

# Probabilistic Methods in Computer System Modeling

Advanced OS course

Spring 2005

# A Model

is a representation of an actual system that allows to manipulate and observe the (expected) behavior of the system without actually modifying the system.

This gives an idea of how the real system would behave, and allows a relatively easy investigation of different alternatives, and review of different “what if” scenarios.

If the system is simple enough (not too many attributes) then an *analytical probabilistic model* could be employed.

If the system is complicated, then *Computer Simulation* is needed.

We will deal with both techniques.

For analytic modeling, we will focus on *Queueing Systems theory*.

# OS is a network of queues

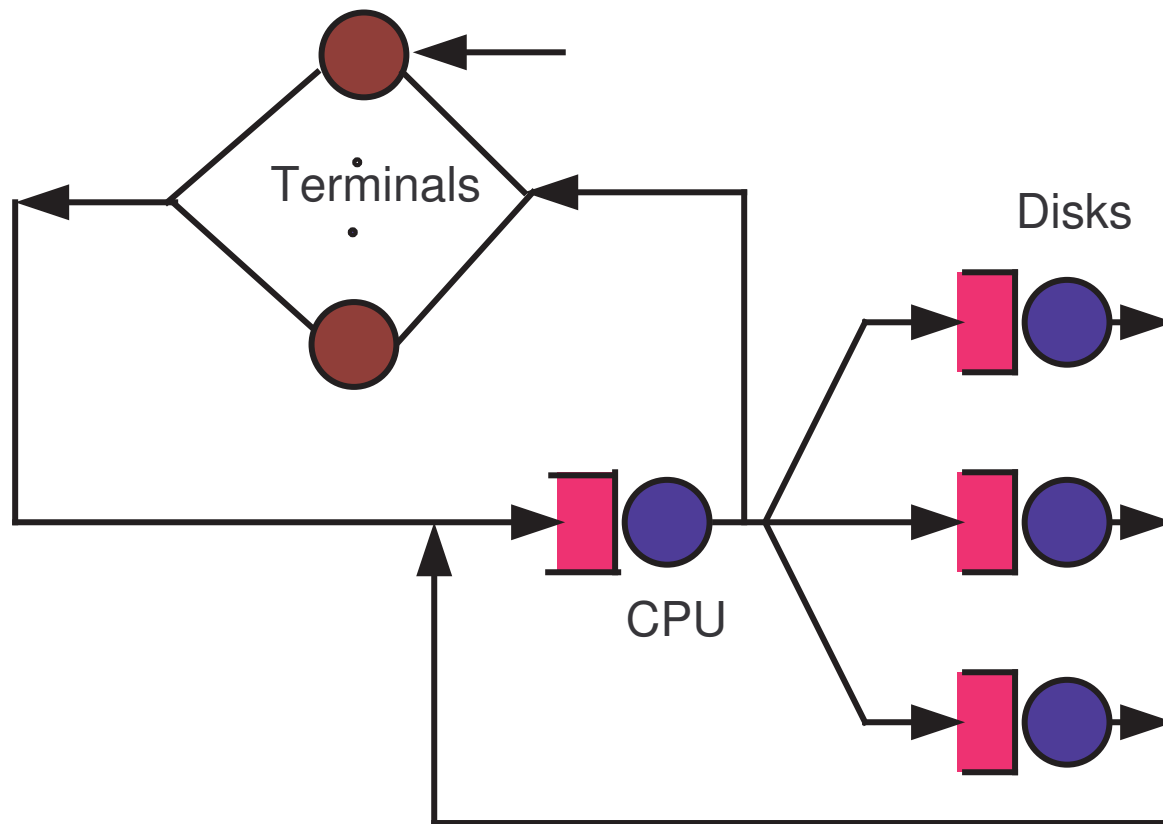
A **network of queues** represent an example of a **System of Flow**: some **commodity flows**, moves, or is transferred through one or more finite capacity **channels** in order to go from one point to another.

OS is a collection of **service centers**: the **system resources** which are the channels, and **customers**: the **users, messages, jobs, or transactions**, which are the commodity that flows in the channels.

E.g., In a packet-switching network, packets arrive asynchronously at nodes, and are processed and released. Typically, a node can not handle all the traffic entering it simultaneously, and packets arriving are *buffered* to await their turn for transmission, according to some service protocol, like first in first out (FIFO), shortest packet first, highest priority first, random, pre-emptive or not, etc..

E.g., If a number of documents needs to be printed, the OS (or a special print spooler) queues the documents by placing them in a special area called a *print buffer* or *print queue*. The printer then pulls the documents off the queue one at a time.

# A Model with a Terminal Driven Workload



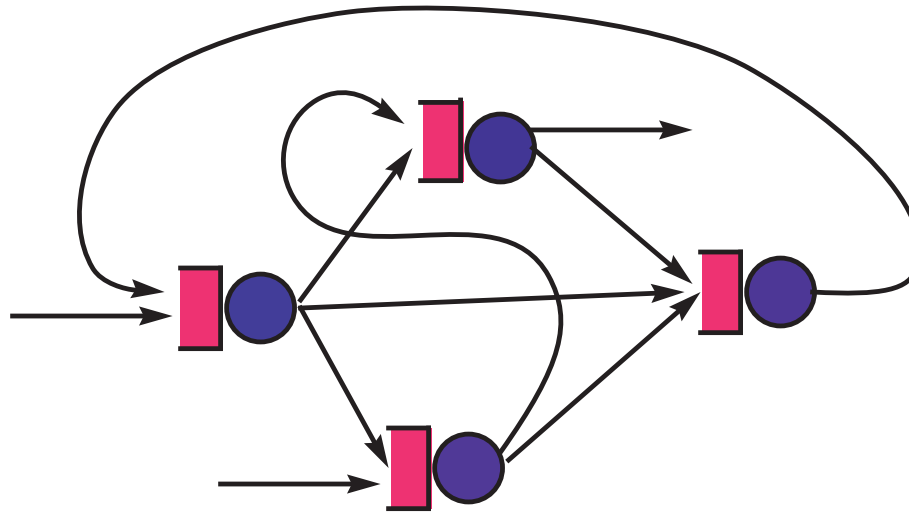
A running program lays demands that go through service by the CPU, memory, screen, storage, communication port, etc.

**How** does response time depend on number of Interactive users?

**What** if the CPU upgrades to 30% faster?

**Will** splitting to more disks do any good?

## A Simple Communication Network Model



The queue for a node service could be split according to different layers of the communication protocol. The servers of these queues could be time slices of same CPU, or specially dedicated communication controllers.

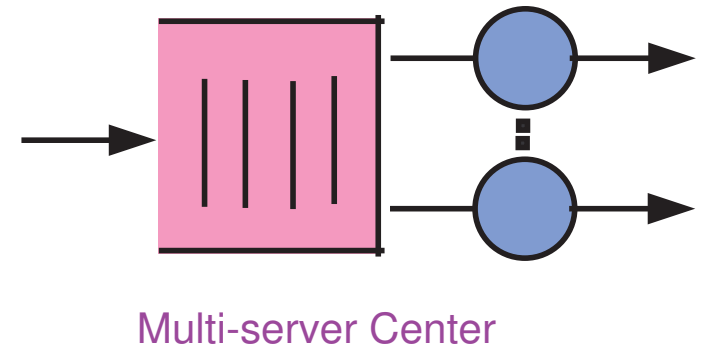
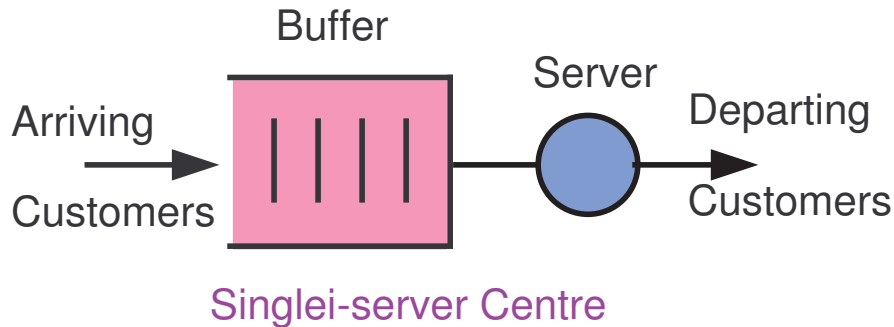
There could be different queues to different priorities of messages.

**Are** messages lost because queues are too short?

**What** is the utilization of the different CPUs?

**Where** are the bottlenecks?

# Queues and Service Centers



A queue models any service center with: One or more servers, and a waiting area (or buffer).

- Customers arrive to receive service. Each customer has its own demand for service time. A Customer that upon arrival does not find a free server, waits in the waiting area.
- All customers can go into the same queue and wait together for the next available server, or split the queue by the servers, or by customer priority, or by any criterion.
- Upon becoming available, the server immediately selects one of the customers waiting in the queue, and starts to serve it. The selection is done in accordance with the queue scheduling discipline (FIFO, LIFO, shortest job first, random, pre-emptive or not, etc.)

# Specification of Queueing Systems

**Average Rate of Customer Arrivals,  $\lambda$ :** the average number of customer arrivals per one unit of time. (Later, we will also specify whole distribution, not only its average.)

**E.g.**, average number of transactions arriving at a data base in one second.

The **average customer interarrival time** is thus  $1/\lambda$ , the average number of time units between two successive arrivals.

**Average service time  $1/\mu$ ,** which is the average time, measured in time units, of service demanded by the customers. (Later – whole distribution.)

**E.g.**, average number of seconds it takes to process a data base transaction.

The **average service rate,  $\mu$ ,** is thus the average number of customers a server can service in one unit of time, *if working non-stop*.

We assume that the server's yield is constant, and that the duration of time it serves a customer depends only on the amount of service demanded by the customer.

## Specification of Queueing Systems (cont.)

We assume **independent, identically distributed** (along the line of arriving customers) interarrival and service time.

**Scheduling Discipline:** FIFO: First In First Out; Possibly several FIFOs with priorities. Can be LIFO, shortest job first, most profitable customer first, or anything else.

**Number of Servers** (service stations).

**Length of Buffer** to accommodate waiting and being serviced customers. Possibly infinite.

**Size of Customer Population** which can be limited or not.

# Measurements of Performance

**Stability:** Does the system reach a steady state? Does the number of waiting customers always grow?

**Utilization:** The proportion of time the server is busy.

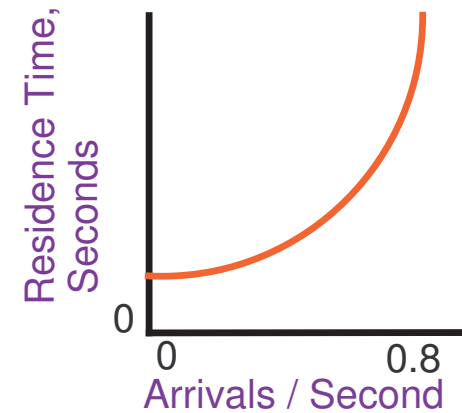
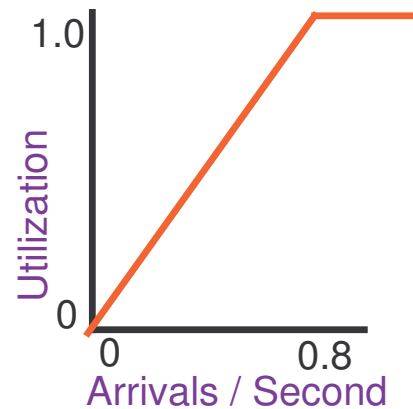
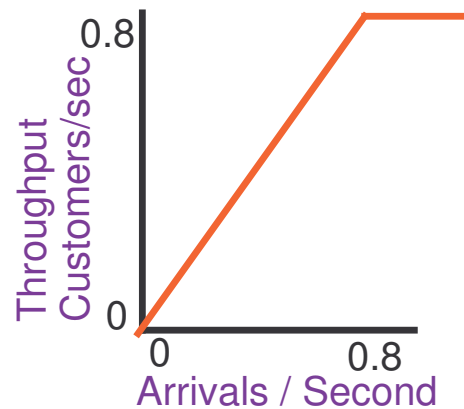
**Residence Time (Delay):** The total time spent by a customer on waiting and on receiving service

**Queue Length:** The number of customers at the service center both waiting and receiving service

**Throughput:** The rate at which customers pass through the service center: number of customers per unit of time.

We look for their average, and if possible: exact distribution.

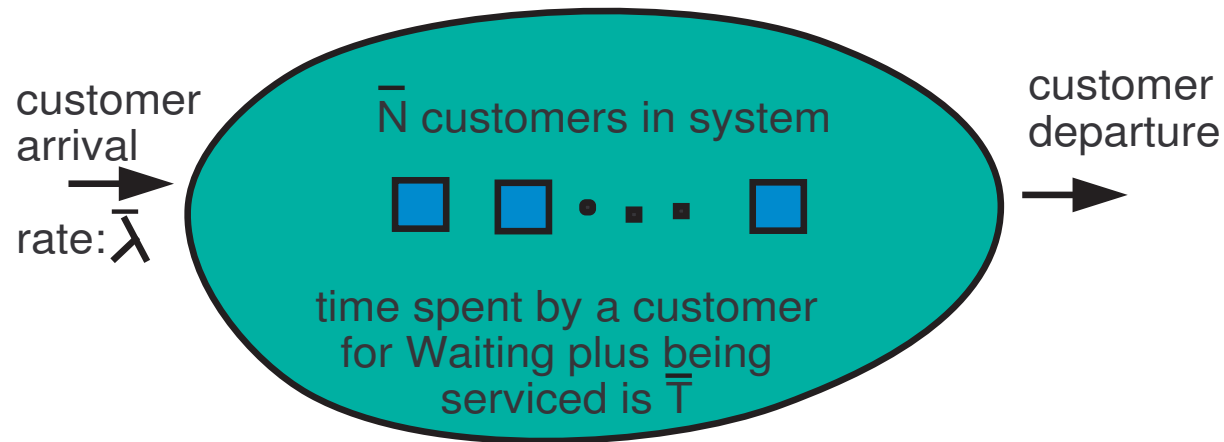
## E.g.: Performance Measures for a Single Server



The average service requirement of a customer is 1.25 seconds of server time. Hence, on the average, the server can not serve more than 0.8 customers in one second.

As load (i.e., rate of arrival) grows, residence time increases at a faster and faster rate.

# Little's Theorem - the "folk theorem"



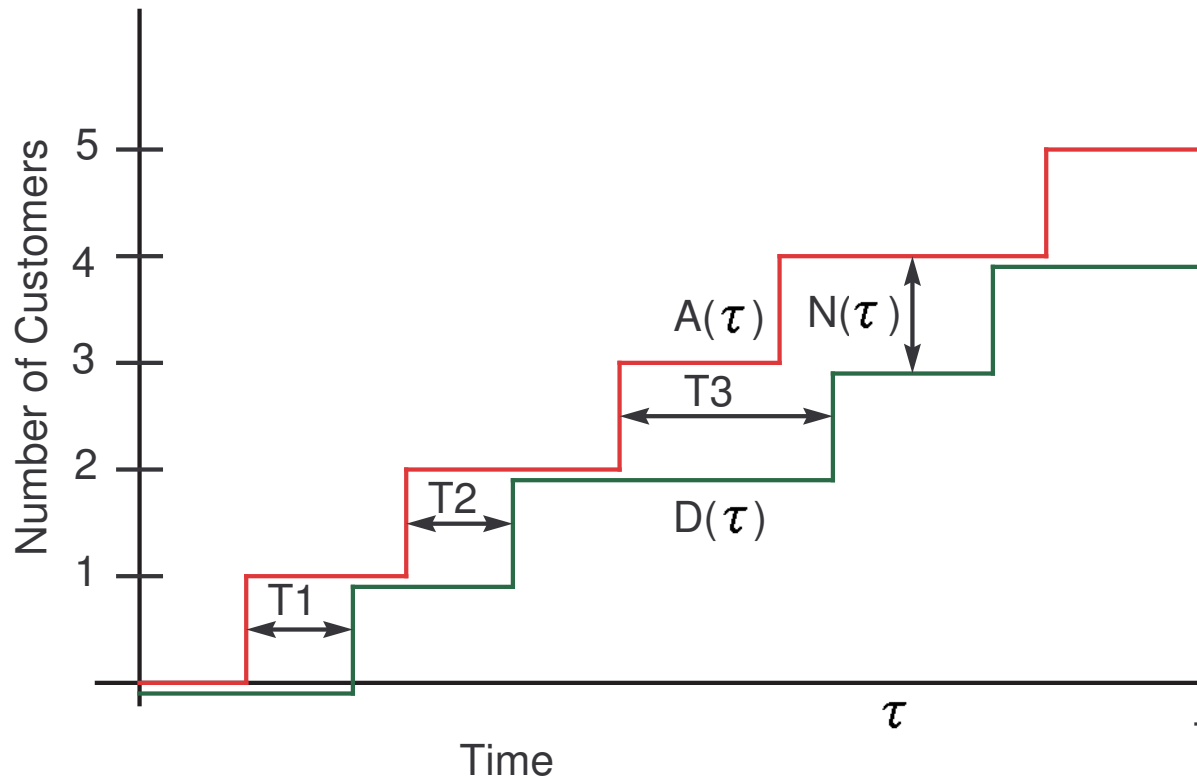
$\bar{\lambda}$ : the mean, steady state customer arrival rate (average number of customers arriving in one unit of time, say one second)

$\bar{N}$ : the average number of customers in the system (both in buffer and in service)

$\bar{T}$ : the mean time (in seconds) spent by each customer in the system (time in queue + time in service)

In a steady state,  $\bar{N} = \bar{\lambda}\bar{T}$

# Proof of $\bar{N} = \bar{\lambda}\bar{T}$



$A(\tau)$ : number of arrivals in the time interval  $[0, \tau]$ .

$D(\tau)$ : number of departures in that time interval.

$N(\tau) = A(\tau) - D(\tau)$ : number of customers present in the queueing environment (both in queue and in service) at time  $\tau$ .

$T_i$ : time spent in the system by the  $i$ -th customer.

## Proof of $\bar{N} = \bar{\lambda}\bar{T}$ (cont.)

The time average (up to  $t$ ) of the instantaneous (snapshot) number of customers is

$$\widehat{N}(t) = \frac{1}{t} \int_0^t N(\tau) d\tau = \frac{1}{t} \sum_{i=1}^{A(t)} T_i = \frac{A(t)}{t} \frac{1}{A(t)} \sum_{i=1}^{A(t)} T_i$$

The time average (up to  $t$ ) of customer arrival rate and the average time spent by the first  $A(t)$  customers in the system are:

$$\widehat{\lambda}(t) = \frac{A(t)}{t}, \quad \widehat{T}(t) = \frac{1}{A(t)} \sum_{i=1}^{A(t)} T_i$$

$$\text{Together: } \widehat{N}(t) = \widehat{\lambda}(t)\widehat{T}(t).$$

As  $t \rightarrow \infty$ , since the system reaches a steady state, the **averages tend to fixed steady values**:  $\widehat{N}(t) \rightarrow \bar{N}$ ,  $\widehat{\lambda}(t) \rightarrow \bar{\lambda}$ ,  $\widehat{T}(t) \rightarrow \bar{T}$ , and we have

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

Note that  $\lambda$ ,  $N$  and  $T$  **need not be constant**, nor does the server have to work all the time. Only **their averages need to converge**.

## Little's Theorem is very general

- Nothing is assumed about the system: the theorem applies to the service area only, or to the waiting area only, or to the whole system, or to any part of the system
- The scheduling discipline does not need to be FIFO
- No constrain on the number of servers
- Each of: the arrival process, the number of customers in the system, and the duration of time spent by the customers in the system, can be any process that reaches a steady average (stationary)

## An Application of $N = \lambda T$ (bank)

### Scenario:

- \* Customers wait in a **single** queue for service from  $n$  servers.
- \* From head of waiting queue customer proceeds to first available server (like in our banks).
- \* Limited number  $N$  of customers in system (queue + service). Queue is always full (many customers want in, and one gets in as soon as one other departs).
- \* Average service time for a customer is  $X$ .

**What** is the average time  $T$  a customer can expect to spend in the system once he enters it?

Let  $\lambda$  denote the steady state mean arrival rate of customers. By Little, for the whole system:  $N = \lambda T$ . By Little, for the **subsystem** consisting only of the  $n$  server counters,  $n = \lambda X$  (in a steady state, both systems have the same arrival rate). Hence:

$$T = N/\lambda = N/(n/X) = NX/n$$

(what we could have expected intuitively)

## 2nd Application of $N = \lambda T$ (Supermarket)

**Scenario:** Like before, but each server has his own queue, and a customer entering the system chooses a server uniformly at random, and awaits in his queue. No limits on the spaces of the individual queues, except that the total space of all queues together is  $N$ .

For  $i$ -th server-and-queue, by Little:  $T_i = N_i/\lambda_i$ .

Denote the average fraction of time that server  $i$  is busy by  $\rho_i$ .

Little for the  $i$ -th counter:  $\rho_i = \lambda_i X$ . Together,  $T_i = N_i X / \rho_i$ .

Now, because of the uniform choice of servers by the customers,

$N_i = N/n$ ,  $\rho_i = \rho$ , and all servers have the same average customer delay:  $T_i = T$ .

Finally, the probability  $1 - \rho_i$  that the  $i$ -th server is idle is  $(1 - 1/n)^N$ : the probability that each of the  $N$  customers in the room have selected a server other than  $i$ . Hence:

$$T = T_i = N_i X / \rho_i = \frac{N}{n} X / (1 - (1 - 1/n)^N)$$

Which is greater than before, because servers can be idle.

## In Our Supermarket, What If

The entering customer would chose the shortest queue and join there? Because we assumed that upon the departure of a customer, a new one enters the system, the servers would always be busy, and the average time from arrival to departure will remain the same as in a bank. (The deviation should increase.)

If we did not deal with a closed waiting hall which is always full, and simply assumed a steady system with converging average of number of customers in the system  $\widehat{N}(t)$ , then depending on the distribution of the arrival rate  $\lambda$ , and the distribution of their service demand  $X$ , it could still be that although when customers arrived they picked the shortest queue, this queue was full of customers with very heavy demands, and while they were still waiting, some other queue became empty, and its server – idle.

•• Allocating a resource in advance, you may lose.

(Why don't customers jump between queues?)

# A Brief Refresher to Theory of Probability

**Random Phenomenon:** A process producing potentially different results each time it is carried out.

**E.g.** Flip a fair coin 5 times in succession.

**Outcome:** A particular result of a random phenomenon.

**E.g.,** HHTTH.

**Sample Space:** The set,  $\mathcal{S}$ , of all possible outcomes.

**E.g.**  $\mathcal{S} = \{HHHHH, HHHHT, \dots, TTTTT\}$ .

**Event:** A collection of outcomes in a sample space.

**E.g.** Flipping exactly 3 heads:  $E = \{HHHTT, HHTHT, \dots, TTHHH\}$ .

**Occurrence of an event:** One of the members of the collection occurs when the random phenomenon is carried out.

**E.g.** A coin flipped 5 times, resulting in  $HTHTH$ , means  $E$  has occurred.

# Probability

is a function that assigns a value between 0 and 1 to an event.

The intention is to measure the event's likelihood of occurrence.

Two common definitions of the probability  $P(A)$  of event  $A$ :

- its long run frequency over repeated trials of the random phenomenon
- the degree of belief placed in the event occurring.

## Kolmogorov's Axioms of Probability

(1) For any  $A \subseteq \mathcal{S}$ ,  $P(A) \geq 0$

(2)  $P(\mathcal{S}) = 1$

(3) For any collection of mutually exclusive events  $A_1, A_2, \dots$ ,  
 $P(\cup A_i) = \sum P(A_i)$ .

## Probabilities of events when $\mathcal{S}$ is uncountable

- If event  $A$  consists of only one outcome, then  $P(A) = 0$ .
- If event  $A$  consists of finitely many outcomes, then  $P(A) = 0$ .
- If event  $A$  consists of countably infinite number of outcomes, then  $P(A) = 0$ .
- $P(A) > 0$  only if  $A$  is uncountable!

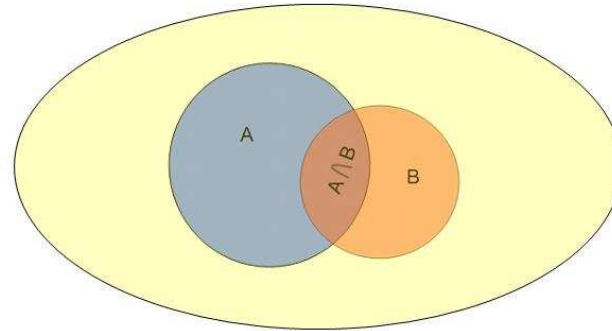
**E.g.:** The probability of randomly selecting a specific rational number from the interval  $[0, 1]$  is 0. However, assuming uniform probability of selection, the probability of selecting a number in the range  $[0.2, 0.2 + \delta]$  is  $\delta$ .

**A conclusion:**  $P(A) = 0$  does not imply  $A = \Phi$ .

# Conditional Probability and Independence

Let  $A, B \subseteq \mathcal{S}$  and  $P(B) > 0$ . The **conditional probability** of  $A$  given that  $B$  has occurred is

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$



In words: when conditioning on  $B$  having occurred, the sample space,  $\mathcal{S}$ , is reduced to  $B$ .

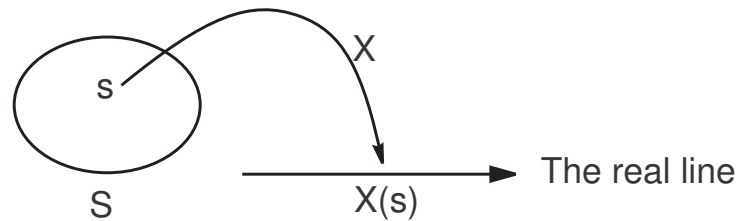
**Corollary:**  $P(A \cap B) = P(A|B) P(B)$

**Corollary:**  $P(B) = P(A) P(B|A) + P(A^c) P(B|A^c)$

Two events are **independent** if they are unrelated. Knowing whether one occurs does not affect the knowledge of whether the other occurs:  $P(A|B) = P(A)$ . Hence:  $P(A \cap B) = P(A) P(B)$ .

# Random Variables

A **Random Variable (RV)** is a function that maps outcomes of a sample space to real values.



**E.g.**, Roll 2 dice. The sample space  $\mathcal{S} = \{(1, 1), (1, 2), \dots, (6, 6)\}$ .

Let  $X$  be the sum of the faces on the two dice:

$X((1, 1)) = 2$ ,  $X((5, 3)) = 8$ , etc.

The random variable then has its own sample space, and we can talk about  $P(X = 4)$ , or  $P(3 \leq X < 6)$ , etc.

In the example,  $P(X = 4)$ , denoted  $P_X(4)$ ,  $= P\{s \in \mathcal{S} : X(s) = 4\} = P\{(1, 3), (2, 2), (3, 1)\} = 3/36 = 1/12$ .

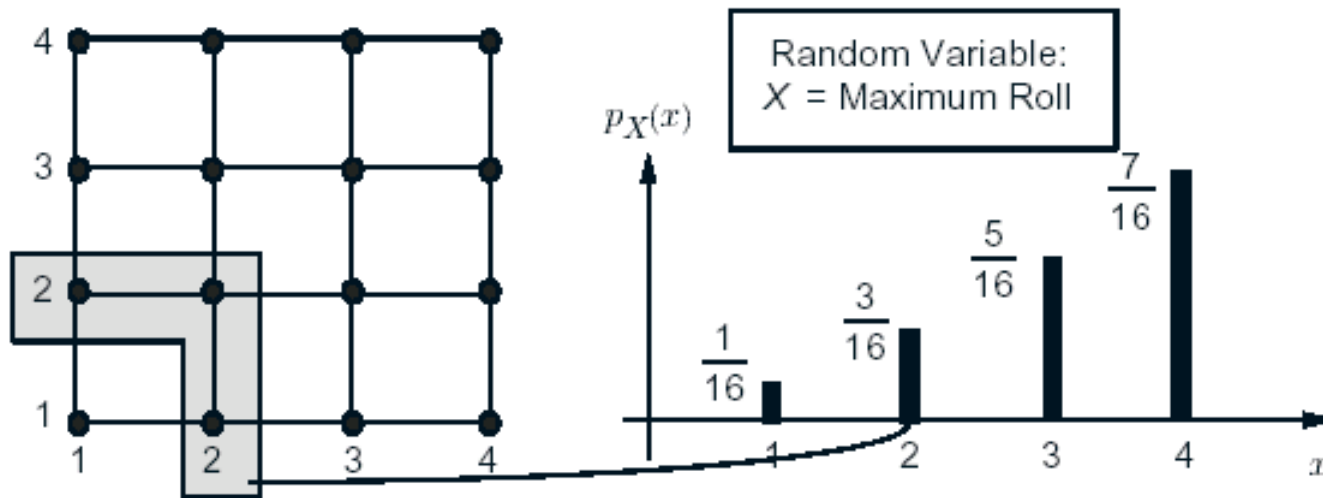
A function of a random variable defines another random variable.

# Discrete Random Variables

A RV is **discrete** if its range consists of finite or countably infinite values. **E.g.**, Number of heads appearing in 20 coin flips; Number of phone messages awaiting me.

The **Probability Mass Function (pmf)**  $p_X$  of a discrete random variable  $X$  is  $p_X(x) = P(X = x)$ .

Because  $P(\mathcal{S}) = 1$ , we have  $\sum_x p_X(x) = 1$ .



Sample Space:  
Pairs of Rolls of a fair 4-sided die

# Continuous Random Variables

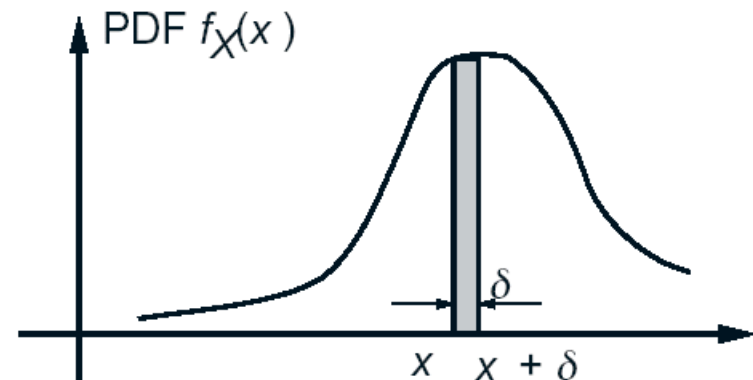
A RV is **continuous** if its range includes segments of the real line, and hence its sample space is uncountable.

**E.g.**, the temperature tomorrow morning (expressed with infinite precision). The **Probability Density Function (pdf)**  $f_X(x)$  of a continuous random variable  $X$  defines for any  $a \leq b$  the probability that the value of  $X$  falls within the interval  $(a, b)$ :

$$P(a \leq X \leq b) = \int_a^b f_X(x) dx \quad .$$

Because  $P(S) = 1$ , we have  $\int_{-\infty}^{\infty} f_X(x) dx = 1$ .

The probability that  $X$  takes value in the interval  $[x, x + \delta]$  is the shaded area. As  $\delta$  tends to 0, this approaches  $f_X(x)\delta$ .



# The Cumulative Distribution Function

The **cumulative distribution function (cdf)** of a **RV  $X$**  in a given real number  $x$  is

$$F_X(x) = P(X \leq x)$$

For discrete RV,  $F_X$  is a step function:

$$F_X(x) = \sum_{u \leq x} p_X(u).$$

For continuous RV,

$$F_X(x) = \int_{-\infty}^x f_X(u) du.$$

## Properties of a cdf:

- (1) for  $a < b$ ,  $F_X(a) \leq F_X(b)$  and  $P(a < X \leq b) = F_X(b) - F_X(a)$
- (2)  $\lim_{x \rightarrow \infty} F_X(x) = 1$ , and  $\lim_{x \rightarrow -\infty} F_X(x) = 0$
- (3)  $F_X$  is right-continuous, i.e.,  $F_X(x)$  is very close to  $F_X(x^+)$ , but not necessarily to  $F_X(x^-)$ .

## Expected Value (Mean) and Variance

The **Expected Value**  $E(X)$  of a random variable  $X$  is the long-run average value of  $X$  upon repeated applications of the random phenomenon.

The **Variance**  $Var(X)$  is the expected deviation from the mean value.

$$\text{Disc. : } E(X) = \sum_x x p_X(x), \quad \text{Cont. : } E(X) = \int_{-\infty}^{\infty} x f_X(x) dx$$

**Variance of Discrete RV  $X$ :**

$$Var(X) = E\left((X - E(X))^2\right) = \sum_x (x - E(X))^2 p_X(x)$$

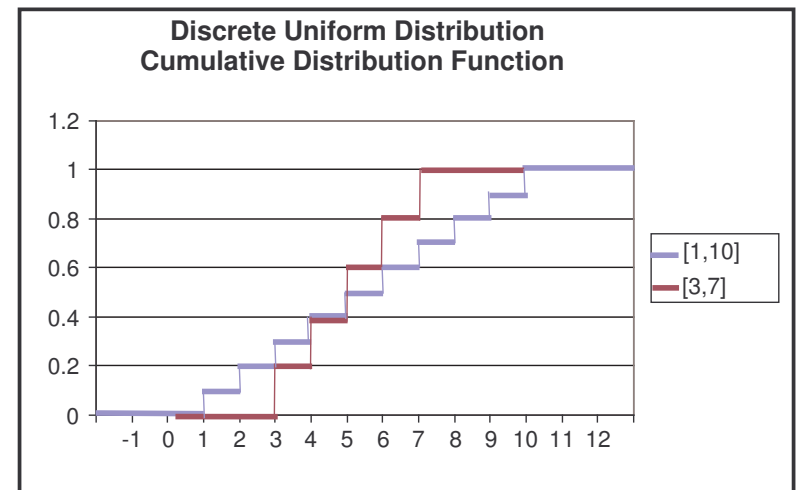
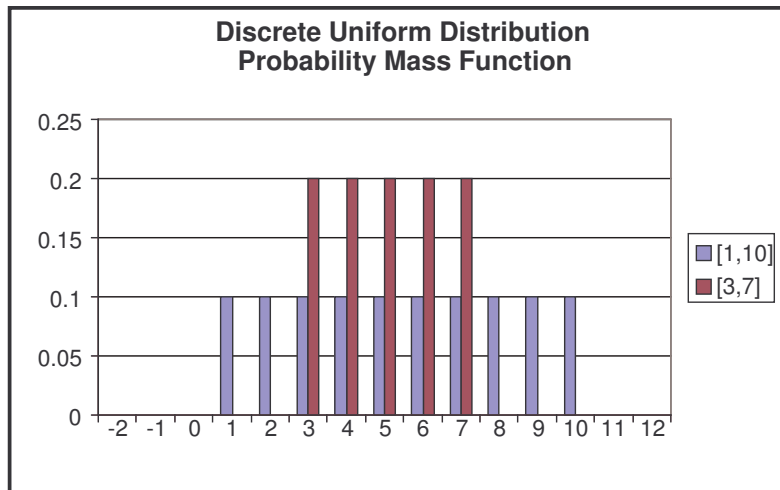
**Variance of Continuous RV  $X$ :**

$$Var(X) = E\left((X - E(X))^2\right) = \int_{-\infty}^{\infty} (x - E(X))^2 f_X(x) dx$$

**Proposition:**

$$Var(X) = E(X^2) - (E(X))^2$$

# Discrete Uniform Distribution over $[a, b]$

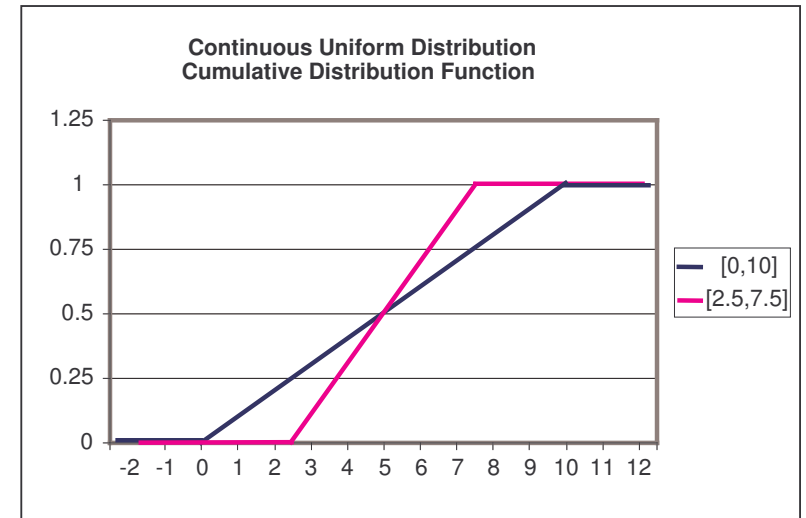
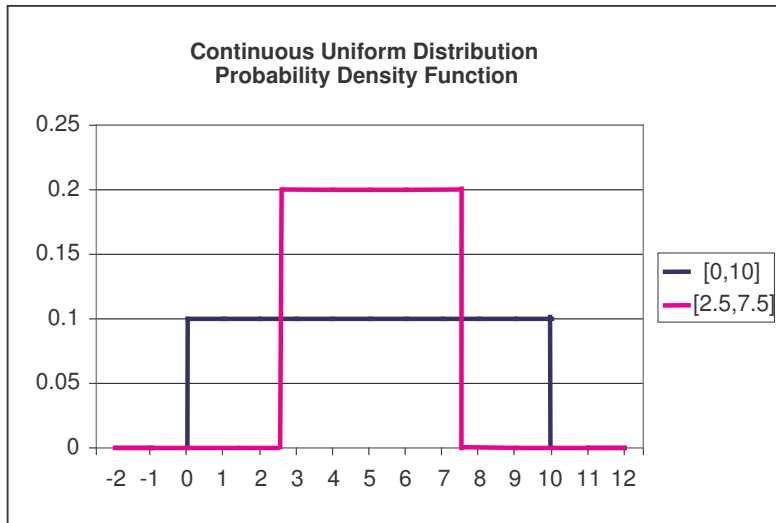


$$P(X = x) = \begin{cases} \frac{1}{b-a+1} & x = a, a+1, a+2, \dots, b \\ 0 & \text{otherwise} \end{cases}$$

$$F_X(x) = P(X \leq x) = \begin{cases} 0 & x < a \\ ([x] - a + 1) \times \frac{1}{b-a+1} & a \leq x < b \\ 1 & x \geq b \end{cases}$$

$$E(X) = \frac{a+b}{2}, \quad \text{Var}(X) = \frac{(b-a)(b-a+2)}{12}$$

# Continuous Uniform Distribution over $[a, b]$



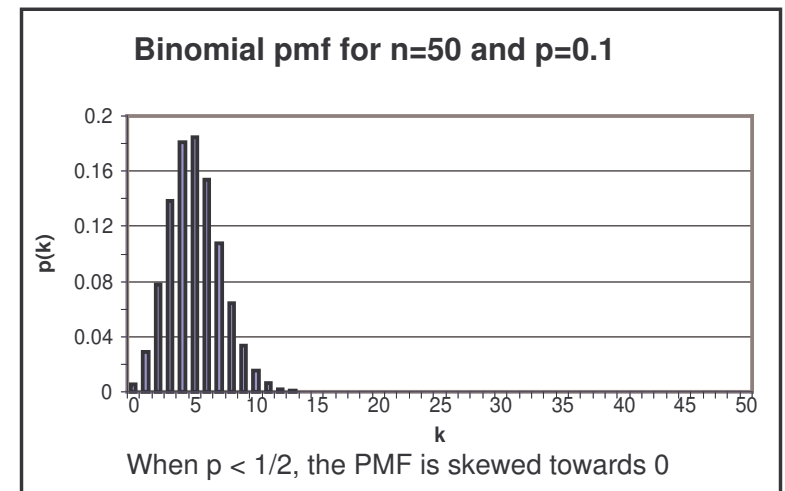
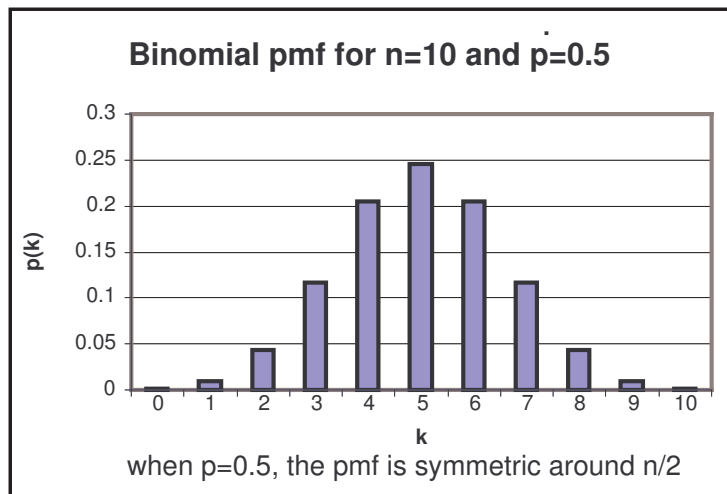
$$f_X(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{otherwise} \end{cases}$$

$$F_X(x) = P(X \leq x) = \begin{cases} 0 & x < a \\ (x - a)/(b - a) & a \leq x < b \\ 1 & x \geq b \end{cases}$$

$$E(X) = \frac{a + b}{2}, \quad \text{Var}(X) = \frac{(b - a)^2}{12}$$

## Binomial Distribution with parameters $n$ and $p$

A biased coin is tossed  $n$  times. At each toss, head shows with probability  $p$ , and tail with probability  $1 - p$ , independently of prior tosses. The number of heads in the  $n$ -toss sequence is distributed binomially with parameters  $n$  and  $p$ .



$$p_X(k) = P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}, \quad k = 0, 1, 2, \dots, n$$

$$E(X) = np, \quad \text{Var}(X) = np(1 - p)$$

## Modeling Customer Arrival Process

with average number of arrivals per second =  $\lambda$

To spread the arrivals evenly and independently, we take an interval that corresponds to one second, and divide it to  $n$  equal segments.

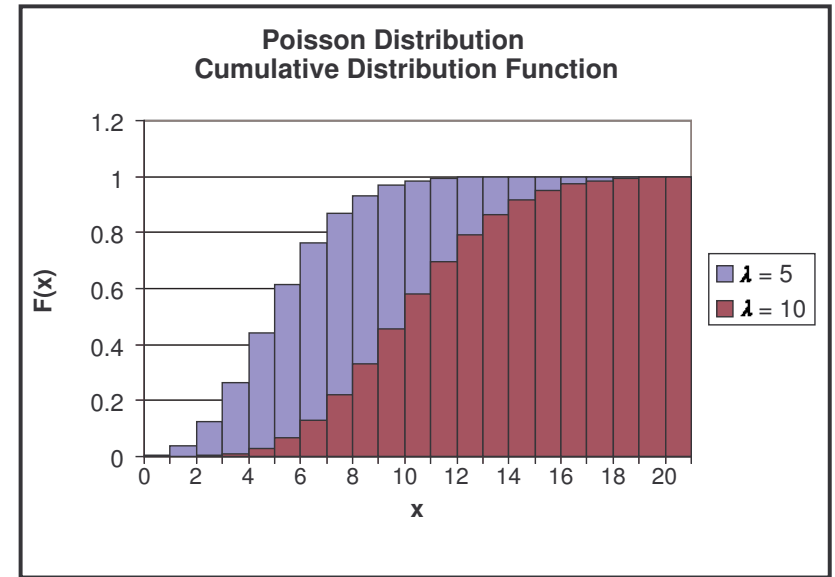
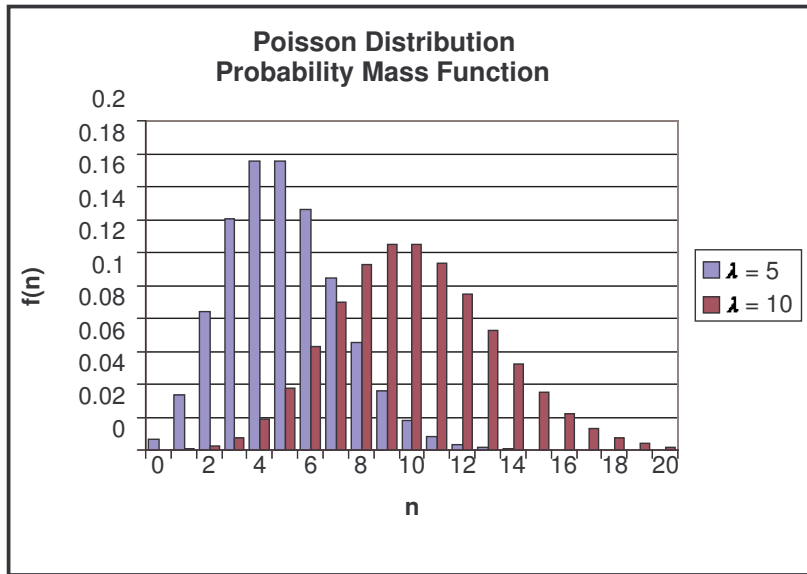
We say that the probability that a customer arrives in a given segment is  $p$ . For large  $n$ , the probability that two or more customers arrive in a given segment tends to 0. Arrival probabilities in two different segments are **independent**.

The number of arrivals in one second is thus distributed binomially with parameters  $n$  and  $p$ . To ensure average number of arrivals =  $\lambda$ , we take  $np = \lambda$  which implies  $p = \lambda/n$ .

To better smooth our model, we increase  $n$ , and accordingly decrease  $p$ , so that  $np$  remains equal  $\lambda$ .

As  $n$  tends to  $\infty$ , the distribution of the number of arrivals in one second approaches Poisson Distribution with parameter  $\lambda$ .

# Poisson Distribution with parameter $\lambda$



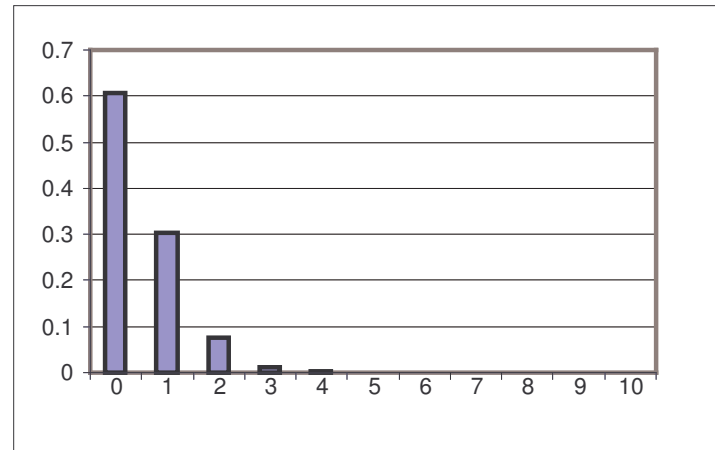
$$P(X = x) = e^{-\lambda} \frac{\lambda^x}{x!}, \quad x = 0, 1, 2, 3, \dots$$

$$P(X \leq x) = \sum_{k=0}^{\lfloor x \rfloor} e^{-\lambda} \frac{\lambda^k}{k!}, \quad x \geq 0$$

$$E(X) = \lambda, \quad \text{Var}(X) = \lambda$$

## Poisson Distribution with $\lambda < 1$

pmf (Probability Mass Function) of a Poisson Distribution  
with  $\lambda = 0.5$



Note that  $p(0) > 0.6$ .

## Approaching Poisson – the Proof

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k} =$$

$$\underline{\underline{np = \lambda}} \quad \frac{(n - k + 1)(n - k + 2) \cdots (n - 1)n}{k!} \left(\frac{\lambda}{n}\right)^k \left(1 - \frac{\lambda}{n}\right)^{n-k}$$

Now, as  $n \rightarrow \infty$ ,

$$\frac{(n - k + 1)(n - k + 2) \cdots (n - 1)n}{n^k} \rightarrow 1$$

and also

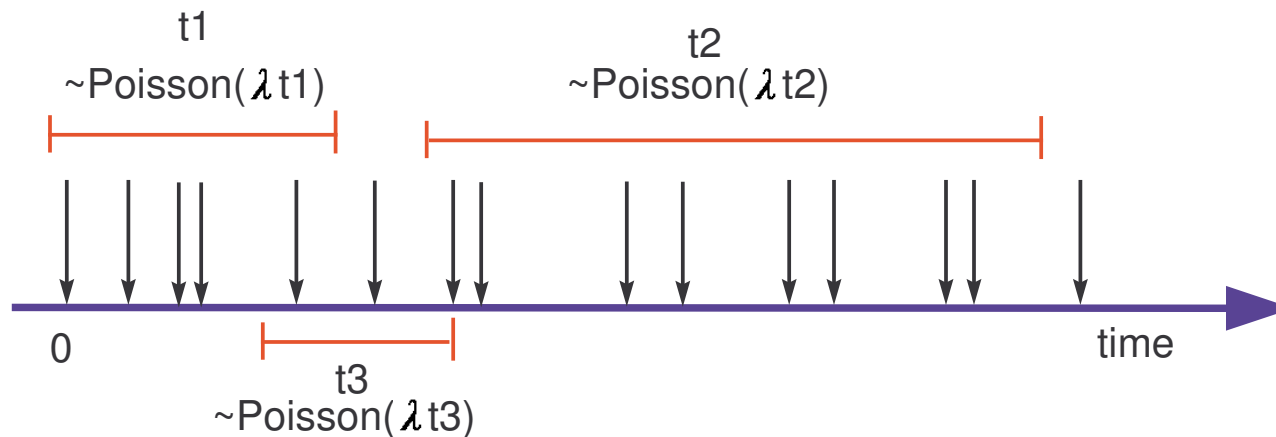
$$\left(1 - \frac{\lambda}{n}\right)^{-k} \rightarrow 1 \quad \text{and} \quad \left(1 - \frac{\lambda}{n}\right)^n \rightarrow e^{-\lambda}$$

Altogether,

$$p(X = k) \rightarrow e^{-\lambda} \frac{\lambda^k}{k!}$$

# Poisson Process

Constructing the arrival process over an interval of  $T$  seconds, by dividing it to  $Tn$  units and assigning with each, independently of all the others, probability  $\lambda/n$  for a customer arrival, we receive, as  $n \rightarrow \infty$ , a **Poisson Process**:



In such a process, the number of arrivals  $N(t)$  in a finite interval of length  $t$  seconds is distributed as Poisson with parameter  $\lambda t$ :

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

Moreover, the number  $N(t_1, t_2)$  of arrivals in the time interval  $(t_1, t_2)$ , and the number of arrivals  $N(t_3, t_4)$  in a non-overlapping interval  $(t_3, t_4)$ , are independent.

# Properties of Poisson Process

- A good model for arrivals that originate from a large population of independent users.
- The most "random" process with a given average arrival rate  $\lambda$ .
- By the Taylor series expansion of Poisson, with parameter  $\lambda\tau$ , the probability  $P(i, \tau)$  for  $i$  arrivals in a time interval of very small length  $\tau$  is:

$$P(0, \tau) = e^{-\lambda\tau} = 1 - \lambda\tau + O(\tau^2)$$

$$P(1, \tau) = \lambda\tau e^{-\lambda\tau} = \lambda\tau - \lambda^2\tau^2 + O(\tau^3) = \lambda\tau + O(\tau^2)$$

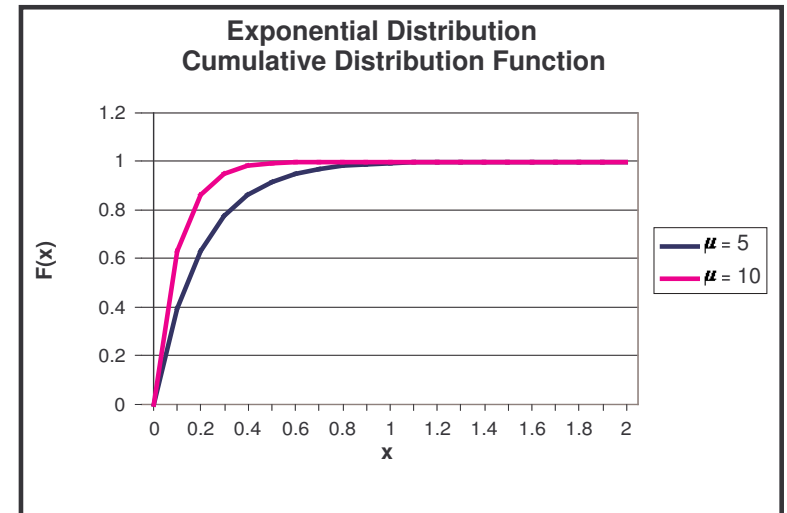
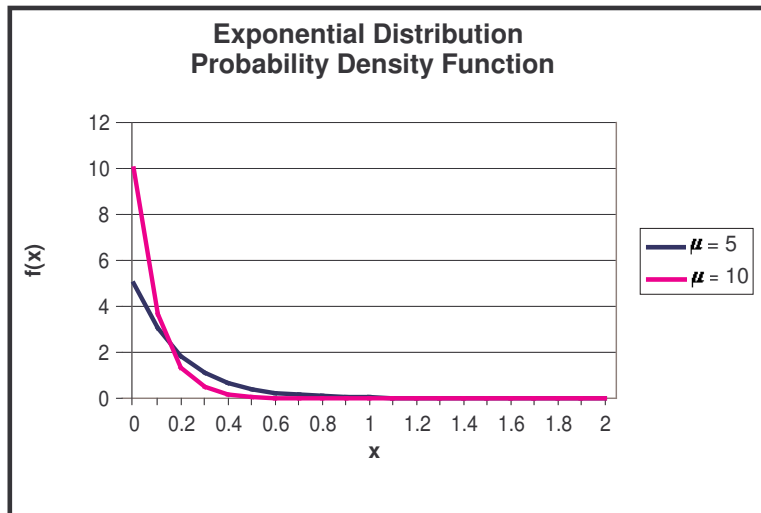
I.e., for a very small duration  $\tau$ , either 0 or 1 arrivals.

Probability  $\rightarrow 0$  for 2 arrivals or more.

- (Equivalent Definition:) The interarrival times are independent, and obey the exponential distribution with parameter  $\lambda$ :

$$P\{\text{interarrival time} > t\} = e^{-\lambda t}$$

# Exponential Distribution with parameter $\mu$



$$f_X(x) = \begin{cases} \mu e^{-\mu x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$F_X(x) = \int_0^x f_X(u) du = \begin{cases} 1 - e^{-\mu x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$E(X) = \frac{1}{\mu}, \quad \text{and} \quad \text{Var}(X) = \frac{1}{\mu^2}$$

## Key Property of Exponential Distribution: It is Memoryless

$$P\{X > r + s \mid X > r\} = P\{X > s\} \quad \text{for } r, s > 0$$

**Proof:**

$$\begin{aligned} P\{X > r + s \mid X > r\} &= \frac{P\{X > r + s, X > r\}}{P\{X > r\}} = \\ &= \frac{P\{X > r + s\}}{P\{X > r\}} = \frac{e^{-\mu(r+s)}}{e^{-\mu r}} = \\ &= e^{-\mu s} = P\{X > s\} \quad \text{QED} \end{aligned}$$

It can be shown that the Exponential distribution is the only memoryless distribution.

We model the amount of service time that customers need as distributed exponentially with parameter  $\mu$ . (A customer is already sitting an hour at the server, the probability that he leaves in the coming 10 seconds is the same as given he is only sitting there for 1 minute).

## Interarrivals in Poisson Processes are distributed exponentially

Let  $t_0$  be a time of one arrival in a Poisson process with parameter  $\lambda$ .

The probability that the time interval  $\tau$  until the following arrival is shorter than  $s$ , equals 1 minus the probability that no arrival occurs in the  $s$  seconds following  $t_0$ .

Since the number of arrivals in a time interval of length  $s$  is distributed by Poisson with parameter  $\lambda s$  (by the definition of a Poisson Process), we have:

$$P\{\tau \leq s\} = 1 - P\{0 \text{ arrivals in duration } s\} = 1 - e^{-\lambda s}$$

Differentiating:

$$p_\tau(s) = \frac{dP\{\tau \leq s\}}{ds} = \lambda e^{-\lambda s}$$

Thus, the interarrival times of a Poisson Process with rate  $\lambda$  are exponentially distributed with parameter  $\lambda$ .

## The Reverse Property

The reverse of Poisson  $\rightarrow$  Exponential is also true: if the interarrival times of events are exponentially distributed with parameter  $\lambda$  then the event counting process is Poisson with rate  $\lambda$ .

(we do not prove).

## The Hitchhiker's Paradox

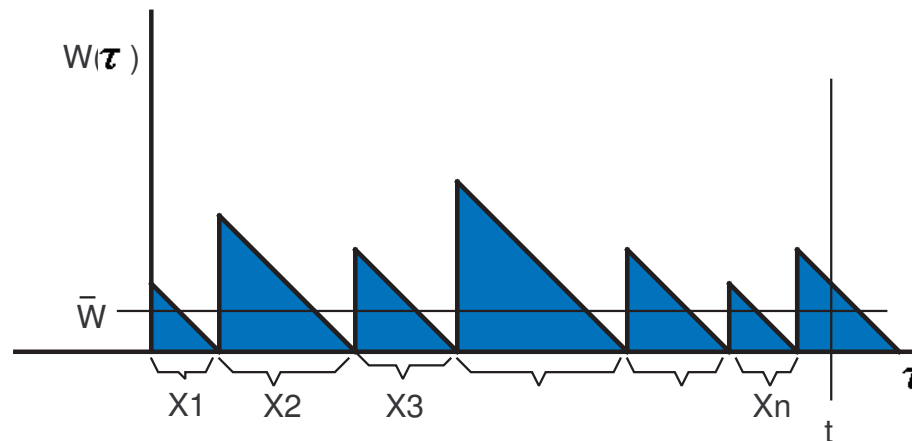
- Cars are passing a point of a road according to a Poisson process.
- The mean interval between the cars is 10 min.
- A hitchhiker arrives to the roadside point at random instant of time.
- **What** is the mean waiting time until the next car?

The interarrival times in a Poisson process are exponentially distributed. From the memoryless property of the exponential distribution, it follows that the (residual) time to the next arrival has the same exponential distribution and the expected time is thus 10 min.

But: if the mean length of an interarrival interval is 10 minutes, and the hitchhiker arrives at random, the expected time until the end of the interarrival interval should be 5 minutes.

## Explanation for the Hitchhiker's Paradox

Consider a long period of time  $t$ . The waiting time to the next car arrival  $W(\tau)$  as a function of the arrival instant  $\tau$  of the hitchhiker is represented by:



where the  $X_i$  are the interarrival intervals. The mean waiting time  $\bar{W}$  is the average value of this sawtooth curve:

$$\bar{W} = \frac{1}{t} \int_0^t W(\tau) d\tau \approx \frac{1}{t} \sum_{i=1}^n \frac{1}{2} X_i^2$$

## Explanation for the Hitchhiker's Paradox (cont.)

Note that long interarrival intervals contribute much more than short ones to the average waiting time.

As  $t$  grows,  $t/n \rightarrow \bar{X}$ , hence,

$$\bar{W} = \frac{1}{\bar{X}n} \sum_{i=1}^n \frac{1}{2} X_i^2 = \frac{1}{2} \overline{X^2}$$

For exponential distribution (as the  $X_i$  are distributed),

$$\text{Var}(X) = \frac{1}{\mu^2} = \bar{X}^2$$

Therefore,

$$\overline{X^2} = \text{Var}(X) + \bar{X}^2 = 2\bar{X}^2$$

Altogether,  $\bar{W} = \bar{X} = 10\text{min}$ .

# Stochastic Processes

A *Stochastic Process* is a function in time  $X(t)$  whose values are random variables. The sample space of these random variables is called *the state space* of the process.

A stochastic process can be described by a CDF:

$$F_X(x, t) \triangleq P(X(t) \leq x)$$

By intuitive notion of process, the random variable  $X(t_2)$  may depend on the random variable  $X(t_1)$ , for some  $t_1 < t_2$ . So we define the joint distribution that rules the process

$F_{\mathbf{X}}(\mathbf{x}; \mathbf{t}) = F_{X_1 \dots X_n}(x_1, \dots, x_n; t_1, \dots, t_n) \triangleq P(X(t_1) \leq x_1, \dots, X(t_n) \leq x_n)$   
 $n$  depends on the process (and our ability to analyse it).

A process is *stationary* if for every  $\tau$ ,  $F_{\mathbf{X}}(\mathbf{x}; \mathbf{t} + \tau) = F_{\mathbf{X}}(\mathbf{x}; \mathbf{t})$ .

A stochastic process can have a **discrete or continuous state space**, and it can run in a **discrete or continuous time**. With discrete time, the process can be represented as a sequence of RVs:  $X_1, X_2, \dots$ ; with continuous time we use the more general notation  $X(t)$ .

# Markov Chain

is a stochastic process with discrete, finite or countable, states set.

Chain “jumps” from state to state, with the **Memoryless (Markov) Property**: Future jumps of the chain depend only on the present state; they are totally independent of the history prior to arrival at the present state, or the time spent so far in the present state.

**Discrete time Markov Chain**: jumps occurs only on discrete points of time.

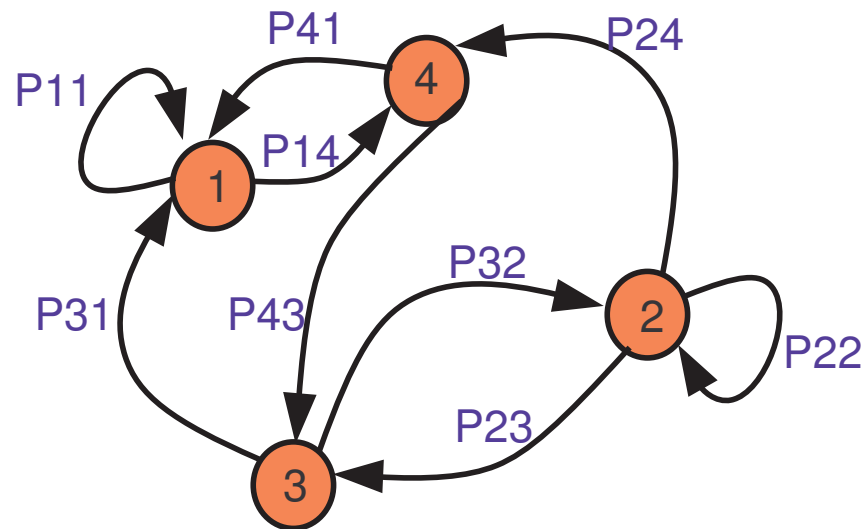
**Continuous time** – jump can happen at any time. However, the way in which history can affect the future of the process is described by the present state of the process and *not by how long the process has been there*. When the process changes state depends only upon which state it is in, not on how long it has been there. This induces exponential distribution of time spent between change of states; the parameter of the distribution may depend on the current state.

# Discrete-time Markov Chains

A sequence of random variables  $X_k \in \{0, 1, 2, \dots\}$  which by the definition of Markov Chain satisfies for every  $k$  and every states  $x_0, x_1, \dots, x_{k+1}$ ,

$$\begin{aligned} P(X_{k+1} = x_{k+1} | X_k = x_k, \dots, X_0 = x_0) &= \\ &= P(X_{k+1} = x_{k+1} | X_k = x_k) \triangleq P_{x_k, x_{k+1}} \end{aligned}$$

Describing the possible **states** as nodes, and the (positive) **transition probabilities**  $P_{i,j}$  as directed arcs, we have, e.g.,



# Transition Probabilities

Satisfy

$$P_{i,j} \geq 0 \quad \text{and} \quad \sum_{j=0}^{\infty} P_{i,j} = 1$$

Transition probability matrix  $\mathbf{P} \triangleq [P_{i,j}]$ .

Multi step transitions are built from the given one step transitions:

$$P_{i,j}^{(m)} \triangleq P(X_{n+m} = j | X_n = i), \quad n, m, i, j \geq 0$$

Chapman-Kolmogorov equations

$$P_{i,j}^{(m)} = \sum_{k=0}^{\infty} P_{i,k} P_{k,j}^{(m-1)}$$

$P_{i,j}^{(m)}$  is element  $(i, j)$  in the matrix  $\mathbf{P}^m$ .

# State Probabilities – Stationary Distribution

Time-dependent state probabilities:

$$\pi_j^n \triangleq P(X_n = j), \quad \pi^n \triangleq (\pi_0^n, \pi_1^n, \dots) = \pi^0 \mathbf{P}^n$$

If, independently of  $\pi^0$ , the time-dependent probability converges to a limit:

$$\pi = \lim_{n \rightarrow \infty} \pi^n$$

that limit is called the **stationary state distribution** of the process, and it satisfies

$$\pi = \pi \mathbf{P} \quad \text{and} \quad \pi_j = \lim_{n \rightarrow \infty} \mathbf{P}_{i,j}^n \quad (\text{for all } i)$$

The existence of this limit depends on the structure of the Markov Chain.

# Classification of Markov Chains

## Ergodic Chains

A state  $j$  can be reached from state  $i$  if for some  $m \geq 0$ ,  $P_{i,j}^{(m)} > 0$ .  
The chain is irreducible if  $j$  is reachable from  $i$  for all pairs  $i, j$  (including  $j = i$ ).

A state  $i$  is periodic if there is  $d > 1$  such that  $P_{i,i}^{(m)} > 0 \Rightarrow m = ad$ .  
A chain is aperiodic if none of its state is periodic.

An irreducible and aperiodic chain is called ergodic.

**Theorem:** With an ergodic chain, the limit  $\pi$  exists, and is unique.

When the chain reaches equilibrium, the frequency at which it visits any state  $j$  is  $\pi_j$ .

In other words: the relative number of appearances of  $j$  in any long sequence of state transitions approaches  $\pi_j$ .

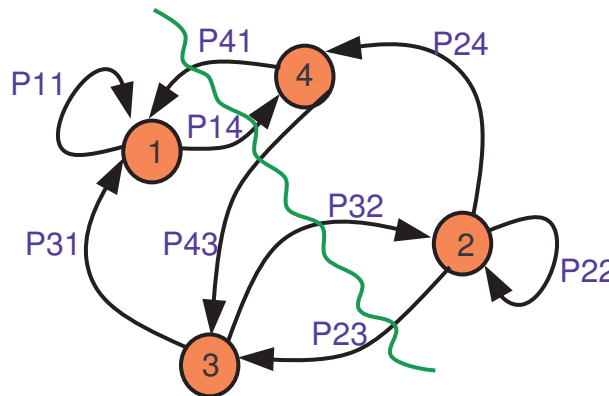
## At Equilibrium

The frequency of transitions into any given state equals the frequency of transitions out of that state:

$$\sum_i \pi_i P_{i,j} = \pi_j = \pi_j \sum_i P_{j,i} = \sum_i \pi_j P_{j,i}$$

(using the fact that  $1 = \sum_i P_{j,i}$ ).

This generalizes to the **Ballance Equation**: At equilibrium, the frequency of transitions into any given **set of states** equals the frequency of transitions out of that set.



At equilibrium, the net flow of probability mass across a membrane that surrounds some states is zero.

# The professor, the Umbrella, and Markov

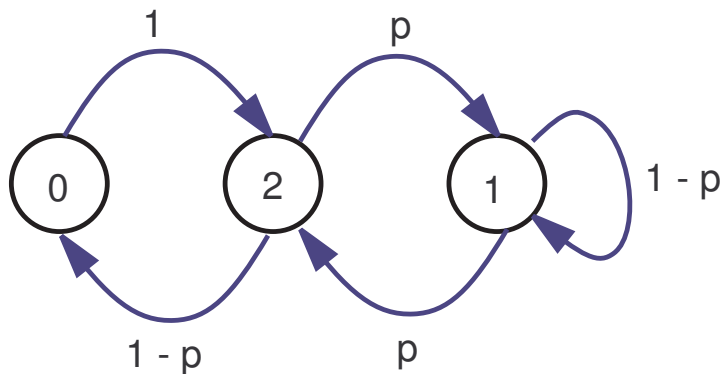
Absent-minded professor uses two umbrellas when commuting between home and office.

If it rains and an umbrella is available at her location, she takes it.

If it does not rain, she always forgets to take an umbrella.

Let  $p$  be the probability of rain each time she commutes.

What is the probability that she gets wet on any given day?

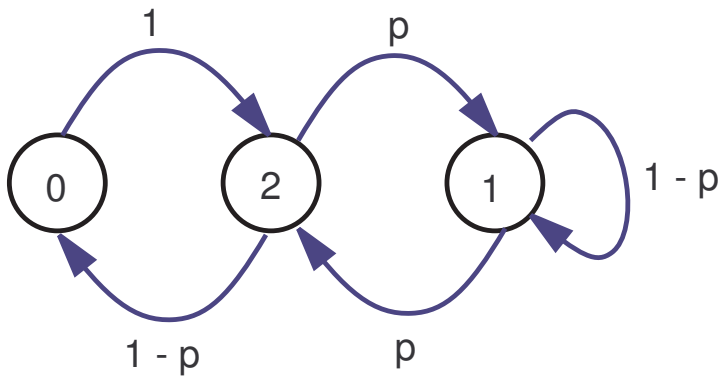


Transition Matrix:

$$P = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1-p & p \\ 1-p & p & 0 \end{bmatrix}$$

$P_{i,j}$ : the probability to go to state  $j$ , given now in state  $i$ .  $i = 0, 1, 2$ .

## Umbrella (cont.)



Transition Matrix:

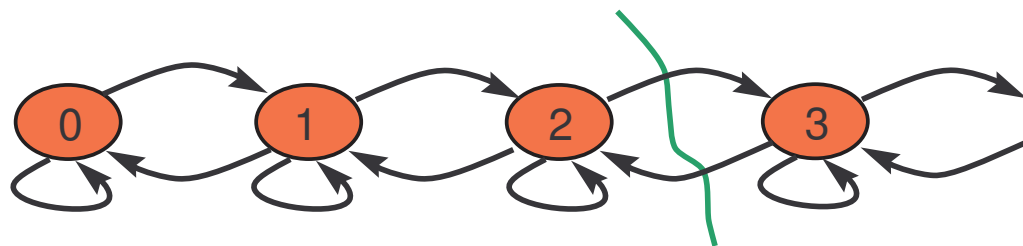
$$P = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1-p & p \\ 1-p & p & 0 \end{bmatrix}$$

$$\begin{cases} \pi = \pi P \\ \sum_i \pi_i = 1 \end{cases} \Rightarrow \begin{cases} \pi_0 = (1-p) \pi_2 \\ \pi_1 = (1-p) \pi_1 + p \pi_2 \\ \pi_2 = \pi_0 + p \pi_1 \\ \pi_0 + \pi_1 + \pi_2 = 1 \end{cases} \Rightarrow \begin{cases} \pi_0 = \frac{1-p}{3-p} \\ \pi_1 = \frac{1}{3-p} \\ \pi_2 = \frac{1}{3-p} \end{cases}$$

$$P(\text{gets wet}) = \pi_0 p = p \frac{1-p}{3-p}$$

## Birth-Death Processes

A special type of Markov chain where if  $X_k = i$  then  $X_{k+1} = i + 1, i, \text{ or } i - 1$ :



When reaches equilibrium, the balance equation for the green line yields:  $\pi_{n-1}P_{n-1,n} = \pi_n P_{n,n-1}$ , and by induction

$$\pi_n = \prod_{i=1}^n \frac{P_{i-1,i}}{P_{i,i-1}} \pi_0.$$

Together with  $\sum_{m=0}^{\infty} \pi_m = 1$  : 
$$\pi_n = \frac{\prod_{i=1}^n \frac{P_{i-1,i}}{P_{i,i-1}}}{\sum_{m=0}^{\infty} \prod_{i=1}^m \frac{P_{i-1,i}}{P_{i,i-1}}} .$$

When  $P_{n-1,n} = p$  and  $P_{n,n-1} = q$  for all  $n \geq 1$ , then with  $\rho \triangleq p/q$  we have

$$\pi_n = \rho^n / \sum_{m=0}^{\infty} \rho^m = \rho^n (1 - \rho). \quad \text{Convergence only possible for } \rho < 1.$$

# Succinct notation of a Queueing System

$A/S/m$  or  $A/S/m/c$  or  $A/S/m/c/p$

**A:** Arrival statistics. Popular values:

**M** – exponential interarrival distribution (M = Markovian, memoryless);  
Poisson process

**D** – deterministic, constant interarrival times, or

**G** – general (unspecified) distribution.

**S:** Customer's service time statistics: service time demanded from a single server.

Popular values: Same as for  $A$ .

**m:** Number of servers. A positive integer. All servers assumed identical.

**c:** System Capacity: maximal number of customers in the system – waiting and being serviced. A positive integer, or infinite (for theoretical interest). When omitted – assumed infinite.

**p:** Size of customer population. Infinite by default.

# Succinct notation, Examples

*M/M/1* :

Poisson Arrival Process

Exponential service time distribution

single server

unlimited number of waiting places

unlimited customer population

*M/M/m/m* :

Poisson Arrival Process

Exponential service time distribution

$m$  servers and  $m$  system places. No waiting room. Loss system.

## Discrete-Time M/M/1 Queue

We partition the time line into small time-slots, which are short time intervals of length  $\delta$ .

When the Arrival Process is a Poisson process with parameter  $\lambda$ , the probability  $P(k)$  that  $k$  arrivals occur in a given time slot is:

$$P(k) = e^{-\lambda\delta} \frac{(\lambda\delta)^k}{k!}$$

Hence,

$$P(0) = e^{-\lambda\delta} = 1 - \lambda\delta + \frac{(\lambda\delta)^2}{2} - \dots = 1 - \lambda\delta + o(\delta)$$

$$P(1) = e^{-\lambda\delta} \lambda\delta = \lambda\delta \left( 1 - \lambda\delta + \frac{(\lambda\delta)^2}{2} - \dots \right) = \lambda\delta + o(\delta)$$

$$P(k \geq 2) = 1 - P(0) - P(1) = o(\delta)$$

## Discrete-Time M/M/1 Queue (cont.)

When customer's service time distributed exponentially with parameter  $\mu$ , if the queue system is not empty (i.e., the server works non-stop), the departure process is a Poisson process with parameter  $\mu$ . Hence, for non-empty queue, the probability  $Q(k)$  that  $k$  departures occur in a given time-slot of length  $\delta$  is:

$$Q(0) = e^{-\mu\delta} = 1 - \mu\delta + o(\delta)$$

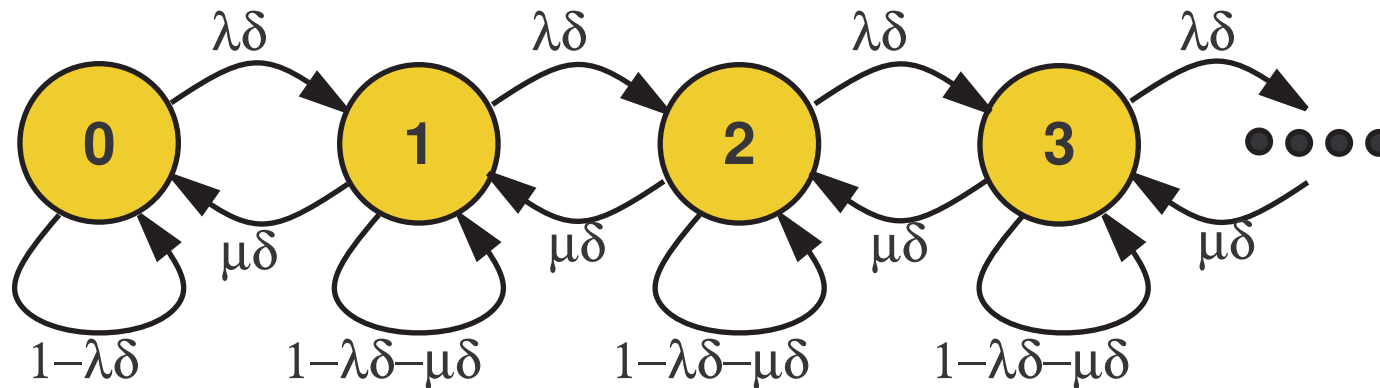
$$Q(1) = e^{-\mu\delta} \mu\delta = \mu\delta + o(\delta)$$

$$Q(k \geq 2) = 1 - Q(0) - Q(1) = o(\delta)$$

Altogether, in each time slot of length  $\delta$ :

- One arrival occurs with probability  $p = \lambda\delta$  or zero arrivals with probability  $1 - p = 1 - \lambda\delta$ .
- The customer in service departs with probability  $q = \mu\delta$ , or stays with probability  $1 - q = 1 - \mu\delta$ .
- Independent arrivals and service times; both arrival and departure occur with probability  $pq = o(\delta)$ .

## Discrete-Time M/M/1 Queue – A Birth-Death Process



We found above that when  $P_{n-1,n} = \lambda\delta$  and  $P_{n,n-1} = \mu\delta$  for all  $n \geq 1$ , then with  $\rho \triangleq \lambda\delta/\mu\delta = \lambda/\mu$  we have

$$\pi_n = \rho^n(1 - \rho) \quad \text{for all } n \geq 0$$

Where convergence occurs only for  $\rho < 1$ ,

i.e.,  $\lambda/\mu < 1 \Rightarrow \lambda < \mu$ , which we could expect:

**service rate should exceed arrival rate if the system is to be stable.**

$\pi_0 = 1 - \rho$  is the probability that the server is idle.

It thus works with probability  $1 - \pi_0 = \rho$ , called **server utilization**.

## Delay and Congestion of M/M/1

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

We have here a derivative of a geometric series:

$$\sum_{n=0}^{\infty} n\rho^{n-1} = \sum_{n=0}^{\infty} \frac{d}{d\rho} \rho^n = \frac{d}{d\rho} \sum_{n=0}^{\infty} \rho^n = \frac{d}{d\rho} \frac{1}{1-\rho} = \frac{1}{(1-\rho)^2}$$

Hence

$$N = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda}$$

By Little

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

## Delay and Congestion of M/M/1 (cont.)

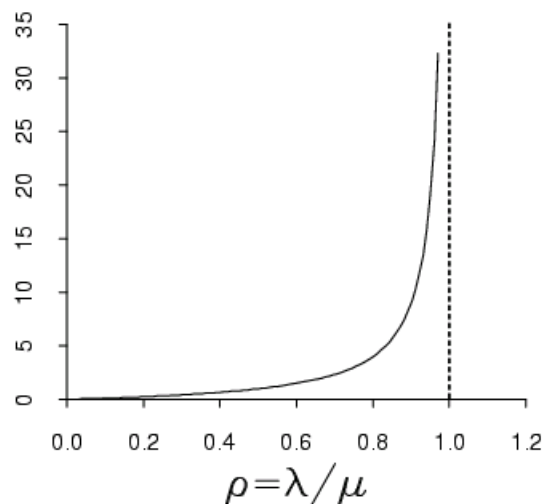
The average number of customers in the system is made of:

$$\rho \text{ \{customer being serviced\}} \quad \text{and} \quad N - \rho = \frac{\rho^2}{1 - \rho} \text{ \{waiting customers\}}$$

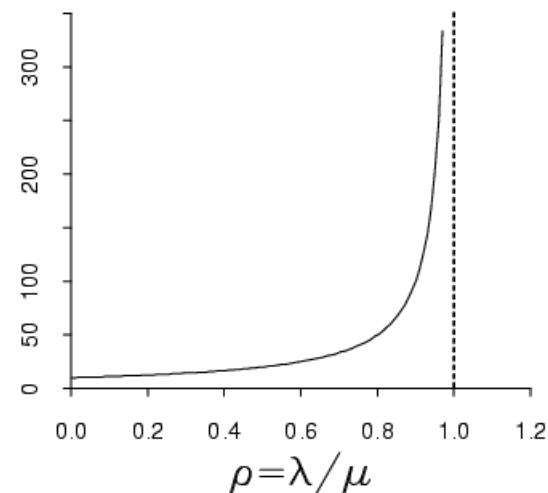
The average residence time can be partitioned:

$$\frac{1}{\mu} \text{ \{service time\}} \quad \text{and} \quad T - \frac{1}{\mu} = \frac{\rho}{\mu - \lambda} \text{ \{waiting time\}}$$

Expected number in the system



Expected time spent in the system



# Delay and Congestion of M/M/1

## What If

**What if** arrival rate  $\lambda$  of jobs grows by  $g$ , and consequently, the system manager increases the CPU power  $\mu$  by  $g$ . **Will that suffice?**

**Yes!!!**: Because  $\rho = \lambda/\mu$  stays the same, so does  $N = \rho/(1 - \rho)$ , and there is **no need** to increase the queue buffer.

Residence time  $T = 1/(\mu - \lambda)$  even shrinks by  $g$ .

(economy of scale..).

## $m$ Servers in the Service Center

Two servers now working. What is the probability distribution of the time until the next departure from the center?

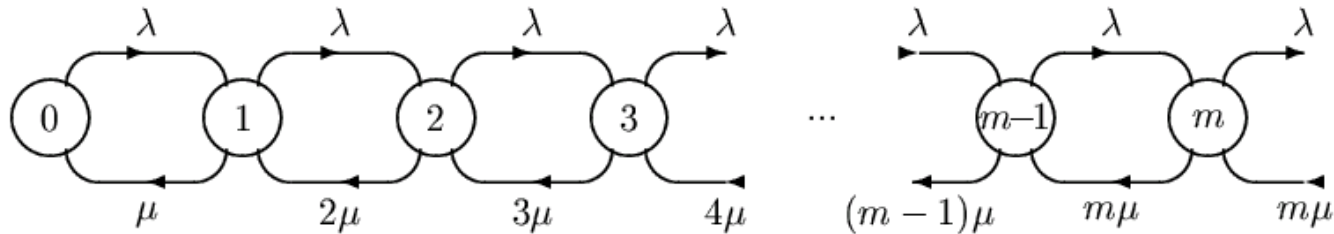
$$\begin{aligned}F_T(t) &= P(\min\{T^{(1)}, T^{(2)}\} \leq t) \\&= 1 - P(\min\{T^{(1)}, T^{(2)}\} > t) \\&= 1 - P(T^{(1)} > t, T^{(2)} > t) \\&\stackrel{\text{ind}}{=} 1 - P(T^{(1)} > t) P(T^{(2)} > t) \\&= 1 - e^{-\mu t} e^{-\mu t} \\&= 1 - e^{-2\mu t}\end{aligned}$$

i.e., that time is distributed exponentially with parameter  $2\mu$ .

Similarly, for  $m$  working servers – the time until the next departure from the center is distributed exponentially with parameter  $m\mu$ :

$$F_T(t) = 1 - e^{-m\mu t} \quad \text{and} \quad f_T(t) = m\mu e^{-m\mu t}.$$

## M/M/m systems – $m$ Servers



(self loops omitted) With  $\rho \triangleq \frac{\lambda}{m\mu}$  we have:

$$\begin{cases} n\mu\pi_n = \lambda\pi_{n-1} & (1 \leq n \leq m) \\ m\mu\pi_n = \lambda\pi_{n-1} & (n > m) \end{cases} \Rightarrow \pi_n = \begin{cases} \frac{m^n}{n!} \rho^n \pi_0 & (0 \leq n \leq m) \\ \frac{m^m}{m!} \rho^n \pi_0 & (n > m) \end{cases}$$

With  $\sum_{n=0}^{\infty} \pi_n = 1$  we derive  $\pi_0$ , and complete the formulas:

$$\pi_0 = \left[ \sum_{n=0}^{m-1} \frac{(m\rho)^n}{n!} + \frac{(m\rho)^m}{m!(1-\rho)} \right]^{-1}$$

## The Probability of the Need to Wait

$$P_Q = \sum_{n=m}^{\infty} \pi_n = \sum_{n=m}^{\infty} \frac{m^m}{m!} \rho^n \pi_0 = \frac{\pi_0 m^m \rho^m}{m!} \sum_{n=0}^{\infty} \rho^n$$

This is the Erlang formula:

$$P_Q = \frac{\pi_0 (m\rho)^m}{m!(1-\rho)}$$

## Delay and Congestion of M/M/m

The expected number of customers in the queue is:

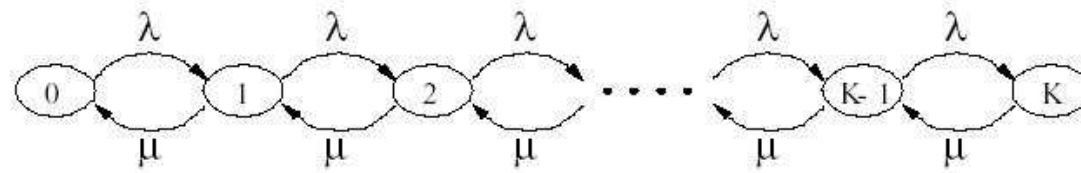
$$\begin{aligned} N_Q &= \sum_{n=m+1}^{\infty} (n-m)\pi_n = \frac{m^m \pi_0}{m!} \sum_{n=m+1}^{\infty} (n-m)\rho^n \\ &= \frac{\pi_0 m^m \rho^{m+1}}{m!} \sum_{n=0}^{\infty} n\rho^{n-1} = P_Q \frac{\rho}{1-\rho} \end{aligned}$$

By Little we get that the average time waiting is  $N_Q/\lambda$ , and the average residence time is thus  $T = N_Q/\lambda + 1/\mu$ .

Little once again:

$$N = \lambda T = P_Q \frac{\rho}{1-\rho} + \frac{\lambda}{\mu} = P_Q \frac{\rho}{1-\rho} + m\rho$$

## M/M/1/K – Finite Queue Capacity



This is very much like M/M/1, except that for  $k > K$ ,  $\lambda_k = 0$ , and  $\pi_k = 0$ . Writing, again,  $\rho = \lambda/\mu$  we have:

$$\pi_k = \pi_0 \rho^k \quad \text{for } k \leq K$$

and

$$\pi_0 = \frac{1}{\sum_{k=0}^K \rho^k} = \frac{1 - \rho}{1 - \rho^{K+1}}$$

which gives for  $k \leq K$

$$\pi_k = \frac{1 - \rho}{1 - \rho^{K+1}} \rho^k$$

From here, we can compute the average number of customers in the system:  $N = \sum_k k \pi_k$ , and continue with Little to get the average residence time.

## Merging of two Poissons is a Poisson As Well

Given two independent Poisson random variables  $X_1$  and  $X_2$  with parameters  $\lambda_1$  and  $\lambda_2$ , the random variable  $X = X_1 + X_2$  is also Poisson, with parameter  $\lambda_1 + \lambda_2$ :

$$\begin{aligned} P(X = k) &= \sum_{i=0}^k P(X_1 = i) P(X_2 = k - i) \\ &= \sum_{i=0}^k e^{-\lambda_1} \frac{\lambda_1^i}{i!} e^{-\lambda_2} \frac{\lambda_2^{k-i}}{(k-i)!} \\ &= e^{-(\lambda_1 + \lambda_2)} \sum_{i=0}^k \frac{\lambda_1^i \lambda_2^{k-i}}{i!(k-i)!} \\ &= e^{-(\lambda_1 + \lambda_2)} \sum_{i=0}^k \binom{k}{i} \frac{\lambda_1^i \lambda_2^{k-i}}{k!} \\ &= e^{-(\lambda_1 + \lambda_2)} \frac{(\lambda_1 + \lambda_2)^k}{k!} \quad \text{QED} \end{aligned}$$

## Splitting of a Poisson Process

Given a Poisson customer arrival process  $X$  of rate  $\lambda$ .

Each arriving customer is classified as **red** with probability  $p$ , independently of all the others. Then, the arrival process of red customers,  $X_p$ , is also Poisson with rate  $p\lambda$ :

The probability that out of  $i$  arriving customers,  $k$  would be classified red is  $\binom{i}{k} p^k (1-p)^{i-k}$ . Hence,

$$\begin{aligned}
 P(X_p = k) &= \sum_{i=k}^{\infty} P(X = i) \binom{i}{k} p^k (1-p)^{i-k} \\
 &= \sum_{i=k}^{\infty} e^{-\lambda} \frac{\lambda^i}{i!} \binom{i}{k} p^k (1-p)^{i-k} \\
 &= e^{-\lambda} \frac{p^k \lambda^k}{k!} \sum_{i=k}^{\infty} \frac{\lambda^{i-k} (1-p)^{i-k}}{(i-k)!} = e^{-\lambda} \frac{p^k \lambda^k}{k!} \sum_{j=0}^{\infty} \frac{\lambda^j (1-p)^j}{j!} \\
 &\stackrel{\text{Taylor}}{=} e^{-\lambda} \frac{(p\lambda)^k}{k!} e^{\lambda(1-p)} = e^{-p\lambda} \frac{(p\lambda)^k}{k!} \quad \text{QED}
 \end{aligned}$$

# Burke's Theorem

## Poisson In $\Rightarrow$ Poisson Out

Departure process of M/M/1 queue is Poisson with rate  $\lambda$  independent of actual arrival process.

Proof by reversing the Markov Chain.

Hence, Two cascaded, independently operating M/M/1 systems can be analyzed separately.

# Jackson Theorem

## A Network of M/M/1-s

For an arbitrary network of  $k$  M/M/1 queueing systems, the probability  $P(n_1, n_1, \dots, n_k)$  that there are  $n_i$  customers at the  $i$ -th queueing system satisfies:

$$P(n_1, n_1, \dots, n_k) = P_1(n_1)P_2(n_2) \cdots P_k(n_k)$$

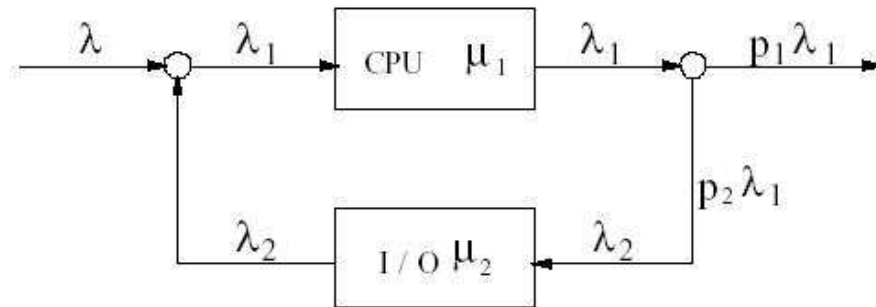
where

$$P_j(n_j) = \rho_j^{n_j} (1 - \rho_j)$$

That is, in terms of the number of customers in each system, individual systems act as if they are independent M/M/1 queues (they may not).

Generalizes to any type of queues (not only M/M/1).

## Application of Jackson



Given  $\lambda$ ,  $p_1$ , and  $p_2 = 1 - p_1$ , and the service rates  $\mu_1$  and  $\mu_2$ , find total number of customers in the system, and average residence time.

By the merging property of Poisson processes:

$$\lambda_1 = \lambda + \lambda_2, \quad \lambda_2 = p_2 \lambda_1$$

$$\Rightarrow \lambda_1 = \lambda / p_1, \quad \lambda_2 = \lambda p_2 / p_1$$

## Application of Jackson (cont.)

Denote  $\rho_1 = \lambda_1/\mu_1$ , and  $\rho_2 = \lambda_2/\mu_2$ .

By Jackson and M/M/1 properties:

$$P(i, j) = \rho_1^i (1 - \rho_1) \rho_2^j (1 - \rho_2)$$

and

$$N_1 = \frac{\rho_1}{1 - \rho_1}, \quad \text{and} \quad N_2 = \frac{\rho_2}{1 - \rho_2}$$

and

$$N = N_1 + N_2$$

By Little:

$$T = N/\lambda$$

All these are expressed using only the given parameters.

## Simulation: Continuous-valued simulation

- The system state at any point in time is described by values for a set of continuous-valued state variables (voltage, velocity, position, etc.)
- The evolution of the system state in time is usually characterized by a set of partial differential equations

For the simulation, continuous values are quantized (digitized) and the differential equations are approximated by difference equations. Time typically advances in fixed increments or clock ticks. When time is incremented, the simulation computes new values for all system variables by the difference equations. If the time increment is too large, state variable values may not converge, or may violate some model-specific constraints. If this occurs, the simulator may try reducing the size of the clock tick and recompute the new values.

## Simulation: Discrete-event simulation

- System state variables only take on discrete values
- Time may advance by fixed or variable amounts, and state variables do not change within any interval over which time advances in a single step

E.g., a model of the activity in the lobby of a bank. State variables are length of lines at each teller station. These lengths change only by integer values, and changes can occur at any time. The simulation advances to the next event of joining or leaving of a client to any queue.

## Simulation: Discrete-event simulation (cont.)

Computer systems are often simulated using one of two types of discrete-event simulation:

- trace driven – by a log of events taken from a run of a benchmark. E.g., recording the time-ordered sequence of memory accesses generated during the execution of some programs, in order to simulate different cache designs.
- stochastic discrete-event simulation – the system workload input is characterized by various probability distributions. During the simulation, these distributions are used to produce random values which are the inputs to the simulation model.

Both kinds are event-driven: simulation advances time only by the occurrence of events (= state change). The simulation computes new values for affected state variables when an event occurs, and then advances time to the time for the next event to occur. This is repeated until the simulation reaches some user-specified condition (like number of events, stability, etc.)

Needs random number generators.

# Linear Congruential Generator

This generator produces  $X_i$  according to:

$$X_i = a X_{i-1} + b \pmod{m}$$

$X_0$  is called the **seed** of the sequence.

The number of distinct values before the sequence begins to repeat is called the **period** of the generator. It is  $\leq m$ . The period depends on  $a, b, m$ , and  $X_0$ .  $a$  should not be too small.

The numbers that are generated are uniformly distributed.

Number-theory conditions can guarantee that a linear congruential random number generator produces a sequence with period  $m$ .

E.g., the following set:

- \*  $b$  and  $m$  are relatively prime
- \* every prime factor of  $m$  is also a prime factor of  $a - 1$
- \* if 4 divides  $m$  then 4 divides  $a - 1$ .

# Shuffling

Produce a “random” sequence using two linear congruential generators, with a period much larger than the modulus of any of them.

Use two linear congruential generators to produce sequences  $X_1, X_2, \dots$  and  $Y_1, Y_2, \dots$  with periods  $m_X$  and  $m_Y$  that are relatively prime. Initialize a table  $T$  of size  $k$  with the values  $X_1, X_2, \dots, X_k$ . Then iterate the following steps:

1. Generate the next values  $X$  and  $Y$  from both sequences
2. Use  $Y$  to produce an index  $i = \lceil kY/m_Y \rceil$  into table  $T$ .
3. Use  $T[i]$  as the next random value, and then update the table by replacing  $T[i]$  with  $X$ .

## Shuffling (cont.)

The effect of this algorithm is to choose randomly, from among the last  $k$  values of the sequence  $X$ , values that have not been previously chosen.  $k = 128$  should be adequate.

There are tests for the extent of randomness in a given sequence, based on the relative frequencies of subsequences of different lengths.

## Generating non-uniform random numbers

Let  $Y$  be a continuous random variable with a cumulative distribution function  $F_Y(y)$  which has an inverse  $F_Y^{-1}(x)$  for  $0 \leq x \leq 1$ . This happens where  $f_Y(y) > 0$ .

Define a new random variable  $Z = F_Y(Y)$ . Then

$$F_Z(z) = \Pr[Z \leq z] = \Pr[F_Y(Y) \leq z], \quad 0 \leq z \leq 1$$

Since  $F_Y^{-1}(x)$  is assumed to exist,

$$\begin{aligned} \Pr[F_Y(Y) \leq z] &= \Pr[Y \leq F_Y^{-1}(z)] = \\ &= F_Y(F_Y^{-1}(z)) = z \end{aligned}$$

Hence,  $F_Z(z) = z$ ,  $0 \leq z \leq 1$ , and thus  $Z$  is uniformly distributed over  $[0, 1]$ , denoted  $z \sim U(0, 1)$ .

Now, generate a value  $z$  from  $U(0, 1)$ . Since  $z = F_Y(y)$ , we have  $F_Y^{-1}(z) = y$  where  $y$  is a value for a random variable with a cumulative distribution function  $F_Y()$ .

## Example: exponential distribution

With the exponential distribution,  $F_Y(y) = 1 - e^{-\lambda y}$ ,  $y \geq 0$ . Now, take  $z$  from  $U(0, 1)$ .

$$z = 1 - e^{-\lambda y}, \quad y \geq 0$$

$$e^{-\lambda y} = 1 - z$$

$$-\lambda y = \ln(1 - z)$$

$$y = -\frac{\ln(1 - z)}{\lambda}$$

Simplification: if  $Z$  is  $U(0, 1)$  then so is  $1 - Z$ . Hence we can compute a value for an exponentially distributed random variable as

$$y = -\frac{\ln z}{\lambda}$$

Since  $-\ln z \rightarrow \infty$  as  $z \rightarrow 0$ , a practical implementation of inversion of the exponential distribution must test for the case  $z = 0$  and return then the largest representable positive number, or configure the uniform random number generator to prevent it from producing 0.

# Generating non-uniform random numbers

## Discrete RV

Need to produce  $k \in 1, 2, \dots$  with probability  $p_k$ .

Divide the unit interval into segments of size  $p_k$ ,  
pick a number  $z$  at random, uniformly distributed  $(0,1)$ .

Output  $k$  such that the interval that  $z$  falls in is associated with  $k$ .

# Final Exam

Definitions, Formulae, and Theorems are given as needed. E.g.:

- Poisson Dist. with parameter  $\lambda$ :  $f_X(x) = e^{-\lambda}\lambda^x/x!$ , for  $x = 0, 1, 2, 3, \dots$
- Exponential Dist. with parameter  $\mu$ :  $f_X(x) = \mu e^{-\mu x}$  for  $x \geq 0$ ;  $= 0$  otherwise; and  $F_X(x) = 1 - e^{-\mu x}$  for  $x \geq 0$ ;  $= 0$  otherwise.
- In a Poisson process with arrival rate  $\lambda$ , the number of arrivals in any time interval of length  $t$  is Poisson distributed with parameter  $\lambda t$ .
- The inter-arrival times of a Poisson Process with rate  $\lambda$  are exponentially distributed with parameter  $\lambda$ .
- In M/M/1 queueing system, there is one server, arrivals are by a Poisson process of rate  $\lambda$ , and the demand for service time is exponentially distributed with parameter  $\mu$ .
- The probability that there are  $n$  customers in a stable M/M/1 queueing system is  $\rho^n(1 - \rho)$ .  $\rho$  is defined to be  $\lambda/\mu$ .
- The average number  $N$  of customers in a steady state M/M/1 system is  $N = \lambda/(\mu - \lambda) = \rho/(1 - \rho)$ .
- Departure process of M/M/1 queueing system is Poisson with rate  $\lambda$ .

## Definitions, Formulae, and Theorems are given as needed (cont.)

- Little's Theorem: The average number of customers in the system = average customer arrival rate (in number of customers per second) multiplied by the average number of seconds a customer spends in the system:  $N = \lambda T$ .
- Given two independent Poisson customer arrival processes  $X_1$  and  $X_2$  with rates  $\lambda_1$  and  $\lambda_2$ , the arrival rate of  $X = X_1 + X_2$  (the merger of the two arrival processes) is also Poisson process, of rate  $\lambda_1 + \lambda_2$ .
- Given a Poisson customer arrival process  $X$  of rate  $\lambda$ . Each arriving customer is classified as red with probability  $p$ , independently of all the others. Then, the arrival process of red customers is also Poisson with rate  $p\lambda$ .
- Jackson Theorem for a Network of M/M/1-s: For an arbitrary network of  $k$  M/M/1 queueing systems, the probability  $P(n_1, n_1, \dots, n_k)$  that there are  $n_i$  customers at the  $i$ -th queueing system satisfies:  $P(n_1, n_1, \dots, n_k) = P_1(n_1)P_2(n_2) \cdots P_k(n_k)$ .

## Example Questions

Consider an M/M/1 queueing system with arrival rate of  $\lambda = 2$  customers per hour, and service demand of  $\mu = 3$ . Questions (1) - (4) refer to this queueing system, in its steady state.

(1) How long, on the average, a customer stays in this queueing system (waiting and being served)?

- a. 1 / 3 hour - as the average service time he needs
- b. 1 / 2 hour - as the average interarrival time
- c. The average number of customers is  $N = \lambda/(\mu - \lambda)$ , and with Little's Theorem, the average residence time is  $N/\lambda = 1/(\mu - \lambda) = 1$  hour.
- d. 2 hours:  $\lambda(\mu - \lambda)$
- e. 3 hours:  $\mu(\mu - \lambda)$

(2) Snap-shooting this system at a random point in time, what is the probability to find it empty of customers?

- a. 0: The average number of customers in the system when in steady state is  $N = \lambda/(\mu - \lambda) = 2/(3 - 2) = 2$ . Hence, with probability 0 the

number of customers is different from 2.

b. 1:  $\mu - \lambda = 3 - 2 = 1$ .

c. 2 / 3:  $\rho = \lambda/\mu = 2/3$

d. 1 / 3:  $\rho^0(1 - \rho) = (1 - 2/3) = 1/3$

e. 1 / 2:  $1/\lambda = 1/2$ .

(3) Suppose that a customer has just arrived at 08:00. What is the probability that the next arrival will occur after 9:00?

a. By the inter-arrival distribution:  $1 - (1 - e^{-\lambda 1}) \approx 0.135$ .

b. By the inter-arrival distribution:  $1 - (1 - e^{-\mu 1}) \approx 0.05$ .

c. 0.5: independent arrivals.

d. 0: the next arrival will surely occur before 09:00.

e. 1: In a steady state, customer arrival occurs every half an hour.

(4) What is the throughput, i.e., the average number of customers served in one hour?

a. 3: the server can serve 3 customers per hour

b. 2: on average, 2 customers arrive per hour

c. 2.5: the average of a and b above

d. 5: the sum of a and b above

e. less than 2: 2 customers arrive per hour, but they get stuck in the waiting queue for a while.