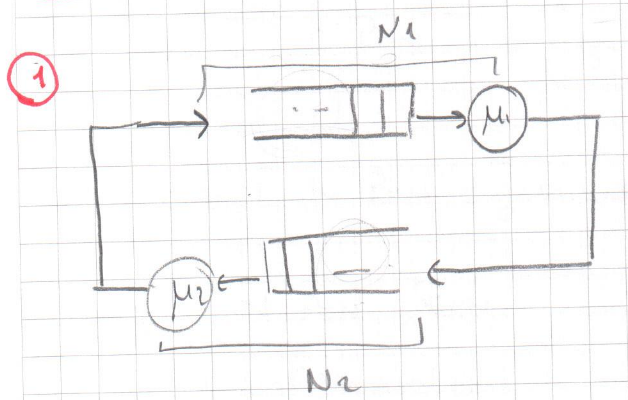
remember ⇝ utilization = prob. of being busy

$$\begin{cases} \rho_1 = 1 - (P_{00} + P_{01}) \\ \rho_2 = 1 - (P_{00} + P_{10}) \end{cases}$$

*

Sınav Soruları (20.11.2008)

1

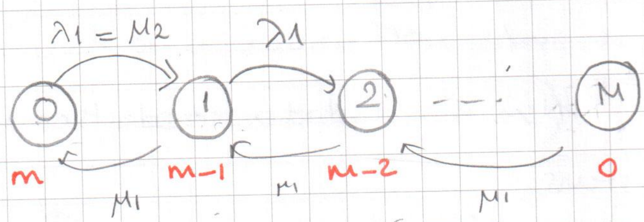


$N_1 + N_2 = M$ (constant)

* İlk kuyruğu modellersek sistemi modellemiş oluruz.

P_0^{1}

P_M^{2}



2nd queuing system

$$P_0 . \mu_2 = P_1 . \mu_1$$

$$P_1 = \frac{\mu_2}{\mu_2} P_0$$

$$P_2 . \mu_1 = P_1 . \mu_2$$

$$P_2 = \frac{\mu_2}{\mu_1} P_1 = \left(\frac{\mu_2}{\mu_1}\right)^2 P_0$$

$$P_n = \left(\frac{\mu_2}{\mu_1}\right)^2 . P_0 \quad \& \quad \sum_{n=1}^{m} P_n = 1$$

$$P_0 \left(1 + \left(\frac{\mu_2}{\mu_1}\right)^1 + \left(\frac{\mu_2}{\mu_1}\right)^2 + \dots \left(\frac{\mu_2}{\mu_1}\right)^m \right) = 1$$

$$\left(\sum_{n=0}^{m} x^n = \frac{x^{m+1} - 1}{x - 1} \right.$$

$$\left(\sum_{k=0}^{n} r^k = \frac{1 - r^{n+1}}{1 - r} \right.$$

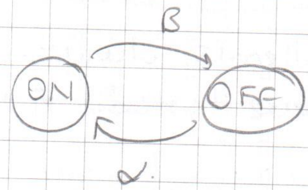
$$\left. \text{for } -1 < r < 1 \quad \sum_{k=0}^{\infty} r^k = 1/1-r \right)$$

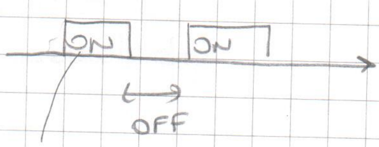
$$P_0 \cdot \frac{1}{\dfrac{\left(\frac{\mu_2}{\mu_1}\right)^{m+1}-1}{\left(\frac{\mu_2}{\mu_1}\right)-1}} = 1.$$

$P_1, P_2 \ldots$ P_0 dan hesaplanır.

② Voice communication systems

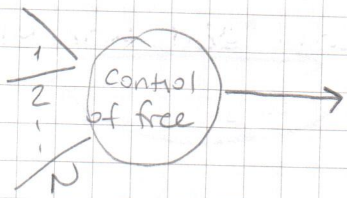
a)





R_{ON} ⇝ fixed data sized streams

⇝ nice model for burst systems



$$P_{ON} + P_{OFF} = 1$$

$$P_{ON} \cdot \beta = P_{OFF} \cdot \alpha \qquad P_{ON}\left(1 + \frac{\beta}{\alpha}\right) = 1$$

$$P_{ON} = \frac{\alpha}{\alpha+\beta} \qquad P_{OFF} = \frac{\beta}{\alpha+\beta}$$

$$R_i = X_i \cdot R_{ON}$$

$$X_i = \begin{cases} 1 & \text{if } P_{ON} \\ 0 & \text{if } P_{OFF} \end{cases}$$

Binomial Distribution

$$\boxed{\binom{N}{n} P_{ON}^{\,n} (1 - P_{ON})^{N-n}}$$

$P\{$ n active voice resources out of N $\}$

FABER-CASTELL

maximum $N \cdot R_{ON} \rightsquigarrow \binom{N}{N} P_{ON}^{N} (1-P_{ON})^{0}$

min $0 \cdot R_{ON} \rightsquigarrow \binom{N}{0} P_{ON}^{N} (1-P_{ON})^{N}$

ortalaması $N \cdot R_{ON}$ diyebilirsin yada hesaplarsın.

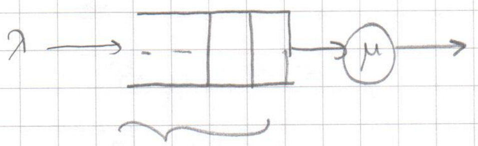
average $\underbrace{N \cdot P_{ON}}_{\text{avg of binomial dist}} \cdot R_{ON}$

b) Divide your channel into N frames.

R_{ON}'u sabitlediğin için $N\, R_{ON}$'u gönderebileceğini garantilemelisin (reserve)

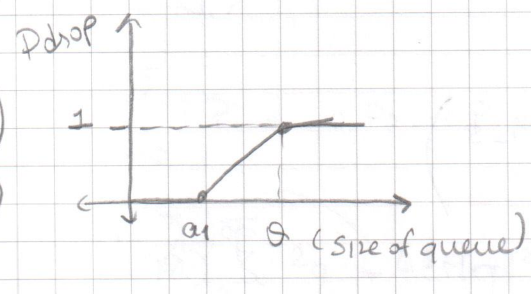
$BW < N \cdot B_N \rightsquigarrow$ a sıkı daha iyi: hepsinin ON state'te olması çok düşük ihtimal dersin. $N\, R_{ON}$ bandwidth allocate edeceğine daha az bw reserve ederim hepsi ON olunca bazı paketler atılır dersin)

3



Burada S'ten fazla paket olunca, gelen paketler p olasılığı ile kabul ediliyor.

Bu sistem, aslında RED sistemine bir örnek (Random Early Detection)

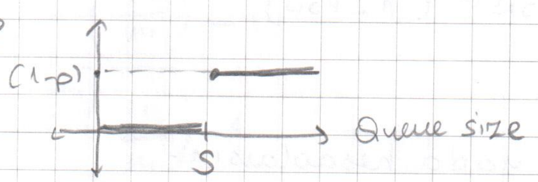


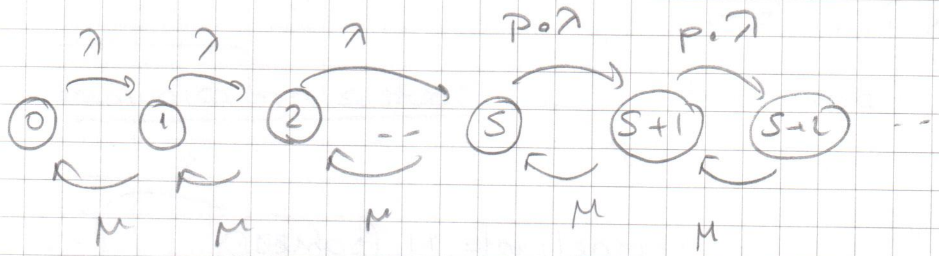
Meaningful if you have TCP like resources. $S \rightarrow \cdots \rightarrow D$

S waits ACK for specific duration. If ACK does not arrive, source understand packet is dropped. Reason for drop: some routers are very busy! S then decreases its rate.

Sorudaki RED

P drop





Stability condition $p \cdot \frac{\lambda}{\mu} < 1$

$\frac{\lambda}{\mu}$ 1 den büyük olsa bu stabiliteyi bozmaz sadece S e kdr ola adımları hızlı geçmenizi sağlar.

$P_1 = \frac{\lambda}{\mu} P_0$

$P_2 = \left(\frac{\lambda}{\mu}\right)^2 P_0$

$P_S = \left(\frac{\lambda}{\mu}\right)^S P_0$

$$P_{S+1} = P_S \cdot p\left(\frac{\lambda}{\mu}\right) = \left(\frac{\lambda}{\mu}\right)^{S+1} \cdot p \cdot P_0$$

$$\sum_{i=0}^{\infty} P_i = 1$$

$$1 = P_0 \left(1 + \sum_{i=1}^{S-1} \rho^i + \sum_{i=S}^{\infty} \rho^i p^{i-S} \right)$$

$$\rho^S \sum_{i=S}^{\infty} \rho^{i-S} \cdot p^{i-S} = \rho^S \sum_{i=S}^{\infty} (\rho p)^{i-S}$$

$$= \rho^S \sum_{i=0}^{\infty} (\rho p)^i \qquad \rho p < 1$$

$$= \rho^S \frac{1}{1-\rho p}$$

$$1 = P_0 \Big(\underbrace{1 + \sum_{i=1}^{S-1} \rho^i} + \rho^S \cdot \frac{1}{1-\rho p} \Big)$$

$$\sum_{i=0}^{S-1} \rho^i = \frac{\rho^S - 1}{\rho - 1}$$

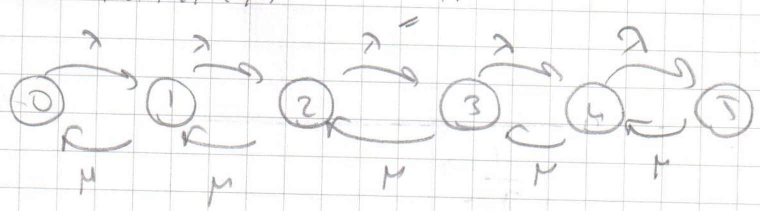
$$P_0 = \frac{1}{\dfrac{\rho^S - 1}{\rho - 1} + \dfrac{\rho^S}{1-\rho p}}$$

$$P\{\text{blocked}\} = \sum_{i=S}^{\infty} P_i (1-p)$$

---

4. Tarif edilen sistem: $M/M/1/5$ $\quad = 1/60$ pas/sec

#of passengers



$\mu = 2/60$ pas/sec

$$\frac{\lambda}{\mu} = \frac{1}{2}$$

$$\left.\begin{aligned} P_1 \mu &= P_0 \lambda \\ P_1 &= \left(\frac{\lambda}{\mu}\right) P_0 \\ P_2 &= \left(\frac{\lambda}{\mu}\right)^2 P_0 \end{aligned}\right\} \quad P_0 \left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} \right) = 1$$

$$\Downarrow$$

$$P_0 = \frac{32}{63}$$

$T = ? = N \cdot \lambda$

$$N = \sum_{i=0}^{5} i \cdot P_i = 0 \cdot P_0 + 1 \cdot P_1 + 2 \cdot P_2 + 3 \cdot P_3 + 4 \cdot P_4 + 5 \cdot P_5 = \frac{57}{63}$$

$N = \bar{\lambda} \cdot T$

Actual Arrival Rate $= ?$ $\quad \underline{\lambda(1 - P_5)}$

$$T = \frac{57}{63} \cdot \frac{1}{\bar{\lambda}}$$

FABER-CASTELL

Poisson arrivals λ

Exponential service times μ

$n>0$ $\quad p(n)\,[\lambda+\mu] = p(n-1)\,\lambda + \mu\, p(n+1)$

$$p(0)\,\lambda = \mu\, p(1)$$

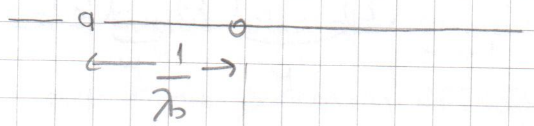
$$p(n) \sim \rho^n \qquad \rho = \frac{\lambda}{\mu}$$

$$p(n) = \left(\frac{\lambda}{\mu}\right)^n \left(1-\frac{\lambda}{\mu}\right)$$

No arrivals $\lambda = 0$

RESET mechanism in G networks.

2002



exp. dist

n I am going to add a batch of size n

$\tau(n)$ = Prob { Batch = n }

Batch size distribution.

If you want to say that when queue is empty, after an exp. distributed time (with λ_0) I'll give a batch

Very special case for G networks.

$$p(0).\lambda_0 = \mu\, p(1)$$

does not happen

$n \geq 0$ $\quad p(n).\mu = \lambda_0\, \tau(n) . p(0) + \mu\, p(n+1)$

when $n>0$ no arrivals.

$$\frac{p(0)}{p(1)} = \frac{\mu}{\lambda_0} \qquad \rho \sim \frac{\lambda_0}{\mu}$$

assume $p(n) \sim \rho^n$

$$\mu = \lambda_0 \tau(n) \frac{p(0)}{\underset{\rho^n}{p(n)}} + \frac{\mu\, p(n+1)\,\rho^{n+1}}{p(n)\,\rho^n}$$

$$\mu = \frac{\lambda_0 \tau(n)}{\rho^n} + \mu\rho$$

$$\frac{\lambda_0 \tau(n)}{\rho^n} = \mu(1-\rho)$$

$$\tau(n) = \overset{\rho^{-1}}{\frac{\mu}{\lambda_0}} \rho^n (1-\rho)$$

$$= [1-\rho]\, \rho^{n-1}$$

$$p(n) = \rho^n (1-\rho)$$

<u>RESETS</u> : you reset to steady state.

What if we want mix both of them?

$$p(0)\,[\underbrace{\lambda}_{\text{no normal arrival}} + \underbrace{\lambda_0}_{\text{no batch arrival}}]\, \underbrace{\mu}_{\text{no dep.}} = \lambda\, p(n-1) + \mu\, p(n+1) + \lambda_0 \tau(n)\, p(0)$$

$$p(0)\,[\lambda + \lambda_0 + \mu] = \mu\, p(1) \qquad \left(\begin{array}{c} p(n) \sim \rho^n \\ \rho \sim \frac{\lambda + \lambda_0}{\mu} \end{array} \right)$$

$$\lambda + \lambda_0 + \mu = \frac{\lambda}{\rho} + \mu\rho + \frac{\lambda_0 \tau(n)}{\rho^n}$$

$$\lambda\left(1 - \frac{1}{\rho}\right) + \mu(1-\rho) = \frac{\lambda_0 \tau(n)}{\rho^n}$$

$$\tau(n) = \frac{\rho^n(\rho-1)}{\rho}\frac{\lambda}{\lambda_0} + \frac{\mu}{\lambda_0}\rho^n(1-\rho)$$

$$= \frac{\rho^n(1-\rho)}{\lambda_0}\left[\mu - \frac{\lambda}{\rho}\right]$$

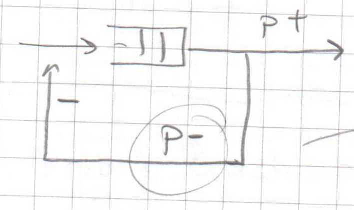
$$= \frac{\rho^n(1-\rho)}{\lambda_0}\left[\frac{\rho\mu-\lambda}{\rho}\right]$$

$$= \frac{\rho^{n+1}(1-\rho)}{\lambda_0}\lambda_0$$

$$\boxed{\tau(n) = \rho^{n-1}(1-\rho)}$$

$$\left(\rho = \frac{\lambda+\lambda_0}{\mu}\right)$$

1990 NEGATIVE CUSTOMERS



negative customer (Kuyruktan birini alacak)

$$p(0).\lambda = p(1)\mu[p^+.p^-] + p(2)\mu p^-$$

$n>0$

$$p(n)[\lambda+\mu] = p(n+1)\mu p^+ + p(n+2)\mu p^- + \lambda p(n-1)$$

Let us postulate a solution in form of $p(n) \sim \rho^n$

divide 1st eq with $p(0)$

$$\lambda = \rho\mu[p^+p^-] + \rho^2\mu p^- \qquad \left(\rho = \frac{\lambda^{\text{arrival}}}{\mu + \rho\mu p^-}\right.$$

normal dep. / 1 more if system is utilized !!

FABER-CASTELL

divide eq 2 by $p(n)$

$$[\lambda + \mu] = \rho \mu p^+ + \rho^2 \mu p^- + \frac{\lambda}{\rho}$$

3rd degree but may solve it

$$\left(\begin{array}{l} \rho = 1 \text{ is always a solution} \\ \lambda + \mu = \mu p^+ + \mu p^- + \lambda \\ \lambda + \mu = \mu \underbrace{(p^+ + p^-)}_{1} + \lambda \end{array} \right)$$

$$\lambda + \mu = \rho \mu p^+ + \rho^2 \mu p^- + \frac{\lambda}{\underbrace{\frac{\lambda}{\mu + \rho \mu p^-}}_{\rho}}$$

$$\lambda + \mu = \rho \mu p^+ + \rho^2 \mu p^- + \mu + \rho \mu p^-$$

$$= \rho \mu \cancel{(p^+ + p^-)} + \rho^2 \mu p^- + \mu$$

$$= \rho \mu (1 + \rho p^-) + \mu$$

$$\lambda + \mu = \frac{\lambda}{\cancel{\mu + \rho \mu p^-}} \cdot \cancel{\mu (1 + \rho p^-)}$$

$$\lambda + \mu = \lambda \mu \quad ?$$

$$\left(p(n) = \rho^n (1 - \rho) \qquad \rho = \frac{\lambda}{\mu + \rho \mu p^-} \right)$$

$\left(\lambda + \mu = \rho \mu p^+ + \rho^2 \mu p^- + \frac{\lambda}{\rho} \right.$ yi ρ ile çarp $\left. \right)$

$$\rho^3 \cdot \mu p^- + \rho^2 \mu p^+ - (\lambda + \mu) \rho + \lambda = 0$$

$$(\rho - 1) \cdot (a\rho^2 + b\rho + c) = 0$$

$\rho = 1$ is a solution

$a\rho^3 + (b-a)\rho^2 + (c-b)\rho - c = 0$

$a = \mu p^-$

$c = -\lambda$

$b - a = \mu p^+$

$b = \mu p^+ + \mu p^- = \mu$

$c - b = -\lambda - \mu$

$c = -\lambda - \mu + b = -\lambda$

$a\rho^2 + b\rho + c$

$\rho^2 \mu p^- + \mu \rho - \lambda = 0$ 2 values for ρ

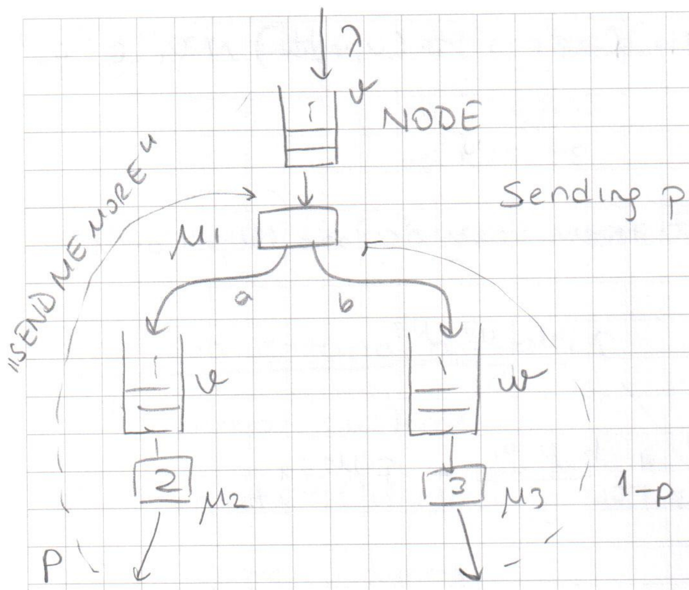
$$\frac{-\mu \mp \sqrt{\mu^2 - 4\lambda\mu p^-}}{2\mu p^-}$$

a negative ρ has no meaning

$$\rho = \frac{-\mu + \sqrt{\mu^2 - 4\lambda\mu p^-}}{2\mu p^-}$$

$\rho = \frac{\lambda}{\mu + \rho\mu p^-}$ non-linear ρ

(You may have the case a neg. customer may remove bunch of people)



Sending packets either to 2 (a)
or 3 (b)

⇒ When 2 finishes, it sends a message "Send me more", when 1 gets this message, sends <u>immediately</u>

⇒ Same for 3

"We can have more complicated scenarios like a network between sender & receiver, multiple receivers etc"

u, v, w #s in queues

$$P[u,v,w]\,[\lambda + \mu_1 + \mu_2 + \mu_3]$$

no arrival, no serv, no serv, no serv.

u'dan çıkan 1 v'ye girecek!

no dep.

$$= \lambda\, p(u-1, v, w) + \mu_1\, a\, p(u+1, v-1, w)$$

$$+ \mu_1\, b\, p(u+1, v, w-1)$$

$$+ \mu_2\, p\, p(u+1, v, w)$$

$$+ \mu_3\, p\, p(u+1, v, w)$$

$$+ \mu_2\, (1-p)\, p(u, v+1, w)$$

$$+ \mu_3\, (1-p)\, p(u, v, w+1)$$

Let us postulate a solution of the form $p(u,v,w) \sim \rho_1^u \rho_2^v \rho_3^w$

(we didn't say $a+b=1$)

(load balancing scenario: send me more degince no degil digerie ponder

$$(\lambda+\mu_1+\mu_2+\mu_3) = \frac{\lambda}{\rho_1} + \frac{a\mu_1\rho_1}{\rho_2} + \frac{b\mu_1\rho_1}{\rho_3} + p\mu_2\rho_1$$

$\lambda+\mu_1+\mu_2+\mu_3$

$$p\mu_3\rho_1 + \mu_2(1-p)\rho_2 + \mu_3(1-p)\rho_3$$

$$\Downarrow$$

$$\rho_1 = \frac{\lambda}{\mu_1(a+b) + \mu_2\rho_2 p + \mu_3\rho_3 p}$$

$$\rho_2 = \frac{\rho_1\mu_1 a + \rho_2\mu_2 p \rho_1}{\mu_2}$$ has to be busy

$$\rho_2\mu_2 = \frac{\rho_1\mu_1 a}{1-p\rho_1}$$

$$\rho_3 = \frac{\rho_1\mu_1 b + \rho_3\mu_3 p \rho_1}{\mu_3}$$

TRIGGERS

$$\lambda+\mu_1 = \mu_1(a+b) + \mu_2\rho_2 + \mu_3\rho_3$$

(suppose $a+b=1$

$$\lambda + \cancel{\mu_1} = \cancel{\mu_1} + \mu_2\rho_2 + \mu_3\rho_3$$

makes sense, everything goes to 2 or 3

$$\boxed{\lambda = \rho_1\mu_1(a+b) + \rho_2\mu_2 p\rho_1 + \rho_3\mu_3 p\rho_1}$$

Network with $\Rightarrow$ Negative Customers
$\Rightarrow$ Triggers
$\Rightarrow$ Resets

N Queues — in french it means tail,

Poisson external arrivals

Λ_i normal customers

Λ_{ri} resets (if queue is NOT empty does nothing, if empty, adds batch)

λ_i : negative customers
(comes & destroys a cust)

Λ_{ti} external trigger arrives queue$_i$

When a customer leaves a queue
P_{ij}^+ normal
P_{ij}^- negative
P_{ij}^0 reset
P_{ij}^t trigger

Q_{ij} when a trigger comes to queue i & makes him to send packet to queue j

triggers coming from outside / inside

λ_i^+ total flow of normal customers coming to queue i

$$\lambda_i^+ = \Lambda_i^+ + \sum P_{ji}^+ \mu_j \rho_j$$

$$+ \sum_j \rho_j \underbrace{\lambda_{tj}^+}_{\text{triggers from queue}_j \text{ to me}} Q_{ji}$$

$$\left(\lambda_{ti}^+ = \Lambda_{ti} + \sum_j \rho_j \mu_j P_{ji}^t\right)$$

$$\lambda_i^- = \Lambda_i^- + \sum_j \rho_j \mu_j P_{ji}^-$$

$$\lambda_{ir} = \Lambda r_i + \sum_j \rho_j \mu_j P_{ji}^{\circ}$$

Brings work

$$\rho_i = \frac{\lambda_i^+ + \lambda_{ir}}{\mu_i + \lambda_i^- + \lambda t_i}$$

removes work.

Again non-linear because λs include ρs

Keep traffic classes.

$\underline{f} = \mathcal{L}(\rho)$ vector equations of ρs (You have to prove existence of a UNIQUE solution)

fixed point iterations method

Does FPI method converge? In practice, typically after 10 iterations... (proof is NOT done yet)

$$p(\underline{k}) = \prod_{i=1}^{N} \rho_i^{k_i}(1-\rho_i)$$

If you remove negative customers ⇒ neural network?

Gelenbe "learning in the recurrent random neural networks"

1993 Backprop. $O(N^3)$

(conventional backprop $O(N^2)$)

Networks & Flows

Network is a finite, connected, directed graph

a vertex with $\underbrace{d^+(x)}_{\text{\# of outgoing edges}} > 0$ is called source

$\underbrace{d^-(y)}_{\text{incoming}} > 0 \rightsquigarrow$ sink

A flow for network N, associates non-negative integer $f(u,v)$ with each edge (u,v) of N such that for all vertices other than x or y

$$\underbrace{\sum_u f(u,v)}_{\text{outgoing flow}} = \underbrace{\sum_v f(v,u)}_{\text{incoming flow}} \rightsquigarrow$$ Conservation of flow at each node (If it is not source / sink)

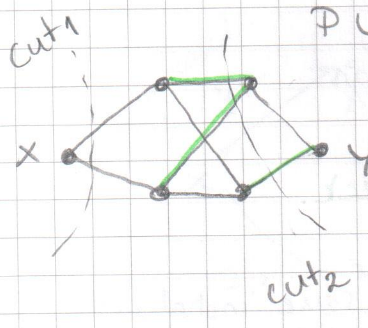
each edge has capacity $c(u,v)$

$$\boxed{0 \leq f(u,v) \leq c(u,v)}$$

A cut of network $N=(V,E)$ is a cut-set of underlying graph

Cut partitions V into 2 subsets P & $\overline{P}$ such that P contains x & $\overline{P}$ contains y

$$P \cap \overline{P} = \emptyset$$
$$P \cup \overline{P} = V$$



Capacity of Cut

$$K(P,\overline{P}) = \sum_{\substack{u \in P \\ v \in \overline{P}}} c(u,v)$$

for cut$_2$ $K(P,\overline{P}) = \sum c(u,v)$

The value of flow $F(N)$ for network N is defined to be netflow leaving source x

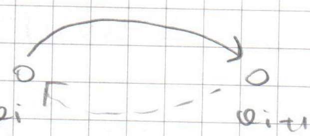
$$F(N) = \sum_{v} F(x,v) - \sum_{v} F(v,x)$$

$$F(N) = \text{flow from } P \text{ to } \overline{P} - \text{flow from } \overline{P} \text{ to } P$$

Maximizing Flow in a Network

$Q = (v_0, \ldots v_k)$ as path
v_0: source, v_k: dest

(v_i, v_{i+1}) forward edge
(v_{i+1}, v_i) reverse edge



(we'll place this edge even if it does not exist in real topology)

For a given flow $F(N)$ an augmenting path is a path Q of N such that for each $(v_i, v_{i+1}) \in Q$

a) If (v_i, v_{i+1}) is a forward edge $\Rightarrow$

$$\Delta_i = c(v_i, v_{i+1}) - f(v_i, v_{i+1}) > 0$$

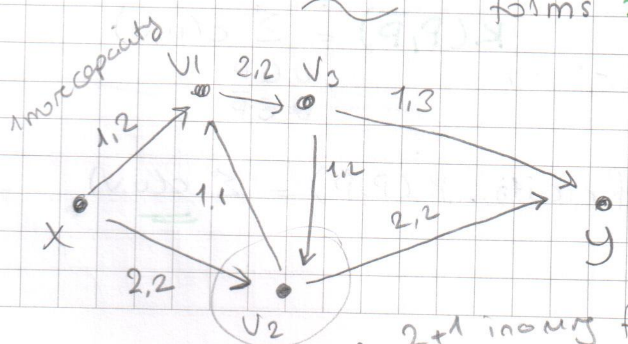
b) If (v_i, v_{i+1}) is a reverse edge

$$\Delta_i = f(v_i, v_{i+1}) > 0$$

If Q is an augmenting path, then we define Δ as

$$\min \Delta_i > 0$$

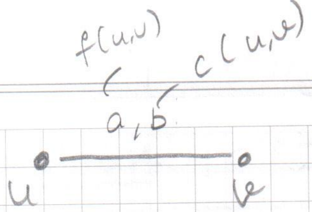
forms bottleneck.



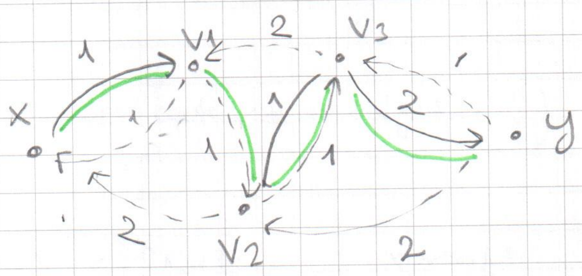
has some initial values

ex v2 has 2+1 incoming flow
1+2 outgoing flow ✓

FABER-CASTELL



we form another graph



Try to find a path from x to y —

Now, look how much you may add

$x - v_1 - v_2 - v_3 - y$

$\Delta = \min\{1, 1, 1, 2\} = 1$ // you will enhance your flow by 1

After finding path, enhancing flow

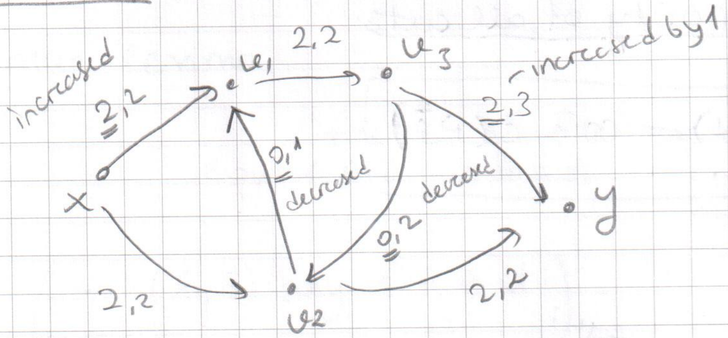
a) if (v_i, v_{i+1}) is a forward edge increase flow value on this link

$$f(v_i, v_{i+1}) \leftarrow f(v_i, v_{i+1}) + \Delta$$

b) if (v_i, v_{i+1}) is a reverse edge, decrease flow value

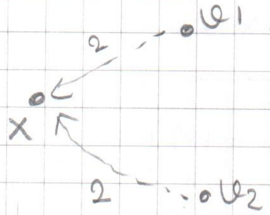
$$f(v_i, v_{i+1}) \leftarrow f(v_i, v_{i+1}) - \Delta$$

apply to example



can we enhance again?

forward ve reverse edgeleri tekrar oluştur.



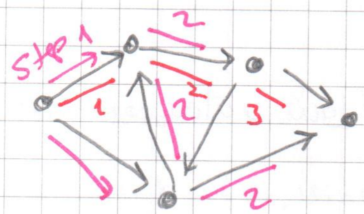
x'ten çıkan edge olmadığı için path bulamayacağız.

$\boxed{\text{maxflow} = 4}$

Note how to represent a graph? Matrix N[4,1] => (no edge)
= 1 (edge)
= C cap(4,1)

Flowu da saklamak için matris elemanı $\frac{c}{f}$ şeklinde de olabilir.

x'ten y'ye path ararken DFS de olur BFS de.



- BFS 2 adımda
- DFS 3 adımda

buldu.

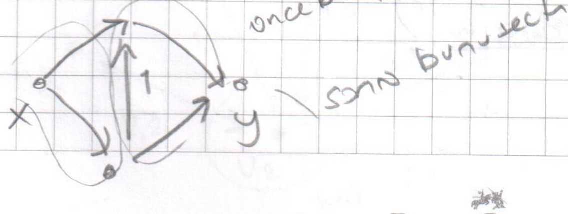
Max-Flow, Min-Cut Theorem

For a given network, the maximum possible value of the flow is equal to the min. capacity of all cuts.

$$\boxed{\max F(N) = \min c(P\bar{P})}$$

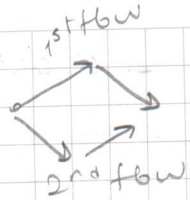
MaxFlow?

ex



Worst case $O(I.E)$ — # of edges

best case



After 2 iterations you find max flow

$2 \cdot I \cdot V$ exactly

for worst case # of edges $\binom{n}{2}$ $\frac{n(n+1)}{2} = O(n^2)$
$= O(V^2)$

algoritmayı enhance etmek için "select path with shortest distance" heuristicini uygulayabilirsin

Edmonds & Karp Algorithm finds maximum flow by using augmentation

capacity of cut $\quad cap(A,B) = \sum_{e \text{ out of } A} c(e)$ $\quad s \in A \quad t \in B$

value of flow $\quad v(f) = \sum_{e \text{ out of } s} f(e)$

For each $e \in E \quad 0 \leq f(e) \leq c(e)$
For each $v \in V - \{s,t\} \quad \sum_{e \text{ into } v} f(e) = \sum_{e \text{ out of } v} f(e)$

flow value lemma

$$\sum_{e \text{ out of } A} f(e) - \sum_{e \text{ into } A} f(e) = v(f)$$

"Net flow sent across cut is equal to the amount leaving s"

- (A,B) any s-t cut
- f be any flow

$v(f) \leq cap(A,B)$ & if $v(f) = cap(A,B)$

max flow $\quad$ min cut

Greedy Alg
- start with $f(e) = 0 \quad \forall e \in E$
- find s-t path P where each edge has $f(e) < c(e)$
- augment flow along path
- repeat until you get stuck.

local optimality $\neq$ global opt

Residual Graph

"Undo" flow sent

residual cap.

$$c_f = \begin{cases} c(e) - f(e) & \text{if } e \in E \\ f(e) & \text{if } e^R \in E \end{cases}$$

$G_f = (V, E_f)$

$E_f = \{ \{e : f(e) < c(e)\} \cup \{e^R : c(e) > 0\} \}$

Ford_Fulkerson (G, s, t, c)

```
foreach e ∈ E  f(e) ← 0
        Gf ← residual graph

while (there exists augmented path P)
    f ← augment (f, c, P)
    update Gf
return f
```

```
Augment (f, c, P)
    b ← bottleneck (P)
    foreach e ∈ P
        if e ∈ E   f(e) ← f(e) + b       ~ forward edge
        else       f(e^R) ← f(e) - b     ~ reverse edge
return f
```

FABER-CASTELL

if max cap is $\underline{C}$, alg may take C iterations

Choose augmenting paths so that ! can find augmenting path efficiently & few iterations

Edmonds - Karp ⇝ choose augmenting paths with → max bottleneck cap.
→ sufficiently large bott. cap.
→ fewest # of edges

19/04/2013
10th week

Solution can be in form of

- Analytical Model
- Numerical Result
- Simulation

λ $\frac{1}{\lambda}$ |————|

neg exp dist having avg $1/\lambda$

How can you obtain this inter-arrival times, if you have uniform r.n. generator?

Cumulative distribution function gives value between 0 1

cdf = random # you generate

for Poisson $F_x(x) = 1 - e^{-x/a}$

$1 - e^{-x/\lambda}$ = random #

$(1 - \text{rand\#}) = e^{-x/\lambda}$

$\ln(1 - \text{rand \#}) = -\frac{x}{\lambda}$

$\Rightarrow p(1 - \text{rand \#}) = p(\text{rand\#})$

$x = -\lambda \ln(1 - \text{rand\#})$

$\boxed{x = -\lambda \ln(\text{rand\#})}$

x = interarrival period

averaging results

avg of several runs.

⇝ you may NOT see flactuations.

Change seed value & get random sequences in different order

CI

with 0.95 probability my value will be in this range.

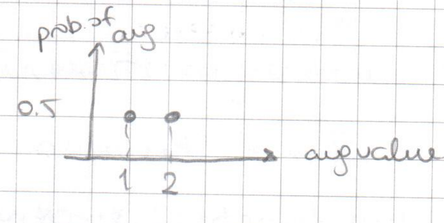
how to calculate confidence interval

Remember central limit theorem (The distribution of an average tends to be Normal even if distribution from which average is computed is non-Normal. Furthermore this normal distribution will have same mean as parent dist & $\frac{\text{variance}}{\text{samplesize}}$ variance.)

If you have 2 values ex 1, 2

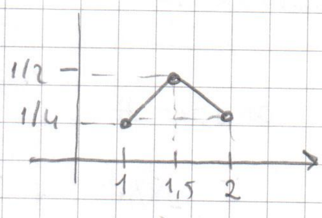
what's prob. of obtaining avg value?

you have 1 pick (you choose 1 or 2)



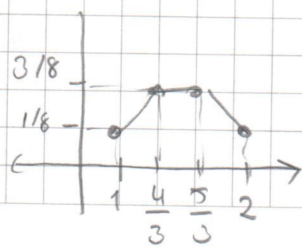
select 2 values & give me avg

11 – 1
12 – 1.5
21 – 1.5
22 – 2



pick 3 values

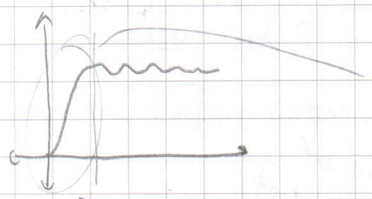
1 1 1	1	prob =	1/8
1 1 2	4/3	=	3/8
1 2 1	4/3	=	3/8
1 2 2	5/3	=	3/8
2 1 1	4/3	=	3/8
2 1 2	5/3	=	3/8
2 2 1	5/3		3/8
2 2 2	2		1/8



As we pick more values, prob of average curve converges to normal distribution.

- How to simulate and getting multiple results?

Replication Method re-run everything (usually exactly same scenario)

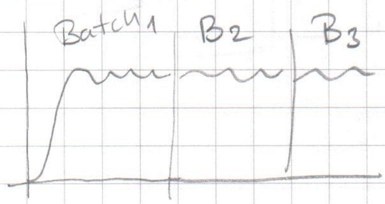


initial transient (initial bias)

↔ to remove bias, discard this part. (ex: queuing system, empty at initial state, mean waiting time very small initially)

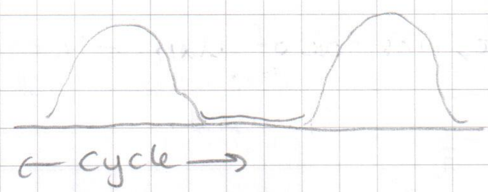
Batch Mean Method

Simulasyonu durdurmadan batchleri ard-arda calıştır



Regenerative Method

In a single program re-run sim code.



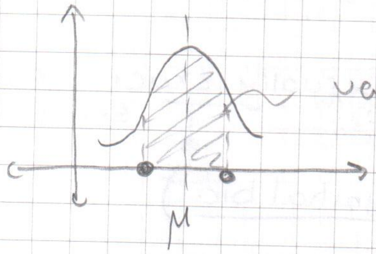
Observed values $x_1, x_2, \ldots x_n$ for value of x, say the # of calls in system are independent, the confidence interval $[\mu_\alpha^+, \mu_\alpha^-]$ of mean $\mu = E[X]$ with prob $1-\alpha$ is calculated as follows:

Let σ^2 be variance of X, from central limit theorem, sample mean

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$ follows normal dist. with mean μ & variance $\frac{\sigma^2}{n}$ as $n \to \infty$

FABER-CASTELL

$N(\mu, \frac{\sigma^2}{n})$ Using this theorem, confidence intervals μ_α^+ & μ_α^- with prob $(1-\alpha)$ %100 are determined by:



values you obtain within this range with $(1-\alpha)$ prob.

$$\left.\begin{matrix}\mu_\alpha^+ \\ \mu_\alpha^-\end{matrix}\right\} \quad \bar{X} \mp \frac{\sigma}{\sqrt{n}} u_{\alpha/2}$$

value of normal dist. such that

$$P(X > u_{\alpha/2}) = \frac{\alpha}{2}$$

Usually σ^2 is unknown & from sim results we obtain <u>sample variance</u>

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x}_i)^2$$

In this case, instead of $u_{\alpha/2}$, the corresponding value of t function $t_{n-1,\, 1-\frac{\alpha}{2}}$ is used.

degree of freedom

<u>example</u> Suppose 10 observations 1.2, 1.5, 1.68, 1.89, 0.95, 1.49, 1.58, 1.55, 0.5 & 1.09 are from a normal distribution with unknown mean μ and our objective is to construct %90 CI for μ

$$\bar{X}(10) = \frac{\sum_{i=1}^{10} x_i}{10} = 1.34$$

$$S^2(10) = \frac{1}{9} \sum_{1}^{10} (\bar{x} - x_i)^2 = 0.179$$

$$(1-\alpha)\, 100 = \frac{90}{1} \qquad \alpha = 0.1$$

$$\mu_{x}^{\pm} = \bar{x}(10) \pm \underset{(n-1)\quad 1-\frac{\alpha}{2}}{t_{9,\,0.95}} \cdot \sqrt{\frac{s^2(10)}{10}}$$

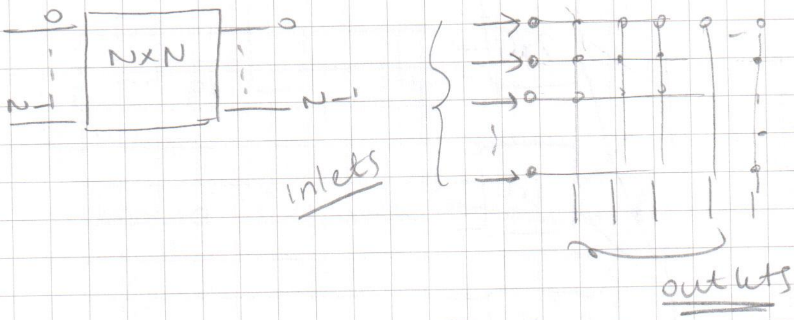
$$= 1.34 \pm 1.833 \sqrt{\frac{0.17}{10}} = 1.34 \mp 0.24$$

1.58
1.34
1.10

(%'yi arttırırsan $\alpha \uparrow$
t'den gelen değer artar, aralığı büyür)

Switching Networks

Interconnection networks



According to address, corresponding outlet switch is activated

# of components = N^2
(complex systems)

Multi-Stage switches

Suppose, inlets have slot based incoming traffic

0 — at a slot, any of inlets can have packets.
1
2
⋮
N-1

p: prob. of a packet occurrence at each inlet
"incoming line is busy with prob. p"

What is the probability of observing packet at any of outlets

Prob of having a packet at outlet 0 = 1 − $\underbrace{\text{observing no packets at outlet 0}}$

$\frac{p}{N}$: selecting any inlet

$\left(1-\frac{p}{N}\right)$ not selecting outlet 0

$$1-\underbrace{\left(1-\frac{\rho}{N}\right)^{N}}$$

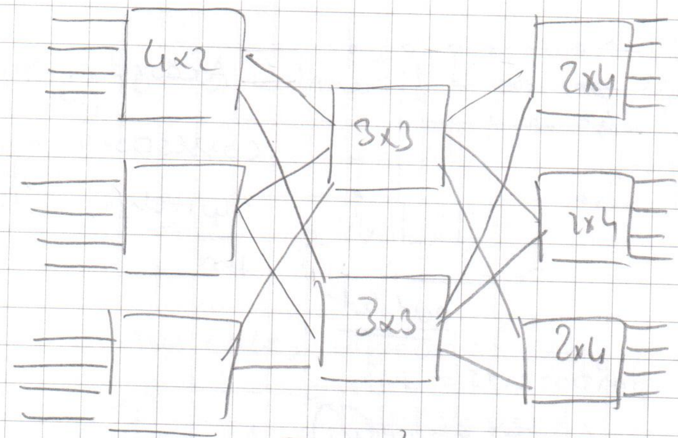
all inlets do NOT select outlets

Crossbar switch •) simple

•) quiet good throughput! If all packets in all inlets choose different outlets, you'll have NO LOSS

Multi-stage Switches

- Banyan
- Clos

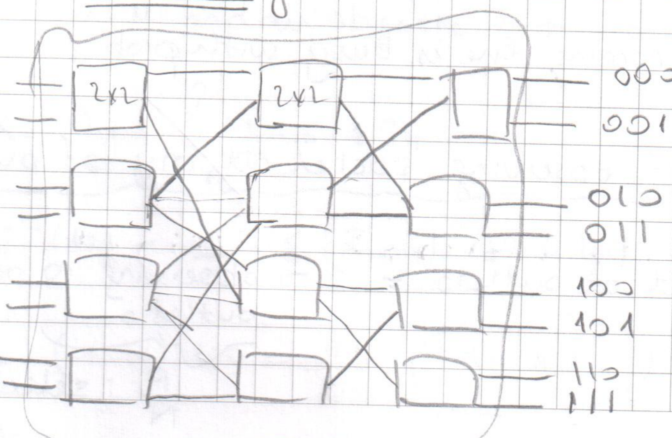


intermediate cross-bars.

* increasing # of these intermediate crossbars increases # of parallel paths.

* Multi-stage switches increases delay (simple cross bar: 1 hop
ex multi stage 3 hops)

Banyan self-routing switch



# of stages
$= \log_2 N$
$= \log_2 8 = 3$

A packet arrives (at any of inlets) with dest (e.g 011) firststage is effected by 1st bit, 2nd stage with 2nd bit...

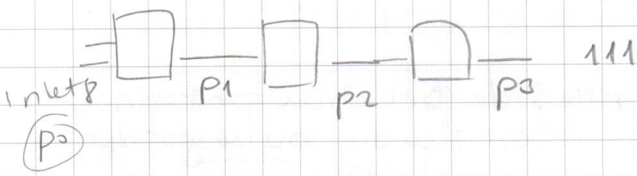
inlet$_0$ —— dest$_1$ = 011 } even if dest addresses are different
inlet$_1$ —— dest$_2$ = 001 } one of them will be LOST.

⇓

internal blocking problem of Banyan networks.

What's prob. of observing packet at inlet8

(we have uniform selecting property →1/2 →1/2)



P_0 is known (eg with Bernoulli dist.)
prob of seeing a packet at a given time is given.

$$P_1 = 1 - \left(1 - \frac{P_0}{2}\right)^2$$

selecting outlet$_1$

not selecting outlet$_1$

no inlet selects outlet$_1$

some selects outlet$_1$

$$P_i = 1 - \left(1 - \frac{P_{i-1}}{2}\right)^2$$

(Sume hoca'dan)

Assume we place some buffers (try to prevent packet loss in intermediate stages)

(But introduce more delay)

FABER-CASTELL

How can we model such switching system?

- each switching node with simple M.C.

<u>state</u>

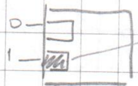
0 ready (idle)
1 blocking (busy)

$q_{k,\ell}(x)$ Steady state prob. that a packet is available for entering x^{th} buffer at beginning of clock period

stage # node # 0: upper buffer 1: lower "

$r_{k,\ell}(x)$ steady state prob. that a blocked packet can move during clock period.

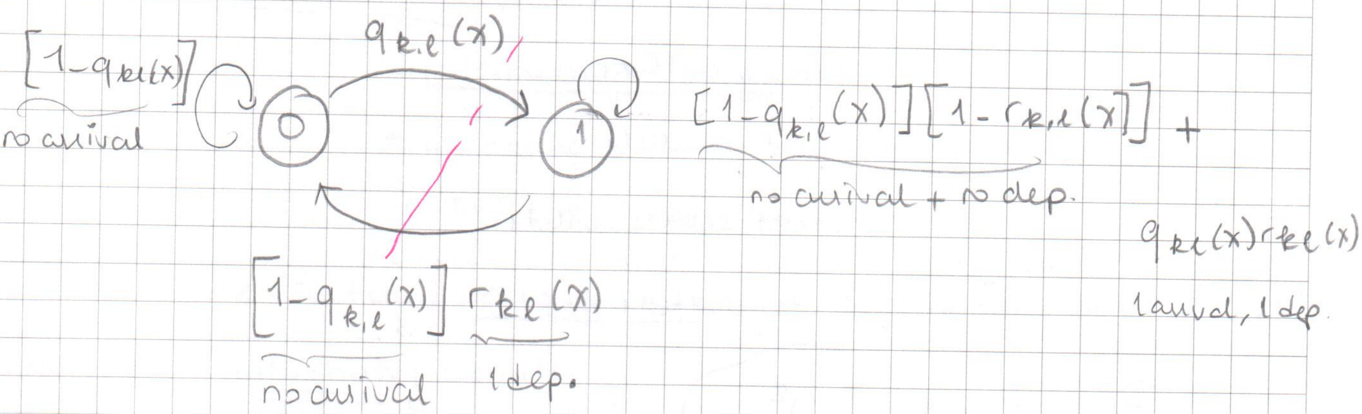
$r_{1,2}(1)$



node 2 de bekletilen paketin cıkıslarda bu ine gonderilme olasılıgı.

stage # 1
node # 2

<u>x^{th} buffer of switching node</u>



•) $P_{k,\ell}(x)^0 + P_{k,\ell}(x)^1 = 1$

•) $P_{k,\ell}(x)^0 = P_{k,\ell}(x)^1 \cdot \dfrac{[1-q_{k\ell}(x)]\,r_{k\ell}(x)}{q_{k\ell}(x)}$

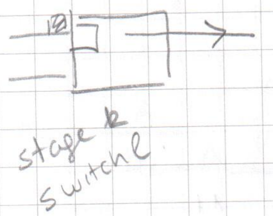
movability of packet

$r_{ke}(x, \overbrace{y}^{0 \text{ or } 1})$ ⇝ conditional prob. that a blocked packet can move given wants to exit via output y

$$r_{ke}(x) = r_{ke}(x,0) \cdot \frac{\overbrace{\lambda_{ke}(x,0)}^{\text{load from } x^{th} \text{ inlet to } y^{th} \text{ outlet}}}{\lambda_{ke}(x,0) + \lambda_{ke}(x,1)}$$

$$+ r_{ke}(x,1) \cdot \frac{\lambda_{ke}(x,1)}{\lambda_{ke}(x,0) + \lambda_{ke}(x,1)}$$

$r_{ke}(0,0)$ a packet buffered at inlet0 of (k,e) can leave outlet0 at next clock cycle when there's no contention from inlet1 & next stage is ready to receive packet.



Possibilities A) inlet1 is at state0 (no packet)

B) inlet1 is at state1 but buffered packet is destined to outlet1

C) inlet1 is at state1, buffered packet destined to outlet0 but contention is resolved in favor of inlet0.

AND (Backpressure sinyaline de bakılmalı)

D) Destination buffer is empty

E) " " " busy but its content is moving out.

$$\boxed{r_{ke}(0,0) = (A \vee B \vee C) \wedge (D \vee E)}$$

A ⇝ $P_{ke}^{0}(1)$

B ⇝ $P_{ke}^{1}(1) \cdot \dfrac{\lambda_{ke}(1,1)}{\lambda_{ke}(1,0) + \lambda_{ke}(1,1)}$

contention is solved in favor of inlet with 1/2 prob!

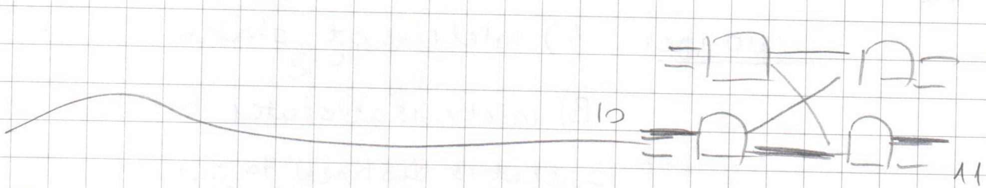
$C \rightsquigarrow \frac{1}{2} \; P^{1}_{kl}(1) \; \frac{\lambda_{kl}(1,0)}{\lambda_{kl}(1,0) + \lambda_{kl}(1,1)}$ desired to outputs

$D \rightsquigarrow P^{0}_{(k+1),m}(i) + P^{1}_{(k-1)m}(i) \cdot r_{(k+1),m}(i)$

next stage
destination node
dest. inlet

$r_{kl}(0,1)$
$r_{kl}(1,0)$
$r_{kl}(1,1)$

} Benzer şekilde yazılabilir.
Neden gerekli?
r'ler sayesinde, path üzerinde delay-i hesap edebilirsin...

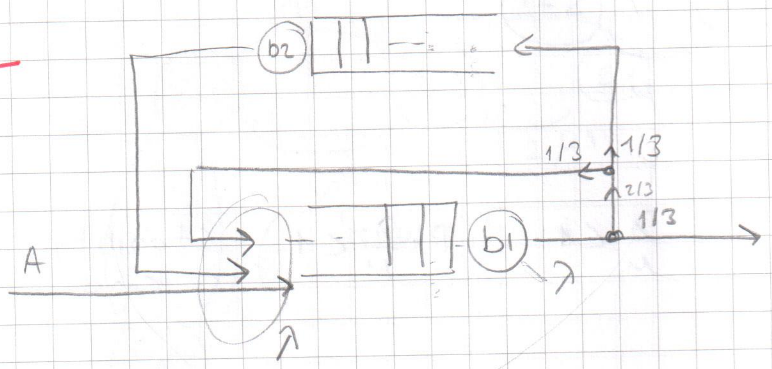


$\underline{\text{Delay}} = \frac{1}{r_{0,1}(0,1)} + \frac{1}{r_{1,1}(1,1)}$

stage 0 switch 1
from inlet 0
outlet 1
stage 1 switch 1
from inlet 1 to outlet 1

MIDTERM

Q1



a) $A + \frac{\lambda}{3} + \frac{\lambda}{3} = \lambda \qquad \underline{A = \frac{\lambda}{3}}$ for stable system

b) Avg # of customers waiting or in service at each queue.

$$N_1 = \frac{\rho_1}{1-\rho_1} = \frac{\lambda_1}{\mu_1 - \lambda_1} = \frac{3A}{b_1 - 3A}$$

$$N_2 = \frac{A}{b_2 - A}$$

c) Total system delay

$N = \lambda_1 . T$

$$T = \frac{N_1 + N_2}{A} = \frac{N_1 + N_2}{A}$$

packet 1/λ sec

1/λ sec

Q2

interarrival ⇝ 1/λ seconds → λ

transmission time ⇝ $1/\mu$

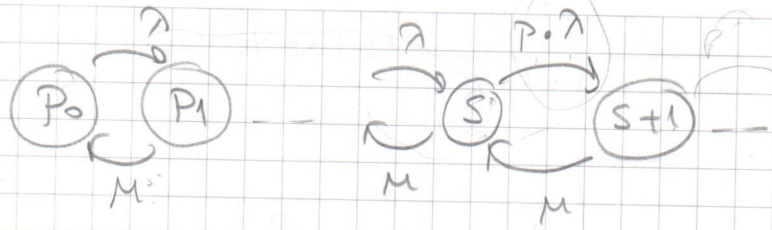
1 pack × λ

1/λ sec

1 sec

Buffer size is greater than S ⇒ new arrival is regarded with (1-p)

- Model this system
- Identify stability condition
- Calculate prob of new arrival to be refused.



$\frac{\text{arrival rate}}{\text{service rate}} < 1$ $\quad \frac{\lambda}{\mu} < 1 \quad \left(p \cdot \frac{\lambda}{\mu} < 1\right)$ stability cond.

$$\begin{cases} \sum p_n = 1 \\ P_k = \left(\frac{\lambda}{\mu}\right)^k P_0 & k < S \\ P_k = \left(\frac{\lambda}{\mu}\right)^S \left(\frac{p\cdot\lambda}{\mu}\right)^{k-S} P_0 & k \geq S \end{cases}$$

$$\sum_{n=0}^{S-1} \rho^n p_0 + \rho^S \sum_{n=S}^{\infty} (\rho p)^{k-S} P_0 = 1$$

A

$$\rho^S \sum_{n=0}^{\infty} (\rho p)^k P_0$$

$(\rho p < 1)$

$$+ \rho^S \frac{1}{1-\rho p} P_0 = 1$$

B

$$P_0 = \frac{1}{A+B}$$

Prob. of blocking $\rightarrow (1-p) \sum_{n=S}^{\infty} P_n$

0.3 M/M/m system

$m_0 < m$

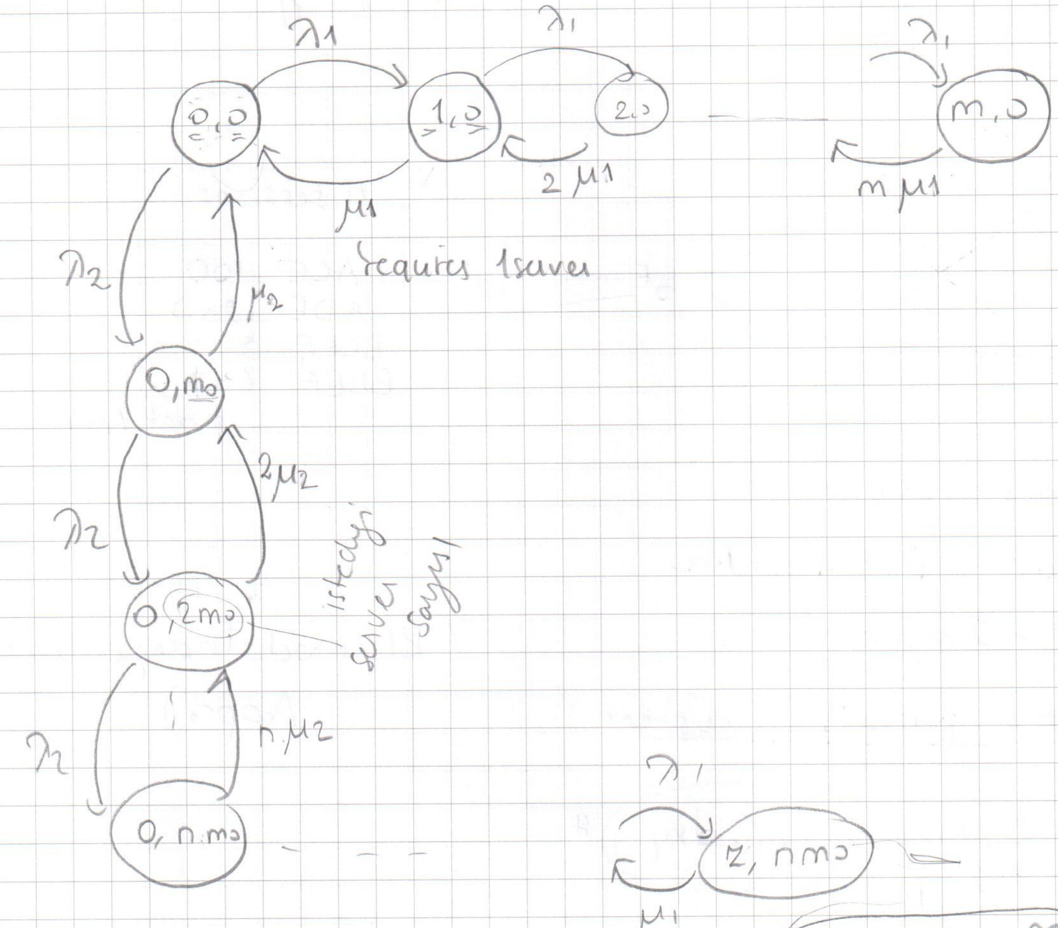
μ rate

2 poisson arrival system $\begin{cases} \lambda_1 \text{ requires 1 server} \rightarrow \mu_1 \\ \lambda_2 \text{ " } m_0 \text{ servers simultaneously} \rightarrow \mu_2 \end{cases}$

if servers are full packets are droped

λ_2 de m_0 eleman geliyor

μ_1/μ_2 are service rates customers require



$n.m_0 \leq m$

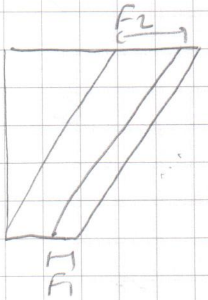
$z + n\,m_0 = m$

$m_0 > z$

snow old i:

$$F_1 = \frac{\lambda_1}{\lambda_1+\lambda_2} \sum_{N_1+N_2=m} P(N_1,N_2) \rightarrow \sum_{i=0}^{\lfloor \frac{m}{m_0} \rfloor} P((m - m_0.i), i.m_0)$$

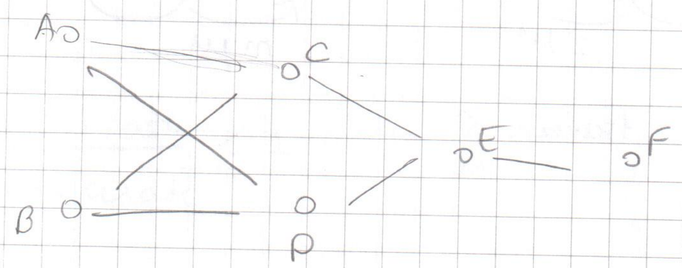
$$F_2 = \frac{\lambda_2}{\lambda_1+\lambda_2} \sum_{m-m_0 \leq N_1+N_2 \leq m} P(N_1,N_2) \rightarrow \sum_{i=0}^{\lfloor \frac{m}{m_0} \rfloor} \sum_{j=0}^{m_0-1} P[(m - i m_0 - j), (i.m_0)]$$



reject god

Q4) Problem 2.8 in Fundamentals of Queueing Theory ?

Q5)



4 sessions

ACE 100
ADE 200
BCEF 500
BDEF 600
pack/min

packet length = 1000 bit
all lines have cap 50 kbit/sec
prop delay = 2 msec.

(Kleinrock Independence Appro)

a) Avg # of packets in system

$$N = \sum_{x,y} N_{xy} + \lambda_{xy} \cdot d_{xy} \quad *$$

$$N_{xy} = \frac{\lambda_{xy}}{\mu - \lambda_{xy}}$$

b) Avg delay per packet (regardless of session)

$$T = \frac{1}{\gamma} N$$

$\gamma \Rightarrow \sum_{s} x_s$

c) Avg delay per packet of each session

⇒ calc delay on each link seperately ?

⇒ sum up for each session

5 + 1/45

55/3

5/3

145/3