Chapter-1

Introduction to Simulation

System, Model and Simulation

System:

- System is a simplified representation of a selected components with specified boundaries and predefined time characteristics.
- System generally indicates a real-world entity over space and time in generalized and simplified form.
- Each system has a time constant that define the time over which the system may change its state.

Model:

- Model is a computerized program that defines the mechanics of the considered system.
- During system modeling, the focus is on the system components, system structure, relationships among the components of the system and the behavior of the modeled system.
- Model must have state which may change on each time step.
- Model represents the system for the purpose of studying the system.

Simulation:

- Simulation is defined as the process of execution of the model of a considered system.
- The conditions required for the simulation are initial conditions of the system under consideration and the specified sets of parameters.
- Simulation involves the generation of the artificial history of a considered system and their observations to draw inferences concerning the operating characteristics of the real system.

A basic example (Coffee Shop):

Consider a real-world system of a coffee shop. It has customers who place orders and staff who process them. At some time, there are few customers and at other time, there are many customers. There are many changes in the system over time, so the time step of 5 minutes will be safer to analyze the system.

A model is generated for the selected system. Customers come and go, orders are placed, coffee is drunk, receipts are paid, messages are received, news is shared, sometimes coffee is spilled. At each time step, the model updates the status of the system.

For simulation, following things should be considered or set:

- When in the day does the modeling start?
- How many customers and staff are already there at that time?
- How much money does the cashier have then?
- What are the prices of the different kinds of coffee?
- What are the rates of inflow and outflow of customers and what determines that?
- What is the rate of inflow of messages?

When Simulation is the appropriate tool?

- To address the dynamic processes.
- To analyze the effect of alterations on the model's behavior.
- To determine the variables that are most important by changing the simulation inputs and observing the outputs.
- To verify the analytic solutions.
- To experiment new designs before actual implementations.

When Simulation is not appropriate?

- Simulation is not appropriate when the problem can be solved by common sense.
- Simulation should not be used if the problem can be solved analytically.
- Simulation should not be used if it is easier to perform direct

implementations.

- Simulation should not be used if the cost exceeds the savings.
- Simulation should not be done if the resources and time are not available.
- Simulation cannot be done if no data is available.
- Simulation is not preferred if the system behavior is too complex to be defined.

Advantages and Disadvantages of Simulation

Advantages

- New procedures or rules can be explored without hampering the ongoing real time system operations.
- New designs can be tested without acquiring resources.
- The hypotheses about certain phenomena can be tested for feasibility.
- Insights can be obtained about the variable interaction and their credit for system performance.
- It provides the better understanding of the operation of the system in easier way.

Disadvantages

- Building model requires specialization which is the art learned through experience.
- In some cases, the results of the simulation may be so complex to interpret.
- It may be time consuming and expensive.

Areas of Application

1. Manufacturing

- Dynamic modeling of continuous manufacturing systems.
- Modeling for quality and productivity.
- Shared resource capacity analysis.

2. Semiconductor Manufacturing

- Constant time interval production planning.
- Design framework for automated material handling systems.
- Making optimal design decisions.

3. Military

- Frequency based design for terminating simulations.
- Support for military based interactive simulations in 3D environments.
- Impact of an automatic logistic system.

4. Transportation Modes

- Simulating aircraft-delay absorption.
- Runway schedule determination.
- Modeling ambulance services.
- Modeling ship arrivals in ports.

Business Process

- Agent based modeling of store performance.
- Visualization of probabilistic business models.
- Baggage screening at airports.
- Human fatigue risk simulation in continuous operations.
- Optimization of telecommunications billing system.

6. Health Care

- Evaluation of hospital operations.
- Estimating maximum capacity in an emergency room.
- Reducing the length of stay in an emergency department.
- Scheduling emergency room staff.

Discrete and Continuous System

- A system is said to be discrete if the state variable(s) change only at a discrete set of points in time. Egg: A bank system, the number of customers changes only when the customer arrives or when service provided is completed.
- A system is said to be continuous if the state variable(s) change continuously over time. Egg: Water head behind dam.

Steps in Simulation Study

- 1. Problem Formulation
- To begin any study, there must be statement of problem.
- Problem statement is formulated either by policy makers or analysts. Whoever formulates the statement, it must be understood by both.
- Problems formulated at the beginning of the study may not be complete, so this process is further iterated whenever there arises a new problem and needs attention to solve it during other stages of simulation study.
- 2. Setting objectives and overall project plan
- Objectives for the simulation must be formulated.
- It indicates what is to be done by the simulation.
- In this phase, it is decided whether or not the simulation is the appropriate tool.
- If simulation is appropriate tool, then the overall project plan should be formulated with alternative systems to be taken into account and methods to evaluate the effectiveness of the alternatives.
- It should also include time, cost and manpower needed along with the expected results of each stage.
- 3. Model Conceptualization
- Model building is the process of construction of a model of a system.
- There is no necessity of one-to-one mapping between model and real system, only the essential components are used in modeling.
- For modeling, essential features of the system are extracted at first, basic assumptions are made and then the assumptions are modified until approximate results are obtained.
- At initial stage, the model should be simple and over time should

develop complexity.

- Excessive complexity should be forbidden to minimize model and computer expenses.

4. Data Collection

- Data plays important role in simulation.
- Data collection is very difficult phase as it requires more time, so it should be started as early as possible.
- The data to be collected depends on the study objectives.

5. Model Translation

- The model of real world generally requires storage, so model must be entered into a computer recognizable format.
- The model programming must be selected by the modeler, whether a simulation language or to use a special purpose simulation software.
- Simulation languages are more powerful and flexible than software's. But the choice depends on model complexity.

6. Verified?

- This stage involves verification of the computer program of modeling.
- Verification is successful if the input parameters and the logical structure of the models are correctly represented in the computer.

7. Validated?

- Validation is performed by calibrating the model against the actual system behavior so as to improve the system model.
- This is iterative process which is successful once the model accuracy is acceptable.

8. Experimental Design

- The alternatives to be simulated are determined.
- For each system design, length of initialization period, length of simulation run and number of replications to be made of each run are decided.

9. Production Runs and Analysis

- It is used to estimate measures of performance for the system design being simulated.

10. More Runs?

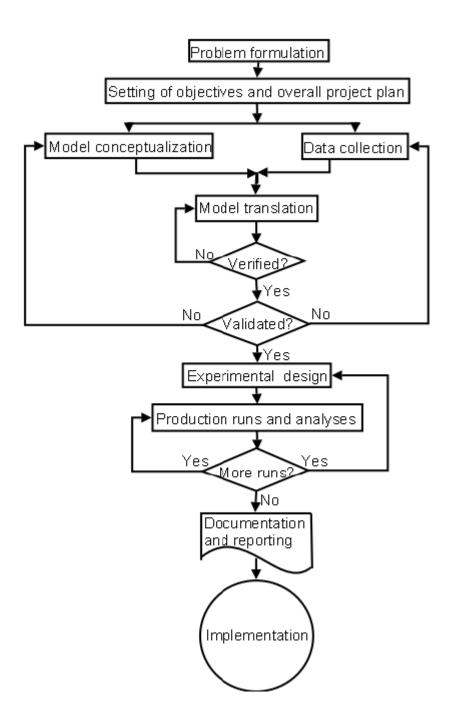
- After the analysis of the runs, it is decided whether or not more runs are needed.

11. Documentation and Reporting

- Program documentation is the document stating the operation of the program. It is necessary to understand how the program works. It facilitates easy modification and maintenance of the program.
- Progress documentation provides the chronology of work done and decisions made. It helps in tracking the course of project towards success.
- The result of the analysis should be clearly stated in a final report.

12. Implementation

- It involves actual implementation of the model.
- Model implementation depends on how well all the other stages are succeeded.
- Proper success of other stages results in good implementation while improper model on implementation may result in hazards.



Chapter-2

Physical and Mathematical models

Physical Model

- Physical model is the smaller or larger physical copy of an object being modeled.
- The geometry of model is just the rescaled view of the object it represents.
- The horizontal and vertical scaling may not be balanced as per the requirements of the model.
- The physical model helps in visualization of the object taken into consideration in an effective way.
- It is also used to solve equations with the particular boundary conditions.

Static Physical Model:

- Static physical model is the physical model which describes relationships that do not change with respect to time.
- Such models only depict the object's characteristics at any instance of time, considering that the object's property will not change over time.
- Egg: An architectural model of a house, scale model of a ship and so on.

Dynamic Physical Model:

- Dynamic physical model is the physical model which describes the time varying relationships of the object properties.
- Such models describes the characteristics of the object that changes over time.
- It rely upon the analogy between the system being studied and some other system of a different nature, but have similarity on forces that

directs the behavior of the both systems.

- Egg: A model of wind tunnel, a model of automobile suspension and so on.

Mathematical Model

- Mathematical model is the model which is composed of a symbols and logic.
- Such model is also called an abstract model.
- It describes the system using mathematical concepts.
- The mathematical model is used to explain the system and to study the effects of different components, and to make predictions about the behavior of the system.
- It generally consists of following major elements:
- a) Governing equations
- b) Defining equations
- c) Constitutive equations
- d) Constraints

Static Mathematical Model:

- Static mathematical model is the mathematical model that represents the logical view of the system in equilibrium state.
- Such models are time-invariant.
- It is generally represented by the basic algebraic equations.
- Egg: An equation relating the length and weight on each side of a playground variation, supply and demand relationship model of a market and so on.

Dynamic Mathematical Model:

- Dynamic mathematical model is the mathematical model that accounts for the time dependent changes in the logical state of the system.

- Such models are time-variant.
- It is generally represented by differential equations or difference equations.
- Egg: The equation of motion of planets around the sun in the solar system.

Examples: Model

Scale Model of a Ship (Static Physical Model):

- A scale model of a ship is a simplified approach to determine the exact measurements of the components of the ship during its construction.
- It removes the necessity of drawing complex 3D shapes to just analyze the ship model and their measurements.
- The measurements on scale model are later converted into the values that would occur in the real system.

Automobile Suspension Model (Dynamic Physical Model):

- Consider the mechanical system in which a mass is subjected to an applied force F(t) that varies with time, a spring whose force is proportional to extension or contraction and a shock absorber that exerts a damping force proportional to the velocity of the mass.
- This represent the suspension of an automobile wheel when the automobile is assumed to be immobile in a vertical direction.
- The problem can be defined by the differential equation:

$$Mx'' + Dx' + Kx = KF(t) \rightarrow 1$$

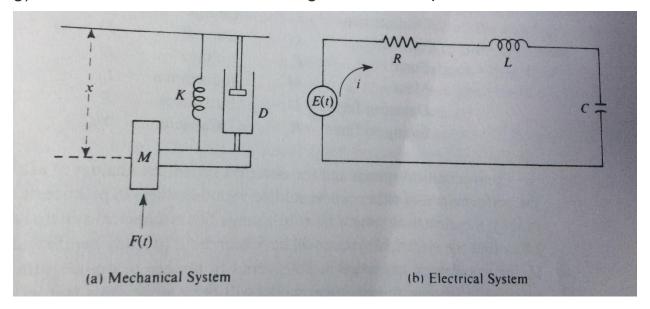
where, M = mass, K = stiffness of the spring and D = damping factor of shock absorber.

- Analyzing this system is very much difficult and expensive. So, it can be modeled with the electrical system as shown:
- The electrical circuit consists of an inductance L, a resistance R and a capacitance C, connected in series with a voltage source that varies with time according to function E(t). If q is the charge in capacitance, it

can be defined as:

Lq" + Rq' + q/C = E(t)/C
$$\rightarrow$$
 2

- Equation 1 and 2 are similar in behavior, so the mechanical system can be simulated using an electrical system with more efficiency and with low cost.
- The equivalence between the two systems are:
- a) Displacement x = Charge q
- b) Velocity x' = Current I, q'
- c) Force F = Voltage E
- d) Mass M = Inductance L
- e) Damping Factor D = Resistance R
- f) Spring stiffness K = Inverse of Capacitance 1/C
- g) Acceleration x" = Rate of change of current q"



Market Model – Supply and Demand (Static Mathematical Model):

- Consider a market model assuming the linear behavior of supply and demand with price.
- Demand for a product will be low when the price is high and will be high when the price is low.
- Supply for a product will be high when the price is high and will be low when the price is low.
- If conditions remain stable, the price will settle to the point at which

demand equals supply.

- The relations can be stated mathematically as:

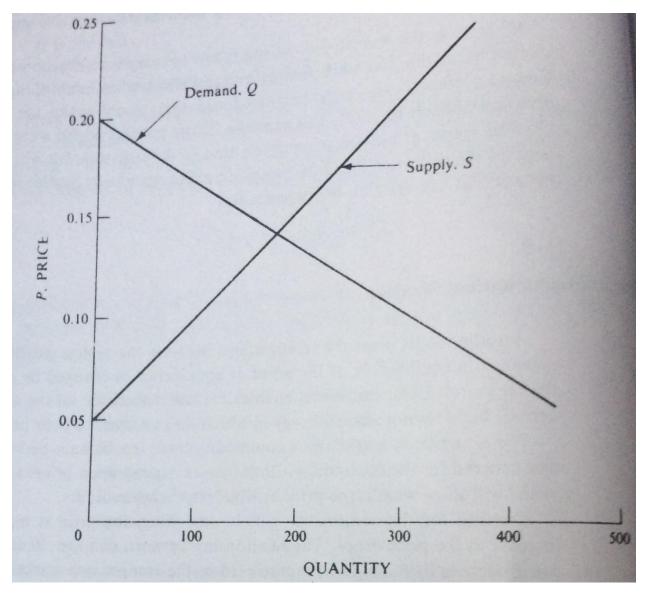
Q = a - bP

S = c + dP

S = Q

where, S = supply, Q = demand and P = price

- The linear market model is shown below:



Physical and Mathematical models

Physical Model

- Physical model is the smaller or larger physical copy of an object being modeled.
- The geometry of model is just the rescaled view of the object it represents.
- The horizontal and vertical scaling may not be balanced as per the requirements of the model.
- The physical model helps in visualization of the object taken into consideration in an effective way.
- It is also used to solve equations with the particular boundary conditions.

Static Physical Model:

- Static physical model is the physical model which describes relationships that do not change with respect to time.
- Such models only depict the object's characteristics at any instance of time, considering that the object's property will not change over time.
- Egg: An architectural model of a house, scale model of a ship and so on.

Dynamic Physical Model:

- Dynamic physical model is the physical model which describes the time varying relationships of the object properties.
- Such models describes the characteristics of the object that changes over time.
- It rely upon the analogy between the system being studied and some other system of a different nature, but have similarity on forces that directs the behavior of the both systems.
- Egg: A model of wind tunnel, a model of automobile suspension and so on.

Mathematical Model

- Mathematical model is the model which is composed of a symbols and logic.
- Such model is also called an abstract model.
- It describes the system using mathematical concepts.
- The mathematical model is used to explain the system and to study the effects of different components, and to make predictions about the behavior of the system.
- It generally consists of following major elements:
- a) Governing equations
- b) Defining equations
- c) Constitutive equations
- d) Constraints

Static Mathematical Model:

- Static mathematical model is the mathematical model that represents the logical view of the system in equilibrium state.
- Such models are time-invariant.
- It is generally represented by the basic algebraic equations.
- Egg: An equation relating the length and weight on each side of a playground variation, supply and demand relationship model of a market and so on.

Dynamic Mathematical Model:

- Dynamic mathematical model is the mathematical model that accounts for the time dependent changes in the logical state of the system.
- Such models are time-variant.
- It is generally represented by differential equations or difference equations.
- Egg: The equation of motion of planets around the sun in the solar system.

Examples: Model

Scale Model of a Ship (Static Physical Model):

- A scale model of a ship is a simplified approach to determine the exact measurements of the components of the ship during its construction.
- It removes the necessity of drawing complex 3D shapes to just analyze the ship model and their measurements.
- The measurements on scale model are later converted into the values that would occur in the real system.

Automobile Suspension Model (Dynamic Physical Model):

- Consider the mechanical system in which a mass is subjected too an applied force F(t) that varies with time, a spring whose force is proportional to extension or contraction and a shock absorber that exerts a damping force proportional to the velocity of the mass.
- This represent the suspension of an automobile wheel when the automobile is assumed to be immobile in a vertical direction.
- The problem can be defined by the differential equation:

$$Mx'' + Dx' + Kx = KF(t) \rightarrow 1$$

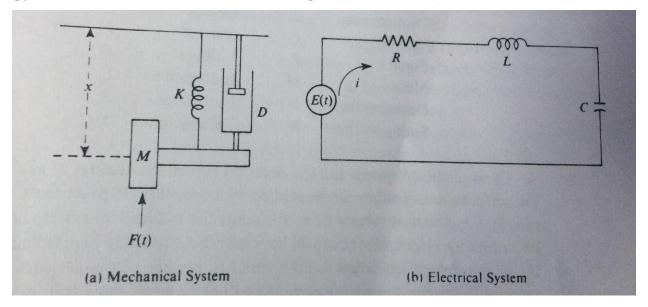
where, M = mass, K = stiffness of the spring and D = damping factor of shock absorber.

- Analyzing this system is very much difficult and expensive. So, it can be modeled with the electrical system as shown:
- The electrical circuit consists of an inductance L, a resistance R and a capacitance C, connected in series with a voltage source that varies with time according to function E(t). If q is the charge in capacitance, it can be defined as:

Lq" + Rq' +
$$q/C = E(t)/C \rightarrow 2$$

- Equation 1 and 2 are similar in behavior, so the mechanical system can be simulated using an electrical system with more efficiency and with low cost.
- The equivalence between the two systems are:

- a) Displacement x = Charge q
- b) Velocity x' = Current I, q'
- c) Force F = Voltage E
- d) Mass M = Inductance L
- e) Damping Factor D = Resistance R
- f) Spring stiffness K = Inverse of Capacitance 1/C
- g) Acceleration x" = Rate of change of current q"



Market Model – Supply and Demand (Static Mathematical Model):

- Consider a market model assuming the linear behavior of supply and demand with price.
- Demand for a product will be low when the price is high and will be high when the price is low.
- Supply for a product will be high when the price is high and will be low when the price is low.
- If conditions remain stable, the price will settle to the point at which demand equals supply.
- The relations can be stated mathematically as:

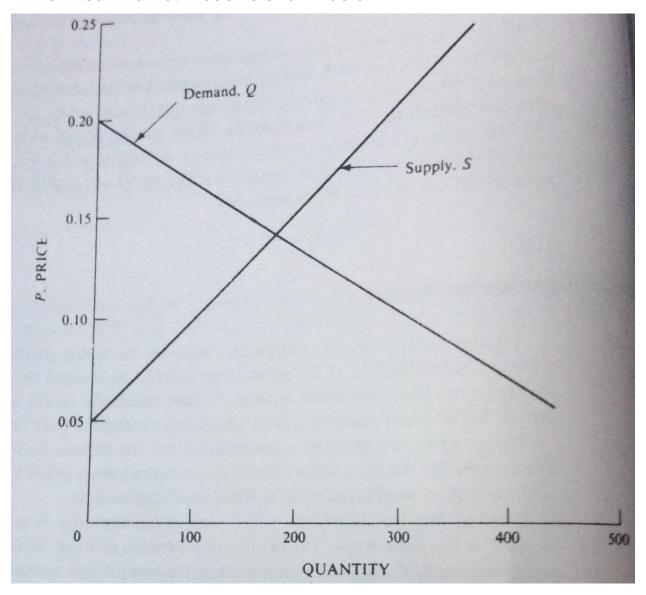
$$Q = a - bP$$

$$S = c + dP$$

$$S = Q$$

where, S = supply, Q = demand and P = price

- The linear market model is shown below:



Automobile Suspension (Dynamic Mathematical Model):

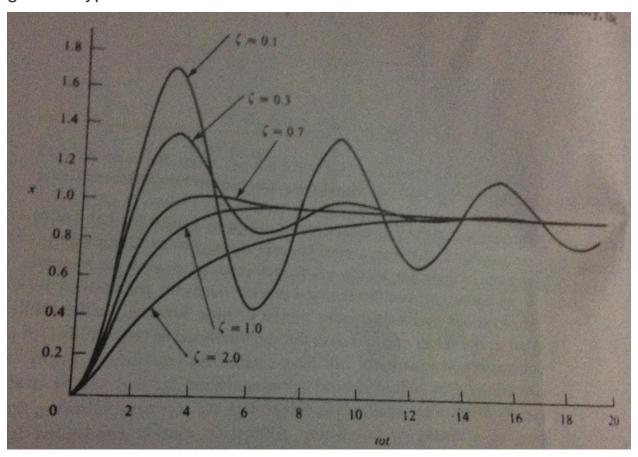
- The automobile suspension equation is written as:

 $x'' + 2dw x' + w^2 x = w^2 F(t)$

where 2dw = D/M and $w^2 = K/M$; d = damping ratio, w = frequency of oscillation

- The solution for several values of d in terms of the variable wt is shown below:

- When d is less than 1, the motion is oscillatory and w = 2 * pi * f
- The relationship guides to select the spring and shock absorber to get that type of motion.



Chapter-3

Continuous system simulation

<u>Differential and Partial Differential</u> Equations

- The equation that consists of the higher order derivatives of the dependent variable is known as differential equations.
- The differential equation is said to be linear if any of the dependent variables and its derivatives have power of one and are multiplied by the constant.

Egg: M x'' + D x' + K x = K F(t)

where, M, D and K are constants; F(t) is the input to the system depending upon the independent variable t; x" and x' are second and first order derivatives of dependent variable x.

- The differential equation is said to be non-linear if the dependent variable or any of its derivatives are raised to a power or are combined in other way like multiplication.
- The differential equation is said to be partial if more than one independent variable occurs in a differential equation.
- Egg: Equation of flow of heat in three-dimensional body. It consists of four independent variables (three dimensions and time) and one dependent variable (temperature).

Necessity of differential equations:

- 1. Most physical and chemical process occurring in the nature involves rate of change, which requires differential equations to provide mathematical model.
- 2. It can be used to understand general effects of growth trends as differential equations can represent a growth rate.

Continuous System Models

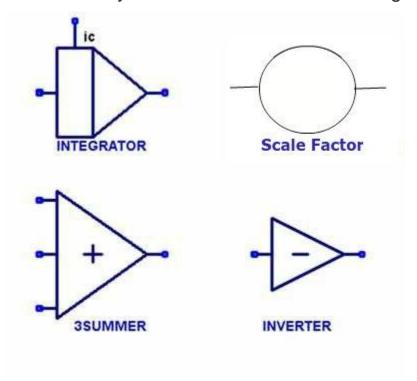
- A continuous system is the system in which the activities of the main elements of the system cause smooth changes in the attributes of the entities of the system.
- On mathematical modeling, the attributes of the system are controlled by a continuous functions.
- In such system, the relationships depicts the rates at which the attributes changes.
- The continuous system is modeled using the differential equations.
- The complex continuous system with non-linearity can be simulated by showing the application to models for linear differential equations to obtain constant coefficients and then generalize to more complex equations.

Analog Computers

- Analog computers are those computers that are unified with devices like adder and integral so as to simulate the continuous mathematical model of the system, which generates continuous outputs.
- The electronic analog computer based on the use of high gain dc amplifiers are widely used analog computer.
- In such analog computer, voltages are equated to mathematical variables and the op amps can add and integrate the voltages.
- The proper configurations can handle addition of several input voltages each representing the input variables.
- The coefficient of the model equations are obtained by using the scale factors.
- The circuit can be arranged to produce integrator that provides integral w.r.t time of a single input voltage or sum of input voltages.
- Sign inverter is used to reverse the sign of the input as per the requirement of the model equation.
- The analog computer provides limited accuracy because op amps have many assumptions which can never be true in reality.

Analog Methods

- The general method to apply analog computers for the simulation of continuous system models involves following components:

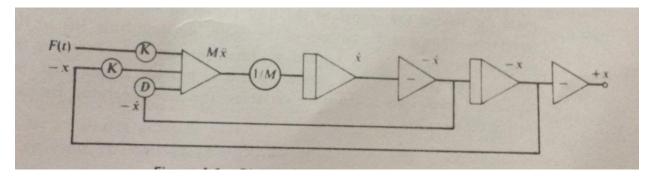


Automobile Suspension Problem: Example

The model is defined by the second order differential equation as: M x'' + D x' + K x = K F(t)

$$=>M x'' = K F(t) - D x' - K x$$

The diagram analog method modeling is shown below:



Hybrid Simulation

- In reality, the system is of neither a pure continuous nor a pure discrete nature.
- For simulating such system, the combination of analog and digital computers is used. Such setup is known as hybrid computers.
- The simulation provided by the hybrid computers is known as hybrid simulation.
- The major difficulty in use of hybrid simulation is that it requires high speed converters to transform signals from analog to digital form and vice versa.

Digital Analog Simulator

- Digital Analog simulators indicates the use of programming languages in digital computer to simulate the continuous system.
- The language is composed of macro-instructions which are able to act as adder, integrator and sign-changer.
- A program is written to link these macro-instructions as per the necessity which are provide virtual connections of op amps in analog computers.

Chapter-4

Queuing system

Elements of Queuing System

Queuing System

- In many real-world scenarios, queue is prevalent. The problem with the queue is that if it is not managed properly, the customers should wait a long time that causes chaos within the customers. To prevent customer from being unsatisfied with the provided service, queuing system is managed. For egg: During cash withdraw in bank, you have to stay in queue and if it is not managed properly then you will surely be disappointed from the bank service even if that bank is one of the finest one in the city.

Elements/Characteristics of Queuing System

- 1. Calling Population:
- The population of potential customers of the service is called calling population.
- It may be finite or infinite.
- The system of the restaurant, bank and so on are considered to be open system i.e. with infinite calling population.
- The system of repairing certain number of machines by a service man is considered to be closed system i.e. with finite calling population.

2. Arrival:

- The way in which the customers enter into the system is called arrival.
- In most of the cases, the arrival of the customer is random, so the inter-arrival between two customers is described by a random distribution of interval known as arrival pattern.

3. Queue:

- Queue represents the number of customers that have entered into the system and are waiting for the service.
- The customer being serviced is not considered to be in queue.
- The two main properties of queue are as follows:
- a) Maximum Size:
- Queue, in practice, is always limited.
- Maximum size represents the maximum number of customers that can accommodate in the queue.
- b) Queue Discipline:
- Queue discipline represents the rules in which the customers are inserted or removed to or from the queue.
- It can be organized in various ways like FIFO, LIFO, Serve In Random Order(SIRO), Priority Queue, etc.

4. Service time:

- Service represents the activity that takes some time and the customers are waiting for.
- Service time represents the time needed to provide service to a customer by a server.
- Service time may be of constant duration or of random duration.

5. Number of Servers:

- Servers represent the entity that provides service to the customer.
- A system may consist of single server or multiple servers.
- A system with multiple servers is able to provide parallel services to the customers.

Types of queuing system

- Channel is the number of parallel servers.
- Phase is the number of servers in sequence.
- Based on numbers of channel and phase, Queuing system is categorized as:
- 1. Single Channel Single Phase
- 2. Multiple Channels Single Phase

- 3. Single Channel Multiple Phases
- 4. Multiple Channels Multiple Phases

Queuing notation (Kendall's Notation)

- Kendall's notation is used for parallel server systems.
- The basic format of this notation is of form: A / B / c / D / N / K
- A represents the inter-arrival time distribution.
- B represents the service time distribution.
- c represents the number of parallel servers.
- D represents the queue discipline.
- N represents the system capacity.
- K represents the size of the calling population.
- The symbols used for A and B are M (exponential or Markov), D (constant or deterministic), EEK (Erlang of order k), G (Arbitrary or general), GI (General Independent) and PH (Phase type).
- If arrival time is Poisson Distribution, then the inter-arrival time is exponential distribution.
- Q. What is the meaning of D/M/1/LIFO/20/510 in queuing system?
- This notation indicates that the system consists of single parallel server with inter-arrival time of constant distribution and service time of exponential distribution.
- It also indicates that the system follows queue of capacity 20 out of 510 calling population and the queue follows Last In First Out queue discipline.
- Q. What is the meaning of M/M/8/15/LIFO in queuing system?
- The system consists of 8 parallel system with inter-arrival and service times exponentially distributed and have queue size of 15 with discipline LIFO.

Measurement of system performance

The various parameters used to measure the performance of queuing system are as follows:

- 1. Average no of customers in the system
- 2. Average no of customers in the queue
- 3. Percentage of time server to be busy
- 4. Percentage of time server to be idle
- 5. Length of queue

Network of queues

- Some system may consists of multiple queues.
- The customers from one queue may be routed to another queue as per necessity.
- Such queue system is known as network of queues.
- For Egg: In a bank with two queues, the customer are routed from queue with large no of customers to queue with less no of customers.

Applications of queuing system

Q. A shop has only one counter. Customers arrive at this counter at random at 4.5 (+ or -) 3.5 mins. Each possible value of inter-arrival time has the same probability of occurrence. The service time may vary from 1 to 6 minutes with the following probabilities. Simulate the arrival and service of six customers and find performance of the system by calculating average waiting time, probability that the customer has to wait, % of server to be idle, average service time, average time between arrivals, average waiting time of waiting customers and average time spent by customer in the system. Use the following sequence for random:

Customer Probability Arrival Time Service Time

1 0.1 913 84

2 0.2 727 10

```
3 0.3 15 74
4 0.25 948 53
5 0.1 309 17
6 0.05 922 79
```

Answer →

Customer Arrival Time = 4.5 (+ or -) 3.5 = 1 to 8 minutes Service Time = 1 to 6 minutes

So, we can get the probability distribution of arrival at different interarrival time is: (table 1)

Time Probability Cumulative Probability Random Digit assignment

1 0.125 0.125 1 - 125

2 0.125 0.250 126 - 250

3 0.125 0.375 251 - 375

4 0.125 0.500 376 - 500

5 0.125 0.625 501 - 625

6 0.125 0.750 626 - 750

7 0.125 0.875 751 - 875

8 0.125 1 875 - 1000

The random probability distribution as given in the question can be stated as: (table 2)

ST P CP RDA

1 0.1 0.1 1-10

2 0.2 0.3 11-30

3 0.3 0.6 31-60

4 0.25 0.85 61-85

5 0.1 0.95 86-95

6 0.05 1 95-100

From the given arrival time and table 1, we get: (table 3)

Customer RD IAT

1 - 0

2 913 8

3 727 6

4 15 1

5 948 8

6 309 3

From the given service time and table 2, we get: (table 4) Customer RD ST

1 84 4

2 10 1

3 74 4

4 53 3

5 17 2

6 79 4

From table 3 and table 4, we can conclude that:

Cust Inter-arrival Arrival Service Start Wait End Time in Queue Idle No Time Time Time Time Time System Length Time

1004004400

2881809104

3614414018405

4 1 15 3 18 3 21 6 1 0

5 8 23 2 23 0 25 2 0 2

6 3 26 4 26 0 30 4 0 1

Now,

- 1. Simulation Run Length = End time of last customer = 30
- 2. Average waiting time = Total Waiting Time / No of Customer = 3 / 6 = 0.5
- 3. P (Customer waits) = Total queue length / No of Customer = 1 / 6 = 0.167
- 4. P (Server idle) = Total idle time / Simulation run length = 12 / 30 = 0.4
- 5. Average Service Time = Total service time / No of Customer = 18 / 6 = 3
- 6. Average Inter-arrival Time = Total IAT / (No of Customer -1) = 26 / 5 = 5.2
- 7. Average waiting time of waiting customer = Total waiting time / Queue length = 3 / 1 = 3
- 8. Average time spent by customer in system = Total time in system / No of Customer = 21 / 6 = 3.5

- Q. In a petrol pump, Customer arrival time is given by Poison distribution within arrival rate of 2 Customer/hr. and they get exponentially served at the rate of 3 Customer/hr. Find:
- a) Probability of zero Customer
- b) Probability of 1 Customer
- c) Probability of 4 or more Customers
- d) Server Busy Time
- e) Server Idle Time
- f) Average no of Customer in system
- g) Average time spent in system
- h) Average waiting time
- I) Average no of customers in queue

We know that: System Utilization (8) = 2/4 = 2/3 = 0.67 <1 1) Probability of zero customer (Po) = 1-3 = 1-0.67 = 0.33 2) Probability of 1 customer (P1) = 3º Po = 0.67 x 0.33 = 0.22 3) Probability of 4 or more customers = 1-Po-P1-P2-P3 = 1-0.33-82p - 82p - 33p = 1-0.33-0.22-0.15-0.10 = 0.2 4) Server Busy Time = 0 3 = 0.67 hour is one hour 5) Server Idle Time = Po = 0.33 hour & and hour 6) Average number of customer in system = 7) Average time spent in system = 8) Average Waiting Time = 3/4-2 = 0.67/(3-2) = 0.67 hour 3) Average no. of customer in queue = 3/1-8 = 0.67/0.33=1.36

Chapter-5

Markov chains

Key features of Markov chains

The key features of Markov chains are as follows:

- 1. The outcome of each experiment is one of the discrete states of the chain.
- 2. The outcome of an experiment depends on only the present state and not on any past states.
- 3. The transition probabilities remain constant from one transition to the next.

Markov process with example

- If the future outcome depends only on the present outcome but not on the past outcomes, then the process is known as Markov process.
- If the states of the Markov process is discrete, then it is known as Markov chain.

Transition Probability Matrix:

- The transition probability matrix is the matrix that shows probabilities of moving from one state to another state from current generation to next generation.
- It is represented by P.
- P^k represents the probabilities of transition from one state to another in k repetitions of the experiment if and only if P is constant for each transition.
- It is a square matrix.
- All entries are between 0 and 1.
- The sum of the entries of any row is equal to 1.

Current Status Distribution Matrix:

- Current status distribution matrix is a row matrix that provides the status of current state of all the discrete states of the Markov chain.
- Each entry must be between 0 and 1 inclusive.
- The sum of entries of each row must be 1.
- It is denoted by Qo.

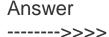
Steps to make predictions:

- 1. Create current status distribution matrix Qo.
- 2. Create probability distribution matrix P.
- 3. Calculate Qo * P ^ n, which represents probability vector after n repetitions of the experiment.

Application of Markov chain

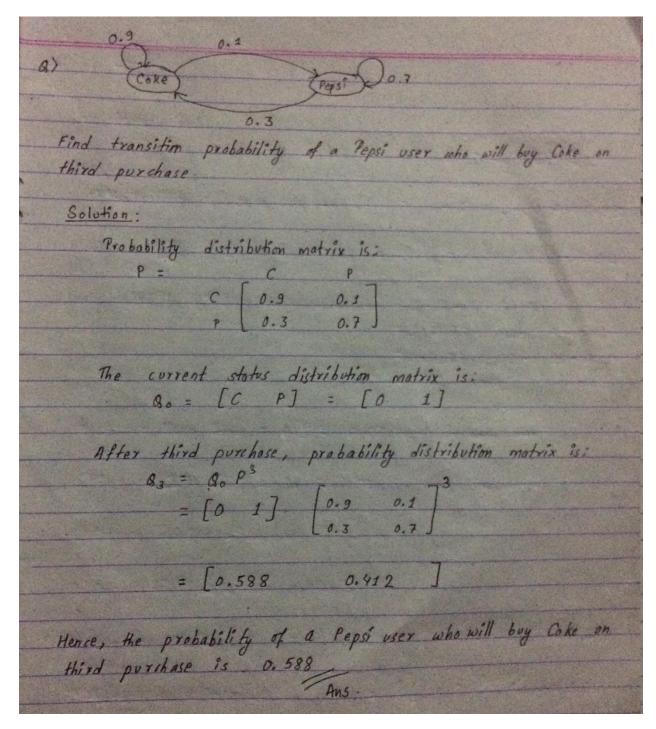
The general applications of Markov chain are as follows:

- 1. Turing System
- 2. Traffic System
- 3. Telephone System
- 4. Weather Forecasting
- Q. Given that chance of a Honda bike user to buy Honda bike at next purchase is 70% and that his next purchase will be Yamaha is 30%. The chance of Yamaha bike user to buy Yamaha bike at next purchase is 80% and that his next purchase will be Honda is 20%. What is the probability to buy Yamaha bike after three purchase of a current Honda Bike user?



Let, H. indicates Honda bike user
Y indicates Yamaha bike user
Then, 0.7
0.3
Then, 0.7 0.3 0.8 0.2
0-2
The probability distribution matrix is:
e H y
$P = H \left[0.7 0.3 \right]$
y 0.2 0.8
The current status distribution matrix is:
$B_0 = \begin{bmatrix} H \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$
y 0
, v 1 like offer three nurchase is:
The probability of or P3
The probability to buy yamaha onto after the probability the probability the probability to buy yamaha onto after the p
1 2 /0.8 / 0]
40.2 /0.3 1/
$Q_3 = Q_0 P^3 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.7 & 0.3 \end{bmatrix}^3$
B3 = 00 T - 1 - 0.2 0.8 J
- [0.475 0.525]
- 0.775
La contraction of the contractio
1/0/0/ 1/0/5
Hence, the required probability is 0.525 Ans.

Question



Chapter-6

Random Number

Properties of Random Numbers

What is random number?

A number chosen from some specified distribution randomly such that selection of large set of these numbers reproduces the underlying distribution is called random number.

General Properties

- 1. Uniformity:
- The random numbers generated should be uniform. That means a sequence of random numbers should be equally probable everywhere.
- If we divide all the set of random numbers into several numbers of class interval then number of samples in each class should be same.
- If 'N' number of random numbers are divided into 'K' class interval, then expected number of samples in each class should be equal to ei = N / K.
- 2. Independent:
- Each random number should be independent samples drawn from a continuous uniform distribution between 0 and 1.
- The probability density function is given by:

$$f(x) = 1, 0 <= x <= 1$$

= 0, otherwise

- 3. Maximum Density:
- The large samples of random number should be generated in a given range.
- 4. Maximum Cycle:
- It states that the repetition of numbers should be allowed only after a large interval of time.

Generation of Pseudo-Random numbers

Pseudo Random Number

- Pseudo random numbers are the random numbers that are generated by using some known methods so as to produce a sequence of numbers in [0,1] that can simulates the ideal properties of random numbers.
- Pseudo random numbers are not completely random as the set of random numbers can be replicated because of use of some known method.

Problems Associated

- The problems associated with pseudo random numbers are as follows:
- 1. The generated numbers might not be uniformly distributed.
- 2. The generated numbers might be discrete valued instead of continuous valued.
- 3. The mean of the generated numbers might be too high or too low.
- 4. The variance of the generated numbers might be too high or too low
- 5. There might be presence of correlation between the generated numbers.

Considerations

- The important considerations that should be made while generating pseudo random numbers are as follows:
- 1. The method used to generate random number should be fast because the simulation problem requires a large set of random numbers which can increase time complexity of the system.
- 2. The method used should be portable to different platform and programming languages so as to generate same results wherever it is

executed.

- 3. The method should have long cycle.
- 4. The random numbers should be replicable. It means that the same set of random numbers should be generated with same starting point.
- 5. The generated random numbers should approximate the uniformity and independence properties.

Random Number generation methods

Linear Congruential Method:

- A sequence of integers X1, X2, X3, are produced between zero and m-1 by using the recursive relation as follows:

```
X(i+1) = (a X(i) + c) \mod m, for I = 0, 1, 2, 3, 4, \dots (Equation 1)
```

- The initial random integer X(0), is known as seed, a is called multiplier, c is increment and m is the modulus.
- The form is called mixed congruential method if c is not equal to 0 in equation 1.
- The form is called multiplicative congruential method if c is equal to 0 in equation 1.
- The random numbers corresponding to each random integer can be obtained as:

```
R(i) = X(i) / m, for I = 0, 1, 2, 3, ...
```

- Example: For the values selection with X(0) = 30, a = 12, c = 21 and m = 100, the sequence of random numbers generated are as follows: X(0) = 30

```
X(1) = (12 * 30 + 21) \mod 100 = 381 \mod 100 = 81
```

$$X(2) = (12 * 81 + 21) \mod 100 = 993 \mod 100 = 93$$

$$X(3) = (12 * 93 + 21) \mod 100 = 1137 \mod 100 = 37$$

and so on...

The random numbers between [0, 1] generated are as follows:

$$R(0) = 0.3$$

$$R(1) = 0.81$$

$$R(2) = 0.93$$

R(3) = 0.37 and so on.....

Combined Linear Congruential Method:

- Combined linear congruential method uses the combination of two or more multiplicative congruential generators so as to provide good statistical properties and a longer period.
- Let X(i, 1), X(i, 2), X(i, 3), are the ith output from k different multiplicative congruential generators, then the combined generator is of form:
- $X(i) = Summation from j = 1 to k [(-1)^(j-1) * X(i,j)] mod m(j) 1$ This generates random integers between 0 and m(j)-2.
- The random numbers are calculated as:

$$R(i) = X(i) / m(j)$$
, if $X(i) > 0$
= $(m(j) - 1) / m(j)$, if $X(i) = 0$

Test for random numbers

Kolmogorov-Smirnov Test:

- This test compares the continuous cdf, F(x), of the uniform distribution with the empirical cdf, S(x), of the sample of N observations.
- Mathematically,

$$F(x) = x , 0 \le x \le 1$$

For samples from random generator be R(1), R(2),, R(N), then empirical cdf is given by:

- It is based on largest absolute deviation between F(x) and S(x) over the range of random variable. It can be given by:

 $D = \max | F(x) - S(x) |$

- The sampling distribution of D is tabulated as a function of N which is standard for comparison purpose.

Algorithm:

- 1. Rank the data from smallest to largest such that $R(1) \le R(2)$ $\le R(N)$
- 2. Compute:

D+ = max [i/N - R(i)] for i = 1 to N

D- = max [R(i) - (i - 1) / N] for i = 1 to N

- 3. Compute D = max(D+, D-)
- 4. Locate in table of sampling distribution of D, the critical value D(alpha), for specified significance level alpha and given sample size N.
- 5. If the sample statistic D is greater than D(alpha), the null hypothesis that the data are a sample from a uniform distribution is rejected. Else, no difference has been detected and the distribution is uniform.

Chi-Square Test:

- The chi-square test uses sample statistic : chi-square = Summation i = 1 to $n [(O(i) E(i))^2 / E(i)]$
- O(i) = Observed number in the ith class
- E(i) = Expected number in the ith class
- n = Number of class
- Degree of freedom = n 1
- If chi-square for sample random numbers is less than standard chi-square at alpha and degree of freedom(n-1), then the null hypothesis is not rejected.

Chapter-7

Verification and validation of simulation models

<u>Verification and validation</u> <u>Model Building:</u>

- The real system and their interactions among various components should be analyzed.
- The domain knowledge can also be acquired from the interaction with concerned people.
- As the model development proceeds, new questions arises and the process of learning system behaviour and structure takes place again.
- Then a conceptual model is constructed with a collection of assumptions and hypotheses.
- Finally, the conceptual model is implemented using various simulation software incorporating assumptions into the worldview.
- Hence, model building is an iterative process of domain knowledge acquisition and model development.

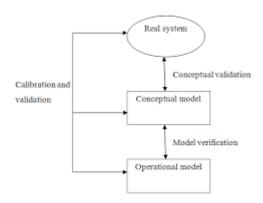
Verification:

- Verification is concerned with building the model correctly.
- It is performed by comparing the conceptual model to the computerized simulation implementation.
- It tests whether the given model is implemented correctly in the simulation software.

Validation:

- Validation is concerned with building the correct model.
- It is performed by calibration of the model.

- Calibration is the iterative process of comparing the model to the actual system behavior.
- It attempts to confirm that a model represents the real system accurately.



Verification of simulation models

Considerations to be used in Verification Process:

- 1. The operational model should be checked by someone other than the developer, preferably an expert in the simulation software being used.
- 2. Make a flow diagram that contains all logically possible action a system can perform when an event occurs.
- 3. Closely examine the model output for reasonableness under variety of input parameters.
- 4. Have the operational model print input parameters when the simulation ends to check if they are not altered.
- 5. Make the operational model as self-documenting as possible.
- 6. Verify animated operational model imitates the actual system.
- 7. Debugger should be used during simulation model building.
- 8. Graphical interfaces are recommended.

Calibration and validation of models

Naylor and Finger Simulation Model Validation:

Naylor and Finger proposed three step approach for validation process. These steps are as follows:

- 1. Face Validity
- 2. Validate Model Assumptions
- 3. Input Output Transformation

Face Validity:

- A model should appear reasonable on its face to model users and to those who knows about the real system that is being simulated.
- A model should be designed with high degree of realism regarding system structure and behavior through reliable data.
- The potential users should also be involved in the validation process to aid in identification of model deficiencies and optimizing those deficiencies to produce better model. This process is termed as structural walkthrough.
- Sensitivity analysis is also used for face validity of the model. It analyses the effect on output when there is change in input parameters.
- Sensitivity analysis is done through appropriate statistical techniques.

Validate Model Assumptions:

- There are many assumptions that are made during model building process.
- These assumptions are of two types: structural assumption and data assumption.
- Structural assumptions involves simplification and abstraction of reality.
- Data assumptions should be done based on collection of reliable data.
- For example Consider a bank system in which customer are queued before providing service.
- The structural assumptions may be Customers form a single queue

which is served by multiple tellers - Customers form one line for each teller - The number of teller should be fixed or variable.

- These structural assumptions should be validated by actual observation during appropriate time periods and also by discussions with the managers and tellers.
- The data assumptions may be interarrival times of customers during peak hour - interarrival times of customers during slack period service time for personal accounts and so on.
- These data assumptions should be validated by consultation with bank managers. The validation is done by using goodness-of-fits tests such as chi-square test or Kolmogorov-Smirnov tests.

Input-Output Transformations:

- It involves validating whether the model can predict the future behavior of the real system when the model input data match the real inputs and when a policy implemented in the model is implemented at some point in the system.
- In this validation phase, the model accepts values of the input parameters and transforms them into output measures of performance.
- The modeler uses the historical data reserved for validation purposes only.
- The main responses of interest should be used as the primary criteria for model validation.
- If in future, the model is used for different purpose, it should be revalidated in terms of new response of interest.
- For the complete input-output validation, at least one set of input conditions should be collected from the system data so as to compare to model prediction.

Iterative Calibration for Validation:

- Iterative calibration means to validate the model with the real system, look out for the places for betterment of the models and revising the

model to form next better model repeatedly until a satisfiable model is not achieved.

- The initial model is developed and is calibrated using Naylor-Finger calibration steps with the real system. It is then revised and a first revision model is generated.
- The first revision model is then calibrated with the real system. It is revised to form a second revision model.
- This process is continued until the model becomes acceptable.

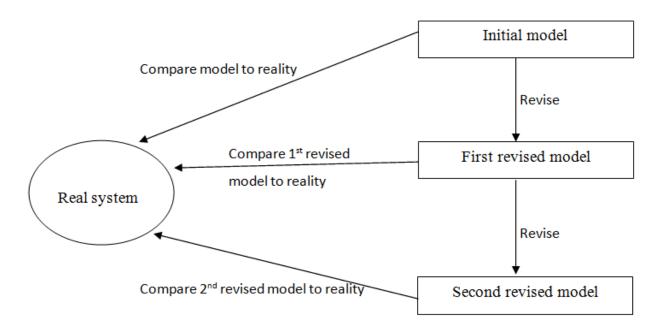


Fig: Iterative process of calibration and validation

Chapter-8

Analysis of simulation output

Estimation methods

Simulation Output: Introduction

- Whenever a random variable is introduced to the simulation model, all the system variables that describe its behavior become random or stochastic.
- The values of the variables involved in the system will fluctuate as the simulation proceeds.
- So, arbitrary measurement of the values of these variables cannot represent the true value.
- For this, some conditions about the probability of the true value falling within a given interval about the estimated value must be made. Such interval is known as confidence interval.
- In simulation study, it is assumed that the observations being made are mutually independent. But, in most of the real-world problems, simulation results are mutually dependent.
- The various methods used to analyze simulation results are as follows:
- 1. Estimation Methods
- 2. Simulation Run Statistics
- 3. Replication of Runs
- 4. Elimination of Initial Bias

Estimation Method

- It is assumed that the random variables are stationary and independent drawn from an infinite population with a finite mean and finite variance.
- Such random variables are independently and identically distributed (IID) random variable.
- The central limit theorem can be applied to IID data. It states that

"the sum of n numbers of IID variables, drawn from a population that has a mean of μ and a variance of σ^2 , is approximately distributed as a normal variable with a mean of $n\mu$ and a variance of $n\sigma^2$."

- The normal distribution can be transformed into a standard normal distribution, that has a mean of 0 and a variance of 1.

Example:

Let us consider x(i) where $i = 1, 2, 3, \dots, n$ be the n number of random variables drawn from a sample of population with mean μ and variance σ^2 .

Using central limit theorem, and transforming to standard normal distribution, we get:

$$Z = (\sum x(i) - n\mu) / \{n^{(1/2)} * \sigma\}$$

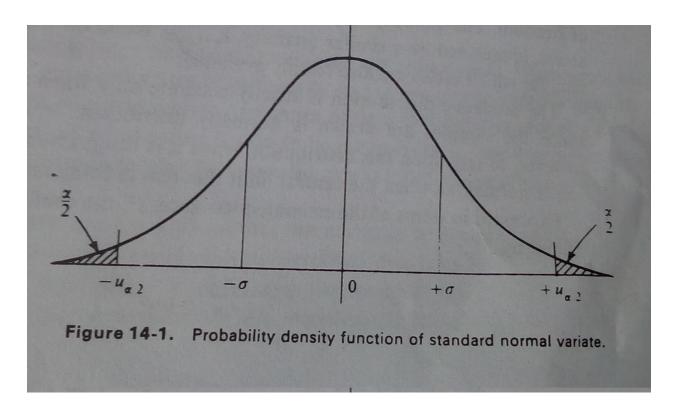
Dividing top and bottom by n, we get:

$$Z = (x - \mu) / (\sigma / n^{(1/2)})$$

Where $x = \sum x(i) / n =$ sample mean

Confidence Interval Estimation

- The sample mean acts as a consistent estimator for the mean of population from which the sample is drawn; which is also a random variable.
- As sample mean is random, confidence interval about its computed value should be established.
- The probability density function of the standard normal variate is shown in given figure:



- For some constant α , known as level of significance, the probability that z lies between -u(α /2) and u(α /2) is given by:

P
$$(-u(\alpha/2) \le z \le u(\alpha/2)) = 1 - \alpha$$

Putting the value of z from point estimation, we get:

$$\begin{array}{l} P(\ -\ u(\alpha/2) <= (\ x\ -\ \mu)\ /\ (\sigma\ /\ n^{(1\!/2)}) <= u(\alpha/2)\) = 1\ -\ \alpha \\ P(\ -\ u(\alpha/2)\ ^*\ (\sigma\ /\ n^{(1\!/2)}) <= (\ x\ -\ \mu) <= u(\alpha/2)\ ^*\ (\sigma\ /\ n^{(1\!/2)})\) = 1\ -\ \alpha \\ P(\ u(\alpha/2)\ ^*\ (\sigma\ /\ n^{(1\!/2)}) + x >= (\mu) >= -\ u(\alpha/2)\ ^*\ (\sigma\ /\ n^{(1\!/2)}) + x \) = 1\ -\ \alpha \end{array}$$

Hence, the confidence interval is given by:

$$X (+ or -) u(\alpha/2) * (\sigma / n^{(1/2)})$$

Simulation run statistics

- In most of the simulation study, the assumptions of stationary and mutually independent observations do not apply. An example of such case is queuing system.

- Correlation is necessary to analyze such scenario.
- In such cases, simulation run statistics method is used.

Example:

- Consider a system with Kendall's notation M/M/1/FIFO and the objective is to measure the mean waiting time.
- In simulation run approach, the mean waiting time is estimated by accumulating the waiting time of n successive entities and then it is divided by n. This measures the sample mean such that:

$$x = \sum x(i) / n$$
 for $i = 1$ to n

- Such series of data in which one value affect other values is said to be autocorrelated.
- The sample mean of autocorrelated data can be shown to approximate a normal distribution as the sample size increases.

Problem:

- The distribution may not be stationary.
- A simulation run is started with the system in some initial idle state.
- In this case, the early arrivals will obtain service quickly deviating from normal distribution.
- Hence, the sample means of the early arrivals is known as initial bias.
- As the sample size increases and the length of run is long, the effect of bias dies and the normal distribution is again established.

Replication of runs

- This approach is used to obtain independent results by repeating the simulation.
- Repeating the experiment with different random numbers for the same sample size n gives a set of independent determinations of the

sample mean x(n).

- The mean of the means and the mean of the variances are then used to estimate the confidence interval.
- Consider that the experiment is repeated p times over the n random variables.
- Let x(i,j) represents jth random variables on ith simulation run.
- We can write:

$$x = \sum x(j) / p$$
 for $j = 1$ to p
Where,
 $x(j) = \sum x(j,i) / n$ for $i = 1$ to n

- The variance can be given by:

$$d^2 = \sum d(j)^2 / p \text{ for } j = 1 \text{ to } p$$

Where,
 $d(j)^2 = \sum \{x(j,i) - x(j)\} / (n-1) \text{ for } i = 1 \text{ to } n$

Elimination of Initial bias

- The ways to eliminate the initial bias are as follows:
- 1. Ignore the initial bias occurred during the simulation run.
- 2. The system should be started in a representative state than in the empty state.
- 3. Start the simulation in the empty state, then stop after initial bias and then start again.
- 4. Run the simulation for such a long period of time so that the initial bias has no any significance in the output result.

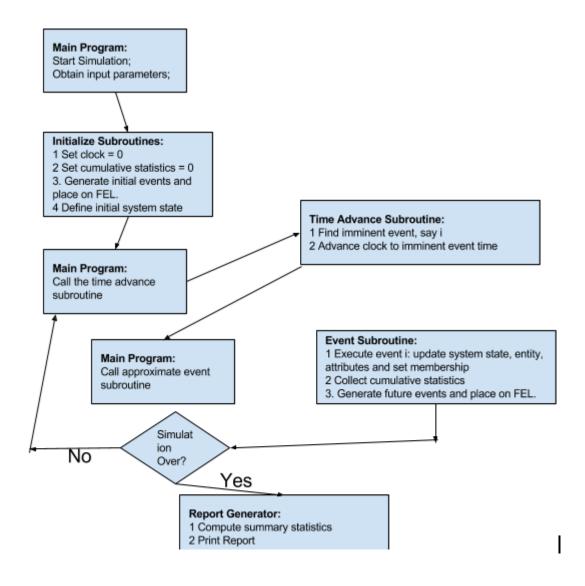
Chapter-9

Simulation software

Simulation in Java

- JAVA is a widely used programming language. But it does not provide any modules directly aimed for simulation system.
- The components that all the simulation models written in JAVA are as follows:
- 1. Clock: It is a variable that defines the simulated time.
- 2. Initialization Method: It is a method to define the system state at initial time.
- 3. Min-time event method: It is a method that identifies the imminent (about to happen) event.
- 4. Event method: It is a method for each event that update the system state when it occurs.
- 5. Random variate generator: It is a method to generate random samples from the desired probability distributions.
- 6. Main program: It is the core of the simulation system that controls the overall event scheduling algorithms.
- 7. Report generator: It is a method that summarizes the collected statistics to give the report at the end of the simulation.

The time advance flowchart of the simulation models in JAVA can be depicted in the given figure:



Explanation

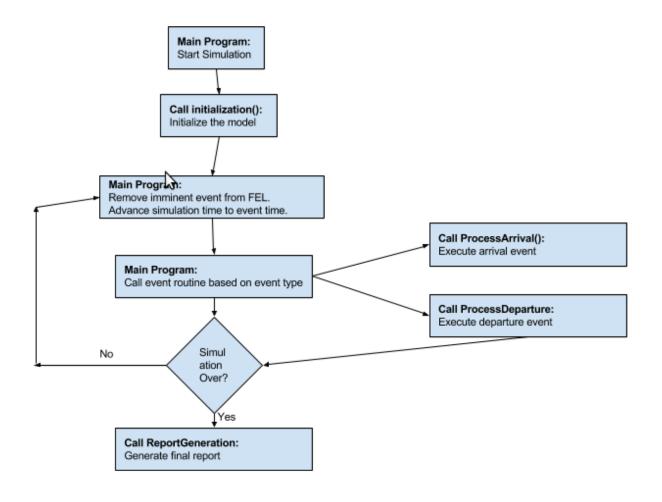
- Simulation begins by setting clock to zero, initializing cumulative statistic to zero, generating any initial events and placing them in the Future Event List (FEL).
- The simulation program then cycles repeatedly passing the current least time event to approximate event methods until the simulation is over.
- At each step, clock is advanced to the time of the imminent event after finding the imminent event but before calling the event method.
- The appropriate event method is called to execute imminent event, update cumulative statistics and generate future events.

- All actions in an event method takes place at one instant of the simulated time.
- When the simulation is over, the control passes to the report generator.

Example: Single Server Queue Simulation

Consider a grocery checkout counter. The simulation will run until 1000 customers have been serviced. It is assumed that the interarrival times are exponentially distributed with mean 4.5 mins and the service times are normally distributed with mean of 3.2 mins and standard deviation of 0.6 mins. When cashier is busy, a queue forms with no customers turned away.

- Class Event represents an event that stores code for arrival or departure and the event time stamp.
- It consists of associated methods for creating an event and accessing its data.
- It also has method compare To that compares the event with another and reports whether the first event should be considered smaller, equal or greater than the argument event.



Variables Used:

Queue Length Number Inservice Customers FutureEventList

MeanInterArrivalTime (4.5 mins) MeanServiceTime (3.2 mins) Sigma (0.6 mins) TotalCustomers (1000)

Clock LastEventTime Total Busy MaxQueueLength SumResponseTime NumberOfDepartures LongService

RHO = BusyTime / Clock AVGR PC4

Functions Used:

exponential(mu) normal(xmu, Sigma)

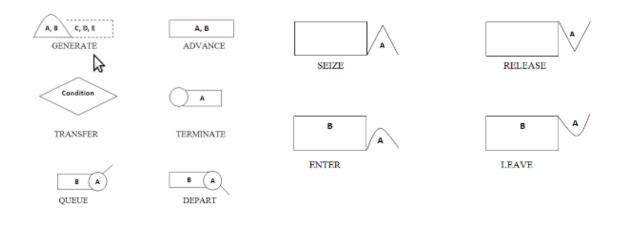
Methods Used:

Initialization()
ProcessArrival()
ProcessDeparture()
ReportGeneration()

Simulation in GPSS

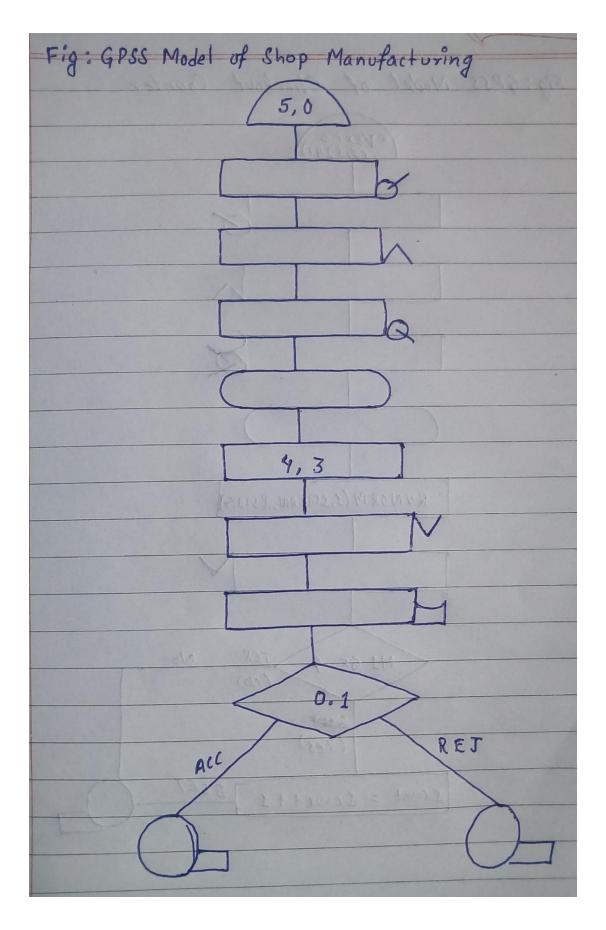
- GPSS (General Purpose Simulation System) is a highly structured and special purpose simulation language based on process interaction approach and oriented toward queuing systems.
- The system being simulated is described by the block diagram using various GPSS blocks.
- Each block represents events, delays or other actions that affect the transaction flow.
- GPSS model is developed by converting the block diagram into block statements and adding the control statements.

The general blocks used in GPSS block diagram is shown in figure below:



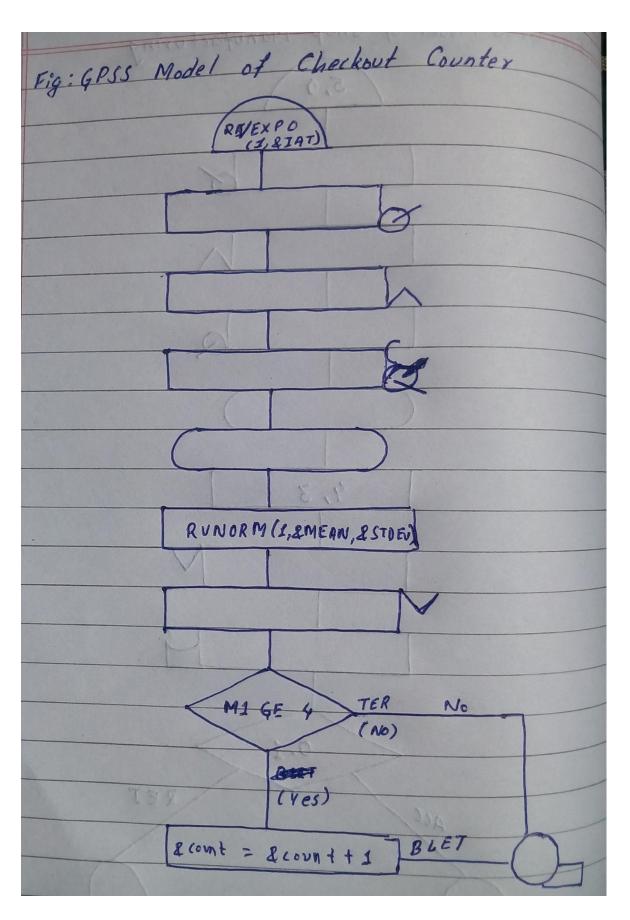
Example 1 - Manufacturing Shop Model Simulation

A machine tool is turning out parse at a rate of 1 per every 5 minutes. As they are finished, the parse goes to an inspector, who takes 4 (+ or -) 3 minutes to examine each one and rejects about 10 % of the parse. Simulate the system using GPSS model.



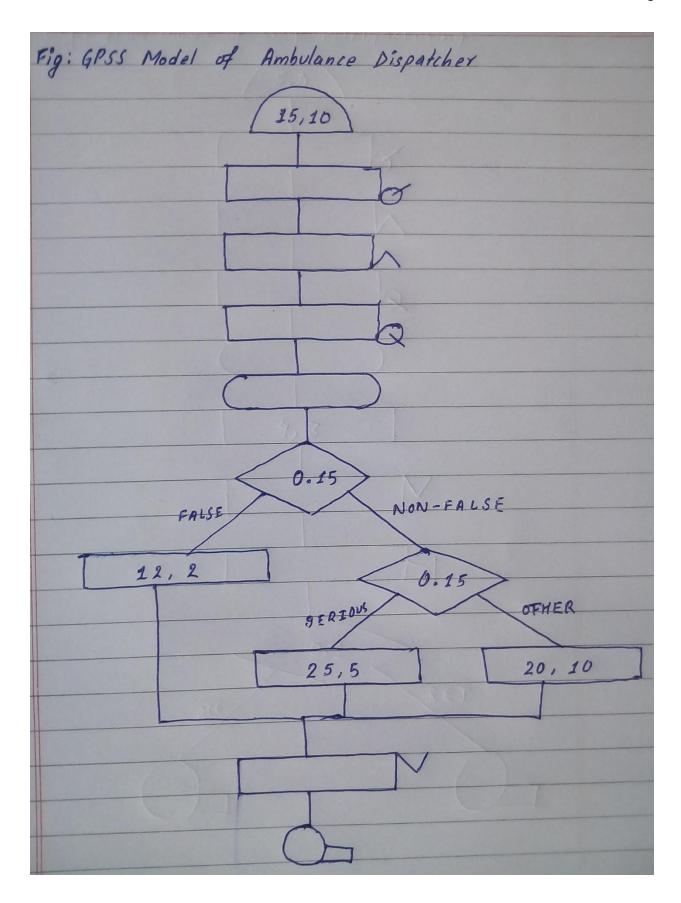
GENERATE 5 0
QUEUE
SEIZE
DEPART
MARK
ADVANCE 4 3
RELEASE
TABULATE
TRANSFER 0.1 ACC REJ
TERMINATE

Example 2 - Single Server Queue System (Checkout Counter)



GENERATE RVEXPO(1, &IAT)
QUEUE
SEIZE
DEPART
MARK
ADVANCE RVNORM(1, &MEAN, &STDEV)
RELEASE
TEST GE M1, 4, TER
BLET &COUNT = &COUNT + 1
TER TERMINATE

Example 3 - Ambulances are dispatched at a rate of one every 15 (+ or -) 10 mins. Fifteen percent of the calls are false alarms, which require 12 (+ or -) 2 mins to complete. All other calls can be one of two kinds. The first kind are classified as serious. They constitute 15% of non-false alarms and take 25 (+ or -) 5 mins to complete. The other calls take 20 (+ or -) 10 mins. Simulate the model using GPSS.



GENERATE 15, 10
QUEUE
SEIZE
DEPART
MARK
TRANSFER 0.15 FALSE NON-FALSE
FALSE ADVANCE 12, 2
NON-FALSE TRANSFER 0.15 SERIOUS OTHER
SERIOUS ADVANCE 25, 5
OTHER ADVANCE 20, 10
RELEASE
TERMINATE

Simulation in SSF

- SSF stands for Scalable Simulation Framework.
- SSF provides a single, unified interface for discrete-event simulation.
- It is an API that describes a set of capabilities for object oriented and process view simulation available for C++ and JAVA.
- It is designed to achieve high performance.
- It defines five base classes; Event, Entity, in Channel, out Channel and process.
- Process implements thread of control in which action method contains the execution body.
- Entity describes simulation objects.
- in Channel and out Channel describes the communication endpoints.
- Event defines messages sent between entities.

SSF Model

1. Starting the Simulation: A simulation starts when any entity's start All () method is called, commonly from the main () routine. The caller

- of start All () specifies two timestamps: the simulation's start time, which defaults to 0, and the end time.
- 2. Initialization: After startAll () has been called, but before any process has started executing, the framework calls the init () routines of all processes and entities. The Entity method now () shall return the simulation start time for the duration of initialization.
- 3. Process Execution: After all processes have been initialized, they become eligible to run at the start time; their actual execution is controlled by the framework, which executes the process's action () callback method. Every time the action () method returns to the caller, the framework executes it again immediately, as many times are necessary to reach the end of the simulation.
- 4. Framework Inner Loop: The SSF framework provides nonprimitive scheduling of processes. Once resumed by the framework, in order to allow the simulation clock to advance, processes must eventually suspend themselves by issuing wait on () or wait for () calls. Within the framework, simulation time advances at a (possibly variable) rate determined by the arrival of events on channels, and the duration of wait for () statements. When simulation time exceeds the end, time specified in the original start All () call, the simulation ends.
- 5. Start, Pause, Resume and Join: The start All (), pause All (), resume All (), and join All () methods may not be called from within process code. The pause All () method allows the simulation to be paused gracefully after some implementation dependent delay. A call to pause All () blocks until the simulation has paused, and then returns the current simulation time. After pause All (), a call to resume All () resumes the simulation's forward progress. At any point, a call to join All () (e.g., from main ()) will block without possibility of pause or resumption until simulation execution is complete.
- 6. Framework Concurrency: Each entity is said to be aligned to some object. Processes that are eligible to execute at each instant of simulation time will be scheduled by the framework: sequentially within an alignment group (but in implementation-dependent order), and concurrently across alignment groups.

7. Simulation Time: The behavior of an SSF model is defined by the collective actions of its processes on its state. Every process resumption takes place at a particular simulation time, and each modification of model state by that resumption of the process takes place at that time.

Example: Single Server Queue System (Checkout Counter)

- 1. SSQueue Class: It is the class that contains the whole simulation experiment. It defines the experimental constants, contains SSF communication endpoints and defines inner class arrival.
- 2. Arrival process: It is a SSF process that stores identity of entity, creates a random number generator and enqueue the generated new arrivals, then blocks for inter arrival time.
- 3. Server process: It is the process that is called when a job has completed service or by a signal from the arrival process. It also updates statistics. Customers are dequeued from the waiting list or the process suspends if no customers were waiting.

Other simulation software

Selection of Simulation Software

- 1. Model Building Features
- It is the features that facilitates the development process of the model.
- It includes modeling worldview, input data analysis capability, graphical model building, simulation programming, syntax, input flexibility, randomness, and so on.
- 2. Runtime Environment
- It is the feature that determines how the model acts during the simulation run.
- It includes execution speed, model size, interactive debugger, model statistics and so on.

3. Animation and Layout Features

- It is the feature that maps the graphical motions of the objects or entities during the simulation run.
- It includes animation type, import files, dimension, movement, motion qualities, navigation, display step, views and so on.

4. Output Features

- It is the feature that optimizes the output of the simulation model.
- It includes scenario manager, run manager, independent replication, optimization, reports, statistical analysis, file export and so on.

5. Vender Support Features

- It is the feature that shows how the user can get support from the vender whenever necessary.
- It includes training, documentation, help system, tutorials, support, upgrades, maintenance and so on.

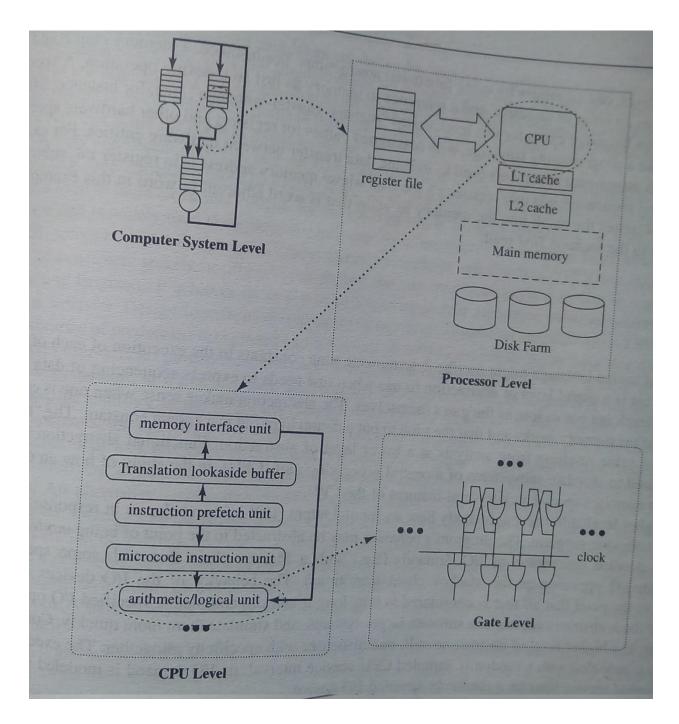
Chapter-9

Simulation of computer systems

Simulation tools

Level of Abstraction in Computer System

- Computer system have complex time scale behavior from time to flipping transistor's state to time for human interaction.
- It is designed hierarchically.
- The high level of abstraction is system level. In this level, one can view computational activity in terms of tasks circulating among servers, queuing for service when a server is busy.
- Below it is Processor level in which one can view components of the processor used.
- Below it is CPU level in which one can view the activity of functional units that together make up a central processing unit.
- The lowest level is Gate level in which one can view the logical circuitry that is responsible for all the computations carried out by the computer system.
- Simulation is used in each level and the results of one level is used by another level.



Simulation Tools

- Simulation tools are the tools that are used to perform and evaluate simulations at different abstraction levels of computer system.
- An important characteristic of a tool is how it supports model building.

- The tools commonly used for simulation are:
- 1. CPU network simulation (Queueing network, Petri net simulators)
- 2. Processor simulation (VHDL)
- 3. Memory simulation (VHDL)
- 4. ALU simulation (VHDL)
- 5. Logic network simulation (VHDL)

Process Oriented Approach

- It implies that the tool must support separately schedulable threads of control.
- It allows continuous description with suspensions.

Event Oriented Approach

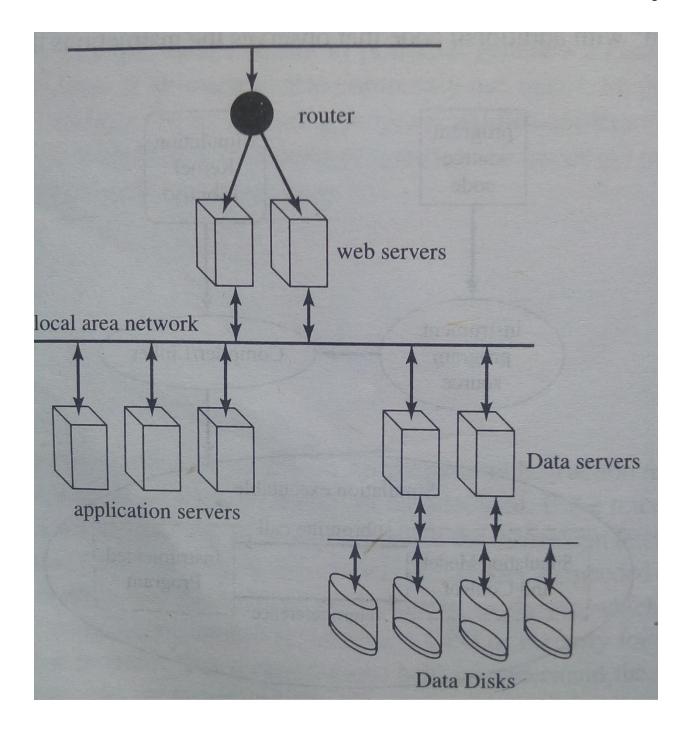
- It implies that the tool must support model description.
- It does not allow continuous description with suspensions.

High Level computer - system simulation

Problem Definition:

Consider a company provides a website for searching and links to sites for certain facilities. At the back end, there is data servers that handles specific queries and updates databases. Data servers receive requests for service from application servers. At front end, there is web servers that manage interaction of applications with the WWW. The whole system is connected with the users through the router. Let us consider that we need to study site's ability to handle load at peak periods i.e. desired output is empirical distribution of the access response time.

Now, for this we need to focus on impact of timing at each level, factors that affect timing and effects of timing on contention for resources for designing this high-level simulation model.



Simulation Model:

- All entries into the system are through dedicated router. It examines the request and forwards it to some web server. It takes some time to decide whether the request is a new request or part of ongoing session. One switching time is assumed for a pre-existing request and different time for a new request. It outputs the web server selection and enqueues request for service to the web server.

- Web server consists of one queue for new requests, one for suspended requests that are waiting for response from application server and one for requests that are ready to process response from application server. It is assumed that web server has enough memory to handle all the requests. It also has queuing policy. Associated application server is identified for each new request. A request for service is formatted and forwarded to the application server and the request joins the suspended queue.
- Application server organizes the request for services. The new request for service joins the new-request queue. An application request is modeled as a sequence of sets of requests (organized in a burst) from data servers. It is assumed that all data requests from a burst must be satisfied before next burst computation begins. For each application, a list of ready to execute and a list of suspended threads are maintained.
- Data servers create a new thread to respond to data request and places it in a queue of ready threads. When service is received, the thread requests data from a disk and then places in a suspended queue.
- Disk completes its operation for data request and the thread in data server on receiving response from disk moves to ready list and reports back to application server associated with the request.
- The thread suspended at application server responds and finishes; then reports its completion to the web server.
- The thread in web server that initiates that request then communicates the results back to the Internet.

Response Time Analysis

- Query-response-time distribution is estimated by measuring between the time at which a request first hits the router and the time at which web server thread communicates the result.
- The system can be analyzed by measuring behavior at each server of each type.

- To assess system capacity at peak loads, we would simulate to identify bottlenecks, then look to see how to reduce load at bottleneck devices by changing various settings of simulation like scheduling policy, queue discipline and so on.

CPU simulation

- In CPU simulation, we focus on discovering execution time.
- For CPU simulation, the input is the stream of instructions and the simulation must model the logical design on what happens in response to the instruction stream.

Problem Definition of ILP (Instruction Level Parallelism) CPU:

The stages in an ILP CPU are as follows:

- 1. Instruction fetch The instruction is fetched from memory.
- 2. Instruction decode The memory word holding the instruction is interpreted to discover operations to be performed and registers involved.
- 3. Instruction Issue An instruction is issued if no constraints hold it back from being executed.
- 4. Instruction Execute The instruction operation is performed.
- 5. Instruction Complete The results of instruction are stored in the destination register.
- 6. Instruction Graduate Executed instructions are graduated in the order that they appear in the instruction stream.
- ILP design allows multiple instructions to be represented in some stages. This may cause execution of instruction out of order. The instruction graduate phase will reorder all the executed instructions.

Simulation Model of ILP (Instruction Level Parallelism) CPU:

- Instruction fetch interacts with the simulated memory system if present. If memory system is present, it can look into an instruction cache for the next referenced instruction, stalling if a miss is suffered. This stage makes instruction in the CPU's list of active instructions.
- Instruction Decode stage places an instruction in the list. A logical register that appears as the target of an operation is assigned a physical register. Registers used as operand are assigned physical registers that define their values. Branch instructions are identified and outcomes are predicted. Resources for the instruction execution are committed.
- Instruction Issue stage issue a decoded instruction for execution if values in its input registers are available and a functional unit needed to perform the instruction is available. It can be achieved by marking the registers and functional units as busy or pending. After the state is changed, the instruction waiting for that register or functional unit is reconsidered for issue.
- Instruction execute stage computes the result specified by the instruction. It means the actual operation intended by the instruction is performed.
- Instruction complete stage deposits the result into a register or memory as specified in the instruction.
- Instruction graduate reorders the completed instruction in the same order as instruction stream. This is simulated by knowing the sequence number of the next instruction to be graduated.

Example----

Consider a hypothetical computer with following instructions:

Load \$2, \$6;

Molt \$5, 2;

Add \$4, \$2;

Add \$5, \$2;

Assume 4 cycles needed to load register, addition takes 1 cycle to complete and multiplication takes 2 cycles to complete.

Instruction/Cycle	1	2	3	4	5	6	7
I1	Fetch	Decode	Issue	Execute	Stall	Stall	Stal
I2		Fetch	Decode	Issue	Execute	Stall	Con
I3			Fetch	Decode	Issue	Execute	Con
I4				Fetch	Decode	Stall	Stal

Memory Simulation

Memory is arranged hierarchically with L1 cache, L2 cache, main memory and disks.

Example -- Cache Simulation

- The input is cache parameters and memory access tree.
- The output of simulation is cache hit rate.

Simulation Model

- Maintain cache directory and LRU status of the lines within the set.
- When an access is made, update LRU status.
- If a hit, record it as such.
- If a miss, update the contents of the directory.
- Cache directory is implemented as an array, with array entries corresponding to directory entries.