Pramod Parajuli
Simulation and Modeling, CS-331

Chapter 8
Queueing Systems

Queueing Systems

A queueing system consists of one or more servers that provide service of some kind to arriving customers

If the servers are busy, the customers join one or more queues in front of the servers, hence queueing system

System	Servers	Customers
Bank	Tellers	Customers
Hospital	Doctors, Nurses, beds	Patients
Computer System	CPU, I/O devices	Jobs
Manufacturing systems	Machines, Workers	Parts
Airport	Runways, Gates	Airplanes, travelers
Communications network	Nodes, links	Messages, Packets

Queueing Systems

State of the system; the number of units in the system

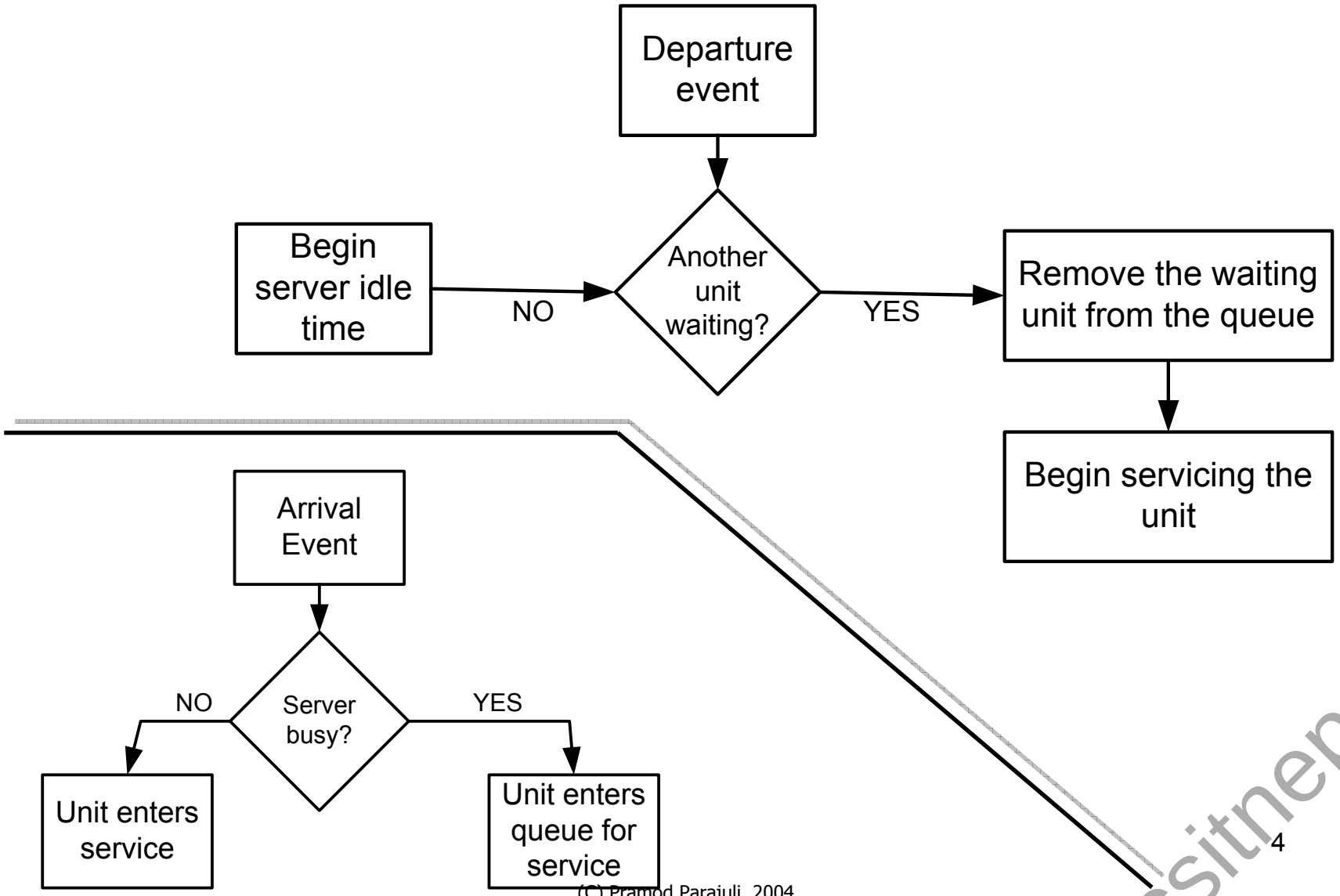
Queues that grow without bound are known as explosive or unstable queues

Queueing system includes; the server, the unit being services (?), and units in the queue

Simulation time is determined by simulation clock

		Queue status	
		Not empty	Empty
Server status	Busy	Enter queue	Enter queue
	Idle	Impossible	Enter service

Queueing Systems



Source: www.csitnepal.com

Characteristics of Queueing Systems

- The calling population
  - -The population of potential customers
  - -In systems with a large population of potential customers, the calling population is considered to be infinite
  - -Although the population in finite, if very small fraction of the potential customers are served by the system, then the population is supposed to be infinite
- System capacity
  - -Number of customers waiting in the queue is limited
- The arrival process
- Queue behavior and descipline
- Service times and service mechanism

Components Queueing Systems

-Arrival process (pattern)

-Service mechanism

-Queue discipline

Arrival Process

The simulation time start when first customer arrives to the system

The arrival time is random and hence inter-arrival time is also random

Customer	Inter-arrival time	Arrival time on clock
1	-	0
2	2	2
3	4	6
4	1	7
5	2	9
6	6	15

Arrival Process

If arrivals vary stochastically, it is necessary to define the probability function of the inter-arrival times

Two or more arrivals may be simultaneous. If n variables are simultaneous, then n-1 of them have zero inter-arrival times

Notation;

Ta	Mean inter-arrival time
λ	Mean arrival time

They are related as; $\lambda = \frac{1}{T_a}$

Arrival Process

Example - office;

- five days a week
- Eight-hour day
- 800 calls a week
- Model the office using time scale of minutes

1 week = 40 working hours. Therefore inter-arrival time
= (40 * 60) / 800
= 3 minutes

i.e. 0.333 calls per minute

- In infinite population model, the arrival rate is not affected by the number of customers
- If the arrival process is homogenous (no rush-hours) then arrival is linear

Arrival Process

On the other hand, the arrival time is very much uncertain and hence its distribution is important

Let's define $A_{0(t)}$ be the arrival distribution to be defined as; ***(different writers use different notations)***

It is the probability that an inter-arrival time is greater than 't'

Again, if the cumulative distribution function F(t) is the probability that an inter-arrival time is less than t, then

$$A_{0(t)} = 1 - F(t)$$

Therefore, takes maximum value of '1' at t = 0

Poisson Arrival Patterns

- Often used to model arrival processes with constant arrival rates
  - Gives the number of events that occur in a given period
- Probability of an arrival in an interval dt is proportional to dt
- If is the mean number of arrivals per unit time, then probability of an arrival in dt is

$$\lambda.\Delta t$$

- Therefore, PDF of inter arrival time is given by;

$$f(t) = \lambda.e^{-\lambda t} \quad (t >= 0)$$

Poisson Arrival Patterns

In other words, a probability function is said to have poisson pattern if

$$p(x) = \begin{cases} \dfrac{e^{-\alpha}\alpha^{x}}{x!}, & x = 0,1,\cdots \\ 0, & otherwise \end{cases}$$

- Where α is the mean rate and must be positive
- E(X) = V(X) = α

Poisson Arrival Patterns

So now, arrival distribution will be;

$$A_0(t) = e^{-\lambda t}$$

$$A = \frac{f(t)}{\lambda}$$

$$A = \frac{\lambda . e^{-\lambda t}}{\lambda}$$

$$A = e^{-\lambda t}$$

Here, λ is mean no. of arrivals per unit time and arrivals in a period of time 't' is again a random variable.

So now, probability of 'n' arrivals occurring in a period of length 't' is given by (exponential)

$$p(n) = \frac{(\lambda t)^n e^{-\lambda t}}{n!}$$

Poisson Arrival Patterns

Example;

Arrival Number	Arrival Time	Inter-arrival Time	Arrival Number	Arrival Time	Inter-arrival time
1	12.5	12.5	11	136.4	21.4
2	15.1	2.6	12	142.7	6.3
3	44.1	29.0	13	151.2	8.5
4	62.6	18.5	14	162.5	11.3
5	65.3	2.7	15	167.2	4.7
6	67.6	2.3	16	172.9	5.7
7	71.0	3.4	17	179.8	6.9
8	92.5	21.5	18	181.6	1.8
9	106.5	14.0	19	185.0	3.4
10	115.0	8.5	20	194.9	9.9

Poisson Arrival Patterns

Example;

The inter arrival time is ranging from 1.8 to 29.0. Look at graph plot at page 149 of text book.

Taking the average of the inter arrival time;

9.74

Therefore, estimated arrival rate is = $\lambda = \frac{1}{T_a}$

= 0.103

Now, let's group the time period into 10 different range of size 20

Poisson Arrival Patterns

Example;

Time Period	No. of Arrivals	Time Period	No. of Arrivals
0-20	2	100-120	2
20-40	0	120-140	1
40-60	1	140-160	2
60-80	4	160-180	4
80-100	1	180-200	3

λ= 0.103, t = 20, and value of p(n) for n = 0, 1, 2, 3, 4, 5 will be;

Poisson Arrival Patterns

Example;

No. of arrivals in 20 mins	Actual No. of Occurrences	p(n)	Expected number of occurrences
0	1	0.128	1.3
1	3	0.266	2.7
2	3	0.272	2.7
3	1	0.187	1.9
4	2	0.096	1.0
5	0	0.040	0.4

Compare the values in 2nd and 4th column

Poisson Arrival Patterns

Example;

Number of people logging onto a computer per minute is Poisson Distributed with mean 0.7

What is the probability that 5 or more people will log onto the computer in the next 10 minutes?

Solution?

Must first convert the mean to the 10 minute period – if mean is 0.7 in 1 minute, it will be (0.7)(10) = 7 in a ten minute period

$P(X >= 5) = 1 - P(0) - P(1) - P(2) - P(3) - P(4)$

$= 1 - F(4)$ (where F is the cdf)

$= 1 - 0.173$ (from the table in the text)

$= 0.827$

The Exponential Distribution

A random variable is said to be exponentially distributed with $\lambda > 0$ if its pdf is given by

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & otherwise \end{cases}$$

Since the exponent is negative, the pdf will decrease as x increases

λ is the arrival rate: number of occurrences per time unit

Ex: arrivals per hour; failures per day

Note that $1/\lambda$ can thus be considered to be the time between events or the duration of events

Ex: 10 arrivals per hour → 1/10 hr (6min) average between arrivals

Ex: 20 customers served per hour → 1/20 hr (3min) average service time

The Exponential Distribution

The CDF of exponential distribution is given by;

$y = 1 - e^{-\lambda t}$ after solving; $\lambda t = -\ln(1 - y)$

Since y is representing cumulative distribution, the term (1-y) gives values between 0 and 1

Within the range of 0-1, the natural logarithm gives –ve value

If numbers y are uniformly distributed, the numbers 1-y are also uniformly distributed

$$t = \frac{-\ln(y)}{\lambda} = -T_a \ln(y)$$

It shows that exponential distribution is dependent on mean value

The Exponential Distribution

The significance about T_a is that it is in multiplicative form. Numbers with any mean can be derived from a generator of mean value 1 by multiplying its output by the required mean

$$E(X) = \frac{1}{\lambda} \qquad V(X) = \frac{1}{\lambda^2}$$

$$F(x) = \begin{cases} 0, & x < 0 \\ \int_0^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x}, & x \geq 0 \end{cases}$$

The Exponential Distribution

Example – Generation;

Let's consider we have 5 random numbers 'y' in the range (0,1). The numbers are 0.135, 0.639, 0.424, 0.010, 0.843

Given $\lambda = 3.8$

$1/\lambda = 1/3.8$

The numbers we get are;

0.526, 0.118, 0.226, 1.213, 0.045

Coefficient of Variation

Applies to arrival times

Ratio of standard deviation of the inter-arrival time to the mean arrival time

Zero value means no variation. If the coefficient increases the data become more dispersed

If exponential distribution of mean value is T and standard deviation is also T, then variance is 1 i.e. if the variance is close to 1, then the exponential distribution may fit the data. In such case, chi-square test should be carried out to test the level of significance

Erlang Distribution

- Originally derived to analyze telephone traffic
- PDF of such distribution is given by;

$$f(t) = (k\lambda)^k \left[\frac{e^{-k\lambda t}}{(k-1)!} \right] t^{k-1}$$

- And arrival distribution will be;

$$A_0(t) = e^{-k\lambda t} . \sum_{n=0}^{k-1} \frac{(k\lambda t)^n}{n!}$$

$$F(x) = \begin{cases} 1 - \sum_{i=0}^{k-1} \frac{e^{-k\theta x} (k\theta x)^i}{i!}, & x > 0 \\ 0, & x \le 0 \end{cases}$$

Where k is a positive integer

Erlang Distribution

Setting k=1, gives exponential distribution.

Standard deviation of Erlang Distribution = $T/k^{1/2}$

Therefore coefficient of variation is = $1/k^{1/2}$

If k > 1, the data cluster around the mean value

If k = ∞, the distribution will be step function

If variation coefficient of variation is significantly less than 1, it is desirable to represent the data using Erlang distribution for which 'k' is closest to the value $(T/s)^2$

In general terms, there are 'k' stages that an entity have to pass and total time required is 'T' then,

$$E(x) = \underbrace{\frac{1}{\lambda} + \frac{1}{\lambda} + \frac{1}{\lambda} + \frac{1}{\lambda} ..... + \frac{1}{\lambda}}_{\text{k-terms}}$$

Erlang Distribution

If β is average time delay of each 'k' constituent exponential distributions, then kβ is the average value of the Erlang distribution.

In this case;

$$f(t) = \frac{1}{(k-1)!}.\left(\frac{1}{\beta}\right)^{k}.t^{k-1}.e^{-t/\beta}$$

If k = 1, then

$$f(t) = \frac{1}{\beta}.e^{-t/\beta}$$

i.e. exponential distribution

Erlang Distribution

Example - Generation;

- Generate 'k' random samples from an exponential distribution with mean time of β
- Add all of the terms

```
float prod = 1.0;
for(counter = 0; counter < k; counter++){
   rn = generateRandom(seed);
   prod *= rn;
}
return –beta * ln(prod);
```

Erlang Distribution

Exercise 5.21

Time to serve a customer at a bank is exponentially distributed with mean 50 sec

1. Probability that two customers in a row will each require less than 1 minute for their transaction
2. Probability that the two customers together will require less than 2 minutes for their transactions

For 1) we are looking at the probability that each of two independent events will be < 1 minute

In this case, the probability overall is the product of the two probabilities, each of which is exponential

λ = rate = 1/mean = 1/50

We want $P(X < 60) = F(60) = 1 - e^{-(1/50)(60)} = 0.6988$

Thus the total probability is $(0.6988)^2 = 0.4883$

Erlang Distribution

For 2) we are looking at the probability that two events together will be < 2 minutes

In this case the probability overall is an Erlang distribution with k = 2 and kθ = 1/50, and we want to determine P(X < 120) = F(120)

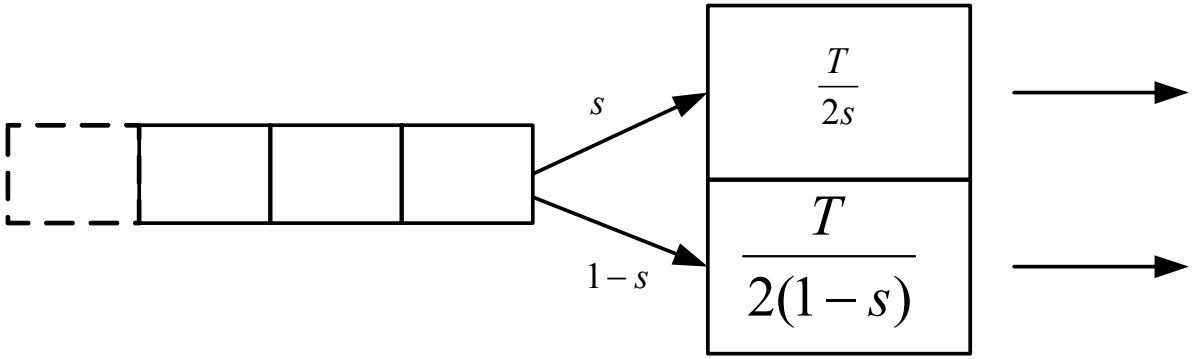
Substituting into our equation for F, we get

$$F(120) = 1 - \sum_{i=0}^{2-1} \frac{e^{-(120/50)}(120/50)^i}{i!}$$

$$= 1 - (0.0907) - (0.2177) = 0.6916$$

Hyper Exponential Distribution

If a set of data have two different exponential patterns, then it known as hyper-exponential distribution



Here, mean service time for first server = T/2s

mean service time for second server = T/2(1-s)

0 < s <= ½

And therefore, the arrival pattern

$$A_0(t) = se^{-2s\lambda t} + (1-s)e^{-2(1-s)\lambda t}$$

Hyper Exponential Distribution

The variance

$$k = \frac{(1 - 2s + 2s^2)}{2s(1-s)}$$

when s = ½, k becomes 1 and hence gives exponential distribution

Steps;

- Generate exponentially distributed rn from uniformly distributed random number with mean value of 1
- Compare the second uniformly distributed random number to 's'
- If less than s, multiply by T/2s
- Else multiply by T/2(1-s)

Service Times

Normally, service time of a process is constant

But if it varies stochastically, it is described by a probability function

Ts = Mean service time

μ = mean service rate

$S_0(t)$ = Probability that service time > t

Normally, exponential distribution is used to represent the service time but if the service time is fluctuating by large variance, it is represented by normal distribution

Service Times

Articulated by specifying the number of servers (denoted by s) whether each sever has its own queue or there is one queue feeding all servers and probability distribution of customers' service times

If S_i is the service time of the i^{th} arriving customer, E(S) is the mean service time of the customer then service rate of the server is

$$= 1/E(S)$$

Example : Class demonstration (from page 210 of Banks, Carson, Nelson, Nicol)

Normal Distribution

A random variable X with mean μ ($-\infty < \mu < \infty$) and variance $\sigma^2 > 0$ has a normal distribution if it has the pdf

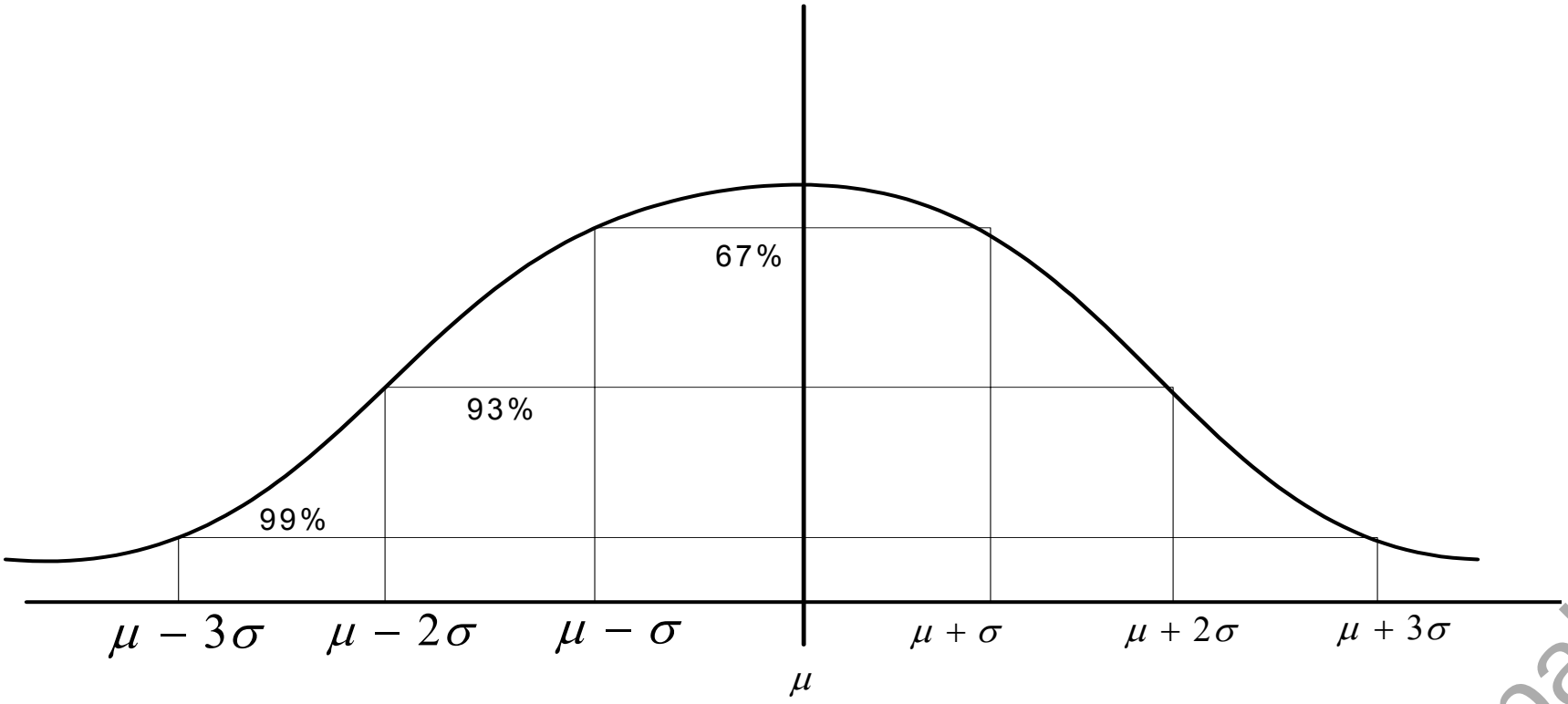
$$f(x) = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, -\infty < x < \infty$$

In normal form

$$f(z) = \frac{1}{\sqrt{2\pi}} . e^{-z^2/2}$$

Normal Distribution

-Property;



Normal Distribution

To generate numbers distributed according to f(z), we use

$x = z \,.\, \sigma + \mu$, z = standard normal variate

If r1 and r2 are two uniform random numbers in the range (0, 1), then to generate random samples from a standardized normal distribution is to use 'Box Mueller Transformation';

$$z = (-2.\ln r_1)^{1/2}.\cos(2\pi r_2)$$

For a set of several uniformly distributed random numbers,

standard deviation of this equation reaches normal distribution as k approaches infinity

x_i = uniformly distributed random numbers

k = small values of k give good accuracy (normally 12)

$$y = \frac{\sum_{i=1}^{k} x_i - \frac{k}{2}}{\left(\frac{k}{12}\right)^{1/2}}$$

$$y = \sum_{i=1}^{12} x_i - 6.0$$

Normal Distribution

The random numbers x_i are generated using inverse transformation method

A more accurate method can be;

Given two independent uniform variates, the method generates two outputs which are independent and is from a normal distribution with a mean of 0 and deviation of 1. It provides more accuracy than the previous one

$$X_1 = (-2\ln U_1)^{1/2}.\cos 2\pi U_2$$

$$X_2 = (-2\ln U_1)^{1/2}.\sin 2\pi U_2$$

In this case, the numbers X1 and X2 will be IID (independent and identically distributed)

Queueing Notation

Standard notation

A / B / c / N / K

A – preresents interarrival-time distribution

B – represents the service-time distribution

A and B can follow any distribution

D : Constant or deterministic

M : Exponential or Markov

Ek : Erlang of order k

G : Arbitrary or General

H : Hyper exponential

GI : General independent

Queueing Notation

c – represents the number of parallel servers

N – represents the system capacity

Could be "infinite" if the queue can grow arbitrarily large (or at least larger than is ever necessary)

Ex: a queue to go up the Eiffel Tower

Space could be limited in the system

Ex: a bank or any building

This can affect the effective arrival rate, since some arrivals may have to be discarded

K – represents the size of the calling population

- How large is the pool of customers for the system?
- It could be some relatively small, fixed size

Ex: The computers in a lab that may require service

Queueing Notation

It could be very large (effectively infinite)

Ex: The cars coming upon a toll booth

The size of the calling population has an important effect on the arrival rate

If the calling population is infinite, customers that are removed from the population and enter the queueing system do not affect the arrival rate of future customers (∞-1 = ∞)

If the calling population is finite, removal of customers from the population (and putting them into and later out of the system) must affect future arrival rates

Ex: In a computer lab with 10 computers, each has a 10% chance of going down in a given day. If a computer goes down, the repair takes a mean of 2 days

Queueing Notation

In the first day, the expected number of failures is 10*0.1=1

However, once a failure occurs, the faulty computer is out of the calling population, so the expected number of failures in the next day is 9 * 0.1 = 0.9

Clearly this changes again if another computer fails

Queueing Notation

Example;

1. there are 's' servers in parallel and one FIFO queue feeding all servers
2. A1, A2, . . . are IID random variables
3. S1, S2, . . . are IID random variables
4. A_i's and S_i's are independent

Such system is represented by GI/G/s queue

GI – distribution of Ai

G – distribution of Si

M/M/1/∞/∞

Here, the interarrival time and service time are exponentially distributed, the server is of single-server, the system have infinite capacity and infinite population of potential arrivals

This kind of server often represented by M/M/1

Queueing Disciplines

The queueing discipline describes in which order next entity is selected from the queue for service

- FIFO – first in first out
- LIFO – Last in first out
- SIRO – service in random order
- SPT – shortest processing time first
- PR – service according to priority

The order of picking entities from the queue does not mean that the entities will leave the system in the same order. This is all because of random service time for each entity

Queueing Disciplines

Sometimes, the entity may leave the queue even before being served. Such process is known as '***reneging***'. The rules for reneging must be defined clearly

If an entity refuses to join the queue (because the queue is relatively long), the process is known as '***balking***'

If there are multiple lines i.e. queues but a single server, the entities are selected by the process of polling

In case of priority queue, if the recently arrived entity have the greatest priority, then the new arrival will interrupt or preempt the service

Measure of Performance for Queues

Let T_a be mean arrival time, T_s be mean service time, λ be arrival rate, and μ be service rate then;

The ratio of mean service time to the mean inter-arrival time is called the ***traffic intensity (u)***

$$u = T_s / T_a$$

Let's denote rate of all of the arrivals (including loses) by λ' and λ to denote arrival rate of entities that are served. In this case;

$$u = \lambda'.T_s$$

And ***server utilization (ρ)*** is defined as;

$$\rho = \lambda'.T_s = \lambda / \mu$$

If 'n' number of servers;

$$\rho = \lambda / (n.\mu)$$

Measure of Performance for Queue

Notations for parallel server systems

P_n	steady state probability of having 'n' customers in system
$P_n(t)$	Probability of 'n' customers in system at time 't'
λ	Arrival rate
λ_e	Effective arrival rate
μ	Service rate of one server
ρ	Server utilization
A_n	Inter-arrival time between customers 'n-1' and 'n'
S_n	Service time of the nth arriving customer

Measure of Performance for Queue

Notations for parallel server systems

W_n	Total time spent in system by the nth arriving customer
or W	Mean time to complete service (including the wait for service)
W_n^Q	Total time spent in the waiting line by customer 'n'
L(t)	The no. of customers in system at time 't'
or W_w	Mean time spent waiting for service to begin
$L_Q(t)$	The no. of customers in queue at time 't'
L	Long-run time-average no. of customers in system (Mean no. of entities in the system)

Measure of Performance for Queue

Notations for parallel server systems

- L_Q Long-run time-average no. of customers in queue
  or L_w Mean number of entities in the waiting line
- W Long-run average time spent in system per customer
- w_Q Long-run average time spent in queue per customer

The probability that an entity will have to wait more than a given time is known as ***delay distribution***

P(t)- Probability that the time to complete the service is greater than 't'

P_w(t)-Probability that the time spent waiting for service to begin is greater than 't'

Mathematical Solutions of Queuing Problems

Some times, the queuing problems can be formulated by some mathematical solutions

Single server, poisson arrival

The mean number of entities in the system when ρ < 1 are;

$$L = \frac{\rho(2-\rho)}{2(1-\rho)}$$ Constant

$$L = \frac{\rho}{1-\rho}$$ Exponential

$$L = \frac{2k\rho - \rho^2(k-1)}{2k(1-\rho)}$$ Erlang

$$L = \frac{\rho 2 + \rho(1-\rho)4s(1-s)}{4s(1-s)(1-\rho)}$$ Hyper-Exponential

Mathematical Solutions of Queuing Problems

In above cases, the total number of entities waiting for service (L_w), mean times spent waiting for service to begin (W), and mean times spent waiting for service to be completed (W_w) can be derived as;

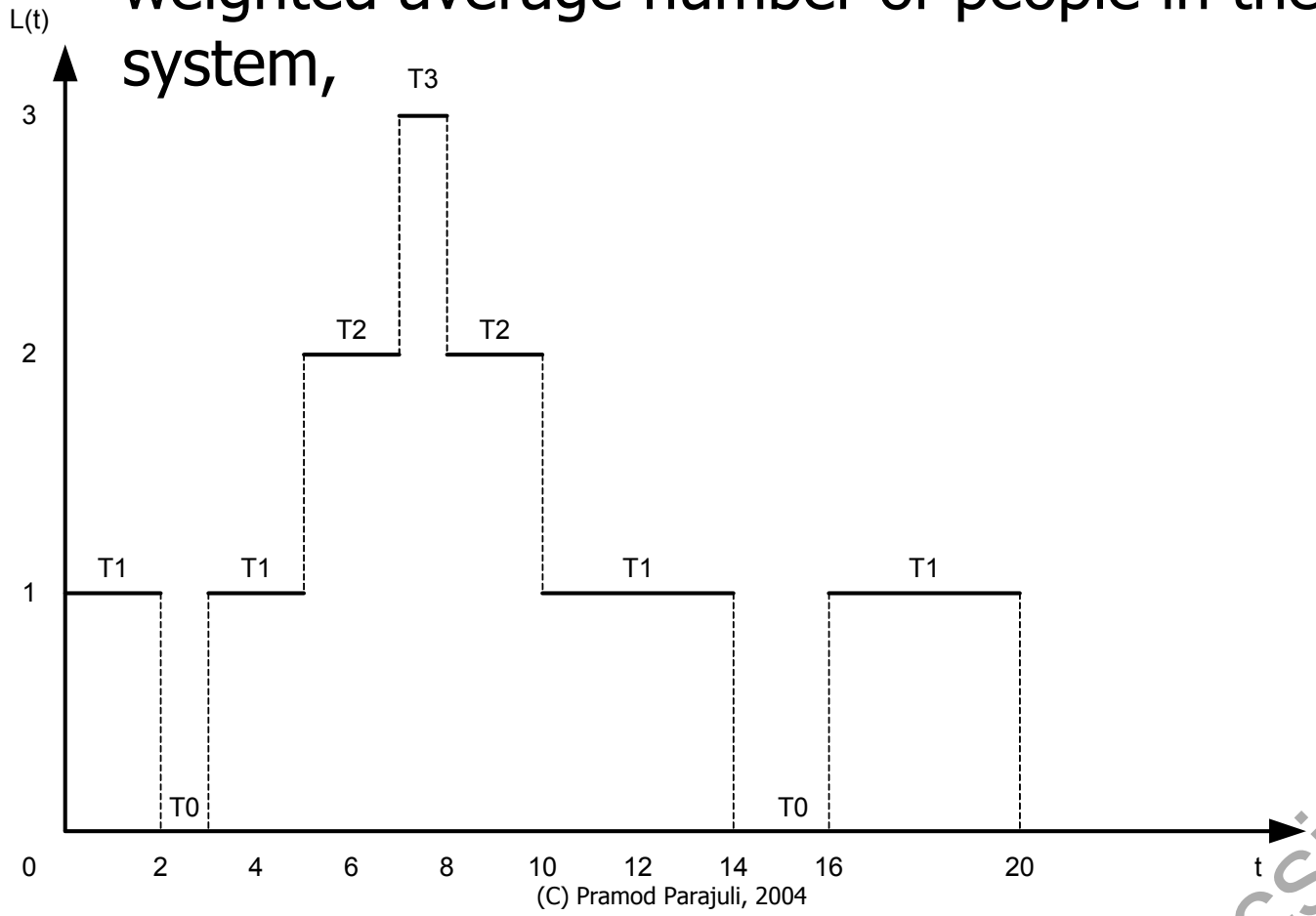
$$L_w = L - \rho$$

$$W = \frac{L}{\lambda} = L.T_a$$

$$W_w = \frac{L_w}{\lambda} = L_w T_a$$

Time-Average Number in the System, L

Given a queueing system operating for some period of time, T, we'd like to determine the time-weighted average number of people in the system,



Time-Average Number in the System, L

We'd also like to know the time-weighted average number of people in the queue,

Note that for a single queue with a single server, if the system is always busy,

However, it is not the case when the server is idle part of the time

The "hats" indicate that the values are "estimators" rather than analytically derived long-term values

Time-Average Number in the System, L

We can calculate $\hat{L}$ for an interval [0,T] in a fairly straightforward manner using a sum

$$\hat{L} = \frac{1}{T} \sum_{i=0}^{\infty} i T_i$$

Note that each T_i here represents the total time that the system contained exactly i customers

These may not be contiguous

'i' is shown going to infinity, but in reality most queuing systems (especially stable queuing systems) will have all $T_i = 0$ for i > some value

In other words, there is some maximum number in the system that is never exceeded

Time-Average Number in the System, L

Let's think of this value in another way:

- Consider the number of customers in the system at any time, t
  - L(t) = number of customers in system at time t
- This value changes as customers enter and leave the system
- We can graph this with t as the x-axis and L(t) as the y-axis
- Consider now the area under this plot from [0, T]
  - It represents the sum of all of the customers in the system over all times from [0, T], which can be determined with an integral

$$Area = \int_0^T L(t)\,dt$$

Time-Average Number in the System, L

Now to get the time-average we just divide by T, or

$$\hat{L} = \frac{1}{T}\sum_{i=0}^{\infty} iT_i = \frac{1}{T}\int_0^T L(t)\,dt$$

For many stable systems, as T →∞ (or, practically speaking, as T gets very large) $\hat{L}$ approaches ***L***, the long-run time-average number of customers in the system

However, initial conditions may determine how large T must be before the long-run average is reached

The same principles can be applied to $\hat{L}_Q$, the time-average number in the queue, and $\boldsymbol{L_Q}$, the long-run time average number in the queue

Time-Average Number in the System, L

From fig. in slide no. 51,

$$\hat{L} = [0(3) + 1(12) + 2(4) + 3(1)] / 20$$

$$= 23/20$$

$$= 1.15 \text{ customers}$$

Average Time in System Per Customer, w

This is also a straightforward calculation during our simulations

$$\hat{w} = \frac{1}{N}\sum_{i=1}^{N} W_i$$

where N is the number of arrivals in the period [0,T]

where each W_i is the time customer i spends in the system during the period [0,T]

If the system is stable, as N → ∞ , $\hat{w} \rightarrow w$

w is the long-run average system time

We can do similar calculations for the queue alone to get the values $\hat{w}_Q$ and w_Q

We can think of these values as the observed delay and the long-run average delay per customer

Example: from 216 - Nicol

Average Time in System Per Customer, w

For simple queueing systems such as those we have been examining, stability can be determined in a fairly easy way

The arrival rate, λ, must be less than the service rate i.e. customers must arrive with less frequency than they can be served

Consider a simple single queue system with a single server (G/G/1/∞/∞)

- Define the service rate to be μ
- This system is stable if $\lambda < \mu$

If $\lambda > \mu$, then, over a period of time there is a net rate of increase in the system of $\lambda - \mu$

This will lead to increase in the number in the system (L(t)) without bound as t increases

Average Time in System Per Customer, w

Note if $\lambda == \mu$ some systems (ex: deterministic) may be stable while others may not be

If we have a system with multiple servers (ex: G/G/c/∞/∞) then it will be stable if the net service rate of all servers together is greater than the arrival rate

If all servers have the same rate μ, then the system is stable if $\lambda < c\mu$

Average Time in System Per Customer, w

Note that if our system capacity or calling population (or both) are fixed, our system can be stable even if the arrival rate exceeds the service rate

Ex: G/G/c/k/∞

With a fixed system capacity, in a sense the system is unstable until it "fills" up to k. At this point, excess arrivals are not allowed into the system, so it is stable from that point on

The idea here is that the net arrival rate decreases once the system has filled

Ex: G/G/c/∞/k

With a fixed calling population, we are in effect restricting the arrival rate

As arrivals occur, the arrival rate decreases and the system again becomes stable

Conservation Law

An important law in queueing theory states

$$L = \lambda w$$

where L is the long-run number in the system, λ is the arrival rate and w is the long-run time in the system

Often called "Little's Equation"

This holds for most queueing systems

Server Utilization

What fraction of the time is the server busy?

Clearly it is related to the other measures we have discussed (we will see the relationship shortly)

As with our other measures we can calculate this for a given system (G/G/1/∞/∞)

$$\hat{\rho} = \frac{1}{T}\sum_{i=1}^{\infty} T_i$$

We assume that if at least 1 customer is in the system, the server will be busy (which is why we start at T_1 rather than T_0)

However, we can also calculate the server utilization based on the arrival and service rates

Server Utilization

G/G/1/∞/∞ systems

Consider again the arrival rate λ and the service rate μ

Consider only the server (without the queue)

With a single server, it can be either busy or idle

If it is busy, there is 1 customer in the "server system", otherwise there are 0 customers in the "server system" (excluding the queue)

Thus we can define $L_s = \rho$ = average number of customers in the "server system"

Using the conservation equation this gives us

$L_s = \lambda_s w_s$

where λ_s is the rate of customers coming into the server and w_s is the average time spent in the server

Server Utilization

For the system to be stable, $\lambda_s = \lambda$, since we cannot serve faster than customers arrive and if we serve more slowly the line will grow indefinitely

The average time spent in the server is simply $1/\mu$ (i.e. 1/(rate of the server))

Putting these together gives us

$$\rho = L_s = \lambda_s w_s = \lambda\ (1/\mu) = \lambda/\mu$$

Note that this indicates that a stable queueing system must have a server utilization of less than 1

- The closer we get to one, the less idle time for the server, but the longer the lines will be (probably)
- Actual line length depends a lot not just on the rates, but also on the variance

Server Utilization

G/G/c/∞/∞ systems

Applying the same techniques we did for the single server, recalling that for a stable system with c servers:

$$\lambda < c\mu$$

we end up with the the final result

$$\rho = \lambda/c\mu$$