Computer Networks Performance Evaluation

10

Chapter 10
Markov Models

Performance by Design:
Computer Capacity Planning by Example

Daniel A. Menascé, Virgilio A.F. Almeida, Lawrence W. Dowdy

Prentice Hall, 2004

Outline

10.1. Introduction
10.2. Modeling Context
10.3. Motivating Examples
10.4. Model Construction
10.5. Model Solution
10.6. Model Interpretation
10.7. Model Assumptions and Limitations
10.8. Generalized Birth-Death Models
10.9. Beyond the Basics
10.10. Chapter Summary
10.11. Exercises
Bibliography

In part I

- Performance terms have been introduced.
  - response time
  - throughput
  - Availability
  - reliability
  - security
  - Scalability
  - extensibility
- Performance results based on the operational laws have been defined and applied to sample systems.
  - Utilization Law
  - Service Demand Law
  - Forced Flow Law
  - Little's Law
  - Interactive Response Time Law
- Simple performance bounding techniques and basic queuing network models have been established as tools to evaluate and predict system performance .

The Goal of Part II

- Motivate, establish, and explain the basic building blocks and underpinnings of the analytical techniques introduced in Part I.

- To be able to understand what is inside "the black box“

System Modeling

- Prototype model
  - The physical construction of a scaled version of the actual system and executing a typical workload on the prototype
- Simulation model
  - writing of detailed software programs which (hopefully accurately) emulate the performance of the system
- Analytical model
  - capturing the key relationships between the architecture and the workload components in mathematical expressions

System Modeling

- Prototype model
  - high accuracy
  - high costly
- Simulation model
  - less accurate
  - much less costly than prototype models
- Analytical model
  - flexible, inexpensive, and easily changed
  - lack of detail and more difficult to validate

Markov Model

- In this chapter we focus in Markov model.
- Markov model is an analytical model.

Why Markov Model?

- Markov models are often used to explain the current interactions between various system components.

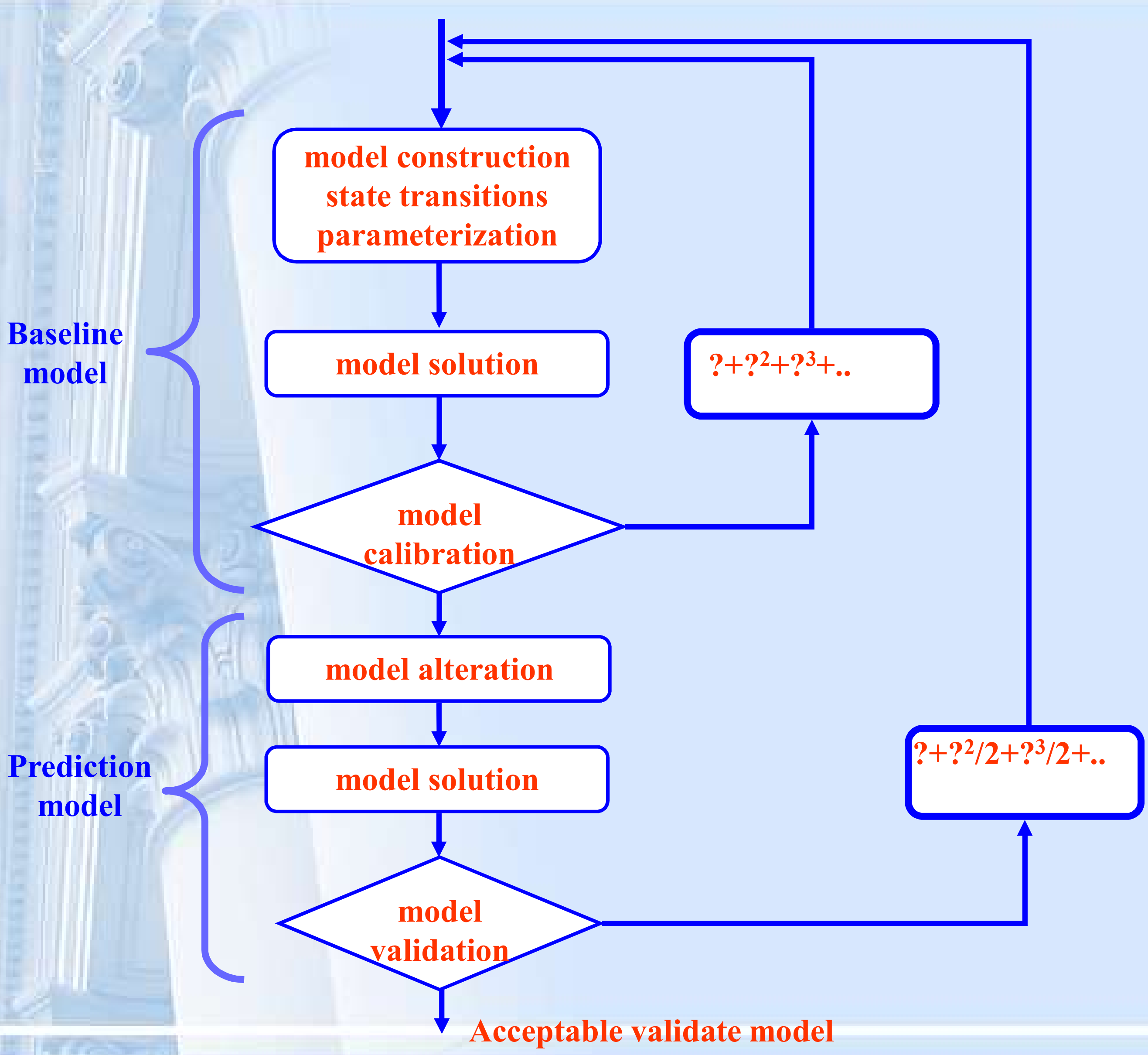
- It can be altered to predict (hopefully, accurately) what would happen if various aspects of the system's hardware or of the system's workload change.

Thus, Markov models can be used for both descriptive and predictive purposes.

To Create Markov Model

- Constructing the state diagram by identifying all possible states that the modeled system may find itself.
- Identifying the state connections (i.e., transitions)
- Parameterzing the model by specifying the length of time spent in each state once it is entered (or, equivalently, the probability of transitioning from one state to another within the next time period).
- Solving the model.
  - Abstracting a set of linear "balance" equations from the state diagram (linear algebra)
  - Solving them for long term "steady state" probabilities of being in each system state.

Modeling Paradigm



Baseline model

model construction
state transitions
parameterization

model solution

model calibration

$?+?^2+?^3+..$

Prediction model

model alteration

model solution

model validation

$?+?^2/2+?^3/2+..$

Acceptable validate model

Model Construction

- Model construction identifying
  - the parameter type and
  - parameter value of model

Model solution

- After construction, the model must be solved.
- prototype models
  - running an experiment on the newly constructed hardware and monitoring its performance
- simulation models
  - running a software package (i.e., the simulator) and recording the emulated performance results
- analytical models
  - solving a set of mathematical equations and interpreting the performance expressions correctly

Model Calibration

- After construction and solution, the model must be calibrated.
- Calibration involves comparing the performance results obtained from the model against those observed in the actual system.
- Often, one must return to a previous step since modeling errors may be discovered during calibration.
- It is not atypical to cycle between the various steps before an acceptable model is found.
- The modeled system is calibrated to match the actual system on a finite set of previously observed (i.e., baseline) performance measures

Accountability

- Accountability is a validity check that the prediction is accurate.
- Accountability is the final step.
- Too often, this step is ignored. It is more normal to make a prediction, collect a consultant's fee, go on to another project (or vacation!), and never return to see if one's prediction actually came true.
- The overall modeling paradigm is improved and the resulting prediction model is truly validated, only by:
  - completing this final check.
  - answering those harder series of questions when the predictions are incorrect.
  - returning to a previous step in the modeling process.

Motivating Example #1: RWTE

- Random Walk Through England
- Consider a lad who is given a year to spend in England.
- The only condition is that every day at 3:00 p.m., the lad must use his mobile phone to call his mother and simply let her know where he is and what he is doing
- After only a couple of months, the mother deduces a definite pattern to her son's behavior.

RWTE: Assumptions

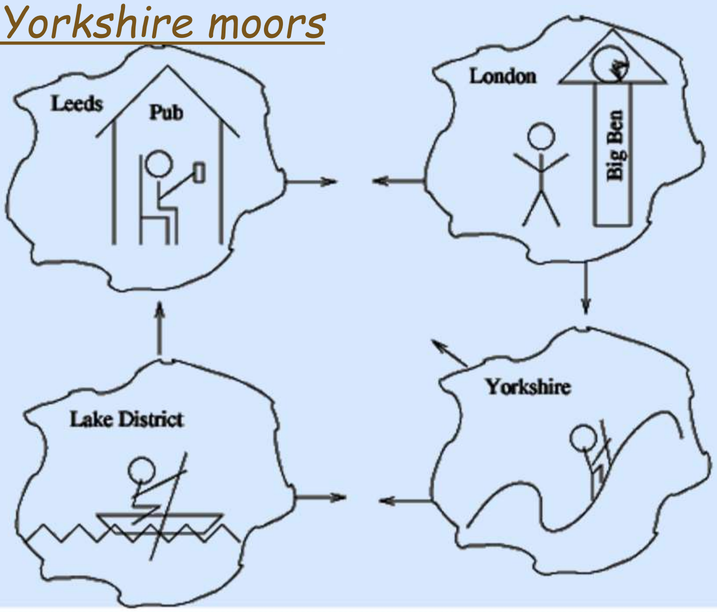
- The lad is always doing something in one of four locations

  drinking in a Leeds pub

  sightseeing in London

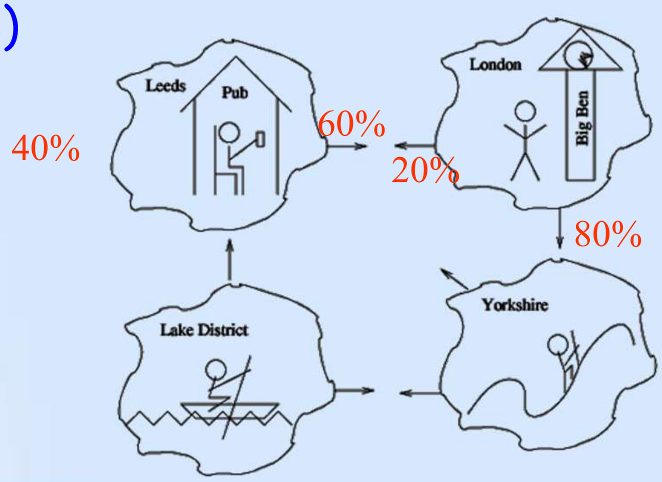
  kayaking in the Lake District

  hiking in the Yorkshire moors



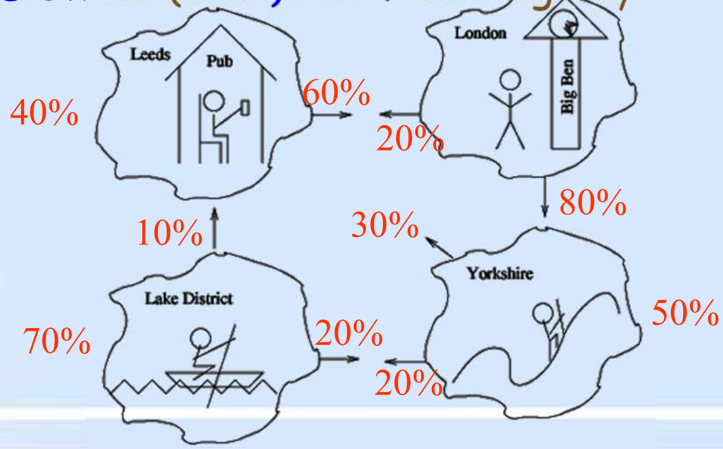
RWTE: Parameters1

- If the lad is in a Leeds pub, he is either likely to go sightseeing in London the following day (60%), or he will still be found in a Leeds pub (40%)
- If the lad is in London, he is likely to be found in a Leeds pub the following day (20%) or will decide to go hiking in the Yorkshire moors (80%)



RWTE: Parameters2

- Once found in the Lake District, there is very good chance that the lad will still be found kayaking the following day (70%), but there is a possibility that he will next be found hiking the moors (20%) or back in a Leeds pub (10%_)
- When found hiking in the moors, there is a good chance that he will still be hiking the following day (50%). However, he sometimes goes to a Leeds pub (30%) and sometimes decides to go kayaking in the Lake District (20%) the following day

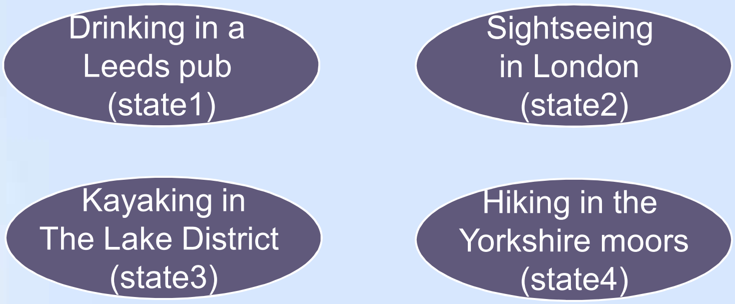


RWTE: Questions

- Question1: What percentage of days is the son actually not drinking in Leeds?
- Question2: Once the son finishes a day of kayaking in the Lake District, how long will it typically be before he returns?
- Question3: How many days each month can the bobbies expect to see the son driving to London after drinking in Leeds?
- Question4: How many visits each month does the son typically visit their shop and typically how long does the son keep their kayak out each visit?

RWTE:
Model Construction-States

1. Enumerate all the possible states:
   - state 1: Drinking in a Leeds pub
   - state 2: Sightseeing in London
   - state 3: Kayaking in the Lake District
   - state 4: Hiking in the Yorkshire moors



Drinking in a Leeds pub (state1)

Sightseeing in London (state2)

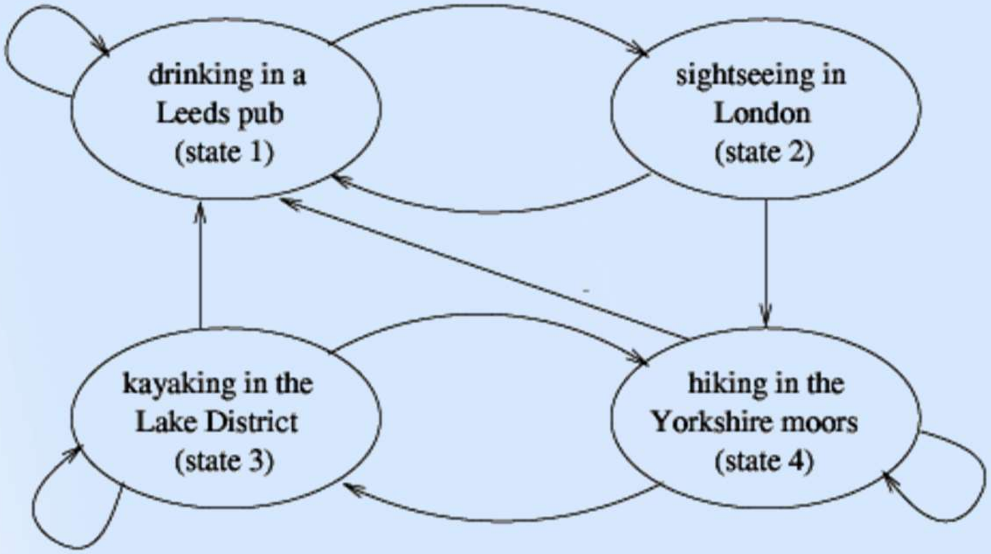
Kayaking in The Lake District (state3)

Hiking in the Yorkshire moors (state4)

RWTE:
Model Construction-Transitions

2. State transitions:
   - If in state 1, the lad may find himself next in state 1 or state 2
   - If in state 2, the lad may find himself next in state 1 or state 4
   - If in state 3, the lad may find himself next in states 1, 3, or 4
   - If in state 4, the lad may find himself next in state 1, 3, or 4

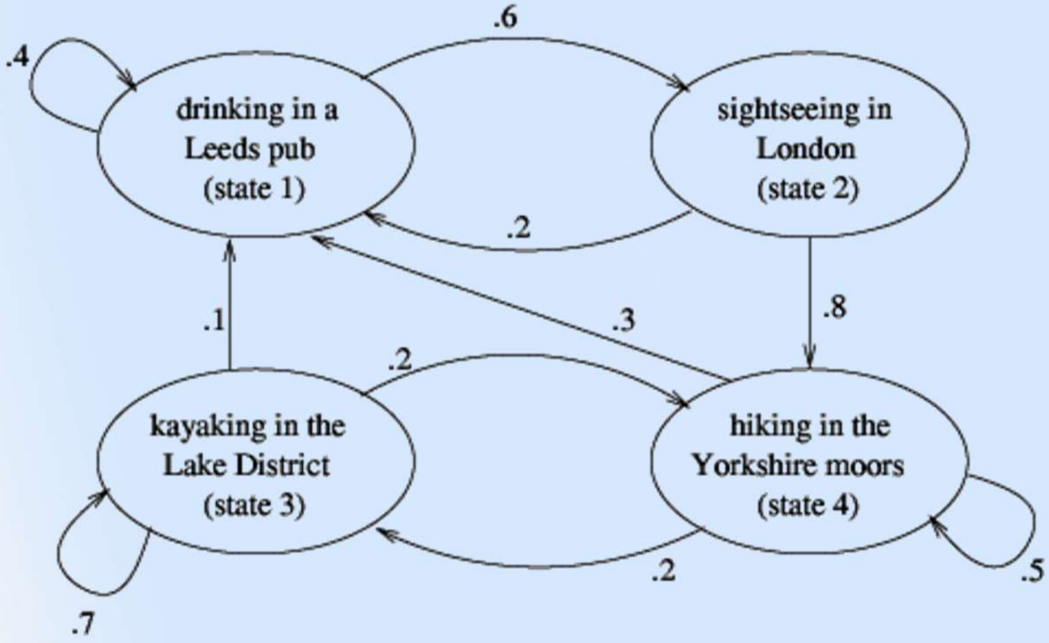
State Transitions



RWTE:
Model Construction-Parameters

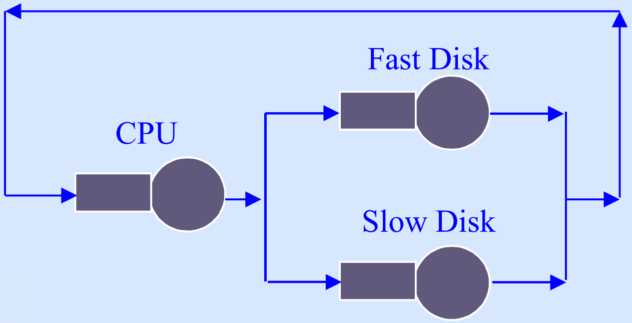
- Model parameterization
  - State 1 goes directly back to state 1 with probability 0.4 and to state 2 with probability 0.6.
  - State 2 goes directly to states 1 or 4 with probabilities 0.2 and 0.8, respectively.
  - State 3 goes directly to states 1, 3, or 4 with probabilities 0.1, 0.7, and 0.2, respectively.
  - State 4 goes directly to states 1, 3, or 4 with probabilities 0.3, 0.2, and 0.5, respectively

Parameters


.4
drinking in a
Leeds pub
(state 1)
.6
sightseeing in
London
(state 2)
.2
.8
.1
.3
.2
kayaking in the
Lake District
(state 3)
hiking in the
Yorkshire moors
(state 4)
.2
.5
.7

Motivating Example #2: Database Server Support

- Consider a computer system with one CPU and two disks used to support a database server.
- Users remotely access the server and typically login, perform some database transactions, and logout.
- Each time 2 request (users) are in the system.



Database Server: Assumptions

- Each transaction alternates between using the CPU and using a disk.
- The two disks are of different speeds, with the faster disk being twice as fast as the slower disk.
- A typical transaction requires a total of 10 sec of CPU time.

D_{cpu} = 10 sec, 6 transactions per minute

Database Server: Assumptions.

- Transactions are equally likely to find the files they require on either disk.
- If a transaction's files are found on the fast disk, it takes an average of 15 seconds to access all the requested files.
- If a transaction's files are found on the slow disk, it takes an average of 30 seconds to access all the requested files.

D_{fdisk}=15 sec, 4 transactions per minute

D_{sdisk}=30sec, 2 transactions per minute

Database Server: Questions

- User's question: What response time can the typical user expect?
- System administrator's question: What is the utilization of each of the system resources?
- Company president's question: If I can capture Company X's clientele, which will likely double the number of users on my system, I will need to also double the number of active users on my system. What new performance levels should I spin in my speech to the newly acquired customers?
- Company pessimist's question: Since I know that the fast disk is about to fail and all the files will need to be moved to the slow disk, what will the new response time be?

Database Server:
Model Construction-States

1. Enumerate all the possible states:
   - State (2,0,0): both users are currently requesting CPU service.
   - State (1,1,0): one user is requesting CPU service and the other is requesting service from the fast disk.
   - State (1,0,1): one user is requesting CPU service and the other is requesting service from the slow disk.
   - State (0,2,0): both users are requesting from the fast disk.
   - State (0,1,1): one user is requesting service from the fast disk while the other user is requesting service from the slow disk.
   - State (0,0,2): both users are requesting from the slow disk.

Database Server
Model Construction-Trans.

2. State transitions:

- If both users are at the CPU (state (2,0,0)), one of the users could complete service at the CPU and go to either the fast disk (state (1,1,0)) or to the slow disk (state (1,0,1))
- If one of the users is at the CPU and the other is at the fast disk (state (1,1,0)), either the user at the fast disk could finish and return to the CPU (state (2,0,0)), or the user at the CPU could finish and go to either the fast disk (state (0,2,0)) or to the slow disk (state (0,1,1)
- if one of the users is at the CPU and the other is at the slow disk (state (1,0,1)), either the user at the slow disk could finish and return to the CPU (state (2,0,0)), or the user at the CPU could finish and go to either the fast disk (state (0,1,1)) or to the slow disk (state (0,0,2).

Database Server: Model Construction-Trans.

- If both users are at the fast disk (state (0,2,0)), one of the users could finish and return to the CPU (state (1,1,0)).

- If one of the users is at the fast disk and the other is at the slow disk (state (0,1,1)), either the user at the fast disk could finish and return to the CPU (state (1,0,1)), or the user at the slow disk could finish and return to the CPU (state (1,1,0)).

- If both users are at the slow disk (state (0,0,2)), one of the users could finish and return to the CPU (state (1,0,1)).

Transitions

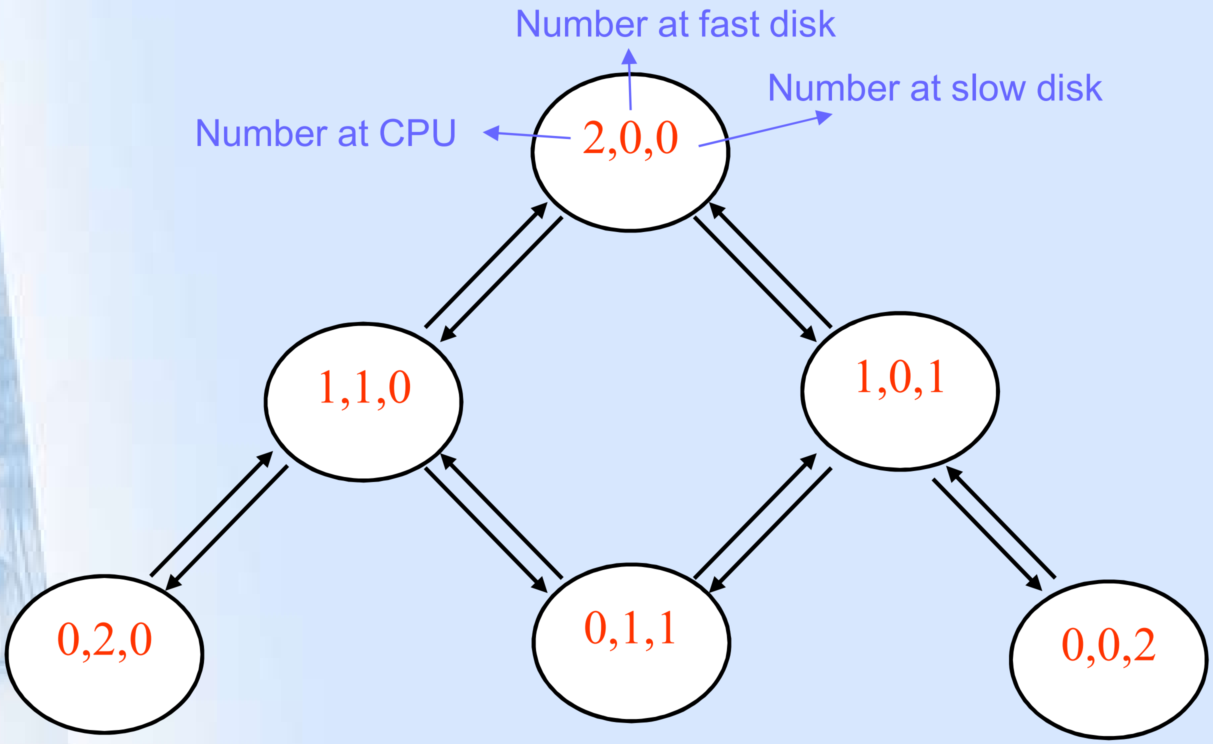


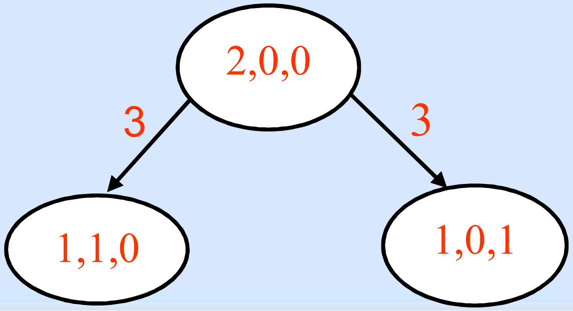
Figure 10.5. Markov Model of the database server example.

Database Server: Model Construction-Parameter

3. Model parameterization
   - In the England example, these weights are given as simple transition probabilities.
   - In the database server example, these are given as "flow rates," which is a generalization of transition probabilities.

Database Server:
Model Construction.

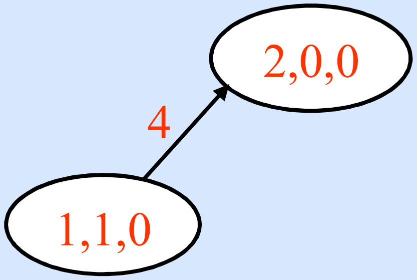
- Suppose the system is in state (2,0,0) where both users are at the CPU
- In this state, the CPU is satisfying user requests at a rate of 6 transactions per minute (an average of 10 seconds for one user's CPU demand)
- Of the 6 transactions per minute that the CPU can fulfill, half of these transactions next visit the fast disk (state (1,1,0)) and half next visit the slow disk (state (1,0,1))



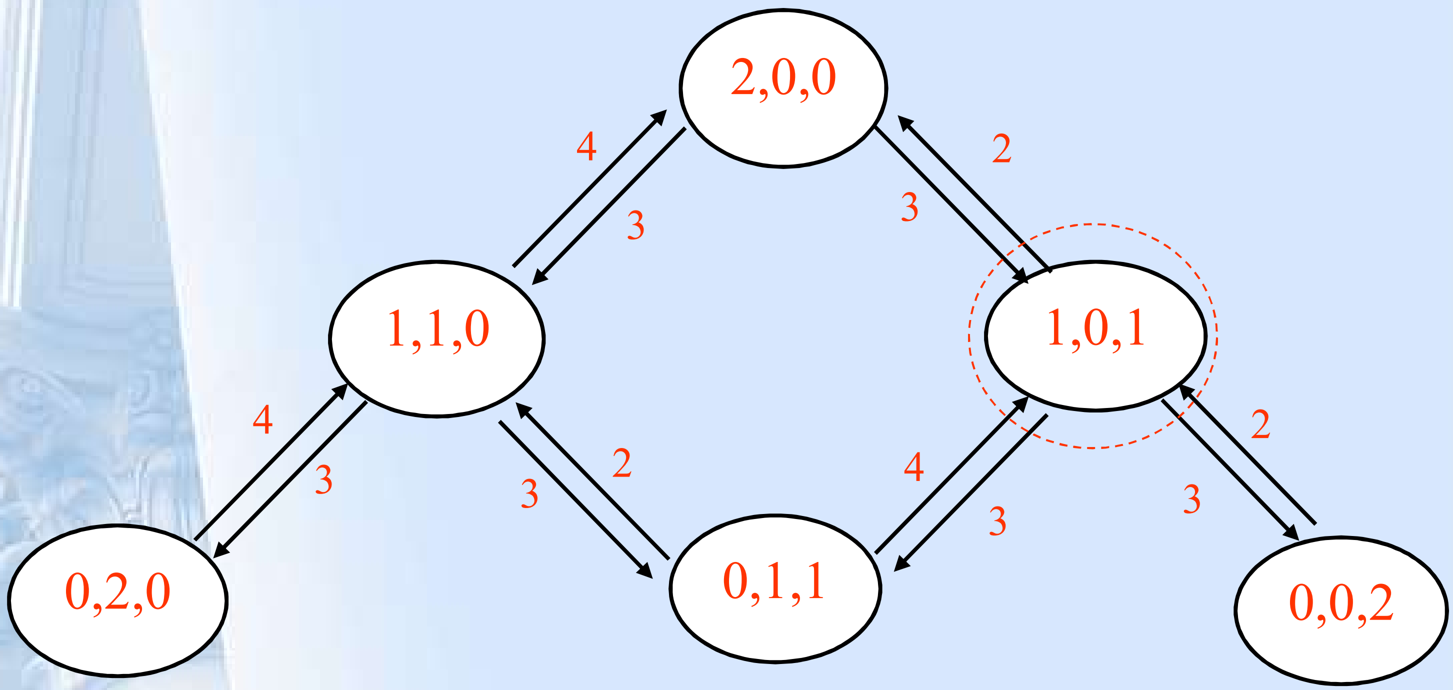
3

Database Server: Model Construction..

- The system is in state (1,1,0) since the fast disk satisfies user requests at a rate of 4 transactions per minute and since all users at the fast disk next visit the CPU, the weight assigned to the (1,1,0)→(2,0,0) transition is 4.
- We can also find other transaction weight like this state.



Database Server:
Model Construction-Model Par.



RWTE: Model Solution

- The model solution step is surprisingly the most straightforward and easiest aspect of the entire modeling paradigm.
- Robust software solution packages based on the mean value analysis (MVA) technique are available.
- The definition of "model solution" is to find the long term (i.e., the "steady state") probability of being in any particular state.
- Steady state is independent of the initial starting state of the system.

State Probability

- Let P_i represent the (steady state) probability of being in state *i*. Thus,
  - P_1 represents the probability that the lad is in the Leeds pub,
  - P_2 represents the probability that the lad is sightseeing in London,
  - P_3 represents the probability that the lad is kayaking in the Lake District, and
  - P_4 represents the probability that the lad is hiking in the Yorkshire moors.

Balance Equations

- The balance equation for each system state is one that represents the fact that:

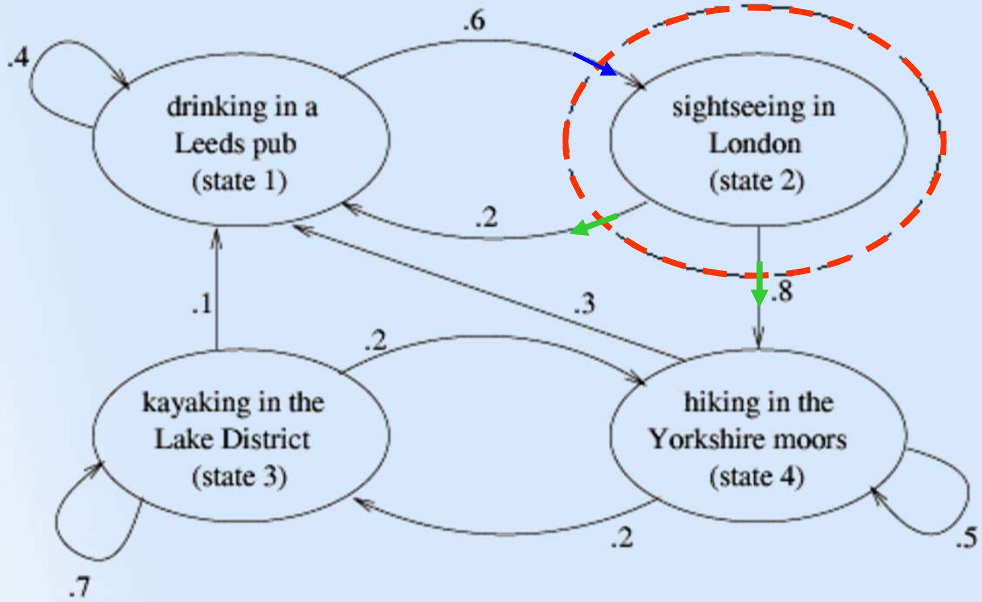
For Each State
Overall flow in = Overall flow out

- That is, on average, the lad must walk out of the Leeds pub as many times as he walks in.
- (The consequences of any other result would reduce the system to one where the lad is either always or never in the pub.)

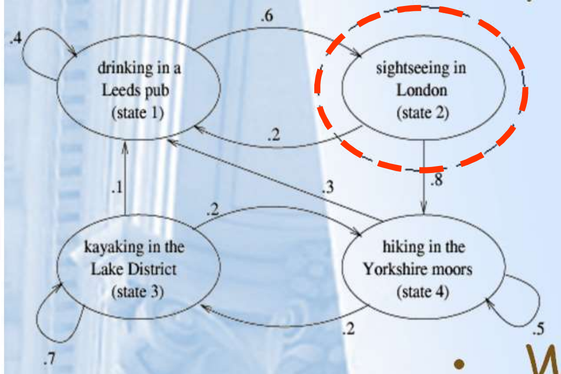
RWTE: Model Solution

- For state 2: the flow in=flow out equation is

$$0.6 \times P_1 = 0.2 \times P_2 + 0.8 \times P_2$$



RWTE: Equilibrium Equations


.4
.6
drinking in a Leeds pub (state 1)
sightseeing in London (state 2)
.2
.1
.3
.8
.2
kayaking in the Lake District (state 3)
hiking in the Yorkshire moors (state 4)
.2
.5
.7

- For all states:

$$0.2\times P_2 + 0.1\times P_3 + 0.3\times P_4 = 0.6\times P_1$$

$$0.6\times P_1 = P_2$$

$$0.2\times P_4 = 0.3\times P_3$$

$$0.8\times P_2 + 0.2\times P_3 = 0.5\times P_4$$

- We also have: $P_1 + P_2 + P_3 + P_4 = 1$

- Results

$$P_1 = 0.2644$$

$$P_2 = 0.1586$$

$$P_3 = 0.2308$$

$$P_4 = 0.3462$$

Equations System

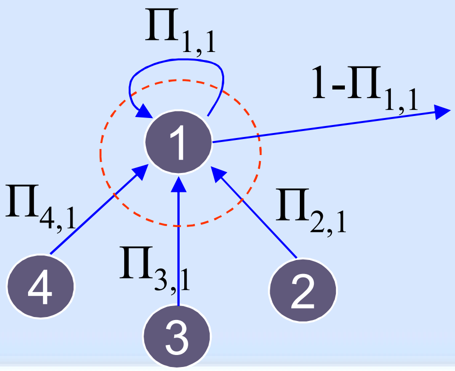
- We could write the equations like follow:

$$\begin{cases} [P_1\ P_2\ P_3\ P_4]\times\begin{bmatrix}\Pi_{1,1} & \Pi_{1,2} & \Pi_{1,3} & \Pi_{1,4}\\ \Pi_{2,1} & \Pi_{2,2} & \Pi_{2,3} & \Pi_{2,4}\\ \Pi_{3,1} & \Pi_{3,2} & \Pi_{3,3} & \Pi_{3,4}\\ \Pi_{4,1} & \Pi_{4,2} & \Pi_{4,3} & \Pi_{4,4}\end{bmatrix} = [P_1\ P_2\ P_3\ P_4] \\ P_1+P_2+P_3+P_4=1 \end{cases}$$

$$\sum_{i=1}^{4}\Pi_{j,i}=1,\quad for\ j=1,\dots,4$$

$$P_1\times\Pi_{1,1}+P_2\times\Pi_{2,1}+P_3\times\Pi_{3,1}+P_4\times\Pi_{4,1}=P_1$$

$$P_2\times\Pi_{2,1}+P_3\times\Pi_{3,1}+P_4\times\Pi_{4,1}=(1-\Pi_{1,1})P_1$$



Equation System.

$$P = [\, P_1 \;\; P_2 \;\; P_3 \;\; P_4 \,], \qquad \Pi = \begin{bmatrix} \Pi_{1,1} & \Pi_{1,2} & \Pi_{1,3} & \Pi_{1,4} \\ \Pi_{2,1} & \Pi_{2,2} & \Pi_{2,3} & \Pi_{2,4} \\ \Pi_{3,1} & \Pi_{3,2} & \Pi_{3,3} & \Pi_{3,4} \\ \Pi_{4,1} & \Pi_{4,2} & \Pi_{4,3} & \Pi_{4,4} \end{bmatrix}$$

$$P^T = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix} \qquad \Pi^T = \begin{bmatrix} \Pi_{1,1} & \Pi_{2,1} & \Pi_{3,1} & \Pi_{4,1} \\ \Pi_{1,2} & \Pi_{2,2} & \Pi_{3,2} & \Pi_{4,2} \\ \Pi_{1,3} & \Pi_{2,3} & \Pi_{3,3} & \Pi_{4,3} \\ \Pi_{1,4} & \Pi_{2,4} & \Pi_{3,4} & \Pi_{4,4} \end{bmatrix}$$

$$P\Pi = P, (P\Pi)^T = P^T$$

$$\Pi^T P^T = P^T$$

P^T is the Eigenvector of Π^T while the Eigenvalue $\lambda = 1$.

Database Server:
Model solution

- For solving example 2 we can do as same as example 1:

$$\begin{aligned}
(4 \times P_{(1,1,0)}) + (2 \times P_{(1,0,1)}) &= 6 \times P_{(2,0,0)} \\
(3 \times P_{(2,0,0)}) + (4 \times P_{(0,2,0)}) + (2 \times P_{(0,1,1)}) &= 10 \times P_{(1,1,0)} \\
(3 \times P_{(2,0,0)}) + (4 \times P_{(0,1,1)}) + (2 \times P_{(0,0,2)}) &= 8 \times P_{(1,0,1)} \\
3 \times P_{(1,1,0)} &= 4 \times P_{(0,2,0)} \\
(3 \times P_{(1,1,0)}) + (3 \times P_{(1,0,1)}) &= 6 \times P_{(0,1,1)} \\
3 \times P_{(1,0,1)} &= 2 \times P_{(0,0,2)}
\end{aligned}$$

$$P_{(2,0,0)} + P_{(1,1,0)} + P_{(1,0,1)} + P_{(0,2,0)} + P_{(0,1,1)} + P_{(0,0,2)} = 1.0$$

Database Server: Results

- And steady state probabilities are:

$$
\begin{aligned}
P_{(2,0,0)} &= \frac{16}{115} = 0.1391 \\
P_{(1,1,0)} &= \frac{12}{115} = 0.1043 \\
P_{(1,0,1)} &= \frac{24}{115} = 0.2087 \\
P_{(0,2,0)} &= \frac{9}{115} = 0.0783 \\
P_{(0,1,1)} &= \frac{18}{115} = 0.1565 \\
P_{(0,0,2)} &= \frac{36}{115} = 0.3131
\end{aligned}
$$

RWTE: Question1

- What percentage of days is the son actually not drinking in Leeds?
  - Answer: $1- P_1$ =74%.
  - Since the steady state probability of being in state 1 (i.e., drinking in a Leeds pub) is 0.2644 (i.e., 26%), the rest of the time ***(74%)*** the lad is sightseeing, kayaking, or hiking.

RWTE: Question2

- Once the son finishes a day of kayaking in the Lake District, how long will it typically be before he returns?
  - Answer: 3.33 days
- The mean time between entering a particular state (i.e., the state's "cycle time") is the inverse of the steady state probability of being in that state.
  - Since the steady state probability of being in state 3 is 0.2308, the cycle time between successive entries into state 3 is 1/0.2308 = 4.33 days.
  - Since it takes one day for the lad to kayak, the time from when he finishes a day of kayaking until he typically starts kayaking again is 4.33 – 1 = ***3.33*** days.

RWTE: Question3

- How many days each month can the bobbies expect to see the son driving to London after drinking in Leeds?
  - Answer: 4.76 days.
  - Consider a 30 day month. The steady state probability of being found drinking in a Leeds pub (i.e., state 1) on any particular day is 0.2644. Thus, out of 30 days, 30 x 0.2644 = 7.93 days will find the lad drinking.
  - However, since the lad decides to go to London with only probability 0.6 after a day of drinking in Leeds, the bobbies can expect to find the lad on the road to London 7.93 x 0.6 = ***4.76*** days each month.

RWTE: Question4

- How many visits each month does the son typically visit the Lake District ?
  - Answer: 2.08 visits per month.
  - The only way to enter state 3 (i.e., kayaking) from another state is from state 4 (i.e., hiking).
  - The steady state probability of being found hiking on any particular day is 0.3462, or 30 x 0.3462 = 10.39 days each month
  - However, after hiking for a day, the lad only decides to go kayaking the next day with probability 0.2. Then, the lad typically starts a new visit to the Lake District 10.39 x 0.2 = ***2.08*** times each month.

RWTE: Question5

- Typically how long does the son stays at the Lake District each visit?
  - Answer: keeping the kayak an average of 3.33 days each visit.
  - Since the steady state probability of being found kayaking is P_3=0.2308 on any particular day, the lad can be expected to be kayaking 30 x 0.2308 = 6.92 days out of each month.
  - If he makes only 2.08 new visits each month, the duration of each visit is typically 6.92 / 2.08 = ***3.33*** days/visit.

RWTE: Question 5.

- An alternative solution for question 5.
  - the lad kayaks for only one day with probability 0.3. He kayaks for exactly two days with probability 0.7×0.3. He kayaks for exactly three days with probability $2(0.7) \times 0.3$ and, in general, he kayaks for exactly n days with probability $(0.7)^{n-1}\ x\ 0.3$.
  - The average time spent kayaking per visit is:

$$\sum_{i=1}^{\infty} i\,(0.7)^{i-1}(0.3) = 3.33\ \ days$$

Database Server: User's Question

- What response time can the typical user expect?
  - Answer: 44.24 seconds per user transaction.
  - The response time can be found via application of the Interactive Response Time Law,
  - $R = M/X_0 - Z$, eq 3.2.10 (chapter 3)
  - Z (think time) $=0$,
  - M (average number of users n the system) $=2$

We must find X_0 (throughput)

Database Server: User's Question'

$$U_{cpu} = D_{cpu} \times X_O$$

$$X_O = U_{cpu} \times \frac{1}{D_{cpu}}$$

$$N = RX_O$$

- The throughput of the system, measured at the CPU, is the product of its utilization and its service rate.
- The CPU is utilized in states (2,0,0),(1,1,0), and (1,0,1). then the CPU utilization is $P(2,0,0) + P(1,1,0) + P(1,0,1) = 0.1391 + 0.1043 + 0.2087 = 0.4521$.
- The service rate of the CPU is 6 transactions per minute.
  - Therefore, the throughput measured at the server is 0.4521 x $6 = 2.7126$ transactions per minute,
- Resulting in an average response time of $2/2.7126 = 0.7373$ minutes per transaction, which equals 0.7373 x $60 = 44.24$ seconds per user transaction.

Database Server: System administrator's question

- How near capacity (i.e., what is the utilization) of each of the system resources?
  - Answer:

    CPU's utilization is 0.4521,

    fast disk's utilization is 0.3391,

    slow disk's utilization is 0.6783.
  - These are found as direct sums of the relevant steady state probabilities.

Database Server: Company president's question

- What new performance levels should be if we double the active users? The results are:
  - Throughput: 3.4768 transactions per minute
  - The user response time: 69.03 seconds per transaction
  - The original system had a state space diagram of 6 states. The new system model has 15 states.

Database Server: Company pessimist's question

- Since I know that the fast disk is about to fail and all the files will need to be moved to the slow disk, what will the new response time be?
  - Answer: 65.00 seconds per transaction
  - This number is a result of resolving the model with only two devices, the CPU and the slow disk.
  - The resulting state diagram has only 3 states, with steady state probabilities: 0.0769, 0.2308, and 0.6923.
  - This leads to a CPU utilization of 0.3077 and a system throughput of 1.8462 transactions per minute.
  - $R = M/X0 - Z= 2/1.8462-0=1.083$ minute = 65 second

Model Assumptions: Memory-Less

- Memory-less Assumption:
  - It is assumed that all the important system information is captured in the state descriptors of a Markov model. That is, simply knowing which state the system is in, uniquely defines all relevant information.
  - Knowing the current state alone is sufficient to determine the next sate. It doesn't matter how one arrives (i.e., by which path) to a particular state.
  - It means, the only thing that is important in determining which state will be visited next is that the system is in a particular state at the current time

Model Assumptions: Memory-Less.

- This assumption also implies that the length of time that the system is in a particular state, by continually taking self loops, is irrelevant.
- This places a nontrivial burden on choosing an appropriate notation for the state descriptor.
- For instance, if the jobs have different characteristics, this must be evident from the state descriptor.
- If the order in which jobs arrive to the systems makes a difference, this, too, must be captured in the state descriptor.
- That is, previous history can be forgotten. This explains the term "memoryless" as it applies to Markov models

Limitations of Memory-Less Assumption

- Resulting Limitation of memory-less assumption:
  - Because everything must be captured in the state descriptor, Markov models are susceptible to state space explosion.
  - For example, if there are 10 customers at the CPU, and the CPU is scheduled first-come-first-serve, then the number of state for the CPU will be 10! = 3,628,800
  - However, if the customers behave similarly and the order of customer arrivals to CPU is not important, then the state descriptor of the CPU can be represented by a single number.

Limitations of Memory-Less Assumption.

- Exponential Assumption:
  - The exponential probability distribution is the only continuous distribution that is memory-less.
  - E.g.: The average number of flips to get a head is two.
    - knowing that the first flip resulted in a tail, the average number of flips still needed to get a head is again two.
  - Another E.g.: suppose the average amount of CPU time required by a customer is 10 seconds.
    - Knowing that the customer has already received 5 seconds worth of CPU time but not yet finished (i.e., previous history, which is irrelevant under the Markov memory-less assumption), the average amount of CPU time still needed is again 10 seconds.

Exponential

- Thus, Markov models assume that:

  the time spent between relevant events, such as job arrival times and job service times, is exponentially distributed.

Exponential Distribution

The density of an exponential distribution with parameter μ is given by

$$f(t) = \mu e^{-\mu t}, \qquad t > 0.$$

The distribution function equals

$$F(t) = 1 - e^{-\mu t}, \qquad t \geq 0.$$

For this distribution we have

$$E(X) = \frac{1}{\mu}, \qquad \sigma^2(X) = \frac{1}{\mu^2}, \qquad c_X = 1.$$

An important property of an exponential random variable X with parameter μ is the *memoryless property*. This property states that for all $x \geq 0$ and $t \geq 0$,

$$P(X > t + x | X > t) = P(X > x) = e^{-\mu x}.$$

So the remaining lifetime of X, given that X is still alive at time t, is again exponentially distributed with the same mean $1/\mu$.

Limitations of Memory-Less Assumption: Other Distributions

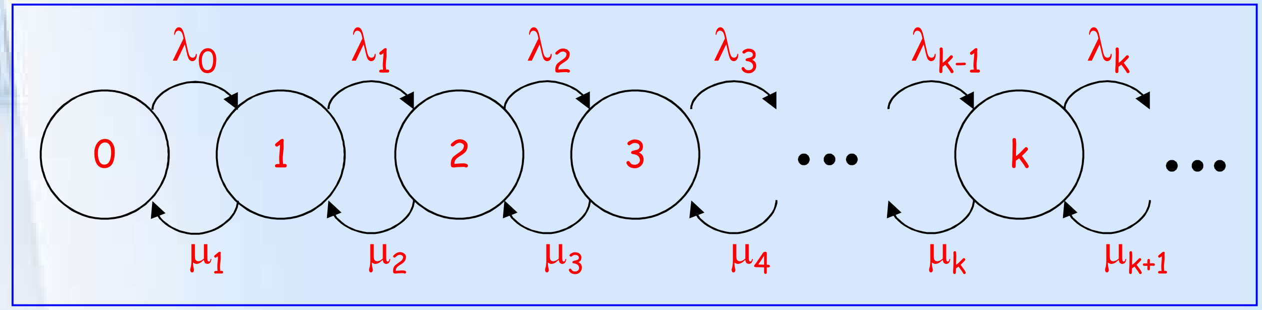
- E.g.: again suppose that the average amount of CPU time required by a customer is 10 seconds.
- By partitioning the total service requirement into two phases of service (i.e., each phase being exponentially distributed with an average of 5 seconds), the CPU state for each customer can be decoupled into two states.
  - That is, each customer can be in either its first stage of service or in its second stage of service.
- This technique opens up a whole host of other distributions (i.e., not simply exponential) that can be closely approximated. However, the price is again a potential state space explosion since the state descriptors must now contain this additional phase information

Final Word

- Theoretically, Markov models can be constructed to any desired level of accuracy.
- The price is
  - the complexity of the state descriptors and
  - the resulting size of the state space.
- Practically, beyond a few 1000 states, the computational complexity is limiting and any intuitive benefits are lost.
- Thus, the system analyst is faced with a tradeoff of determining an acceptable level of aggregation that affords both accuracy and efficiency.

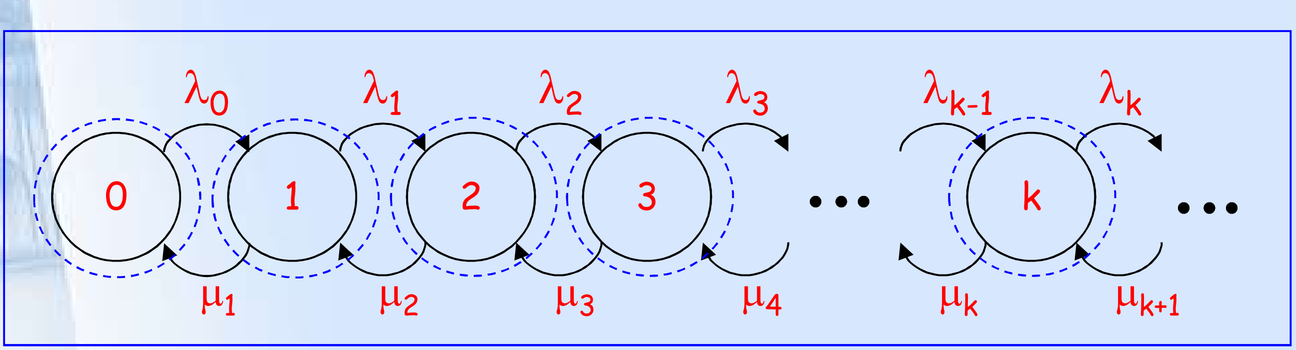
Generalized Birth-Death Models

- Generalized Birth-Death model is a class of Markov models.
- In each state, one of two events can occur:
  - birth: the arrival of another customer
  - death: the departure of a customer
- In state k:
  - birth: entering in state k+1 with rate λ_k
  - death: entering in state k–1 with rate μ_k



Flow balance Equations

$$\textit{flow in} = \textit{flow out} \longrightarrow \begin{cases} \mu_1 P_1 = \lambda_0 P_0 \\ \lambda_0 P_0 + \mu_2 P_2 = \lambda_1 P_1 + \mu_1 P_1 \\ \ldots \\ \ldots \\ \lambda_{k-1} P_{k-1} + \mu_{k+1} P_{k+1} = \lambda_k \; P_k + \mu_k \; P_k \end{cases}$$



Generalized Equations

- Equation 10.8.1

$$P_0 = \left[\sum_{k=0}^{\infty}\prod_{i=0}^{k-1}\lambda_i / \mu_{i+1}\right]^{-1} = \frac{1}{1+\frac{\lambda_0}{\mu_1}+\frac{\lambda_0}{\mu_1}\times\frac{\lambda_1}{\mu_2}+\frac{\lambda_0}{\mu_1}\times\frac{\lambda_1}{\mu_2}\times\frac{\lambda_2}{\mu_3}+\ldots}$$

- Equation 10.8.2

$$P_k = P_0 \times \prod_{i=0}^{k-1}\frac{\lambda_i}{\mu_{i+1}} \qquad k = 0,1,2,\ldots.$$

- If $\lambda_0 = \lambda_0 = \ldots = \lambda_k = \ldots = \lambda$ and $\mu_1 = \mu_2 = \ldots = \mu_k = \ldots = \mu$

$$U = 1 - P_0 \rightarrow P_0 = 1 - \frac{\lambda}{\mu} = 1 - U \qquad P_k = P_0 \times (\frac{\lambda}{\mu})^k$$

Performance Equations

- Equation 10.8.3: $utilization = p_1 + p_2 + ... = 1 - p_0$
- Equation 10.8.4

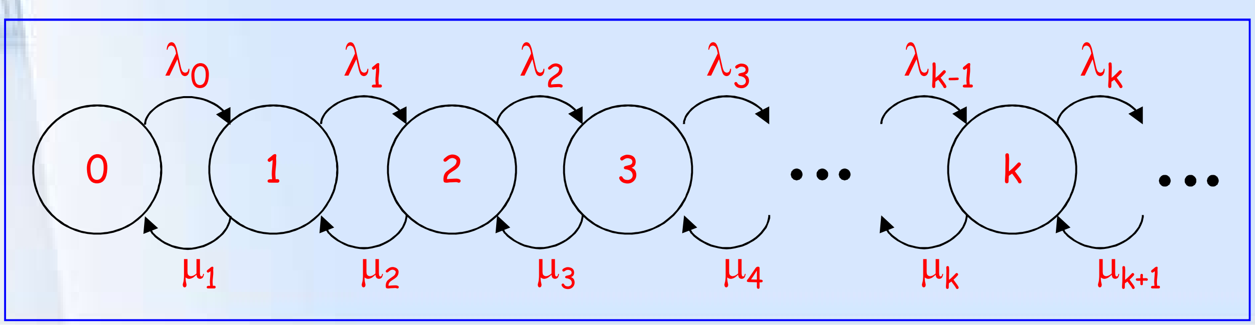
$$throughput = \mu_1 P_1 + \mu_2 P_2 + ... = \sum_{k=1}^{\infty} \mu_k P_k$$

- Equation 10.8.5

$$queue \;\; length = 0P_0 + 1P_1 + 2P_2 + ... = \sum_{k=1}^{\infty} k \, P_k$$

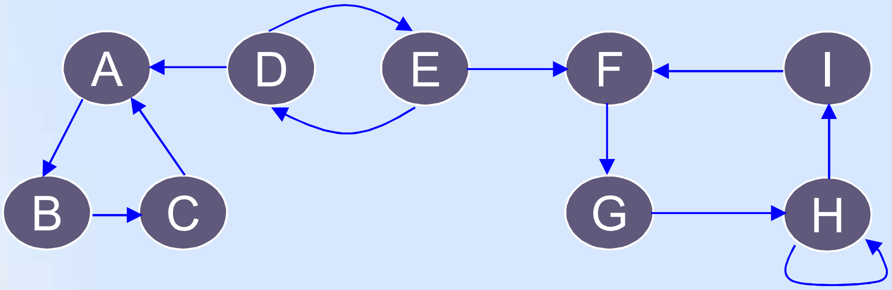
- Equation 10.8.6

$$Littel' sLaw : response\, time = \frac{queue \;\; length}{throughput} = \frac{\sum_{k=1}^{\infty} k \, P_k}{\sum_{k=1}^{\infty} \mu_k \, P_k}$$



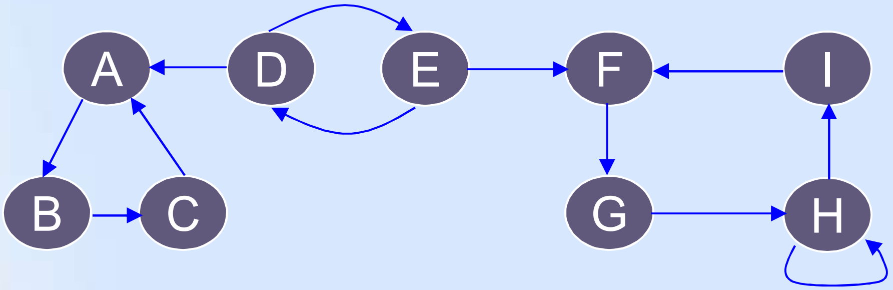
Definitions 1

- Definitions:
  - Recurrent state: A state that can always be revisited in the future after leaving the state (A, B, C, F, G, H, and I)
  - Transient state: A state that may not be possible to return to the state after leaving it, depending on the subsequent states visited. (D and E)



Definitions1.

- Periodic state: A periodic state is a recurrent state where the system can only return to the periodic state in p, 2p, 3p, ..., steps, P>1 , p is the period
- States A, B, and C are periodic with a period of 3.
- The self loop around state H prohibits it (and also states F, G, and I) from being periodic



Facts of Definitions1

- Fact: Each state in the Markov model is either recurrent or transient.
- Fact: The set of recurrent states and the set of transient states is mutually exclusive and collectively exhaustive.
- Fact: All states reachable from a recurrent state are recurrent.
- Fact: All states that can reach a transient state are transient.
- Fact: All states reachable from a periodic state are periodic with the same period.

Definations2

- Chain: A chain is a set of recurrent states that can all reach each other. The set is as large as possible. (A, B, and C) form one chain. (F, G, H, and I) form another chain

- Discrete state, discrete transition: A Markov model is discrete state, discrete transition if the number of states is countable and *the transitions between states can only take place at known intervals*.

- Discrete state, continuous transition: A Markov model is discrete state, continuous transition if the number of states is countable and *the transitions between states can take place at any time*, driven by an exponential distribution.

Facts of Definitions2

- Fact: Discrete state, continuous transition Markov models do not have periodic states. This is since transitions can take place at any time.

- Major fact: Any finite Markov model, with no periodic states and whose recurrent states are all in the same chain, will have limiting state probabilities that are independent of the initial starting state. The example does not have steady state that is independent of the initial starting state because it has two chains, not one.

- Fact: The steady state probability of being in a transient state is 0.

Model Construction Steps

Within the context of Markov models,
model construction consists of three steps:

1. state space enumeration

   specifying all reachable states that the system might enter

2. state transition identification

   identification indicates which states can be directly entered from any other given state

3. parameterization

   making measurements and making assumptions of the original system

Summary1

- This chapter presents a basic, practical, and working knowledge of Markov models. Markov models fit within the general modeling paradigm which involves model construction, solution, calibration, alteration, and validation. Markov models are useful for both descriptive as well as predictive purposes. They are versatile and can model a wide range of applications. Two quite diverse applications are considered in this chapter and each of the modeling steps is demonstrated in detail. The assumptions and limitations of Markov models are summarized. The basics, as well as building blocks for more advanced topics, are presented.
- In general, the primary limitation of Markov models is that they are susceptible to state space explosion. This explosion poses a danger that the computational complexity of solving the balance equations is prohibitive.

Summary2

- Fortunately, there are subclasses of Markov models that lend themselves to efficient, alternative solution techniques. One such subclass of models is known as separable Markov models. (The database server example in this chapter is one example of a separable Markov model.) Separable Markov models can be solved using the Mean Value Analysis (MVA) technique, which is the topic of the following chapter.
- We believe that the best way to learn about and to understand the subtleties of system modeling is not a passive process, but rather an active engagement. Arguably, the primary benefit of system modeling is the development of keen insights and intuition by the system analyst concerning the interdependencies between various modeling parameters. To this end, a rich set of exercises is provided. The reader is encouraged to participate by attempting to solve these exercises. They are not trivial.

Bibliography

[1] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications, John Wiley & Sons, New York, 1998.

[2] P. J. Denning and J. P. Buzen, "The operational analysis of queueing network models", Computing Surveys, vol. 10, no. 3, September 1978, pp. 225–261.

[3] A. W. Drake, Fundamentals of Applied Probability Theory, McGraw-Hill Higher Education (Columbus), 1967.

[4] L. Kleinrock, Queueing Systems, Volume I: Theory, Wiley-Interscience, New York, 1975.

[5] E. D. Lazowska, J. Zahorjan, S. Graham, and K. C. Sevcik, Quantitative System Performance: Computer System Analysis Using Queueing Network Models, Prentice-Hall, Upper Saddle River, New Jersey, 1984.

Appendix:

Distributions

Gamma & Exponential Distributions

- Exponential and gamma distributions find application in queuing theory and reliability studies.
- The exponential distribution is a special case of the gamma distribution.
- Examples:
  - Time between customer arrivals at a terminal.
  - Time to failure of electrical components.

Gamma Distribution

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta}, \quad x > 0$$

where $\alpha > 0$ and $\beta > 0$.

The mean and variance of x are:

$\mu = \alpha\beta$ and $\sigma^2 = \alpha\beta^2$

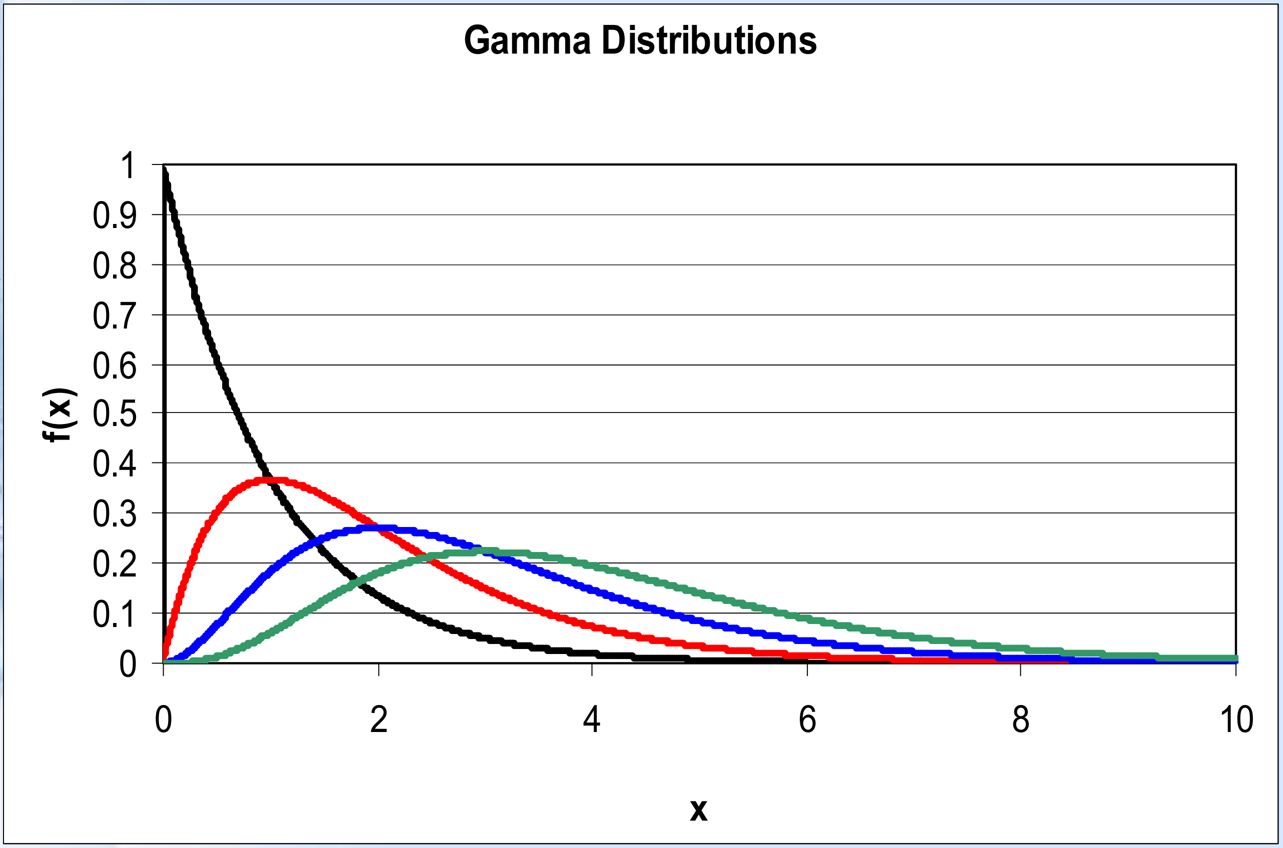
Gamma Distributions

α=1, β=1

α=2, β=1

α=3, β=1

α=4, β=1



Exponential Distribution

- A gamma distribution with α = 1 is called the exponential distribution.

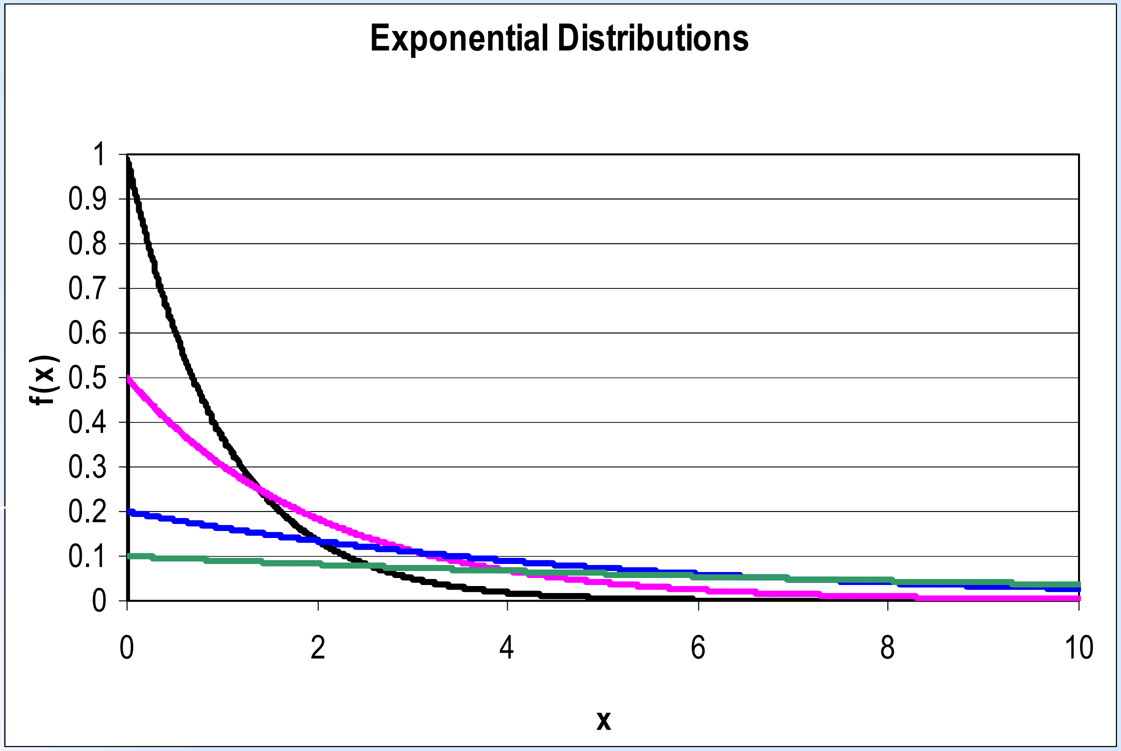
$$f(x) = \frac{1}{\beta} e^{-x/\beta}, x > 0$$

$$F(x) = 1 - e^{-x/\beta}$$

where $\beta > 0,\ \mu = \beta$ and $\sigma^2 = \beta^2$

Exponential Distributions

α=1, β=1
α=1, β=2
α=1, β=5
α=1, β=10



Exponential Applications

- Example: In a certain city, daily consumption of electric power (in millions of kilowatt hours) is a random variable that follows an exponential distribution with $\beta = 3$.
  - (a) What is the mean daily consumption of power?
  - (b) What is the variation of the daily consumption of power?
  - (c) If the city's power plant has a daily capacity of 12 million kilowatt hours, what is the probability that this power supply will be inadequate on any given day?

Relationship between Exponential & Poisson

- Recall that the Poisson distribution is used to compute the probability of a specific number of events occurring in a particular interval of time or space.

  Instead of the number of events being the random variable, what if the time or space interval is the random variable?

EX: We want to model the space between defects in material. (Defect is a Poisson event)

EX: We want to model the time between radioactive particles passing through a counter. (arrival of radioactive particle is a Poisson event)

Relationship between Exponential & Poisson

- Recall:

$$p(x;\lambda t)=\frac{e^{-\lambda t}(\lambda t)^{x}}{x!},x=0,1,2,...$$

- where λ is mean number of events per base unit time or space and t is the number of base units being inspected.

- The probability that no events occur in the span of time (or space) t is:

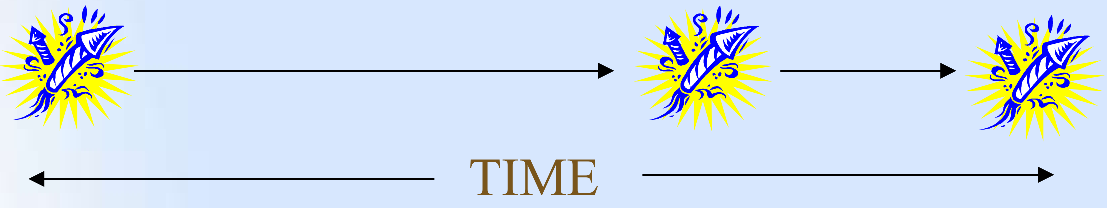
$$p(0;\lambda t)=\frac{e^{-\lambda t}(\lambda t)^{0}}{0!}=e^{-\lambda t}$$

Relationship between Exponential & Poisson

- Let
  - X = the time (or space) to the first Poisson event.
- Note,
  - the probability that the length of time (or space) until the first event > some time (or space),
  - x is the same as the probability that no events will occur in x, which $= e^{-\lambda x}$.
- So, $P(X > x) = e^{-\lambda x}$ and $P(X < x) = 1 - e^{-\lambda x}$
- $1 - e^{-\lambda x}$ is the cumulative distribution function for an exponential random variable with $\lambda = 1/\beta$.

Relationship between Exponential & Poisson

Exponential distribution models time (or space) between Poisson events.



TIME

Note, $\beta = 1/\lambda$ and $\lambda = 1/\beta$

Relationship between Exponential & Poisson

- EX: Radioactive particles passing by a counter are Poisson events. Suppose the particles average 2 per millisecond.

  (a) What is the probability that at least 1 particle will pass in 3 milliseconds?

  (b) What is the probability that more than 1 millisecond will elapse before the next particle passes?

Gamma Application

- Exponential models time (or space) between Poisson events.
- Gamma models time (or space) occurring until a specified number of Poisson events occur with α = the specific number of events and β = mean time (or space) between Poisson events.
- EX: Radioactive particles passing by a counter follow a Poisson process with an average of 4 particles per millisecond. What is the probability that up to 2 millisecond will elapse until 3 particles have passed the counter?

$$g(\, x \leq 2; \alpha=3, \beta=0.25\,) = 0.9862$$

Exponential & Binomial Application

- EX: Let
  - X = the time in 1,000's of hours to failure for an electronic component. X follows an exponential distribution with mean time to failure $\beta = 10$ (remember that is 10,000 hours).
- (a) What is the probability that the electronic component is functioning after 15,000 hours?
- (b) If 6 of these electronic components are installed in different systems, what is the probability that at least 3 are still functioning at the end of 15,000 hours?

Continuous Distribution Summary

True for all continuous distribution:

$$\int_{-\infty}^{+\infty} f(x)dx = 1 \qquad f(x) \geq 0 \qquad F(a) = \int_{-\infty}^{a} f(x)dx$$

$$P(a < x < b) = \int_{a}^{b} f(x)dx = F(b) - F(a)$$

$$\mu = \int_{-\infty}^{+\infty} xf(x)dx$$

$$\sigma^2 = E[X^2] - \mu^2 = \int_{-\infty}^{+\infty} x^2 f(x)dx - \mu^2$$

Continuous Distribution Summary

Continuous Uniform

$$f(x; A, B) = \frac{1}{B - A}, A \le x \le B$$

$$\mu = \frac{A + B}{2}$$

$$\sigma^2 = \frac{(B - A)^2}{12}$$

Continuous Distribution Summary

Normal distribution

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

$$E[X] = \mu$$

$$Var[X] = \sigma^2$$

$$Z = \frac{X - \mu_x}{\sigma_x}$$,where Z is standard normal

Continuous Distribution Summary

Exponential distribution

$$f(x) = \frac{1}{\beta} e^{\frac{-x}{\beta}} \qquad \text{,where } \beta > 0$$

$$F(x) = 1 - e^{\frac{-x}{\beta}}$$

$$\mu = \beta$$

$$\sigma^2 \ = \beta^2$$

Exponential distribution models time (or space) between Poisson events. Note, $\beta = 1/\lambda$ and $\lambda = 1/\beta$

Continuous Distribution Summary

Gamma distribution

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{\frac{-x}{\beta}}, x > 0$$ where $\alpha > 0$ and $\beta > 0$

$$\mu = \alpha\beta$$

$$\sigma^2 = \alpha\beta^2$$

Gamma distribution models time (or space) occurring until a specified number of Poisson events occur with $\alpha =$ the specific number of events and $\beta =$ mean time (or space)between Poisson events.