

# Single queues

Tran, Van Hoai (hoai@hcmut.edu.vn)

Le, Hong Trang (lhtrang@hcmut.edu.vn)

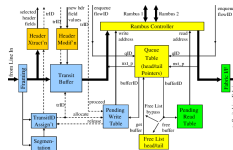
Faculty of Computer Science & Engineering  
HCMC University of Technology

2020-2021/Semester 1

- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues
- 6  $M/M/1$ 
  - Exercise
- 7  $M/M/n$ 
  - Exercise

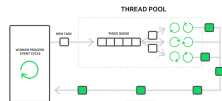
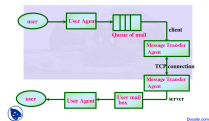
- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues
- 6 M/M/1
  - Exercise
- 7 M/M/n
  - Exercise

# Queues in real world

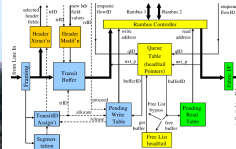


## Simple Mail Transfer Protocol (SMTP)

Out line of Internet Electronic Mail

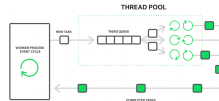
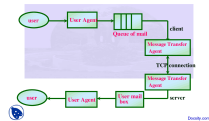


## Queues in real world



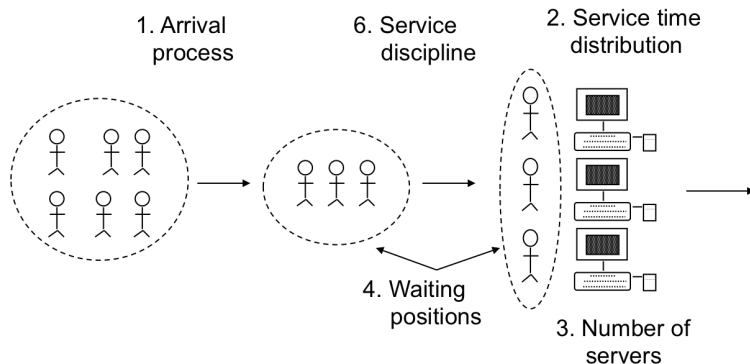
## Simple Mail Transfer Protocol (SMTP)

### Out line of Internet Electronic Mails



Are there any **common** structures of queues?

# Basic components of a queue



- **Customers** can be people, parts, vehicles, machines, jobs,...
- Queue might not be a **physical** line.

- $A/S/m/B/K/SD$ 
  - $A$ : Arrival process,
  - $S$ : Service time distribution,
  - $m$ : Number of servers,
  - $B$ : Number of buffers (system capacity),
  - $K$ : Population size,
  - $SD$ : Service discipline.

- Arrival times:  $t_1, t_2, \dots, t_j$ .
- Interarrival times:  $\tau_j = t_j - t_{j-1}$ .
- $\tau_j$  form a sequence of Independent and Identically Distributed (IID) random variables.
- Exponential + IID  $\rightarrow$  Poisson.
- Notation:
  - $M$  = Memoryless = Poisson,
  - $E$  = Erlang,
  - $H$  = Hyper-exponential,
  - $G$  = General  $\rightarrow$  Results valid for all distributions.



- Time each student spends at the terminal.
- Service times are IID.
- Distribution:  $M$ ,  $E$ ,  $H$ , or  $G$ .
- Device = Service center = Queue.
- Buffer = Waiting positions.

- First-Come-First-Served (FCFS);
- Last-Come-First-Served (LCFS);
- Last-Come-First-Served with Preempt and Resume (LCFS-PR);
- Round-Robin (RR) with a fixed quantum.
- Small Quantum  $\rightarrow$  Processor Sharing (PS);
- Infinite Server: (IS) = fixed delay;
- Shortest Processing Time first (SPT);
- Shortest Remaining Processing Time first (SRPT);
- Shortest Expected Processing Time first (SEPT);
- Shortest Expected Remaining Processing Time first (SERPT).
- Biggest-In-First-Served (BIFS);
- Loudest-Voice-First-Served (LVFS).

- $M$ : Exponential,
- $E_k$ : Erlang with parameter  $k$ ,
- $H_k$ : Hyper-exponential with parameter  $k$ ,
- $D$ : Deterministic  $\rightarrow$  constant,
- $G$ : General  $\rightarrow$  All.
- Memoryless:
  - Expected time to the next arrival is always  $1/\lambda$  regardless of the time since the last arrival,
  - Remembering the past history does not help.

## Example: $M/M/3/20/1500/FCFS$

- Time between successive arrivals is exponentially distributed.
- Service times are exponentially distributed.
- Three servers,
- 20 Buffers = 3 service + 17 waiting,  
After 20, all arriving jobs are lost,
- Total of 1500 jobs that can be serviced.
- Service discipline is FCFS.
- Defaults:
  - Infinite buffer capacity,
  - Infinite population size,
  - FCFS service discipline.
- $G/G/1 = G/G/1/1/1/FCFS$ .

- Bulk arrivals/service.
- $M^{[x]}$ :  $x$  represents the group size.
- $G^{[x]}$ : a bulk arrival or service process with general inter-group times.
- Example:
  - $M^{[x]}/M/1$ : Single server queue with bulk Poisson arrivals and exponential service times;
  - $M/G^{[x]}/m$ : Poisson arrival process, bulk service with general service time distribution, and  $m$  servers.

- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues
- 6 M/M/1
  - Exercise
- 7 M/M/n
  - Exercise

# Typical performance questions

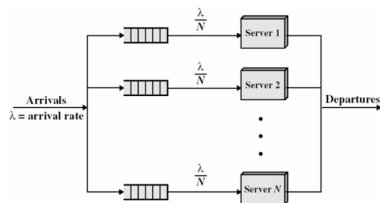
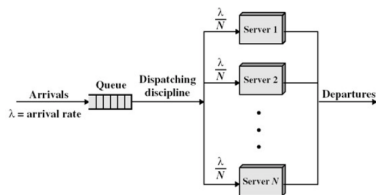
What is the ...

- average number of customers in the system?
- average time a customer spends in the system?
- probability a customer is rejected?
- fraction of time a server is idle?

# Typical performance questions

What is the ...

- average number of customers in the system?
- average time a customer spends in the system?
- probability a customer is rejected?
- fraction of time a server is idle?



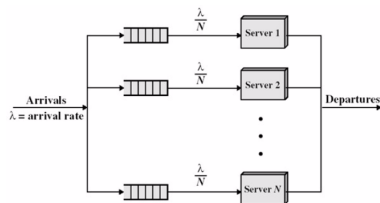
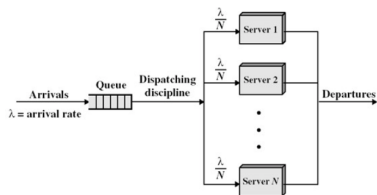
- What is average time waiting in the queue?
- What is variability of time in the queue?



# Typical performance questions

What is the ...

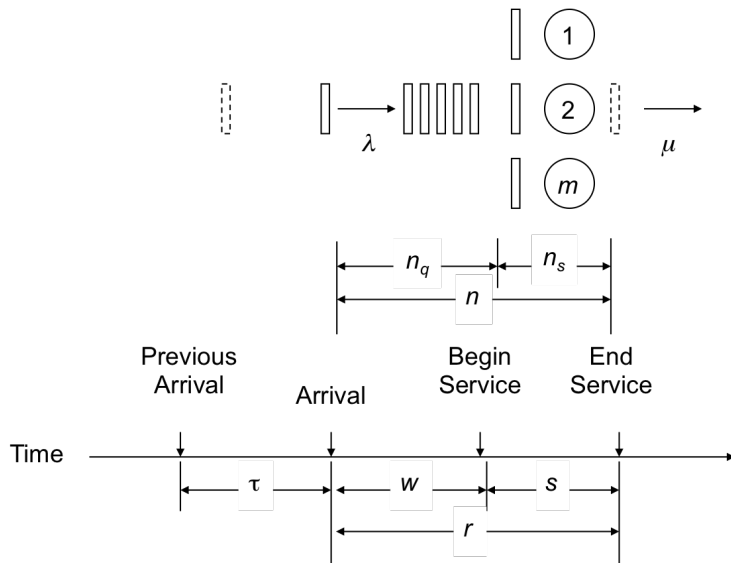
- average number of customers in the system?
- average time a customer spends in the system?
- probability a customer is rejected?
- fraction of time a server is idle?



- What is average time waiting in the queue?
- What is variability of time in the queue?

None of **done incorrectly** outcomes are investigated to produce metrics.

# Key variables (1)



## Key variables (2)

- $\tau$ : Inter-arrival time = time between two successive arrivals.
- $\lambda$ : Average arrival rate =  $1/E[\tau]$ 
  - May be a function of the state of the system  
E.g., number of jobs already in the system.
- $s$ : Service time per job.
- $\mu$ : Average service rate per server =  $1/E[s]$ .
- Total service rate for  $m$  servers is  $m\mu$ .
- $n$ : Number of jobs in the system.  
**Note:**  $n$  includes jobs currently receiving service as well as those waiting in the queue.

## Key variables (3)

- $n_q$ : Number of jobs waiting.
- $n_s$ : Number of jobs receiving service.
- $r$ : Response time or the time in the system (system time)
  - time waiting + time receiving service.
- $w$ : Waiting time
  - Time between arrival and beginning of service.

- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law**
- 4 Birth-death processes
- 5 Rules for All Queues
- 6  $M/M/1$ 
  - Exercise
- 7  $M/M/n$ 
  - Exercise

# Little's Law

Named after Little (1961).

## Little's Law

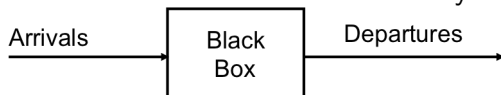
For any queuing system that has a steady state and has an average rate of  $\lambda$ ,

$$E[n] = \lambda E[r]$$

If the average system time is 2 hours, and customers arrive at a rate of 3 per hour then on average, there are 6 customers in the system.

# Discussion on Little's Law

Based on a black-box view of the system.



Little's law can be used for a system or **any** part of the system.

- Average number in queue = arrival rate  $\times$  average waiting time
- Average number in service = arrival rate  $\times$  average service time

Little's law requires **no assumptions** about arrival or service time distribution, the size of population, or limits on the system.

# Example on Little's Law

- Consider problem
  - A monitor on a disk server showed that the average time to satisfy an I/O request was 100 milliseconds.
  - The I/O rate was about 100 requests per second.
  - What was the average number of requests at the disk server?
- Using Little's law, average number in the disk server
  - = Arrival rate  $\times$  system time
  - = 100 (requests/second)  $\times$  (0.1 seconds)
  - = 10 requests.



- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes**
- 5 Rules for All Queues
- 6 M/M/1
  - Exercise
- 7 M/M/n
  - Exercise

## (Recall) Stochastic processes

- Process: sequence (family) of random variables that are functions of time.  
E.g.,  $n(t)$ : number of jobs waiting for CPU of a computer system at time  $t$ .
- Each  $n(t)$  is random variable which can be defined by a probability distribution.  $\Rightarrow$  Stochastic process.

# (Recall) Stochastic processes



- Process: sequence (family) of random variables that are functions of time.  
E.g.,  $n(t)$ : number of jobs waiting for CPU of a computer system at time  $t$ .
- Each  $n(t)$  is random variable which can be defined by a probability distribution.  $\Rightarrow$  Stochastic process.
- Types of stochastic processes
  - Discrete or Continuous State Processes
    - $n(t)$  is a discrete-state process
    - $w(t)$  is a continuous-state process
  - Markov Processes
  - Birth-death Processes
  - Poisson Processes

# Markov processes

- A **Markov process** = A stochastic process in which future states are independent of the past and depend only on the present.  $\Rightarrow$  easier to analyze, not have to remember past trajectory.
- A **Markov chain** = A discrete-state Markov process
- Some remarks:
  - Knowing current (present) state is sufficient
  - **Not necessary** to know how long the process has been in the current state.  $\Rightarrow$  State time has a memoryless distribution.

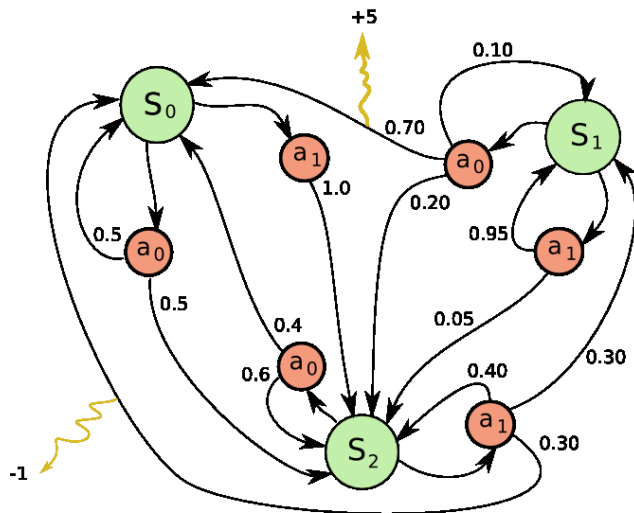
E.g., exponential distribution

$$\begin{aligned}Pr\{X > s + t | X > t\} &= \frac{Pr\{X > s + t \text{ and } X > t\}}{Pr\{X > t\}} \\&= \frac{Pr\{X > s + t\}}{Pr\{X > s\}} \\&= \frac{e^{-\lambda(s+t)}}{e^{-\lambda s}} \\&= e^{-\lambda t}\end{aligned}$$

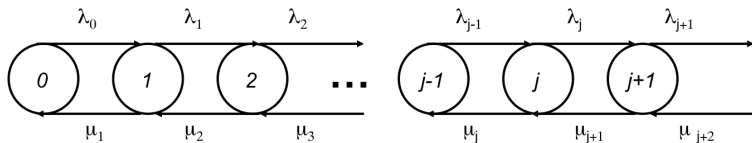
The time spent by a job in such a queue is a Markov process and the number of jobs in the queue is a Markov chain.

# Markov decision process in learning

Vision



# Birth-death processes



- A **Birth-death process** = A discrete-state Markov process in which the transitions are restricted to neighboring states.  
 $\Rightarrow$  Process in state  $n$  can change only to state  $n + 1$  or  $n - 1$ .

Number of jobs in a queue with a single server and individual arrivals (**not bulk** arrivals).

- An arrival (birth): state changed by  $+1$ .
- A departure (death): state changed by  $-1$ .

### Theorem

The steady-state probability  $p_n$  of a birth-death process being in state  $n$  is given by

$$p_n = \frac{\lambda_0 \lambda_1 \dots \lambda_{n-1}}{\mu_1 \mu_2 \dots \mu_n} p_0, n = 1, 2, \dots, \infty$$

Here,  $p_0$  is the probability of being in the zero state.

# Poisson processes (1)

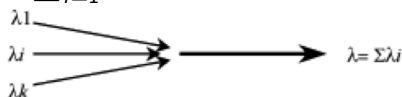
- Inter-arrival time  $\tau = \text{IID}$  (identically and independently distributed) and exponential
  - number of arrivals  $n$  over a given interval  $(t, t + x)$  has a Poisson distribution  $P\{X = k\} = \frac{\lambda^k e^{-\lambda}}{k!}$
  - Arrival process is a **Poisson process** or **Poisson stream**.



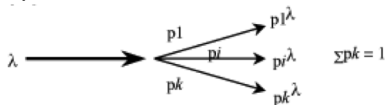
# Poisson processes (1)

- Inter-arrival time  $\tau = \text{IID}$  (identically and independently distributed) and exponential
  - number of arrivals  $n$  over a given interval  $(t, t + x)$  has a Poisson distribution  $P\{X = k\} = \frac{\lambda^k e^{-\lambda}}{k!}$
  - Arrival process is a **Poisson process** or **Poisson stream**.
- Properties

(1) Merging:  $\lambda = \sum_{i=1}^k \lambda_i$ .

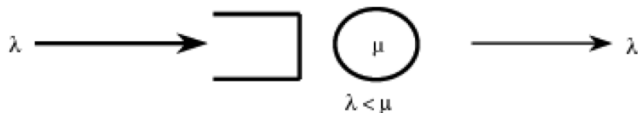


(2) Splitting: If the probability of a job going to  $i^{\text{th}}$  substream is  $p_i$ , each substream is also Poisson with a mean rate of  $p_i \lambda$ .



## ■ Properties (cont.)

- (3) If the arrivals to a single server with exponential service time are Poisson with mean rate  $\lambda$ , the departures are also Poisson with the same rate  $\lambda$  provided  $\lambda < \mu$ .

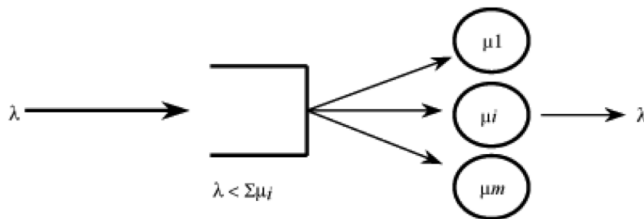


# Poisson processes (3)

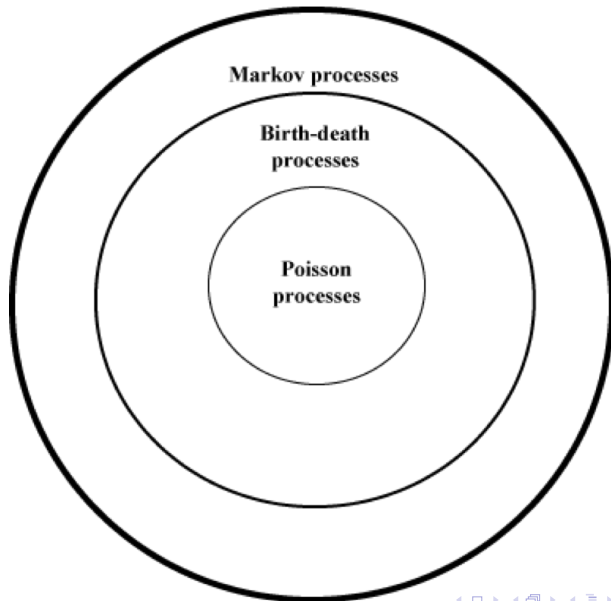
## ■ Properties (cont.)

(4) If the arrivals to a service facility with  $m$  service centers are Poisson with a mean rate  $\lambda$ , the departures also constitute a Poisson stream with the same rate  $\lambda$ , provided  $\lambda < \sum_i \mu_i$ .

- Here, the servers are assumed to have exponentially distributed service times.



# Relationship among stochastic processes



- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues**
- 6  $M/M/1$ 
  - Exercise
- 7  $M/M/n$ 
  - Exercise

# Rules for All Queues: apply to G/G/m queues (1)

## (1) Stability Condition: $\lambda < m\mu$

- Finite-population and infinite-buffer systems are always stable.

## (2) Number in System versus Number in Queue:

- $n = n_q + n_s$ , where  $n$ ,  $n_q$ , and  $n_s$  are random variables;
- $E[n] = E[n_q] + E[n_s]$ ;
- If the service rate is independent of the number in the queue,  $Cov(n_q, n_s) = 0$

$$\text{Var}[n] = \text{Var}[n_q] + \text{Var}[n_s].$$

## Rules for All Queues: apply to G/G/m queues (2)

- (3) Number versus Time: if jobs are not lost due to insufficient buffers,
- Mean number of jobs in the system = Arrival rate  $\times$  Mean response time.
- (4) Similarly,
- Mean number of jobs in the queue = Arrival rate  $\times$  Mean waiting time.
  - This is **Little's law** as mentioned later.
- (5) Time in System versus Time in Queue  $r = w + s$ , where  $r$ ,  $w$ , and  $s$  are random variables.
- $E[r] = E[w] + E[s]$ .
- (6) If the service rate is independent of the number of jobs in the queue,  $Cov(w, s) = 0$

$$Var[r] = Var[w] + Var[s].$$

- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues
- 6 M/M/1**
  - Exercise
- 7 M/M/n
  - Exercise



## Definition

- Interarrival times, service times are **exponentially distributed**.
- **One** server.
- **No limitation** on buffer and population.
- **FCFS** service discipline.

## Definition

- Interarrival times, service times are **exponentially distributed**.
  - **One** server.
  - **No limitation** on buffer and population.
  - **FCFS** service discipline.
- 
- It is the **most commonly used** type of queue.
  - Need to know only arrival rate  $\lambda$  and service rate  $\mu$ .

- A birth-death process with

$$\lambda_n = \lambda, \quad n = 0, 1, 2, \dots, \infty$$

$$\mu_n = \mu, \quad n = 1, 2, \dots, \infty$$

- Probability of  $n$  jobs in the system

$$p_n = \left(\frac{\lambda}{\mu}\right)^n p_0, \quad n = 1, 2, \dots, \infty$$

- Traffic intensity  $\rho = \lambda/\mu$ .

$$p_n = \rho^n p_0$$

We have

$$\begin{aligned}\sum_{i=0}^{\infty} p_i &= 1 \\ p_0(1 + \rho + \rho^2 + \dots) &= 1 \\ p_0 &= \frac{1}{1 + \rho + \rho^2 + \dots} = 1 - \rho\end{aligned}$$

$$\Rightarrow p_n = \rho^n (1 - \rho)$$

- Utilization of the server = Probability of having one or more jobs in the system

$$U = 1 - p_0 = \rho$$

- Mean number of jobs in the system

$$E[n] = \sum_{n=1}^{\infty} np_n = \sum_{n=1}^{\infty} n\rho^n(1 - \rho) = \frac{\rho}{1 - \rho}$$

- Variance of the number of jobs in the system

$$\begin{aligned}\text{Var}[n] &= E[n^2] - (E[n])^2 \\ &= \sum_{n=1}^{\infty} n^2(1 - \rho)\rho^n - (E[n])^2 \\ &= \frac{\rho}{(1 - \rho)^2}\end{aligned}$$

- Probability of  $n$  or more jobs in the system

$$P(\geq n \text{ jobs in the system}) = \sum_{j=n}^{\infty} p_j = \rho^n$$

- By Little's Law

$$E[n] = \lambda E[r]$$

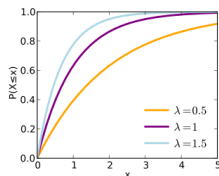
Then,

$$E[r] = \frac{E[n]}{\lambda} = \frac{1/\mu}{1 - \rho}$$

- Cumulative distribution function of the response time

$$F(r) = 1 - e^{-r\mu(1-\rho)}.$$

⇒ The response time is exponentially distributed.



- $q$ -percentile of the response time (i.e.,  $F(r) = q/100$ )

$$1 - e^{-r_q\mu(1-\rho)} = \frac{q}{100}.$$

Hence,

$$r_q = \frac{1}{\mu(1-\rho) \ln \left( \frac{100}{100-q} \right)}.$$

- Cumulative distribution function of the waiting time

$$F(w) = 1 - \rho e^{-w\mu(1-\rho)}.$$

- Mean waiting time

$$E[w] = \rho \frac{1/\mu}{1-\rho} = E[r] - \frac{1}{\mu}$$

- This is a truncated exponential distribution. Its  $q$ -percentile is given by

$$w_q = \frac{1}{\mu(1-\rho) \ln \left( \frac{100\rho}{100-q} \right)}.$$

- The above formula applies only if  $q$  is greater than  $100(1-\rho)$ . All lower percentiles are zero.

$$w_q = \max \left\{ 0, \frac{E[w]}{\rho} \ln \left( \frac{100\rho}{100-q} \right) \right\}.$$

- Mean number of jobs in the queue:

$$E[n_q] = \sum_{n=1}^{\infty} (n-1)p_n = \sum_{n=1}^{\infty} (n-1)(1-\rho)\rho^n = \frac{\rho^2}{1-\rho}.$$
$$E[n_q] = E[n] - \rho$$

### Note

All results for *M/M/1* queues including some for the busy period are summarized in Box 31.1 in the book of R. Jain.



## Problem

On a network gateway, measurements show that the packets arrive at a mean rate of 125 packets per second (pps) and the gateway takes about two milliseconds to forward them.

- Using an  $M/M/1$  model, analyze the gateway.
- What is the probability of buffer overflow if the gateway had only 13 buffers?
- How many buffers do we need to keep packet loss below one packet per million?

## M/M/1: exercise(1)

- Arrival rate  $\lambda = 125$  pps.
- Service rate  $\mu = 1/.002 = 500$  pps.
- Gateway Utilization  $\rho = \lambda/\mu = 0.25$ .
- Probability of  $n$  packets in the gateway:  
 $(1 - \rho)\rho^n = 0.75 \times 0.25\rho^n$ .
- Mean Number of packets in the gateway:  
 $\rho/(1 - \rho) = 0.25/0.75$ .
- Mean time spent in the gateway:  
 $(1/\mu)/(1 - \rho) = (1/500)/(1 - 0.25) = 2.66$  milliseconds.
- Probability of buffer overflow:  
 $P(\text{more than 13 packets in gateway})$   
$$= \rho^{13} = 0.25^{13} = 14.9 \times 10^{-9}$$
$$\approx 15 \text{ packets per billion packets.}$$

## $M/M/1$ : exercise(2)

- An airport runway for arrivals only
- Arriving aircraft join a single queue for the runway
- Exponentially distributed service time with a rate  $\mu = 27$  arrivals/hour.
- Poisson arrivals with a rate  $\lambda = 20$  arrivals/hour.
- Compute
  - Time in the airport runway system
  - Number of aircrafts in the runway system
  - Waiting time for the runway
  - Number of aircrafts waiting for the runway

# M/M/1: exercise(2)

- An airport runway for arrivals only
- Arriving aircraft join a single queue for the runway
- Exponentially distributed service time with a rate  $\mu = 27$  arrivals/hour.
- Poisson arrivals with a rate  $\lambda = 20$  arrivals/hour.
- Compute
  - Time in the airport runway system
$$E[r] = \frac{1}{\mu - \lambda} = \frac{1}{27 - 20} = \frac{1}{7} \text{ hour.}$$
  - Number of aircrafts in the runway system
$$E[n] = \lambda E[r] = \frac{20}{27 - 20} = 2.9 \text{ aircrafts.}$$
  - Waiting time for the runway
$$E[w] = E[r] - 1/\mu = \frac{1}{7} - \frac{1}{27} = 6.4 \text{ min}$$
  - Number of aircrafts waiting for the runway
$$E[n_q] = \dots$$

- 1 Basic structures and components
  - Kendall Notation
- 2 Performance metrics
- 3 Little's Law
- 4 Birth-death processes
- 5 Rules for All Queues
- 6 M/M/1
  - Exercise
- 7 M/M/n**
  - Exercise**

## Definition

- Interarrival times, service times are **exponentially distributed**.
- **$n$  servers.**
- **No limitation** on buffer and population.
- **FCFS** service discipline.

- A birth-death process with

$$\begin{aligned}\lambda_n &= \lambda \\ \mu_n &= \begin{cases} n\mu & n = 1, 2, \dots, m-1 \\ m\mu & n = m, m+1, \dots, +\infty \end{cases}\end{aligned}$$

- Traffic intensity  $\rho = \lambda/(m\mu)$
- Probability of zero job in the system

$$p_0 = \left[ 1 + \frac{(m\rho)^m}{m!(1-\rho)} + \sum_{n=1}^{m-1} \frac{(m\rho)^n}{n!} \right]^{-1}$$

- Probability of  $n$  jobs in the system

$$\mu_n = \begin{cases} \frac{\lambda^n}{n!\mu^n} p_0 & n = 1, 2, \dots, m-1 \\ \frac{\lambda^n}{m!m^{n-m}\mu^n} p_0 & n = m, m+1, \dots, +\infty \end{cases}$$

## Computer center

Students arrive at a computer center in Poisson manner of rate 10 students/hour. Each student spends an average of 20 minutes at a terminal in exponential distribution. The center has 5 terminals. Let analyze the center usage.

- Traffic intensity  $\rho = \lambda/(5\mu) = 0.167/(5 \times 0.05) = 0.67$
- Probability of all terminals being idle is  $p_0 = \dots = 0.0318$
- Probability of all terminals being busy is  $\frac{(m\rho)^m}{m!(1-\rho)} p_0 = 0.33$ .