

EEC 686/785
Modeling & Performance Evaluation of
Computer Systems

Lecture 25

Wenbing Zhao

Department of Electrical and Computer Engineering

Cleveland State University

wenbing@ieee.org

(based on Dr. Raj Jain's lecture notes)



Outline

- Homework #5 and #6
- Review for the final exam
- **Final exam: Dec 12, Monday 4:00-6:00pm**
- **Final deadline for project report: Dec 14, midnight**
- Records of your course work will be posted on the course Web site. If there is any mistake, please contact me as soon as possible

11 December 2005

EEEC686/785

Wenbing Zhao



Basic Components of a Queue



Queueing Notation

- Kendall notation: 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

11 December 2005

EEEC686/785

Wenbing Zhao

11 December 2005

EEEC686/785

Wenbing Zhao

Common Distributions

- M : Exponential
- E_k : Erlang with parameter k
- H_k : hyperexponential with parameter k
- D : deterministic \Rightarrow constant
- G : general \Rightarrow all
- $M^{[x]}$: bulk Poisson arrival or bulk service process with exponential service times
- $G^{[x]}$: a bulk arrival or service process with general intergroup times

Key Variables

Rules for All Queues (G/G/m)

- **Stability condition:** $\lambda < m\mu$ (m : number of servers)
 - Finite-population and the finite-buffer systems are always stable
- **Number in system versus number in queue:**

$$n = n_q + n_s$$
- **Number versus time (Little's law):** if jobs are not lost due to insufficient buffers,

$$n = \lambda \times r, \text{ similarly } n_q = \lambda \times w$$
- **Time in system versus time in queue:**

$$r = w + s$$

Stochastic Processes

- **Process:** function of time
- **Stochastic process:** 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 Processes

- Discrete or continuous state processes
- Markov processes
- Birth-death processes
- Poisson processes

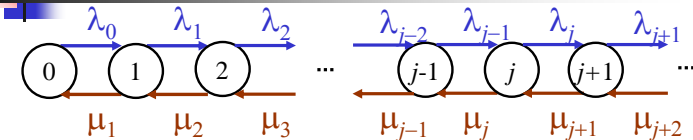
Discrete/Continuous State Processes

- Discrete = finite or countable number of values the state can take
- Number of jobs in a system $n(t) = 0, 1, 2, \dots$
 $n(t)$ is a discrete state process
- The waiting time $w(t)$ is a continuous state process
- **Stochastic chain:** discrete state stochastic process

Markov Processes

- **Markov Processes:** Future states are independent of the past and depend only on the present
- **Markov chain:** discrete state Markov process
- Markov \Rightarrow It is not necessary to know how long the process has been in the current state \Rightarrow 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 and the number of jobs in the queue is a Markov chain

Birth-Death Processes



- **Birth-Death Processes:** 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 Processes

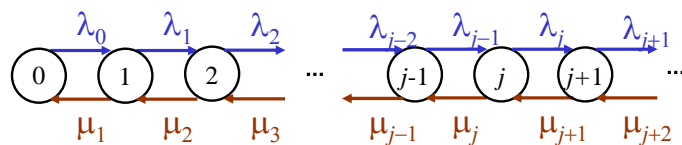
- Interarrival times = IID + exponential
 - Number of arrivals n over a given interval $(t, t+x)$ has a Poisson distribution
 - The arrival process is referred to as a **Poisson process** or a **Poisson stream**

Relationship Among Stochastic Processes

- **Markov process** broader than **birth-death process** broader than **Poisson process**
- The Poisson process can be modeled as a pure birth process with constant birth rate
- All birth-death processes are Markov processes with the restriction that the transitions are restricted to neighboring states

Birth-Death Processes

- State-transition diagram



- In state n :
 - New arrivals take place at a rate λ_n
 - The service rate is μ_n
- Both the interarrival times and service times are assumed exponentially distributed

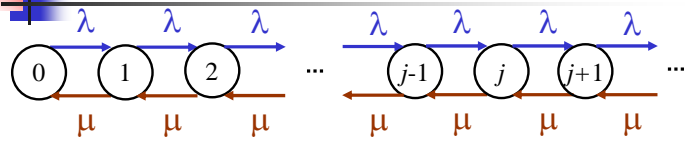
Theorem: State Probability

- The steady-state probability p_n of a birth-death process being in state n (i.e., there are n jobs in the system) is given by:

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

- Here, p_0 is the probability of being in the zero state (no job in system, i.e., system is idle)

Results for M/M/1 Queues



- Birth-death processes 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

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

$\rho = \lambda/\mu$: **traffic intensity**

Results for M/M/1 Queues

- Utilization of the server: $U = 1 - p_0 = \rho$
- Mean number of jobs in the system: $E[n] = \frac{\rho}{1 - \rho}$
- Variance of the number of jobs: $Var[n] = \frac{\rho}{(1 - \rho)^2}$
- Mean response time: $E[r] = \frac{1/\mu}{1 - \rho}$
- Cumulative distribution function of the response time: $F(r) = 1 - e^{-r\mu(1-\rho)}$

Results for M/M/1 Queues

- Cumulative distribution function of the waiting time

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

- 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-ρ). All lower percentiles are 0

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

Results for M/M/1 Queues

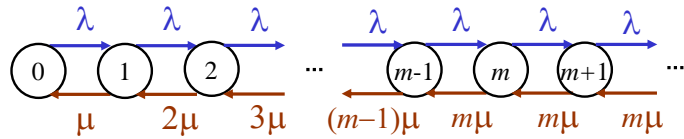
- Mean number of jobs in the queue

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

- Busy period**: the time interval between two successive idle intervals

M/M/m Queue

- The state of the system is represented by the number of jobs n in the system



Analysis of M/M/m Queue

- Number of jobs in the system is a birth-death process:

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

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

- Probability of n jobs in the system:

$$P_n = \begin{cases} \frac{(m\rho)^n}{n!} p_0 & n = 1, 2, \dots, m-1 \\ \frac{\rho^n m^m}{m!} p_0 & n = m, m+1, \dots, \infty \end{cases}$$

Analysis of M/M/m Queue

- Probability of zero jobs 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 that an arriving job has to wait in the queue:

$$\mathcal{G} = P(\geq m \text{ jobs}) = p_0 \frac{(m\rho)^m}{m!(1-\rho)}$$

- This is known as **Erlang's C formula**. Notice that for $m = 1$, $\mathcal{G} = \rho$

Analysis of M/M/m Queue

- Mean number of jobs in the queue $E[n_q]$:

$$E[n_q] = \sum_{n=m+1}^{\infty} (n-m) p_n = \frac{\rho \mathcal{G}}{1-\rho}$$

- Expected number of jobs in service:

$$E[n_s] = \sum_{n=1}^{m-1} n p_n + \sum_{n=m}^{\infty} m p_n = m\rho$$

- Utilization of each server:

$$U = \frac{\text{Busy time per server}}{\text{Total time}} = \frac{(\lambda T / \mu) / m}{T} = \frac{\lambda}{m\mu}$$

Analysis of M/M/m Queue

- Mean response time:

$$E[r] = \frac{1}{\mu} \left(1 + \frac{\rho}{m(1-\rho)} \right)$$

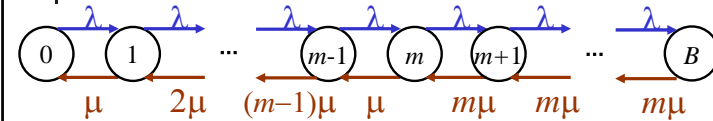
- Mean waiting time:

$$E[w] = \frac{\rho}{m(1-\rho)}$$

- q -percentile of waiting time:

$$w_q = \max \left\{ 0, \frac{1}{m\mu(1-\rho)} \ln \left(\frac{100\rho}{100-q} \right) \right\}$$

Analysis of M/M/m/B Queue



- Birth-death process:

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

$$\mu_n = \begin{cases} n\mu & n = 1, 2, \dots, m-1 \\ m\mu & n = m, m+1, \dots, B \end{cases}$$

Analysis of M/M/m/B Queue

- Probability of n jobs in the system:

$$p_n = \begin{cases} \frac{(m\rho)^n}{n!} p_0 & n = 1, 2, \dots, m-1 \\ \frac{\rho^n m^m}{m!} p_0 & n = m, m+1, \dots, B \end{cases}$$

- Probability of zero jobs in the system:

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

Analysis of M/M/m/B Queue

- Mean number of jobs in the system: $E[n] = \sum_{n=1}^B np_n$
- Mean number of jobs in the queue: $E[n_q] = \sum_{n=m+1}^B (n-m)p_n$
- All arrivals occurring when the system is in the state $n=B$ are lost. Rate of the jobs actually entering the system, called **effective arrival rate**, is:

$$\lambda' = \sum_{n=0}^{B-1} \lambda p_n = \lambda \sum_{n=0}^{B-1} p_n = \lambda(1-p_B)$$

- The difference $\lambda - \lambda' = \lambda p_B$ represents the **loss rate**

Analysis of M/M/m/B Queue

- Mean response time (using Little's law):

$$E[r] = \frac{E[n]}{\lambda'} = \frac{E[n]}{\lambda(1-p_B)}$$

- Utilization of each server:

$$U = \frac{\text{Busy time per server}}{\text{Total time}} = \frac{(\lambda'T/\mu)/m}{T} = \frac{\lambda'}{m\mu} = \rho(1-p_B)$$

- Probability of the system in full capacity = p_B

Queueing Networks

- Network = model in which jobs departing from one queue arrive at another queue (or possibly the same queue)
- Open queueing networks
 - External arrivals departures
- Closed queueing networks
 - No external arrivals or departures
- Mixed queueing networks
 - open for some workloads and closed for others

Product-Form Network

- Any queueing network in which:

$$P(n_1, n_2, n_3, \dots, n_M) = \frac{1}{G(N)} \prod_{i=1}^M f_i(n_i)$$

When $f_i(n_i)$ is some function of the number of jobs at the i th facility, $G(N)$ is a normalizing constant and is a function of the total number of jobs in the system

Non-Markovian Product Form Networks

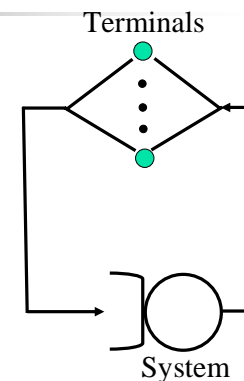
- **1. Job flow balance:** for each class, the number of arrivals to a device must equal the number of departures from the device
- **2. One step behavior:** no bulk arrivals and no simultaneous job-moves
- **3. Device homogeneity:** a device's service rate for a particular class does not depend on the state of the system in any way except for the total device queue length and the designated class's queue length

Non-Markovian Product Form Networks

- The assumption 3 implies the following:
 - **Single resource possession**
 - **No blocking**
 - **Independent job behavior**
 - **Local information**
 - **Fair service**
 - **Routing homogeneity**

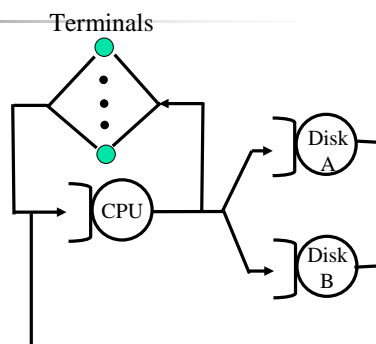
Machine Repairman Model

- One of the earliest model of computer systems
- Originally developed for modeling machine repair shops
 - A number of working machines
 - A repair facility with one or more servers (repairmen)
 - Whenever a machine breaks down, it is put in the queue for repair and serviced as soon as a repairman is available



Central Server Model

- Introduced by Buzen (1973)
- The CPU is the “central server” that schedules visits to other devices
- After service at the I/O devices the jobs return to the CPU



Types of Service Centers

- **Fixed-capacity service centers:** service time does not depend upon the number of jobs in the device
- **Delay centers** or IS (**infinite server**): no queueing. Jobs spend the same amount of time in the device regardless of the number of jobs in it
- **Load-dependent service centers:** service rates may depend upon the load or the number of jobs in the device

Operational Laws

- Relationships that do not require any assumptions about the distribution of service times or interarrival times
- Operationally testable assumptions** => assumptions that can be verified by measurements
 - For example, whether number of arrivals is equal to the number of completions (i.e., job flow balance) is operationally testable
 - A set of observed service times is not operationally testable

Operational Laws

- Operational quantities:** quantities that can be directly measured during a finite observation period
 - T = observation interval
 - A_i = number of arrivals
 - C_i = number of completions
 - B_i = busy time
- Operational quantities are *variables* that can change from one observation period to the next

Operational Laws

- Relationships that hold in every observation period are called **operational laws**

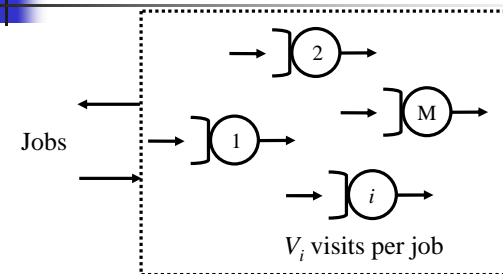
Operational quantities

$$\left\{ \begin{array}{l} \text{Arrival Rate } \lambda_i = \frac{\text{Number of arrivals}}{\text{Time}} = \frac{A_i}{T} \\ \text{Throughput } X_i = \frac{\text{Number of completions}}{\text{Time}} = \frac{C_i}{T} \\ \text{Utilization } U_i = \frac{\text{Busy Time}}{\text{Total Time}} = \frac{B_i}{T} \\ \text{Mean service time } S_i = \frac{\text{Total time served}}{\text{Number served}} = \frac{B_i}{C_i} \end{array} \right.$$

Utilization Law

$$\left\{ \begin{array}{l} U_i = \frac{B_i}{T} = \frac{C_i}{T} \times \frac{B_i}{C_i} \quad \text{or} \quad U_i = X_i S_i \end{array} \right.$$

Visit Ratio



- Each job makes V_i requests for i th device in the system:

$$C_i = C_0 V_i \quad \text{or} \quad V_i = \frac{C_i}{C_0}$$

V_i is called **visit ratio**

Forced Flow Law

- Forced Flow Law relates the system throughput to individual device throughputs.
- If observation period T is such that $A_i = C_i$
=> Device satisfies the assumption of **job flow balance**
- Forced Flow Law
 - System throughput:

$$\text{System throughput } X = \frac{\text{Jobs completed}}{\text{Total time}} = \frac{C_0}{T}$$
 - Throughput of i th device:

$$\text{Device throughput } X_i = \frac{C_i}{T} = \frac{C_i}{C_0} \times \frac{C_0}{T} \quad \text{or} \quad X_i = XV_i$$

Forced Flow Law

- Combining the forced flow law and the utilization law, we get:
 - Utilization of i th device $U_i = X_i S_i = XV_i S_i$
or $U_i = XD_i$
 - Here $D_i = V_i S_i$ is the total service demand on the device for all visits of a job
- The device with the highest D_i has the highest utilization and is the **bottleneck device**

Transition Probabilities

- p_{ij} = probability a job moving to j th queue after service completion at i th queue
- Visit ratios and transition probabilities are equivalent in the sense that given one we can always find the other
- In a system with job flow balance, the number of completions at j th queue:

$$C_j = \sum_{i=0}^M C_i p_{ij}$$
 - Subscript 0 is used to denote visits to the outside link
 - p_{i0} = probability of a job exiting from the system after completion of service at i th device

Visit Ratio Equations

- Dividing both sides by C_0 we get:

$$V_j = \sum_{i=0}^M V_i p_{ij}$$
- Since each visit to the outside link is defined as the completion of the job, we have:

$$V_0 = 1$$
- The above two equations are called **visit ratio equations**

Visit Ratio Equations for Central Server Models

45

- In central server models, after completion of service at every queue, the jobs always move back to the CPU queue:

$$p_{i1} = 1 \quad \forall i \neq 1$$

$$p_{ij} = 0 \quad \forall i, j \neq 1$$

- $1 = V_1 p_{10}$

$$V_1 = 1 + V_2 + V_3 + \dots + V_M$$

$$V_j = V_1 p_{1j} = \frac{p_{1j}}{p_{10}} \quad j = 2, 3, \dots, M$$

- Thus, we can find the visit ratios by dividing the probability p_{1j} of moving to j th queue from CPU by the exit probability p_{10}

11 December 2005

EEEC686/785

Wenbing Zhao

Little's Law

46

- Mean number in the device = arrival rate \times mean time in the device

$$Q_i = \lambda_i R_i$$

- If the job flow is balanced, the arrival rate is equal to the throughput and we can write:

$$Q_i = X_i R_i$$

11 December 2005

EEEC686/785

Wenbing Zhao

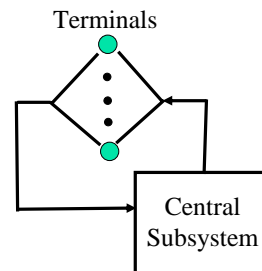
General Response Time Law

47

- There is one terminal per user and the rest of the system is shared by all users
- General response time law:**

$$R = \sum_{i=1}^M R_i V_i$$

- > This law holds even if the job flow is not balanced



11 December 2005

EEEC686/785

Wenbing Zhao

Interactive Response Time Law

48

- If Z = think-time, R = response time, the total cycle time of requests is $R + Z$
 - Each user generates about $T/(R + Z)$ requests in T
 - If there are N users, the system throughput X = total number of requests / total time
- $$= N[T/(R+Z)]/T$$
- $$= N/(R+Z)$$

or

$$R = (N/X) - Z$$

This is the **interactive response time law**

11 December 2005

EEEC686/785

Wenbing Zhao

Bottleneck Analysis

- From forced flow law: $U_i \propto D_i$
the device with the highest total service demand D_i has the highest utilization and is called the bottleneck device
 - Note: delay centers can have utilizations more than one without any stability problems. Therefore, delay centers cannot be a bottleneck device
 - Only queueing centers used in computing D_{max}
- The bottleneck device is the key limiting factor in achieving higher throughput

Asymptotic Bounds

- Throughput and response times of the system are bound as follows:

$$X(N) \leq \min \left\{ \frac{1}{D_{max}}, \frac{N}{D+Z} \right\}$$

$$R(N) \geq \max \{ D, ND_{max} - Z \}$$

Here, $D = \sum D_i$ is the sum of total service demands on all devices except terminals

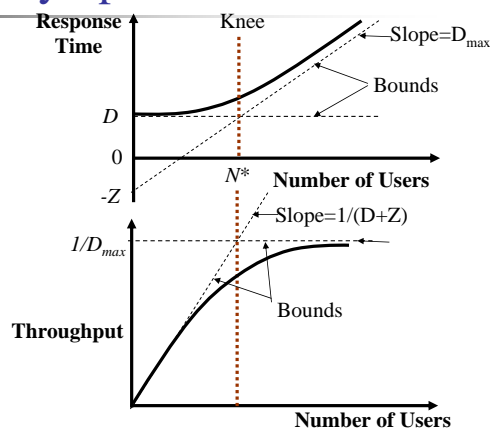
- These are known as **asymptotic bounds**

Asymptotic Bounds

- The asymptotic bounds are based on the following observations:
 - The utilization of any device cannot exceed one. This puts a limit on the maximum obtainable throughput
 - The response time of the system with N users cannot be less than a system with just one user. This puts a limit on the minimum response time
 - The interactive response time formula can be used to convert the bound on throughput to that on response time and vice versa

Typical Asymptotic Bounds

- Both *throughput* and *response time* bounds consist of two straight lines
- The point of intersection of the two lines is called the **knee**
- For both response time and throughput, the **knee** occurs at the same value of number of users



Typical Asymptotic Bounds

- The number of jobs N^* at the knee is given by:

$$D = N^* D_{\max} - Z$$
$$\Rightarrow N^* = \frac{D + Z}{D_{\max}}$$

- If the number of jobs is more than N^* , then we can say with certainty that there is queueing somewhere in the system
- The asymptotic bounds can be easily explained to people who do not have any background in queueing theory or performance analysis