QUEUING SYSTEM

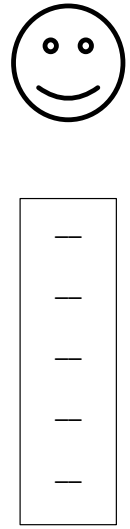
Yetunde Folajimi, PhD

Part 2

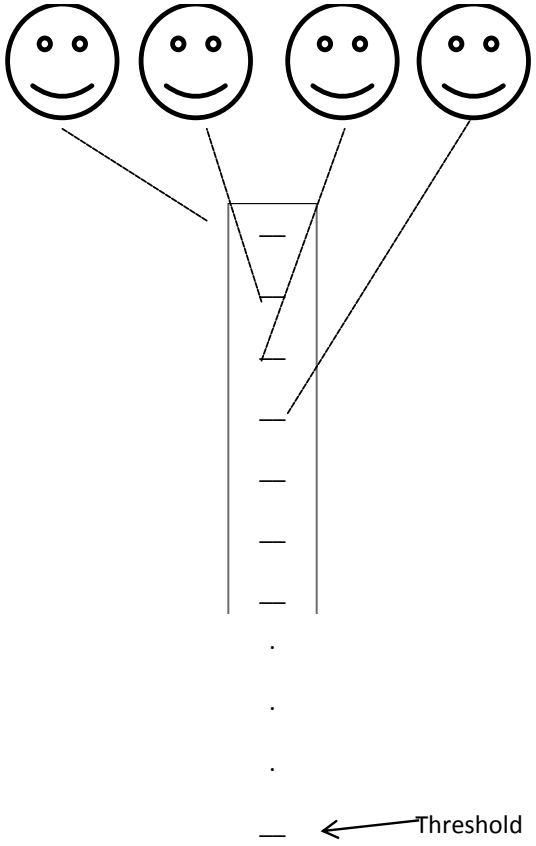
Queuing Models

- Queueing models are constructed so that queue lengths and waiting times can be predicted
- They help us to understand and quantify the effect of variability in the arrival and service processes
- Also called waiting line models.
- Categories:
  - single server queuing model
    - Finite queue length, infinite queue length
  - Multiple server queuing model
    - Finite queue length, infinite queue length

Single server and multiple server with finite/infinite queue length



Single server finite queue length model



Multiple server infinite queue length model

Finite and infinite population models

- Infinite population model
  - No limit to the population size
  - E.g. anybody can come for doctor's service in an hospital.
- Finite population model
- Finite Population model
  - There is restriction on the population size
  - E.g. a factory with thirty machines and a dedicated maintenance team
- infinite population situation are more common than finite population situations

Poisson Distribution and Exponential distribution

- Poisson process describes the distribution of the number of customers arriving in a fixed interval of time length, t. i.e. a process in which events occur continuously and independently at a constant average rate
- **Exponential Distribution** is the probability distribution that describes the time between events in a Poisson process, Given a **Poisson Process**, the following have an **Exponential Distribution**:
  - the time until the first event
  - the time from now until the next occurrence of an event
  - the time interval between two successive events
- The following has a **Poisson Distribution**:
  - the number of events in a given time period

Distribution of Arrival and Service

- Most times, arrivals follow a Poisson/ Markovian distribution
  - Arrival rate = per hour (λ/h)
- Service times are exponential
  - Service time = μ per hour (μ/h)
- Note: For efficiency of queuing system,
  - $\lambda/\mu < 1$, particularly for infinite queue length models.

Kendall's Notation for Classification of Queue Types

- proposed by D. G. Kendall (1953) and exists in several modifications
- The most comprehensive classification uses 6 symbols and are described by:
  - **A/B/s/q/c/p**
- **where:**

A	**The arrival pattern (distribution of intervals between arrivals).**
B	is the service pattern (distribution of service time or duration).
s	Number of servers
q	The queuing discipline (FIFO, LIFO, ...). Omitted for FIFO or if not specified.
c	System capacity or maximum total number of customers which can be accommodated in system. The value is omitted for unlimited queues
p	The population size (number of possible customers). This is omitted for open systems or infinite poplulation

Kendall's Notation for Classification of Queue Types

- The arrival patterns (A) and service patterns (B) can take any of following distribution types:

M	**Poisson (Markovian) process with exponential distribution of intervals or service duration respectively**
D	Degenerate or Deterministic (known) arrivals and constant service duration
E_k	Erlang Distribution of intervals or service duration. (k = shape parameter)
G	General Distribution (arbitrary distribution) GI is a general (any) distribution with independent random values

- Notes: If G is used for **A**, it it sometimes written GI. **c** is usually infinite or a variable, as is **p**. If **c** or **p** are assumed to be infinite for modelling purposes, they can be omitted from the notation (which they frequently are). If **p** is included, **c** must be, to ensure that one is not confused between the two, but an infinity symbol is allowed for **c**

Kendall's Notation: Examples

- **D / M / *n***
  - **Degenerate/deterministic** distribution for the interarrival times of customers, an **exponential** distribution for service times of customers, and *n* servers.
- **E_k / E_l / 1**
  - **Erlang** distribution for the interarrival times of customers (with a shape parameter of **k**), an **exponential** distribution for service times of customers (with a shape parameter of **l**), and a single server.
- **M / M / *m* / K / N**
  - **Exponential** distribution for the interarrival times of customers and the service times of customers, ***m*** servers, a maximum of **K** customers in the queueing system at once, and **N** potential customers in the calling population.
- **D/M/1**
  - **Degenerate/Deterministic** (known) input, **one exponential server**, **one unlimited FIFO or unspecified queue, unlimited** customer population.
- **M/G/3/20**
  - **Poisson** input, **three** servers with **any distribution**, **maximum number of customers (capacity) 20**, unlimited customer population.
- **D/M/1/LIFO/10/50**
  - **Degenerate/Determinist**ic arrivals, **one exponential** server, queue is a stack of the maximum size 10, total number of customers **50**.

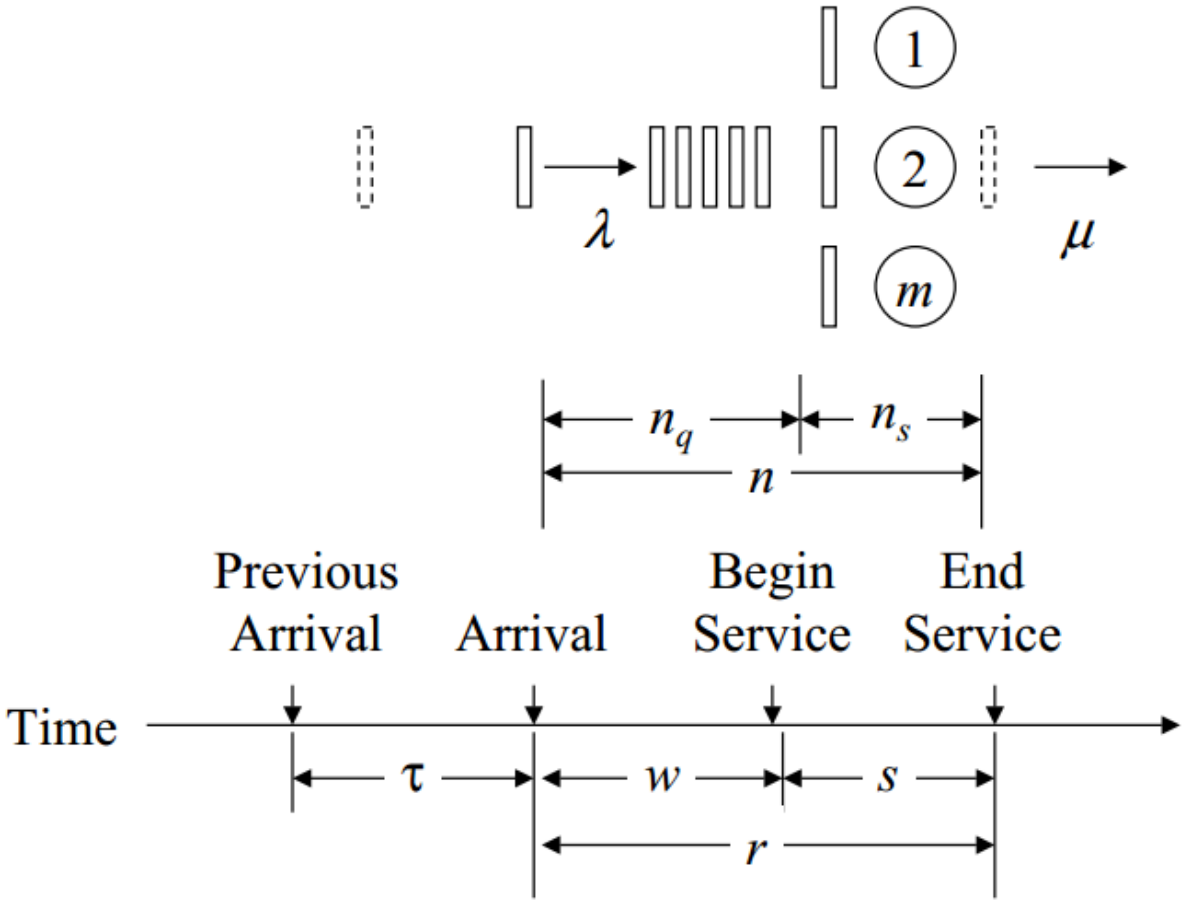
Example: M/M/3/FIFO/20/1500

- 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 first-come-first-served.
- Defaults:
  - Infinite buffer capacity Infinite population size F #CF S service discipline.
  - G/G/1 = G/G/1/oo/oo/
  - FCFS service discipline
- G/G/1 = G/G/l/oo/oo}FCFS

Group Arrivals/Service

- Bulk arrivals/service
- $M^{[x]}$: x represents the group size
- $G^{[x]}$: a bulk arrival or service process with general inter-group times.
- Examples: $M^{[x]}/M/1$: 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.

Key variables



Key variables (contd)

- τ = Inter-arrival time = time between two successive arrivals.
- λ = Mean 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.
- μ = Mean service rate per server = $1/E[s]$
- Total service rate for m servers is $m\mu$
- n = Number of jobs in the system.
  This is also called **queue length**.
- Note: Queue length includes jobs currently receiving service as well as those waiting in the queue.

Key variables (contd)

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

Rules for all queues

Rules: The following apply to *G/G/m* queues

1. Stability Condition:

$$\lambda < m\mu$$

Finite-population and the finite-buffer systems are always stable.

2. Number in System versus Number in Queue:

$n = n_q + n_s$

Notice that 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,

$Cov(n_q,n_s) = 0$

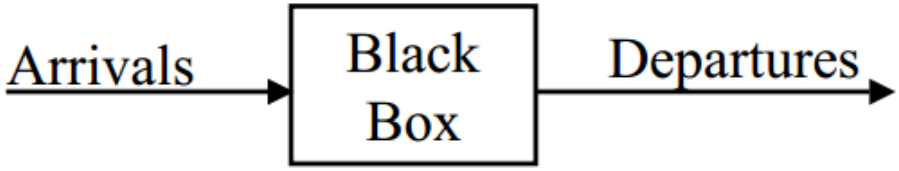
$$\mathrm{Var}[n] = \mathrm{Var}[n_q] + \mathrm{Var}[n_s]$$

Rules for all queues (contd.)

- 3. Number versus Time:
  - If jobs are not lost due to insufficient buffers,
  - Mean number of jobs in the system = Arrival rate × Mean response time
- 4. Similarly,
  - Mean number of jobs in the queue = Arrival rate × Mean waiting time
  - This is known as **Little's law.**
- 5. Time in System versus Time in Queue
  - r = w + s
    - 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]

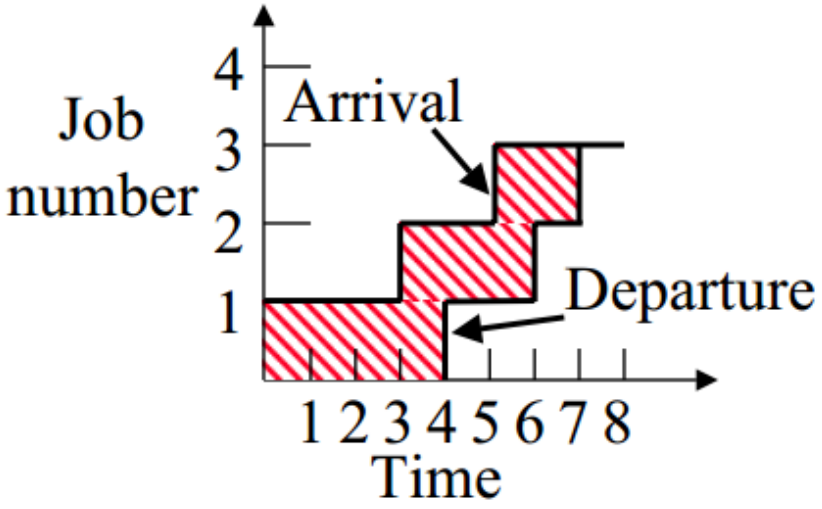
Little's Law

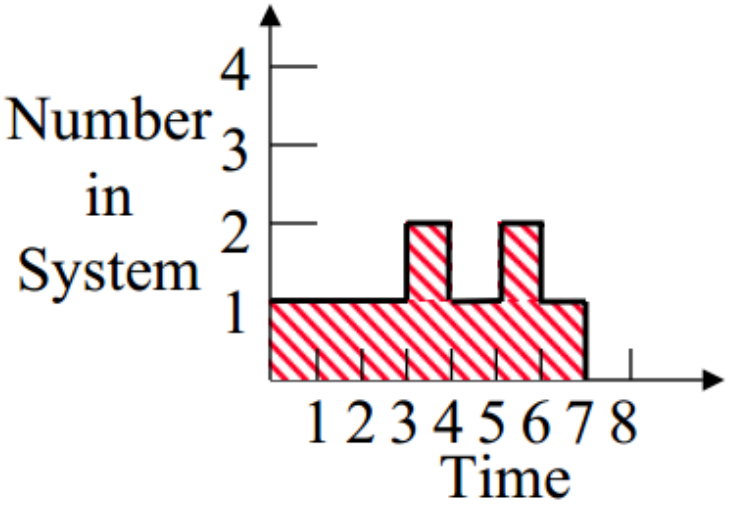
- Mean number in the system
  = Arrival rate × Mean response time
- This relationship applies to all systems or parts of systems in which the number of jobs entering the system is equal to those completing service.
- Named after Little (1961)
- Based on a black-box view of the system:



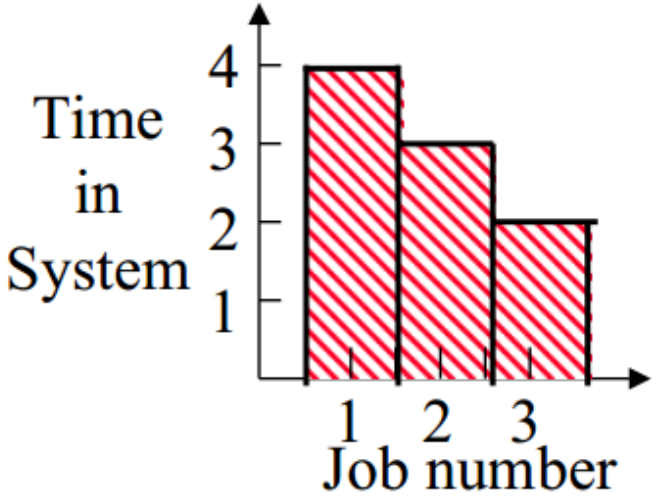
- In systems in which some jobs are lost due to finite buffers, the law can be applied to the part of the system consisting of the waiting and serving positions

Proof of Little's Law

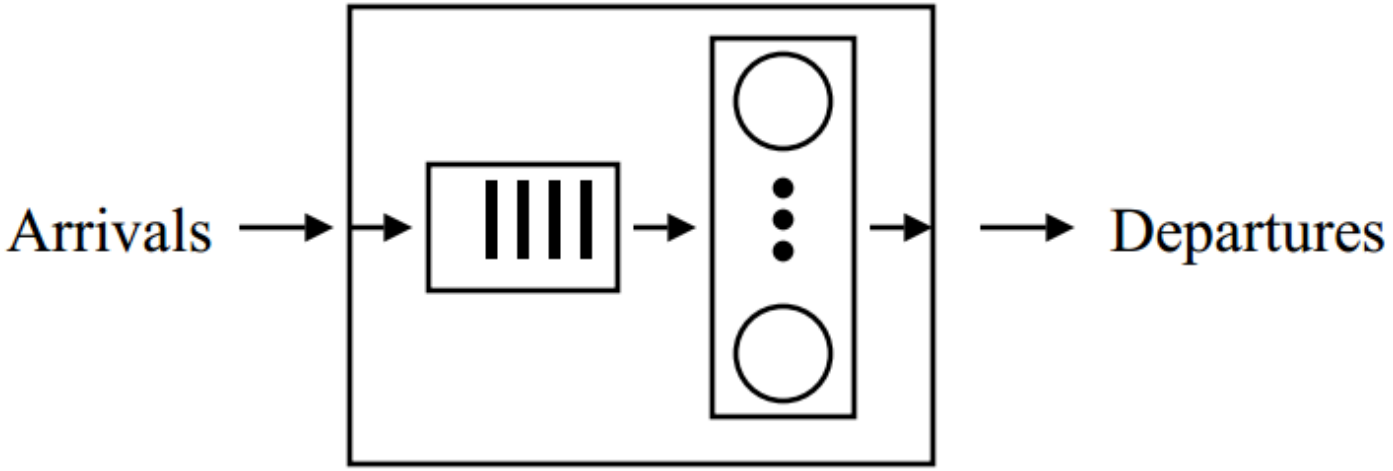




- If T is large, arrivals = departures = N
- Arrival rate = Total arrivals/Total time= N/T
- Hatched areas = total time spent inside the system by all jobs = J
- Mean time in the system= J/N
- Mean Number in the system
  $= J/T = \frac{N}{T} \times \frac{J}{N}$
  = Arrival rate× Mean time in the system



Application of Little's Law



- Applying to just the waiting facility of a service center
- Mean number in the queue = Arrival rate × Mean waiting time
- Similarly, for those currently receiving the service, we have:
- Mean number in service = Arrival rate × Mean service time

Example

- 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 mean number of requests at the disk server?
- Using Little's law:

  Mean number in the disk server = Arrival rate × Response time

  = 100 (requests/second) ×(0.1 seconds)

  = 10 requests

Stochastic Process

- Process is a function of time
- Stochastic Process refers to Random variables, which are functions of time
- Example 1:
  - n(t) = number of jobs at the CPU of a computer system
  - Take several identical systems and observe n(t)
  - The number n(t) is a random variable.
  - Can find the probability distribution functions for n(t) at each possible value of t.
- Example 2:
  - w(t) = waiting time in a queue

Types of Stochastic Process

- Discrete or Continuous State Processes
- Markov Processes
- Birth-death Processes
- Poisson Processes

Discrete or Continuous State Processes

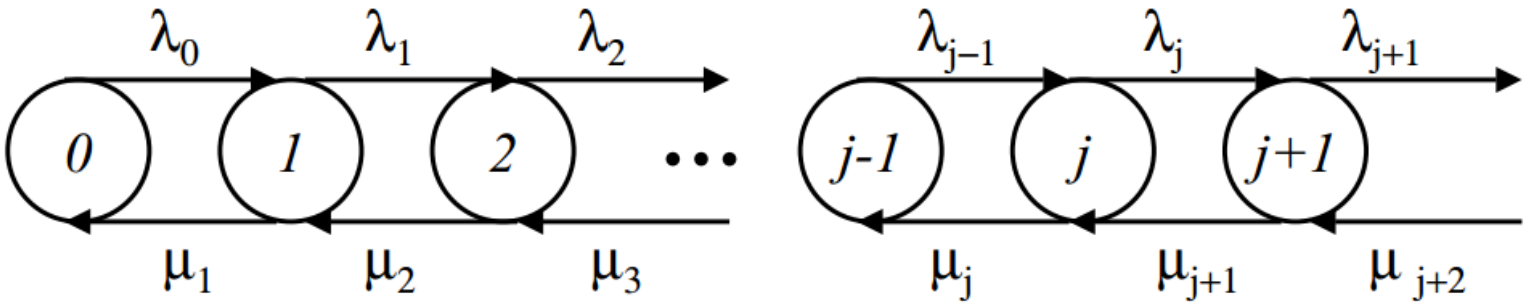
- Discrete = Finite or Countable
- Number of jobs in a system n(t) = 0, 1, 2, ...
- n(t) is a discrete state process
- The waiting time w(t) is a continuous state process.
- **Stochastic Chain:** discrete state stochastic process

Markov Process

- Future states are independent of the past and depend only on the present.
- Named after A. A. Markov who defined and analyzed them in 1907.
- **Markov** Chain: discrete state Markov process
- **Markov** ⇒ It is not necessary to know how long the process
- has been in the current state ⇒ State time has a memoryless (exponential) distribution
- M/M/m queues can be modeled using Markov processes.
- The time spent by a job in such a queue is a Markov process
- the number of jobs in the queue is a Markov chain

Birth-Death Process

- The discrete space Markov processes in which the transitions are restricted to neighboring states
- Process in state n can change only to state n+1 or n-1.
- Example: the number of jobs in a queue with a single server and individual arrivals (not bulk arrivals)



Poisson Process

- Inter-arrival time s = IID and exponential
- ⇒ number of arrivals n over a given interval (t, t+x) has a Poisson distribution
- ⇒ arrival = Poisson process or Poisson stream

Relationship amongst stochastic process