Chapter 1
Introduction to Simulation

1. What is simulation?

Simulation is the imitation of the operation of a real-world process or system over time. Simulation involves the generation of an artificial history of the system, and the observation of that artificial history to draw inferences concerning the operating characteristics of the real system that is represented.

Simulation is the numerical technique for conducting experiments on digital computer, which involves logical and mathematical relationships that interact to describe the behavior and the structure of a complex real world system over extended period of time.

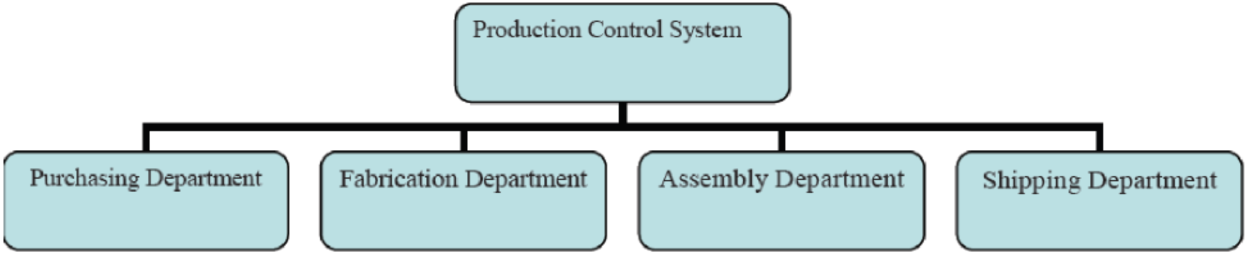
The process of designing a model of a real system, implementing the model as a computer program, and conducting experiments with the model for the purpose of understanding the behavior of the system, or evaluating strategies for the operation of the system.

2. System Concepts

A system is defined as a group of objects that are joined together in some regular intraction or interdependence for the accomplishment of some task. For example: *Production system for manufacturing automobiles.*

A system is usually considered as a set of inter–related factors, which are described as entities activities and have properties or attributes. Processes that cause system changes are called activities. The state of a system is a description of all entities, attributes and the activities at any time.

Example: A factory system shown as follows:



Components of system

2.1 Entity, attribute and activities

An **entity** represents an object that requires explicit definition. An entity can be dynamic in that it moves through the system, or it can be static in that it serves other entities. In the example, the customer is a dynamic entity, whereas the bank teller is a static entity.

An entity may have **attributes** that pertain to that entity alone. Thus, attributes should be considered as local values. In the example, an attribute of the entity could be the time of arrival. Attributes of interest in one investigation may not be of interest in another investigation. Thus, if red parts and blue parts are being manufactured, the color could be an attribute.

Processes that cause system changes are called **activities or events.**

Example

System	Entities	Attributes	Activities
Traffic	Cars, bus, pedestrian	Speed, model	Driving, walking
Bank	Customer	Balance	Depositing, arrival of costomer, ...
Supermarket	Customers	Shopping list	Checking_out, ...

In the bank example, events include the arrival of a customer for service at the bank, the beginning of service for a customer, and the completion of a service.

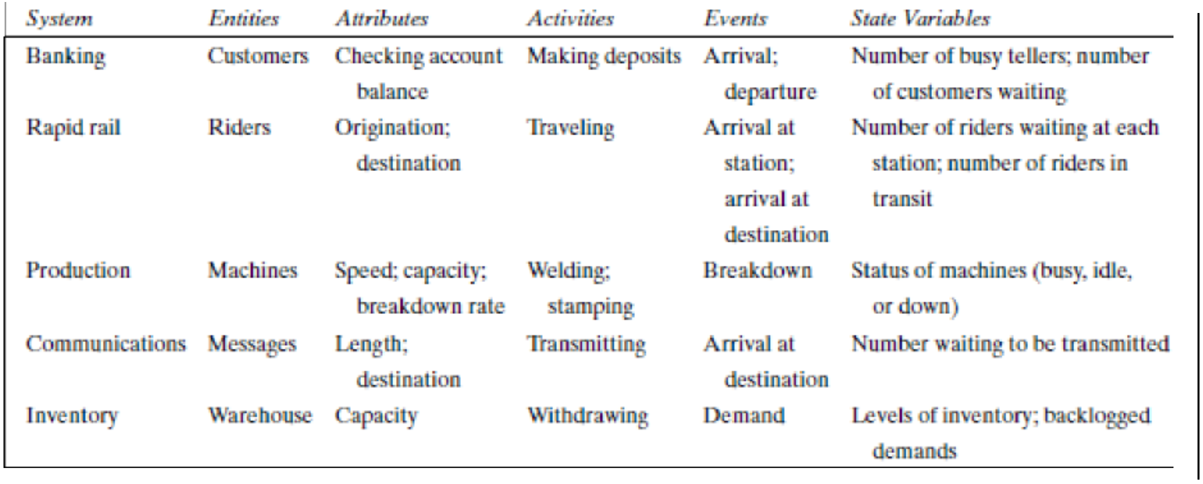
There are both internal and external events, also called endogenous and exogenous events, respectively. For instance, an endogenous event in the example is the beginning of service of the customer since that is within the system being simulated. An exogenous event is the arrival of a customer for service since that occurrence is outside of the simulation.

2.2 State variables

The state of a system is defined to be that collection of variables necessary to describe the system at any time, relative to the objectives of the study. In the study of a bank, possible state variables are the number of busy tellers, the number of customers waiting in line or being served, and the arrival time of the next customer.

So the system state variables are the collection of all information needed to define what is happening within the system to a sufficient level (i.e., to attain the desired output) at a given point in time.

Example



System	*Entities*	*Attributes*	*Activities*	*Events*	*State Variables*
Banking	Customers	Checking account balance	Making deposits	Arrival; departure	Number of busy tellers; number of customers waiting
Rapid rail	Riders	Origination; destination	Traveling	Arrival at station; arrival at destination	Number of riders waiting at each station; number of riders in transit
Production	Machines	Speed; capacity; breakdown rate	Welding; stamping	Breakdown	Status of machines (busy, idle, or down)
Communications	Messages	Length; destination	Transmitting	Arrival at destination	Number waiting to be transmitted
Inventory	Warehouse	Capacity	Withdrawing	Demand	Levels of inventory; backlogged demands

2.3 Open System/Close System

A system with exogenous activities is considered as open system and a system with strict endogenous activities is called a closed system.

2.4 System Environment

The external components which interact with the system and produce necessary changes are said to constitute the system environment. In modeling systems, it is necessary to decide on the boundary between the system and its environment. This decision may depend on the purpose of the study.

Example: In a factory system, the factors controlling arrival of orders may be considered to be outside the factory but yet a part of the system environment. When, we consider the demand and supply of goods, there is certainly a relationship between the factory output and arrival of orders. This relationship is considered as an activity of the system.

Endogenous System

The term endogenous is used to describe activities and events occurring within a system.

Example: Drawing cash in a bank.

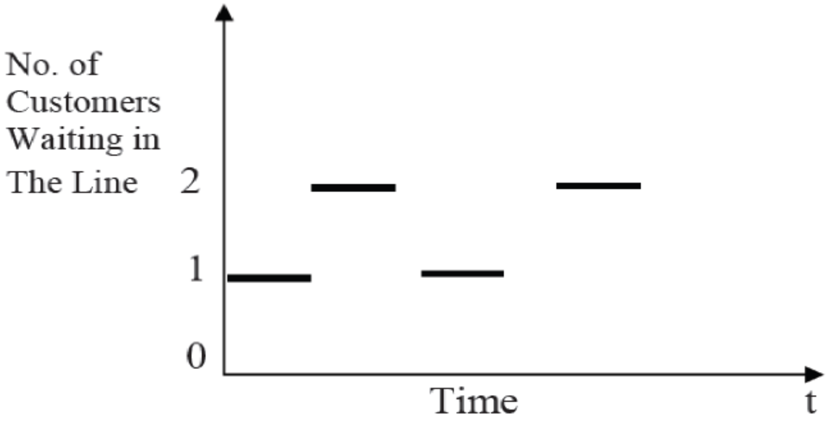
Exogenous System

The term exogenous is used to describe activities and events in the environment that affect the system. Example: Arrival of customers.

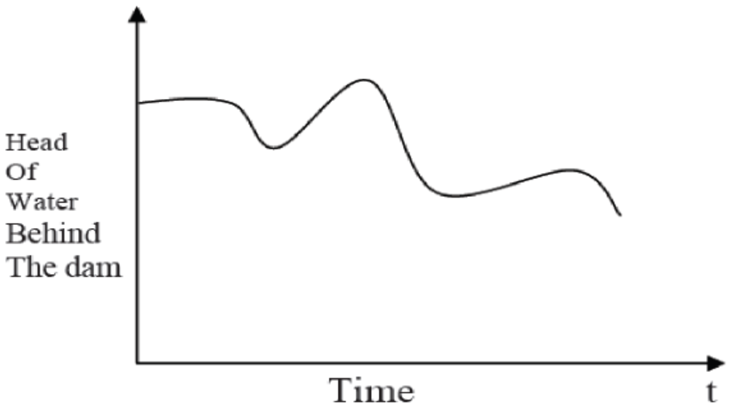
3. Discrete and continuous system

Discrete system is one in which the state variables changes only at a discrete set of time. For example: banking system in which no of customers (state variable) changes only when a customer arrives or service provided to customer i.e customer depart form system.

The figure below show how no of customer changes only at discrete points in time



Continuous system is one in which the state variables change continuously over time. For example, during winter seasons level of which water decreases gradually and during rainy season level of water increase gradually. The change in water level is continuous. The figure below shows the change of water level over time.



4. System Modeling

A model is defined as a representation of a system for the purpose of studying the system. It is necessary to consider only those aspects of the system that affect the problem under investigation. These aspects are represented in a model, and by definition it is a simplification of the system.
The aspect of system that affect the problem under investigation, are represented in a model of the system. Therefore model is the simplification of the real system.
There is no unique model of a system. Different models of the same system will be produced by different system analysts who are interested in different aspect of system.

The task of deriving a model of a system may be divided broadly into two subtasks: Establishing model parameter and supplying data.

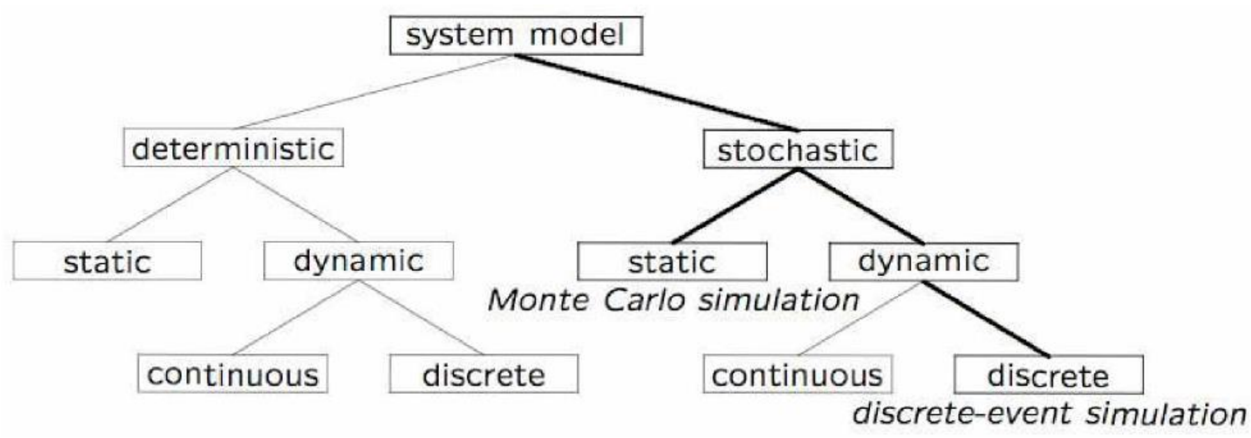
Establishing model structure determines system boundary and identifies the entities, attributes, activities and events of a system.

Supplying data provides value contained an attribute and define relationships involved in the activities.

Types of Model

The various types of models are shown in figure below.
xMathematical and Physical Model
xStatic Model
xDynamic Model
xDeterministic Model
xStochastic Model
xDiscrete Model
xContinuous Model

The chart to represent different model in a hierarchy is as shown below



Physical model

These models are based on some analogy between mechanical and electrical system The system attributes are represented by physical measures such as voltage. The system activities are represented by physical laws.

Physical models are of two types, static and dynamic. Static physical model is a scaled down model of a system which does not change with time. An architect before constructing a building makes a scaled down model of the building, which reflects all it rooms, outer design and other important features. This is an example of static physical model. Similarly for conducting trials in water, we make small water tanks, which are replica of sea, and fire small scaled down shells in them. This tank can be treated as a static physical model of ocean. Dynamic physical models are ones which change with time or which are function of time. In wind tunnel, small aircraft models (static models) are kept and air is blown over them with different velocities and pressure profiles are measured with the help of transducers embedded in the model. Here wind velocity changes with time and is an example of dynamic physical model.

Mathematical Model

It uses symbolic notation and mathematical equation to represent system. The system attributes are represented by variables and the activities are represented by mathematical function.

Example: f(x) = mx+c is a mathematical model of a line.

Static Model

Static models can only show the values that the system attributes value does not change over time.
Example: Scientist has used models in which sphere represents atom, sheet of metal to connect the sphere to represent atomic bonds. Graphs are used to model the various system based on network.
A map is also a kind of graph. These models are sometimes said to be iconic models and are of kind static physical models.

Dynamic Model

Dynamic models follow the changes over time that result from system activities. The mechanical and electrical systems are the example of dynamic system. Generally, dynamic models involve the computation of variable value over time and hence they are represented by differential equations.

Analytical Models:

In mathematical model, we can differentiate the model on the basis of solution technique used to solve the model. Analytical technique means using deductive reasoning of mathematical theory to solve a model. Such models are known as analytical model.

Numerical models
Numerical models involve applying computational process to solve equations. For example: we may solve differential equation numerically when the specific limit of variable is given.

The analytical methods to produce solution may take situation numerical methods are preferred. **Deterministic**

Model

Contains no random variables. They have a known set of inputs which will result in a unique set of outputs. Ex: Arrival of patients to the Dentist at the scheduled appointment time.

Stochastic Model
Has one or more random variable as inputs. Random inputs leads to random outputs. Ex: Simulation of a bank involves random inter-arrival and service times.

Principles used in Modeling

Guidelines used in modeling
- It is not possible provide rule by which models are built. But a number of guidelines can be stated.
- The different viewpoints from which we can judge whether certain info. Should be included as excluded in models are:

1. **Block –Building:** The description of system should be organized as a sequence of blocks. It simplifies the interaction between block within system. Then it will be easy to describe the whole system in terms of interaction between the block and can be represented graphically as simple block diagram. For example:-the block of factory system.

Fig:

Block diagram of factory system

2. **Relevance:** The model should only include relevant information. For example, if the factory system study aims to compare the efficient of different operating rules efficiency it is not relevant to consider the mining of employee as an activity.

   Irrelevant information should not include despite of being no harm because it increases the complexity of model and takes more time and effort to solve model.

3. **Accuracy:** The gathered information should be accurate as well. For example in aircraft system the accuracy as movement of the aircraft depends upon the representations of airframe such as a rigid body.

4. **Aggregation:** It should be considered that to which numbers of individual entities can be grouped into a block. For example in factory system, different department are grouped together handled by production manger.

Distributed lag model

Models that have the property of changing only at fixed interval of time and based on current values of variables on other current values of variables are called distributed lag model.

In economic studies some economic data are collected over uniform time interval such as a month or year.

This model consists of linear algebraic equations that represent continuous system but data are available at fixed points in time.

For example: Mathematical model of national economy

Let

C=consumption
I=investment
T=Taxes
G=government expenditures Y=national income

Then

$C=20+0.7(Y-T)$
$I=2+0.1Y$
$T=0.2Y$
$Y=C+I+G$

All the equation are expressed in billions of rupees. This is static model and can be made dynamic by lagging all the variables as follows

$C=20+0.7(Y_{-1}-T_{-1})$ $I=2+0.1Y_{-1}$
$T=0.2Y_{-1}$
$Y^{=}C_{-1}+I_{-1}+G_{-1}$

Any variable that can be expressed in the form of its current value and one or more previous value is called lagging variable. And hence this model is given the name distributed lag model. the variable in a previous interval is denoted by attaching –n suffix to the variable. Where -n indicate the ^{nthe} interval.

Advantages of distributed lag model

x Simple to understand and can be computed by hand, computers are extensively used to run them.
x There is no need for special programming language to organize simulation task.

When simulation is appropriate Tool?

The availability of special-purpose simulation languages, massive computing capabilities at a decreasing cost per operation, and advances in simulation methodologies have made simulation one of the most widely used and accepted tools in operations research and systems analysis. Simulation can be used for the following purposes:

1. Simulation enables the study of, and experimentation with, the internal interactions of a complex system, or of a subsystem within a complex system.

2. Informational, organizational, and environmental changes can be simulated, and the effect of these alterations on the model's behavior can be observed.

3. The knowledge gained in designing a simulation model may be of great value toward suggesting improvement in the system under investigation.

4. By changing simulation inputs and observing the resulting outputs, valuable insight may be obtained into which variables are most important and how variables interact.

5. Simulation can be used to experiment with new designs or policies prior to implementation, so as to prepare for what may happen.

6. Simulation can be used to verify analytic solutions.

7. By simulating different capabilities for a machine, requirements can be determined.

8. Simulation models designed for training allow learning without the cost and disruption of on-the-job learning.

When the simulation is not appropriate?

1. Simulation should not be used when the problem can be solved using common sense.
2. Simulation should not be used if the problem can be solved analytically.
3. Simulation should not be used if it is easier to perform direct experiments.etc
4. Simulation should be used when the problem cannot be solved using common sense.
5. Simulation should not be used, if the costs exceeds savings.
6. Simulation should not be performed, if the resources or time are not available.
7. if no data is available, not even estimate simulation is not advised.
8. If there is not enough time or the persons are not available, simulation is not appropriate.
9. If managers have unreasonable expectation say, too much soon – or the power of simulation is over estimated, simulation may not be appropriate.
10. If system behavior is too complex or cannot be defined, simulation is not appropriate.

Advantages of simulation

1. Simulation can also be used to study systems in the design stage.
2. Simulation models are run rather than solver.
3. New policies, operating procedures, decision rules, information flow, etc can be explored without disrupting the ongoing operations of the real system.
4. New hardware designs, physical layouts, transportation systems can be tested without committing resources for their acquisition.
5. Hypotheses about how or why certain phenomena occur can be tested for feasibility.
6. Time can be compressed or expanded allowing for a speedup or slowdown of the phenomena under investigation.
7. Insight can be obtained about the interaction of variables.
8. Insight can be obtained about the importance of variables to the performance of the system.
9. Bottleneck analysis can be performed indication where work-in process, information materials and so on are being excessively delayed.
10. A simulation study can help in understanding how the system operates rather than how individuals think the system operates.
11. "what-if" questions can be answered. So it is useful in the design of new systems.

Disadvantage of simulation

1. Model building requires special training.
2. Simulation results may be difficult to interpret.
3. Simulation modeling and analysis can be time consuming and expensive.

4. Simulation is used in some cases when an analytical solution is possible or even preferable.

Applications of Simulation

Manufacturing Applications

1. Analysis of electronics assembly operations
2. Design and evaluation of a selective assembly station for high precision scroll compressor shells.
3. Comparison of dispatching rules for semiconductor manufacturing using large facility models.

4. Evaluation of cluster tool throughput for thin-film head production.
5. Determining optimal lot size for a semiconductor backend factory.
6. Optimization of cycle time and utilization in semiconductor test manufacturing.
7. Analysis of storage and retrieval strategies in a warehouse.
8. Investigation of dynamics in a service oriented supply chain.
9. Model for an Army chemical munitions disposal facility.

Semiconductor Manufacturing

1. Comparison of dispatching rules using large-facility models.
2. The corrupting influence of variability.
3. A new lot-release rule for wafer fabs.
4. Assessment of potential gains in productivity due to proactive retied management.
5. Comparison of a 200 mm and 300 mm X-ray lithography cell.
6. Capacity planning with time constraints between operations.

Military Applications

1. Modeling leadership effects and recruit type in a Army recruiting station.
2. Design and test of an intelligent controller for autonomous underwater vehicles.
3. Modeling military requirements for non war fighting operations.
4. Multi trajectory performance for varying scenario sizes.
5. Using adaptive agents in U.S. Air Force retention.

Steps on simulation Study

1. Problem formulation

Every study begins with a statement of the problem, provided by policy makers. Analyst ensures its clearly understood. If it is developed by analyst policy makers should understand and agree with it.

2. Setting of objectives and overall project plan

The objectives indicate the questions to be answered by simulation. At this point a determination should be made concerning whether simulation is the appropriate methodology. Assuming it is appropriate, the overall project plan should include

- A statement of the alternative systems
- A method for evaluating the effectiveness of these alternatives
- Plans for the study in terms of the number of people involved
- Cost of the study
- The number of days required to accomplish each phase of the work with the anticipated results.

Model conceptualization

The construction of a model of a system is probably as much art as science. The art of modeling is enhanced by ability:

x To abstract the essential features of a problem

x To select and modify basic assumptions that characterizes the system x To enrich and elaborate the model until a useful approximation results

Thus, it is best to start with a simple model and build toward greater complexity. Model conceptualization enhances the quality of the resulting model and increases the confidence of the model user in the application of the model.

Data collection

There is a constant interplay between the construction of model and the collection of needed input data. It is done in the early stages. Objective kind of data are collected.

Model translation

Real-world systems result in models that require a great deal of information storage and computation. It can be programmed by using simulation languages or special purpose simulation software. Simulation languages are powerful and flexible. Simulation software models development time can be reduced.

Verified

It pertains to the computer program and checking the performance. If the input parameters and logical structure are correctly represented, verification is completed.

Validated

It is the determination that a model is an accurate representation of the real system. It is achieved through calibration of the model. The calibration of model is an iterative process of comparing the model to actual system behavior and the discrepancies between the two.

Experimental Design

The alternatives that are to be simulated must be determined. Which alternatives to simulate may be a function of runs? For each system design, decisions need to be made concerning

- Length of the initialization period
- Length of simulation runs
- Number of replication to be made of each run

Production runs and analysis

They are used to estimate measures of performance for the system designs that are being simulated.

More runs

Based on the analysis of runs that have been completed, the analyst determines if additional runs are needed and what design those additional experiments should follow.

Documentation and reporting

Two types of documentation.

- Program documentation
- Process documentation

Program documentation
Can be used again by the same or different analysts to understand how the program operates. Further modification will be easier. Model users can change the input parameters for better performance.

Process documentation
Gives the history of a simulation project. The result of all analysis should be reported clearly and concisely in a final report. This enables to review the final formulation and alternatives, results of the experiments and the recommended solution to the problem. The final report provides a vehicle of certification.

Implementation
Success depends on the previous steps. If the model user has been thoroughly involved and understands the nature of the model and its outputs, likelihood of a vigorous implementation is enhanced. The simulation model building can be broken into 4 phases.

Phase of Simulation Study

I Phase

- Consists of steps 1 and 2
- It is period of discovery/orientation
- The analyst may have to restart the process if it is not fine-tuned
- Recalibrations and clarifications may occur in this phase or another phase.

II Phase

- Consists of steps 3,4,5,6 and 7
- A continuing interplay is required among the steps
- Exclusion of model user results in implications during implementation

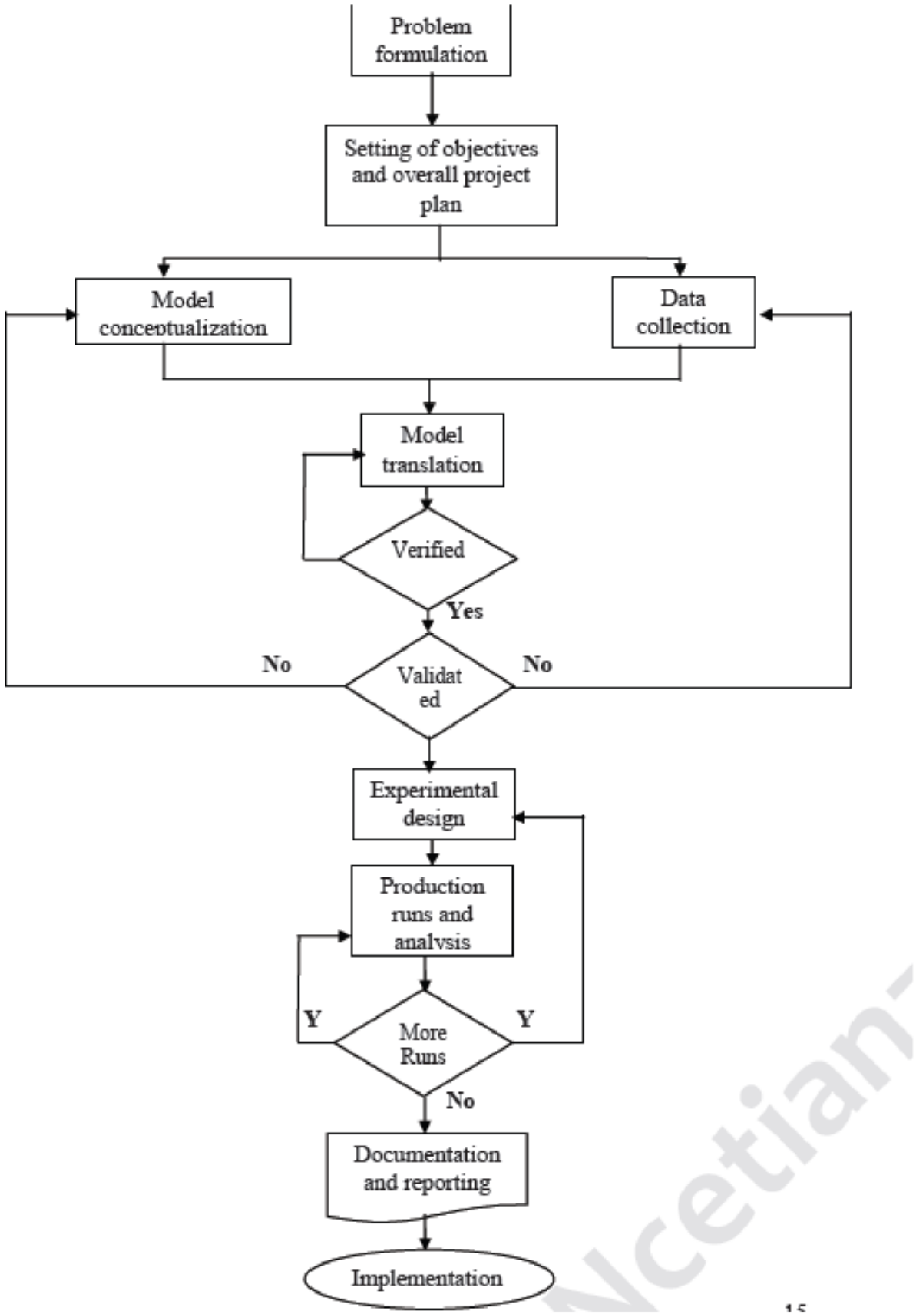
III Phase

- Consists of steps 8,9 and 10
- Conceives a thorough plan for experimenting
- Discrete-event stochastic is a statistical experiment
- The output variables are estimates that contain random error and therefore proper statistical analysis is required.

IV Phase

- Consists of steps 11 and 12

- Successful implementation depends on the involvement of user and every steps successful completion.



Hybrid Simulation: For most studies, the system under study is clearly either of continuous or discrete nature and it is the determining factor in deciding whether to use an analog or digital computer for system simulation.

If the system being simulated is an interconnection of continuous and discrete subsystem, then such system simulation is known as hybrid simulation. Such hybrid system can be digital computer being linked together

Hybrid simulation required high speed converters to transform signals from analog to digital from and vice –versa.

Real time simulation: In real time simulation, actual device (which are part of a system) are used in conjunction with either digital computer or hybrid computer. It provides the simulation of the points of systems that do not exist or that cannot be easily used in an experiment.

i.e. the basic idea of real time simulation is ‟uses the actual part if they are appropriate to use in experiment otherwise use the simulation of the points of the system.”

A well-known examples is “simulation to train pilots”. It uses the devices for training pilots by giving them the impression that is at the control of an aircraft.

It requires real time simulator of the plane its control system, the weather and other environmental conditions. Sometimes, real time simulation also refers to a computer model of a physical system that can execute at the same rate as actual system can. For example: if a machine takes 10 minutes to fill a tank in real world, the simulation also would take 10 minutes.

Real time simulation of an engineering system becomes possible when replace physical device with virtual device are.

Questions: Differentiate between dynamic physical models and static physical models with example.

Chapter 2

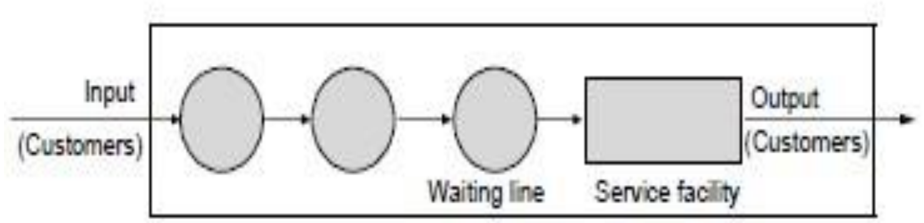
Queuing system and Markov Chains

1. Queuing system: Introduction

Most systems of interest in a simulation study contain a process in which there is a demand for services. The system can service entities at a rate which is greater than the rate at which entities arrives. The entities are then said to join waiting line. The line where the entities or customers wait is generally known as queue.

The combination of all entities in system being served and being waiting for services will be called a **queuing system.**

The general diagram of queuing system can be shown as



A queuing system involves customers arriving at a constant or variable time rate for service at a service station. Customers can be students waiting for registration in college, aero -plane queuing for landing at airfield, or jobs waiting in machines shop.

If the customer after arriving can enter the service center, it is good, otherwise they have to wait for the service and form a queue i.e. **waiting line**. They remain in queue till they are provided the service. Sometimes queue being too long, they will leave the queue and go, it results a loss of customer. Customers are to be serviced at a constant or variable rate before they leave the service station.

2. Characteristics or elements of queuing system

In order to model queuing systems, we first need to be a bit more precise about what constitutes a queuing system. The three basic elements common to all queuing systems are:

1. Arrival Process or patterns
2. Service process or patterns
3. Queuing discipline

a) Arrival Process or patterns

Any queuing system must work on something – customers, parts, patients, orders, etc. We generally called them as **entities or customers**. Before entities can be processed or subjected to waiting, they must first enter the system. Depending on the environment, entities can arrive smoothly or in an unpredictable fashion. They can arrive one at a time or in clumps (e.g., bus loads or batches). They can arrive independently or according to some kind of correlation.

A special arrival process, which is highly useful for modeling purposes, is the **Markov** arrival process. Both of these names refer to the situation where entities arrive one at a time and the times between arrivals are **exponential** random variables. This type of arrival process is *memoryless*, which means that the likelihood of an arrival within the next *t* minutes is the same no matter how long it has been since the last arrival.

Examples where this occurs are phone calls arriving at an exchange, customers arriving at a fast food restaurant, hits on a web site, and many others.

b) Service Process

Once entities have entered the system they must be served. The physical meaning of "service" depends on the system. Customers may go through the checkout process. Parts may go through machining. Patients may go through medical treatment. Orders may be filled. And so on. From a modeling standpoint, the operational characteristics of service matter more than the physical characteristics. Specifically, we care about whether service times are long or short, and whether they are regular or highly variable. We care about whether entities are processed in first-comefirst-serve (FCFS) order or according to some kind of priority rule. We care about whether entities are serviced by a single server or by multiple servers working in parallel etc

Markov Service Process

A special service process is the **Markov** service process, in which entities are processed one at a time in FCFS order and service times are independent and **exponential**. As with the case of Markov arrivals, a Markov service process is memoryless, which means that the expected time until an entity is finished remains constant regardless of how long it has been in service.

For example, in the Marcrohard example, a Markov service process would imply that the additional time required to resolve a caller's problem is 15 minutes, no matter how long the technician has already spent talking to the customer. While this may seem unlikely, it does occur when the distribution of service times looks like the case shown in Figure 1. This depicts a case where the average service time is 15 minutes, but many customers require calls much shorter than 15 minutes (e.g., to be reminded of a password or basic procedures) while a few customers require significantly more than 15 minutes (e.g., to perform complex diagnostics or problem resolution). Simply knowing how long a customer has been in service doesn't tell us enough about what kind of problem the customer has to predict how much more time will be required.

c) Queuing Discipline:

The third required component of a queuing system is a queue, in which entities wait for service. The number of customer can wait in a line is called **system capacity**.

The simplest case is an unlimited queue which can accommodate any number of customers. It is called system with unlimited capacity.
But many systems (e.g., phone exchanges, web servers, call centers), have limits on the number of entities that can be in queue at any given time.

Arrivals that come when the queue is full are rejected (e.g., customers get a busy signal when trying to dial into a call center). Even if the system doesn't have a strict limit on the queue size,

The logical ordering of customer in a waiting line is called Queing discipline and it determines which customer will be chosen for service. We may say that queuing discipline is a rule to chose the customer for service from the waiting line.

The queuing discipline includes:

a) **FIFO (First in First out)** : According to this rule, Service is offered on the basis of arrival time of customer. The customer who comes first will get the service first. So in other word the customer who get the service next will be determine on the basis of longest waiting time.
b) **Last in First Out(LIFO):** It is usually abbreviated as LIFO, occurs when service is next offered to the customer that arrived recently or which have waiting time least. In the crowded train the passenger getting in or out from the train is an example of LIFO.
c) **Service in Random order (SIRO):** it means that a random choice is made between all waiting customers at the time service is offered. I.e a customer is picked up randomly form the waiting queue for the service.
d) **Shortest processing time First(SPT)**: it means that the customer with shortest service time will be chosen first for the service. i.e. the shortest service time customer will get the priority in the selection process.
e) **Priority**: a special number is assigned to each customer in the waiting line and it is called priority. Then according to this number, the customer is chosen for service.

Queuing Behavior

Customers may balk at joining the queue when it is too long (e.g., cars pass up a drive through restaurant if there are too many cars already waiting). It is called balking.

Customer may also exit the system due to impatience (e.g., customers kept waiting too long at a bank decide to leave without service) or perishability (e.g., samples waiting for testing at a lab spoil after some time period). It is called *reneging.*

When there is more than one line forming for the same service or server, the action of moving customer from one line to another line because they think that they have chosen slow line. It is called Jockeying.

3) Queuing Notations (or KENDALL'S NOTATION)

We will be frequently using notation for queuing system, called Kendall's notation, i.e

A/B/c/N/K,

where, A, B, c, N, K respectively indicate arrival pattern, service pattern, number of servers, system capacity, and Calling population.

The symbols used for the probability distribution for inter arrival time, and service time are, D for deterministic, M for exponential (or Markov) and Ek for Erlang.

If the capacity Y is not specified, it is taken as infinity, and if calling population is not specified, it is assumed unlimited or infinite

For example

a) $M/D/2/5/\infty$ stands for a queuing system having exponential arrival times, deterministic service time, 2 servers, capacity of 5 customers, and infinite population.
b) If notation is given as M/D/2 means exponential arrival time, deterministic service time, 2 servers, infinite service capacity, and infinite population.

4) Single server queuing system

For the case of simplicity, we will assume for the time being, that there is single queue and only one server serving the customers. We make the following assumptions.

- First-in, First-out (FIFO): Service is provided on the first come, first served basis.

- Random: Arrivals of customers is completely random but at a certain arrival rate.

- Steady state: The queuing system is at a steady state condition.

The above conditions are very ideal conditions for any queuing system and assumptions are made to model the situation mathematically.

First condition only means irrespective of customer, one who comes first is attended first and no priority is given to anyone.

Poison arrival Patterns

Second condition says that **arrival of a customer** is completely random. This means that an arrival can occur at any time and the time of next arrival is independent of the previous arrival. With this assumption it is possible to show that the distribution of the inter-arrival time is exponential. This is equivalent to saying that the number of arrivals per unit time is a random variable with a Poisson's distribution. This distribution is used when chances of occurrence of an event out of a large sample is small.

i.e. if X = number of arrivals per unit time, then, probability distribution function of arrival is given as

$$f(x) = \Pr(X = x) = \frac{e^{-\lambda}\lambda^x}{x!}, \begin{cases} x = 0, 1, 2, \ldots \\ \lambda > 0 \end{cases}$$

$$E(X) = \lambda$$

, where □ λ. is the average number of arrivals per unit time ($1/\tau$), and x is the number of customers per unit time. This pattern of arrival is called **Poisson's arrival pattern**.

Illustrative example

In a single pump service station, vehicles arrive for fueling with an average of 5 minutes between arrivals. If an hour is taken as unit of time, cars arrive according to Poison's process with an average of λ = 12 cars/hr.

The distribution of the number of arrivals per hour is,

$$f(x) = \Pr(X = x) = \frac{e^{-\lambda}\lambda^x}{x!} = \frac{e^{-12}12^x}{x!}, \begin{cases} x = 0, 1, 2, \ldots \\ \lambda > 0 \end{cases}$$

$$E(X) = 12 \text{ cars/hr}$$

5) Measure of Queues

We have already defined the mean inter arrival time T_a and the mean service time T_s and the corresponding rates; Arrival rate $\lambda=1/T_a$

Service rate $\mu=1/T_s$

The following measures are used in the analysis of queue system

Traffic intensity: the ratio of the mean service time to the mean inter arrival time is called traffic intensity.

I.e. u= λ“T_s or u=T_s/T_a

If there is any balking or reneging, not all arriving entities get served. It is necessary therefore to distinguish between actual arrival rate and the arrival rate of entities that get served. Here λ“ denoted the all arrivals including balking or reneging.

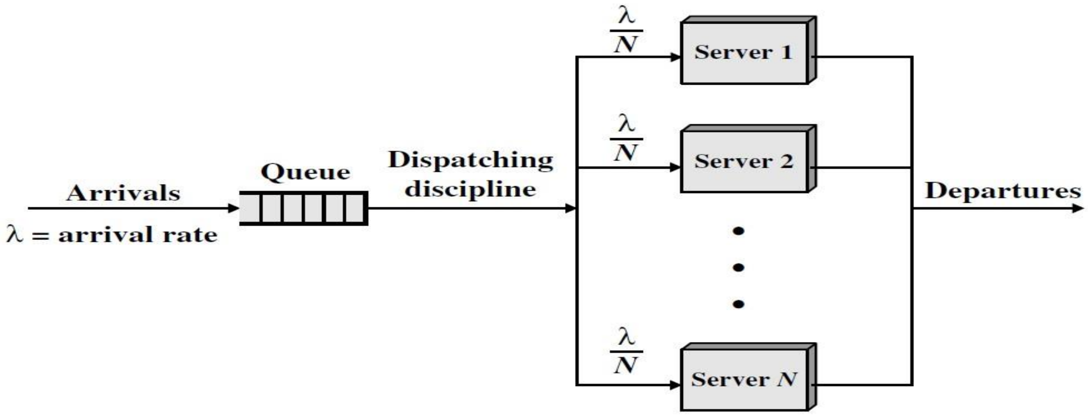
Server utilization: It consists of only the arrival that gets served. It is denoted by ρ and defined as
$\rho = \lambda T_s = \lambda / \mu$ (server utilization for single server).
This is also the average number of customers in the service facility.
Thus probability of finding service counter free is
$(1 - \rho)$

That is there is zero customers in the service facility.

6) Concept of Multi-server Queue

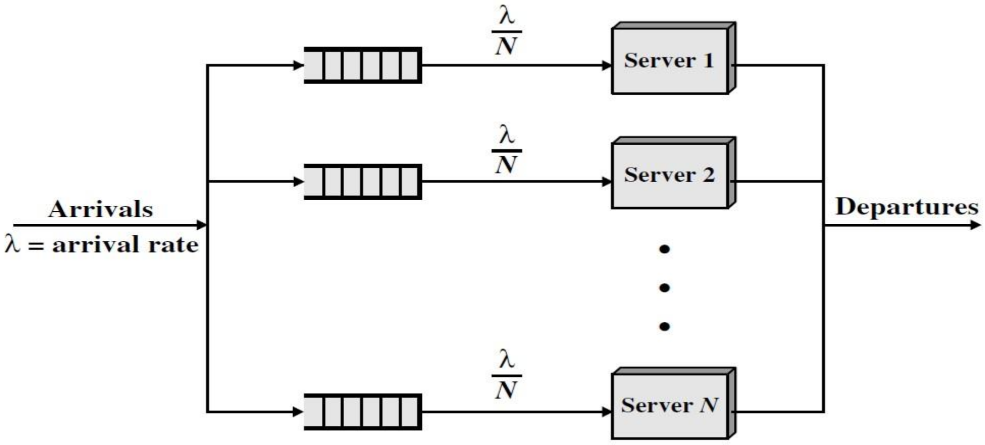
Figure 3a shows a generalization of the simple model we have been discussing for multiple servers, all sharing a common queue. If an item arrives and at least one server is available, then the item is immediately dispatched to that server. It is assumed that all servers are identical; thus, if more than one server is available, it makes no difference which server is chosen for the item. If all servers are busy, a queue begins to form. As soon as one server becomes free, an item is dispatched from the queue using the dispatching discipline in force.

The key characteristics typically chosen for the multi-server queue correspond to those for the single-server queue. That is, we assume an infinite population and an infinite queue size, with a single infinite queue shared among all servers. Unless otherwise stated, the dispatching discipline is FIFO. For the multi-server case, if all servers are assumed identical, the selection of a particular server for a waiting item has no effect on service time.



The total server utilization in case of Multi-server queue for N server system is $\rho = \lambda/c\mu$ where µ is the service rate and lemda is the arrival rate.

There is another concept which is called multiple single server queue system as shown below



7) Some notation or Formula used to Measure the different parameter of queue

Two principal measures of queing system are;

a) The mean number of customers waiting and

b) The mean time the spend waiting

Bothe these quantities may refer to the total number of entities in the system, those waiting and those being served or they may refer only to customer in the waiting line.

Average number of customers in the queue L_Q is same as expected number in the system – the expected number in the service facility:

$$\overline{L}_Q = \overline{L}_S - \rho = \frac{\lambda}{\mu - \lambda} - \frac{\lambda}{\mu} = \frac{\lambda^2}{\mu(\mu-\lambda)} = \frac{\rho^2}{(1-\rho)}$$

Average time a customer spends in the system is denoted by W_S , and is equal to expected number of customers in the system at time t, divided by number of customers arrived in unit time i.e.,

$$\overline{W}_S = \frac{\lambda}{\mu-\lambda} \cdot \frac{1}{\lambda} = \frac{1}{(\mu-\lambda)}$$

Average time a customer spends in the queue ( ) W_Q is same as average time a customer spends in the system – average time a customer spends in the server i.e.,

$$\overline{W}_Q = \overline{W}_S - \frac{1}{\mu} = \frac{\lambda}{\mu(\mu-\lambda)}$$

Example

At the ticket counter of football stadium, people come in queue and purchase tickets. Arrival rate of customers is 1/min. It takes at the average 20 seconds to purchase the ticket.

(a) If a sport fan arrives 2 minutes before the game starts and if he takes exactly 1.5 minutes to reach the correct seat after he purchases a ticket, can the sport fan expects to be seated for the tip-off ?

Solution:

(a) A minute is used as unit of time. Since ticket is disbursed in 20 seconds, this means, three customers enter the stadium per minute, that is service rate is 3 per minute.

Therefore,

λ = 1 arrival/min μ

= 3 arrivals/min

W_S = waiting time in the system=1/(μ- λ)=0.5

The average time to get the ticket and the time to reach the correct seat is 2 minutes exactly, so the sports fan can expect to be seated for the tip-off.

Example 2

Customers arrive in a bank according to a Poisson"s process with mean inter arrival time of 10 minutes. Customers spend an average of 5 minutes on the single available counter, and leave. Discuss (a) What is the probability that a customer will not have to wait at the counter?

(b)What is the expected number of customers in the bank?

(c) How much time can a customer expect to spend in the bank?

Solution: We will take an hour as the unit of time. Thus

λ = 6 customers/hour, μ = 12 customers/hour.

The customer will not have to wait if there are no customers in the bank. Thus

P0 = 1 – λ/μ= 1− 6/12 = 0.5

Expected numbers of customers in the bank are given by

$L_S = \lambda$ /(μ - λ)=6/6=1

Expected time to be spent in the bank is given by W_S=1/(

μ – λ)= 1/(12-6) = 1/6 hour = 10 minutes.

8) Markov Chains and its applications

a) Markov chains and Markov Process

Important classes of stochastic processes are Markov chains and Markov processes. A Markov chain is a discrete-time process for which the future behavior, given the past and the present, only depends on the present and not on the past. A Markov process is the continuous-time version of a Markov chain. Many queuing models are in fact Markov processes. This chapter gives a short introduction to Markov chains and

Markov processes focusing on those characteristics that are needed for the modeling and analysis of queuing problems.

A Markov chain

A Markov chain, named after Andrey Markov, is a mathematical system that undergoes transitions from one state to another, between a finite or countable number of possible states. It is a random process characterized as memoryless: the next state depends only on the current state and not on the sequence of events that preceded it. This specific kind of "memorylessness" is called the Markov property. Markov chains have many applications as statistical models of real-world processes.

Formally

A Markov chain is a sequence of random variables X1, X2, X3, ... with the Markov property, namely that, given the present state, the future and past states are independent. i.e

$$\Pr(X_{n+1}=x|X_1=x_1,X_2=x_2,\ldots,X_n=x_n)=\Pr(X_{n+1}=x|X_n=x_n).$$

Example; A simple whether model

The probabilities of weather conditions (modeled as either rainy or sunny), given the weather on the preceding day, can be represented by a transition matrix:

$$P=\begin{bmatrix}0.9 & 0.1\\ 0.5 & 0.5\end{bmatrix}$$

The matrix P represents the weather model in which a sunny day is 90% likely to be followed by another sunny day, and a rainy day is 50% likely to be followed by another rainy day. The columns can be labelled "sunny" and "rainy" respectively, and the rows can be labeled in the same order.

(P)i j is the probability that, if a given day is of type i, it will be followed by a day of type j.

Notice that the rows of P sum to 1: This is because P is a stochastic matrix.

The weather on day 0 is known to be sunny. This is represented by a vector in which the "sunny" entry is 100%, and the "rainy" entry is 0%:

$$\mathbf{x}^{(0)}=\begin{bmatrix}1 & 0\end{bmatrix}$$

The weather on day 1 can be predicted by:

$$\mathbf{x}^{(1)}=\mathbf{x}^{(0)}P=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}0.9 & 0.1\\ 0.5 & 0.5\end{bmatrix}=\begin{bmatrix}0.9 & 0.1\end{bmatrix}$$

The weather on day 2 can be predicted in the same way:

$$\mathbf{x}^{(2)}=\mathbf{x}^{(1)}P=\begin{bmatrix}0.9 & 0.1\end{bmatrix}\begin{bmatrix}0.9 & 0.1\\ 0.5 & 0.5\end{bmatrix}=\begin{bmatrix}0.86 & 0.14\end{bmatrix}$$

Or

$$\mathbf{x}^{(2)}=\mathbf{x}^{(1)}P=\mathbf{x}^{(0)}P^2=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}0.9 & 0.1\\ 0.5 & 0.5\end{bmatrix}^2=\begin{bmatrix}0.86 & 0.14\end{bmatrix}$$

In general

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} P = \begin{bmatrix} 0.9 & 0.1 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.86 & 0.14 \end{bmatrix}$$

General rules for day *n* are:

$$\mathbf{x}^{(n)} = \mathbf{x}^{(n-1)} P$$
$$\mathbf{x}^{(n)} = \mathbf{x}^{(0)} P^n$$

b) Markov chain or process Applications

Physics

Markovian systems appear extensively in thermodynamics and statistical mechanics, whenever probabilities are used to represent unknown or unmodelled details of the system, if it can be ssumed that the dynamics are time-invariant, and that no relevant history need be considered which is not already included in the state description.

Queueing theory

Markov chains are the basis for the analytical treatment of queues (queueing theory). Agner Krarup Erlang initiated the subject in 1917. This makes them critical for optimizing the performance of telecommunications networks, where messages must often compete for limited resources (such as bandwidth).

Internet applications

The Page Rank of a webpage as used by Google is defined by a Markov chain. It is the probability to be at page i in the stationary distribution on the following Markov chain on all (known) web pages

Statistics

Markov chain methods have also become very important for generating sequences of random numbers to accurately reflect very complicated desired probability distributions, via a process called Markov chain Monte Carlo (MCMC) And many more.

9) Differential and partial deferential equations

Continuous model

When continuous system is modeled mathematically, the variables of model representing the attribute of system are controlled by continuous functions. The distributed lag model is an example of a continuous model. Since in continuous system, the relationship between variables describe the rate at which the value of variable change, these system consist of differential equations.

Continuous system simulation uses the notation of differential equation to represent the change on the basic parameter of the system with respect to time. Hence the Mathematical model for continuous system simulation is usually represented by differential and partial differential equations.

Differential Equations

An example of a linear differential equation with constant coefficients to describe the wheel suspension system of an automobile can be given as

$$M\ddot{x} + Dx + Kx \doteq KF(t)$$

Here the dependent variable x appears together with first and second derivatives single dot and double dot respectively.

The simple differential equation can model the simplest continuous system and they an have one or more linear differential equation with constant coefficients. It is then often possible to solve the model without using simulation technique i.e. we can solve such equations using analytical methods as (we have done in Numerical methods)

However when non linearity involves into the model, it may be impossible or at least very difficult to solve such model without simulation.

Partial Differential Equations

When more than one independent variable occurs in a differential equation the equation is sait to be partial differential equations. It can involve the derivatives of the same dependent variable with respect to each of the independent variable.

Differential equations both linear and nonlinear occur frequently in scientific and engineering studies. The reason for this is that most physical and chemical process involves rates of change, which require differential equation to represent their mathematical descriptions.

Since partial deferential equation can also represent a growth rate, continuous model can also be applied to the problems of a social or economic nature.

Analog Computer

Before the invention of digital computer, there existed devices whose behavior is equivalent of mathematical operation such as addition or subtraction ot integration. Putting together these device in a manner specified by a mathematical model or equation of a system , allowed us to simulate the system.

Some devices have been created for simulation continuous system and called analog computer or differential analyzer.

Digital analog simulators

To avoid the disadvantages of analog computers, many digital computer programming language have been written to produce digital-analog simulators. They allow or facilitate a continuous model to be programmed on a digital computer in essentially the same way as it is solved on analog computer. The language contains micro instructions that carry the action of addition, integration and sign changer. A program is written to link together these micro instructions in the same way as operational amplifiers are connected in analog computer.

Since more powerful digital computer and programming language have been developed for this purpose of simulating continuous system on digital computer, the digital-analog simulators are now in extensive use.

Exercise

(Please find the answer from above notes and prepare it on your copy) Long questions

1) What do you mean by Queuing system? Explain the characteristics of Queuing system with example.

OR

Define the queuing system. Explain the elements of queuing system with example.

2) Explain about the Poison arrival process and Service process with example 3)

Short questions

1) Define a Markov chains and its application OR

What are the key features of Markov chains?

2) Explain about the server utilization and Traffic intensity.
3) What do you mean by multi server queues?
4) What are the kendall notations of queuing system? OR

What do you mean by Queuing notation? Explain with example 5)
Explain about the Queuing Discipline and behaviors. 6) Explain
about the uses of differential equations in simulations.

Chapter 3

Random Number Generations

3.1 Introductions

Random numbers are a necessary basic ingredient in the simulation of almost all discrete systems. Most computer languages have a subroutine, object, or function that will generate a random number. Similarly simulation languages generate random numbers that arc used to generate event limes and other random variables.

3.2 Random Number Tables

A table of numbers generated in an unpredictable, haphazard that are uniformly distributed within certain interval are called random number table.

The random number in random number table exactly obey two random number properties: uniformity and independence so random number generated form table also called true random numbers.

Table of random numbers are used to create a Radom sample. A random number table is also called random sample table.

There are many physical devices or process that can be used to generate a sequence of uniformly distributed random numbers i.e. true random numbers. For example: An electrical pulse generator can be made to drive a counter cycling from 0 to 9. Using an electronic noise generator or radioactive source the pulse can be generated as random numbers.

3.3 Pseudo Random Numbers

Pseudo means false, so false random numbers are being generated. The goal of any generation scheme is to produce a sequence of numbers between zero and 1 which simulates, or imitates, the ideal properties of uniform distribution and independence as closely as possible. When generating pseudo-random numbers, certain problems or errors can occur.

Some examples of errors includes the following

1. The generated numbers may not be uniformly distributed.

2. The generated numbers may be discrete -valued instead continuous valued

3. The mean of the generated numbers may be too high or too low.

4. The variance of the generated numbers may be too high or low

5. There may be dependence. The following are examples:

(a) Autocorrelation between numbers.

(b) Numbers successively higher or lower than adjacent numbers.

(c) Several numbers above the mean followed by several numbers below the mean.

3.4 Properties of Good random Number Generators

Usually, random numbers are generated by a digital computer as part of the simulation.

Numerous methods can be used to generate the values. In selecting among these methods, or

routines, there are a number of important considerations.

1. The routine should be fast. . The total cost can be managed by selecting a computationally efficient method of random-number generation.

2. The routine should be portable to different computers, and ideally to different programming languages .This is desirable so that the simulation program produces the same results wherever it is executed.

3. The routine should have a sufficiently long cycle. The cycle length, or period, represents the length of the random-number sequence before previous numbers begin to repeat themselves in an earlier order. Thus, if 10,000 events are to be generated, the period should be many times that long, A special case cycling is degenerating. A routine degenerates when the same random numbers appear repeatedly. Such an occurrence is certainly unacceptable. This can happen rapidly with some methods.

4. The random numbers should be replicable. Given the starting point (or conditions), it should be possible to generate the same set of random numbers, completely independent of the system that is being simulated. This is helpful for debugging purpose and is a means of facilitating comparisons between systems.

5. Most important, and as indicated previously, the generated random numbers should closely approximate the ideal statistical properties of uniformity and independences

3.5 Method to Generate Random Numbers

3.5.1 Linear Congruential Method

The linear congruential method, initially proposed by Lehmer [1951], produces a sequence of integers, *X\*, *X2,...* between zero and *m* — 1 according to the following recursive relationship:

Xi+1 = (a Xi + c) mod m, i = 0,1, 2,…………………………….Equation (3.1)

The initial value X0 is called the seed, *a* is called the constant multiplier, c is the increment, and *m* is the modulus.

Case 1:

If *c* ≠ 0 in Equation (7.1), the form is called the *mixed congruential method.*

Case 2

When *c* = 0, the form is known as the *multiplicative congruential method.* The selection of the values for a, *c,* m and Xo drastically affects the statistical properties and the cycle length. . An example will illustrate how this technique operates.

EXAMPLE 3.1

Use the linear congruential method to generate a sequence of random numbers with *X0* = 27, *a*= 17, *c* = 43, and *m* = 100. Here, the integer values generated will all be between zero and 99 because of the value of the modulus. These random integers should appear to be uniformly distributed the integers zero to 99.

Random numbers between zero and 1 can be generated by Ri =Xi/m, i= 1,2,…… **equation (3.2)**

The sequence of Xi and subsequent Ri values is computed as follows:

X0 = 27

X1 = (17.27 + 43) mod 100 = 502 mod 100 = 2

R1=2⁄100=0. 02

X2 = (17 • 2 + 43) mod 100 = 77 mod 100 = 77

R2=77 ⁄100=0. 77

X3 = (17•77+ 43) mod 100 = 1352 mod 100 = 52
R3=52 ⁄100=0. 52

First, notice that the numbers generated from Equation (7.2) can only assume values from the set I = {0,1 /m, 2/m,..., (m — l)/m), since each *Xi* is an integer in the set {0,1,2,..., m -1}. Thus, each *Ri* is discrete on I, instead of continuous on the interval [0, 1], This approximation appears to be of little consequence, provided that the modulus m is a very large integer.

(Values such as *m* = 2^{31} -1 and *m* = 2^{48} are in common use in generators appearing in many simulation languages.)

By maximum density is meant that the values assumed by *Ri* = 1,2,..., leave no large gaps on [0,1].

EXAMPLE 3.2

Let *m* = 102 = 100, *a* = 19, *c* = 0, and *X0* = 63, and generate a sequence c random integers using

Xi+1 = (a Xi + c) mod m.

X0 = 63

X1 = (19)(63) mod 100 = 1197 mod 100 = 97

X2 = (19) (97) mod 100 = 1843 mod 100 = 43

X3 = (19) (43) mod 100 = 817 mod 100 = 17

. .

When m is a power of 10, say *m* = 10b *,* the modulo operation is accomplished by saving the *b* rightmost (decimal) digits.

EXAMPLE 4.4

Let a = 7^5 = 16,807, *m* = 2^{31}-1 = 2,147,483,647 (a prime number), and c= 0. These choices satisfy the conditions that insure a period of *P = m-1.* Further, specify a seed, X_0 = 123,457.

The first few numbers generated are as follows:

X1= 75(123,457) mod (2^{31} - 1) = 2,074,941,799 mod (2^{31} - 1)
X1 = 2,074,941,799

$R1= X1 / 2^{31}$

$X2 = 75(2,074,941,799) \bmod (2^{31} - 1) = 559,872,160$ $R2$

$= X2 / 2^{31}= 0.2607$

$X3 = 75(559,872,160) \bmod (2^{31} - 1) = 1,645,535,613$

$R3 = X3 / 2^{31}= 0.7662$

3.5.2 Non Uniform Transformation Method

3.5.3 Acceptance /rejection method

3.5 Testing for Randomness

The desirable properties of random numbers — uniformity and independence To insure that these desirable properties are achieved, a number of tests can be performed (fortunately, the appropriate tests have already been conducted for most commercial simulation software}. The tests can be placed in two categories according to the properties of interest.

a) Testing for uniformity
b) Testing for independence.

Testing for uniformity
The testing for uniformity can be achieved through different frequency test. These tests use the Kolmogorov-Smirnov or the chi- square test to compare the distribution of the set of numbers generated to a uniform distribution.
Hence in this category we will discuss two types of test

a) Kolmogorov-Smirnov test
b) Chi- square test

Test for independence includes the five types of tests as given below

a) Autocorrelation test Tests the correlation between numbers and compares the sample correlation to the expected correlation of zero.

b) Gap test. Counts the number of digits that appear between repetitions of particular digit and then uses the Kolmogorov-Smirnov test to compare with the expected size of gaps,

c) **Poker test** . Treats numbers grouped together as a poker hand. Then the hands obtained are compared to what is expected using the chi-square test.

The detail description of each of these tests is given below. In testing for uniformity, the hypotheses are as follows:

H0: Ri ~ U/[0,1] H1:
Ri ~U/[0,l]

The null hypothesis, H0 reads that the numbers are distributed uniformly on the interval [0, 1]. Failure to reject the null hypothesis means that no evidence of non-uniformity has been detected on the basis of this test. This does not imply that further testing of the generator for uniformity is unnecessary.

For each test, a level of significance *a* must be stated. The level *a* is the probability of rejecting the null hypothesis given that the null hypothesis is true, or *a* = *P* (reject H_0 |H_0 true)
The decision maker sets the value of *&* for any test. Frequently, *a* is set to 0.01 or O.05.

1. The Kolmogorov-Smirnov test.

This test compares the continuous cdf, F(X), of the uniform distribution to the empirical cdf, $S_N(x)$, of the sample of N observations.

By definition,

F(x) = x, 0 <= x <= 1

If the sample from the random-number generator is R_1 R_2, ,• • •, RN, then the empirical cdf, $S_N(X)$, is defined by

$S_N(X)$ =(number of R_1 R_2, ,• • •, Rn which are <= x)/N

As N becomes larger, $S_N(X)$ should become a better approximation to F(X) , provided that the null hypothesis is true.

The **Kolmogorov-Smirnov test** is based on the largest absolute deviation or difference between F(x) and $S_N(X)$ over the range of the random variable. I.e. it is based on the statistic D
= max | F(x) - $S_N(x)$| ……………………………equation (3.1)

For testing against a uniform cdf, the test procedure follows these steps:

Algorithm for K-S test

Step 1. Rank the data from smallest to largest. Let $R_{(i)}$ denote the i th smallest observation, so that R $_{(1)}$<= R $_{(2)}$<= • • • <= R $_{(N)}$

Step 2. Compute

$$D^{+} = \max\left\{\frac{i}{N} - R_i\right\}$$

$$D^{-} = max\left\{R_i - \frac{(i-1)}{N}\right\}$$

Step 3: Compute D=max{D+, D^-}

Step 4. Determine the critical value, Dα, from Table A.8(in your Text book) for the specified significance level α and the given sample size N.

Step 5.

If the sample statistic D is greater than the critical value, Dα, the null hypothesis that the data are a sample from a uniform distribution is rejected.

If D <= Da, conclude that no difference has been detected between the true distribution of **{** R1 R2, ,• • •, Rn **}** and the uniform distribution. Hence the null hypothesis is accepted.

Example

Suppose that the five numbers 0.44, 0.81, 0.14, 0.05, 0.93 were generated, and it is desired to perform a test for uniformity using the Kolmogorov-Smirnov test with a level of significance a of 0.05.

Solution

First, the numbers must be ranked from smallest to largest. I.e the given numbers 0.05, 0.14 0.44 , 0.81, 0.93. The calculations can be facilitated by use of Table below.

R_i	0.05	0.14	0.44	0.81	0.93
i/N	0.2	0.4	0.6	0.8	1.0
i/N-R_i	0.15	0.26	**0.16**	-	0.07
R_i –(i-1)/N	0.15	-	**0.04**	0.21	0.13

For example

At $R_{(3)}$ the value of D_+ is given by 3/5 - $R_{(3)}$ = 0.60 - 0.44 =0.16 and of D_- is given by $R_{(3)}$ = 2/5 = 0.44 - 0.40 = 0.04. and other value also can be computed similarly.

Now The statistics are computed as D+ = 0.26 (Maximum of the row i/N-Ri) and D- = 0.21 (maximum of the row R_i –(i-1)/N) . Therefore, D = max{0.26, 0.21} = 0.26.

The critical value of **D,** obtained from Table A.8 (in Text book) for a = 0.05 and N = 5, is 0.565.

Since the computed value, 0.26, is less than the tabulated critical value, 0.565, the hypothesis of no difference between the distribution of the generated numbers and the uniform distribution is not rejected.

2. Chi- Square Test

The chi-square test uses the sample statistic

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

where Oi; is the observed number in the *i* th class, *Ei* is the expected number in the ith class, and *n* is the number of classes. For the uniform distribution, *Ei* the expected number in each class is given by

Ei = N/n for equally spaced classes, where *N* is the total number of observations. It can be shown that the sampling distribution of X_0^2 is approximately the chi-square distribution with *n* - 1 degrees of freedom.

Example 2.1

Use the chi-square test with α = 0.05 to test whether the data shown below are uniformly distributed.

0.34, 0.83, 0.96, 0.47, 0.79, 0.99, 0.37, 0.72, 0.06, 0.18,
0.90, 0.76, 0.99, 0.30, 0.71, 0.17, 0.51, 0.43, 0.39,0.26
0.25 0.79 0.77, 0.17 0.23 0.99 0.54 0.56 0.84 0.97 0.89
0.64 ,0.67 0.82 0.19 0.46 0.01 0.97 0.24 0.88 0.87
0.70 0.56 0.56 0.82 0.05 0.81 0.30 0.40 0.64
0.44 0.81 0.41 0.05 0.93 0.66 0.28 0.94 0.64
0.47 0.12 0.94 0.52 0.45 0.65 0.10 0.69 0.96
0.40 0.60 0.21 0.74 0.73 0.31 0.37 0.42 0.34
0.58 0.19 0.11 0.46 0.22 0.99 0.78 0.39 0.18
0.75 0.73 0.79 0.29 0.67 0.74 0.02 0.05 0.42
0.49, 0.49 0.05 0.62 0.78

Solutions

The table for chi square statistics is

Class interval(i)	Oi	Ei	(Oi – Ei)	$(Oi-Ei)^2$	$(Oi-Ei)^2/Ei$
1	8	10	-2	4	0.4
2	8	10	-2	4	0.4

3	10	10	0	0	0.0
4	9	10	-1	1	0.1
5	12	10	2	4	0.4
6	8	10	-2	4	0.4
7	10	10	0	0	0.0
8	14	10	14	16	0.16
9	10	10	0	0	0.0
10	11	10	1	1	0.1
Total	N=100	N=100			$\sum = 3.4$

Above Table contains the essential computations for chi square test.The test uses n = 10 intervals of equal length, namely [0.0, 0.1), [0.1, 0.2), . . . , [0.9, 1.0). The value of X^2 is 3.4.

Here degree of freedom is n-1=10-1=9 and α=0.05. The tabulated value of $X^2_{0.05,9}$=16.9.Since X_0^2 is much smaller than the tabulated value of chi square, the null hypothesis of a uniform distribution is not rejected.

3. Tests for Autocorrelation

The tests for autocorrelation are concerned with the dependence between numbers in a sequence. As an example, consider the following sequence of numbers:

0.12 0.01 0.23 0.28 0.89 0.31 0.64 0.28 0.83 0.93 0.99 0.15 0.33 0.35 0.91 0.41 0.60 0.27 0.75 0.88 0.68 0.49 0.05 0.43 0.95 0.58 0.19 0.36 0.69 0.87

From a visual inspection, these numbers appear random, and they would probably pass all the tests presented to this point. However, an examination of the 5th, 10th, 15th (every five numbers beginning with the fifth), and so on. Indicates a very large number in that position.

Now, 30 numbers is a rather small sample size to reject a random-number generator, but the notion is that numbers in the sequence might be related. In this particular section, a method for determining whether such a relationship exists is described. The relationship would not have to be all high numbers. It is possible to have all low numbers in the locations being examined, or the numbers may alternately shift from very high to very low.

Autocorrelation Test

Autocorrelation test is a statistical test that determines whether a random number generator is producing independent random number in a sequence.

The test for the auto correlation is concerned with the dependence between number in a sequence. The test computes the auto correlation between every m numbers (m is also known as lag) starting with ith index.

The variables involved in this test are:

m→ is the lag, the space between the number being tested.
i-→ is the index or number form we start.
N→ is the number of random numbers generaeted. M→
is the largest integer such that
i+(M+1)m<=N

Now the autocorrelation ρ_{im} between R_i, Ri+m, Ri+2m, ...Ri+(M+1)m is computed as

$$\rho_{im} = \frac{1}{M+1}\left[\sum_{k=0}^{M} R_{i+km}R_{i+(k+1)m}\right] - 0.25$$

Now the test statistics is

$$Z_0 = \frac{\rho_{im}}{\sigma_{im}}$$

Where

$$\sigma_{im} = \frac{\sqrt{13M+7}}{12(M+1)}$$

After computing Z0, do not reject the null hypothesis of independence if - **$z_{a/2}$** <= **Z0** <= **$z_{a/2}$**, where *a* is the level of significance.

Example 3.1

Test whether the 3rd, 8th, 13th, and so on, numbers in the sequence at the beginning of this section are auto-correlated. (Use *a* = 0.05.) Here, *i* = 3 (beginning with the third number), *m* = 5 (every five numbers), *N* = 30 (30 numbers in the sequence).

Solution:

First we calculate the value of *M using the condition* i+(M+1)m<=N

since i=3, m=5, and N=30 we have

3 + (M +1)5 <= 30.

i.e 3+5M+5<=30 I.e.5M<=22 i.eM<=22/5$\approx$4 hence M=4 Then, ρ_{35} = 1/ 4 + 1**[**

(0.23)(0.28) + (0.28)(0.33) + (0.33)(0.27) + (0.27)(0.05) + (0.05)(0.36) **]** **=** -0.1945

And σ_{35}= √ (13(4) + 7) / 12(4 + 1) =

0.1280

Then, the test statistic assumes the value

Z0 = -0.1945/0.1280 = -1.516

Now, the critical value isZ0.025 = 1.96 (Za/2 is taken in this test)

Therefore, the hypothesis of independence cannot be rejected on the basis of this test.

4. Gap test

The gap test is used to determine the significance of the interval between the recurrences of the same digit. A gap of length *x* occurs between the recurrences of some specified digit. The following example illustrates the length of gaps associated with the digit 3:

4, 1, **3**, 5, 1, 7. 2, 8, 0, 7, 9, 1, **3**. 5, 2, 7, 9, 4, 1, 6, **3** ,**3**, 9, 6, **3**, 4, 8, 2, **3**, 1, 9, 4. 4, 6. 8, 4, 1, **3**.

There are 7 three's are there. Thus only six gap can occurs. The first gap is of length 10 and second gap of length 7 and third gap of length zero. And so on.

Similarly the gap associated with other digits can be calculated. The theoretical probability of first gap (of lenth 10 for digit 3) can be calculated as

$$P(\text{gapof l0}) = \underbrace{P(\text{not }3)\times p(\text{not }3)\ldots\ldots\ldots\ldots\ldots P(\text{no3})}_{\text{10 of these terms}}\ P(3)$$

$$=0.9\times0.9\times\ldots\ldots\ldots\ldots\ldots\times0.9\times0.1$$

$$= (0.9)^{10}\times0.1$$

Hence in general the probability of length x is

P(t followed by exactly *x* non-r digits) = $(0.9)^x(0.1)$, X = 0,1,2………….equation 4.1

Gap Test Algorithm

The procedure for the test follows the steps below. When applying the test to random numbers, class intervals such as [0, 0.1), [0.1,0.2),. . . play the role of random digits.

Step 1. Specify the cdf for the theoretical frequency distribution given by Equation (4.1) based on the selected class interval width.

Step2.Arrange the observed sample of gaps in a cumulative distribution with these same classes.

Step 3. Find D, the maximum deviation between *F(x)* and *SN(X)* as in K-V test.

Step 4. Determine the critical value, Da, from Table for the specified value of *a* and the sample size *N.*

Step 5. If the calculated value of *D* is greater than the tabulated value of Da, the null hypothesis of independence is rejected.

5. Poker Test

The poker test for independence is based on the frequency with which certain digits are repeated in a series of numbers. The following example shows an unusual amount of repetition:

0.255, 0.577, 0.331, 0.414, 0.828, 0.909, 0.303,0.001, ...

-the poker test uses the chi square statistics to accept or reject the null hypothesis.

In each case, a pair of like digits appears in the number that was generated. In three-digit numbers there are only three possibilities, as follows:

1. The individual numbers can all be different.
2. The individual numbers can all be the same.
3. There can be one pair of like digits.

The probability associated with each of these possibilities is given by the following

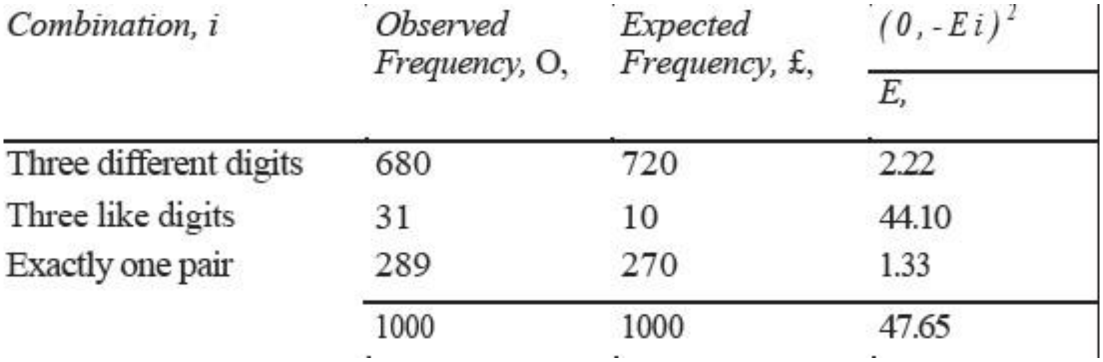
P(three different digits) = *P*(second different from the first) x P(third different from the first and second) = (0.9)(0.8) = 0.72

P(three like digits) = P(second digit same as the first) x *P* (third digit same as the first)
= (0.1)(0.1) = 0.01

P(exactly one pair) = 1 - 0.72 - 0.01 = 0.27

Example 5.1
A sequence of 1000 three-digit numbers has been generated and an analysis indicates that 680 have three different digits, 289 contain exactly one pair of like digits, and 31 contain three like digits. Based on the poker test, are these numbers independent? Let *α* = 0.05. Test these numbers using **poker test for three digit.**

The test is summarized in Table 7.7.



Combination, i	*Observed Frequency, O,*	*Expected Frequency, £,*	$\frac{(0,-Ei)^2}{E,}$
Three different digits	680	720	2.22
Three like digits	31	10	44.10
Exactly one pair	289	270	1.33
	1000	1000	47.65

The appropriate degrees of freedom are one less than the number of class intervals. Since 47.65 > $X^2_{0.05, 2}$=5.99 (tabulated value), the independence of the numbers is rejected on the basis of this test. Here 2 or n-1 is the degree of freedom since there are only 3 (n) class.

Example 5.2
Explain the independence test. A sequence of 1000 four digit numbers has been generated and an analysis indicates the following combinations and frequencies.

Combination (i)	**Observed frequency (Oi)**
Four different digits	560
One pair	394
Two pair	32
Three digits of a kind	13
Four digit of a kind	1
	1000

Based on poker test, test whether these numbers are independent. Use α=0.05 and N=4 is 9.49.

Solution

In four digit number, there are five different possibilities

a. All individual digits can be different
b. There can be one pair of like digit
c. There can be two pair of like digits
d. There can be three digits of a kind
e. There can be four digits of a kind

The probabilities associated with each of the possibilities is given by

P(four different digits)=$^4c_4$ x10/10x9/10x8/10x7/10=0.504

P(one pair)= $^4c_2$ x10/10x1/10x9/10x8/10=0.432

P(two pair)= $^4c_2$ x10/10x1/10x9/10x1/10=0.027

P(three digits of a kind)= $^4c_3$ x10/10x1/10x1/10x9/10=0.036

P(four digits of a kind)= $^4c_4$ x10/10x1/10x1/10x1/10=0.001

Now the calculation table for the Chi-square statistics is

Combination (i)	**Observed frequency (Oi)**	**Expected frequency(Ei)**	**(oi-Ei)**	**(oi-Ei)^2/Ei**
Four different digits	560	0.504x1000=504	56	6.22
One pair	394	0.432x1000=432	-38	3.343
Two pair	32	0.027x1000=27	5	0.926
Three digits of a kind	13	0.036	-23	14.694
Four digit of a kind	1	0.0001x1000=1	0	0.000
	1000	1000		25.185

Here the calculated value of chi-square is 25.185 which is greater than the given value of chisquare so we reject the null hypothesis of independence between given numbers.

Exercise Short Questions

1. What do you mean by Pseudo random numbers?
2. Explain non-uniform random number generation.

3. Use the linear congruential method to generate a sequence of three two-digit random integers. Let X0=29, a=9, c=49 and m=100.
4. Use the mixed congruential method to generate a sequence of three two digit random numbers with Xo=37, a=7,c=29 and m=100.
5. Explain the congruence method of generating random numbers.

Long Questions

1. What is the main objective of gap test? Explain gap test algorithm with example. 2.
Explain the process of Poker test for four digit numbers.

Chapter 4

Verification and validation of simulation Model

One of the most important and difficult tasks facing a model developer is the verification and validation of the simulation model. · It is the job of the model developer to work closely with the end users throughout the period (development and validation to reduce this skepticism and to increase the credibility.

The goal of the validation process is twofold:

1. To produce a model that represents true system behavior closely enough for the model to be used as a substitute for the actual system for the purpose of experimenting with system.
2. To increase an acceptable, level the credibility of the model ,so that the model will be used by managers and other decision makers.

The verification and validation process consists of the following components:-

1. Verification is concerned with building the model right. It is utilized in comparison of the conceptual model to the computer representation that implements that conception. It asks the questions: Is the model implemented correctly in the computer? Are the input parameters and logical structure of the model correctly represented?

2. Validation is concerned with building the right model. It is utilized to determine that a model is an accurate representation of the real system. It is usually achieved through the calibration of the model.

4.1 Model Building

The first step in model building consists of observing the real system and the interactions among its various components and collecting data on its behavior. Operators, technicians, repairs and maintenance personnel, engineers, supervisors, and managers under certain aspects of the system which may be unfamiliar to others. As model development proceeds, new questions may arise, and the model developers will return, to this step of learning true system structure and behavior.

The second step in model building is the construction of a conceptual model – a collection of assumptions on the components and the structure of the system, plus hypotheses on the values of model input parameters, illustrated by the following figure.

The third step is the translation of the operational model into a computer recognizable form- the computerized model.

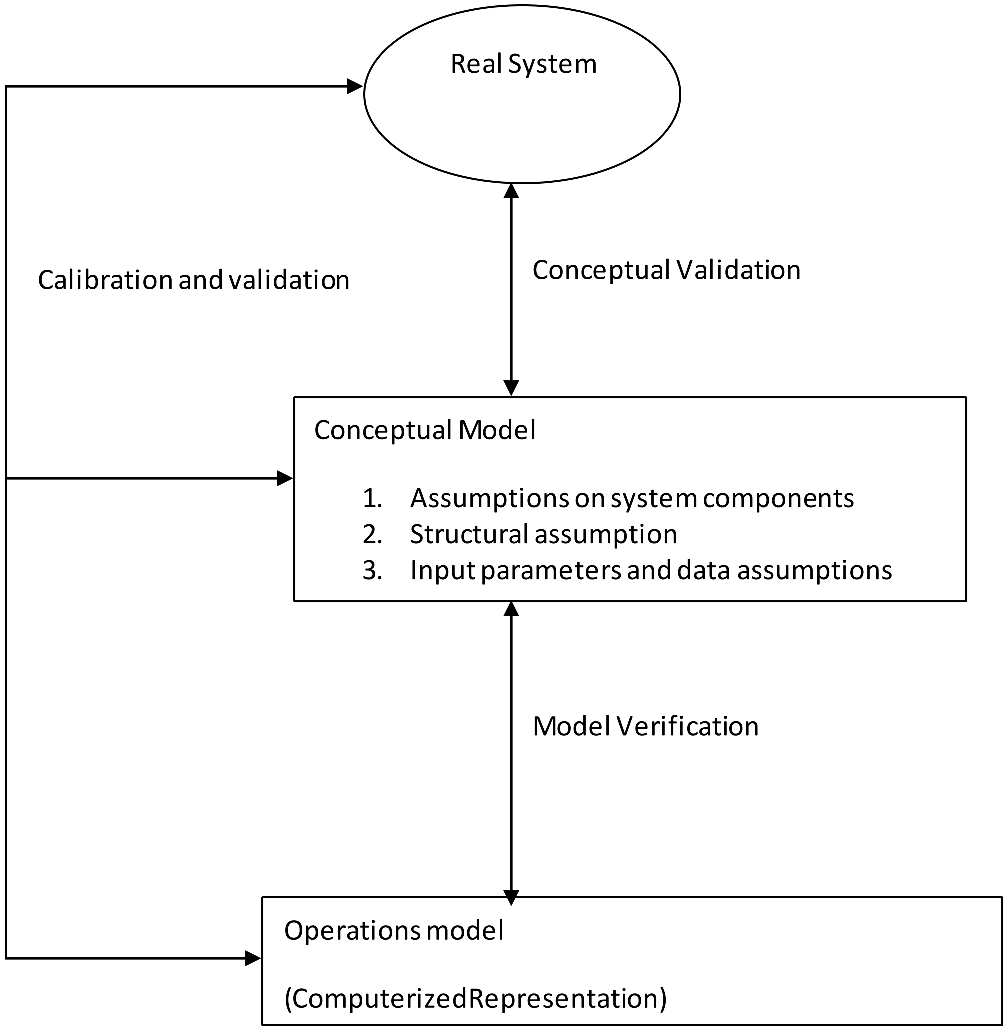

Real System
Calibration and validation
Conceptual Validation
Conceptual Model
1. Assumptions on system components
2. Structural assumption
3. Input parameters and data assumptions
Model Verification
Operations model
(ComputerizedRepresentation)

Figure: Model building, Verification and Validation

Verification of Simulation Models

The purpose of model verification is to assure that the conceptual model is reflected accurately in the computerized representation. The conceptual model quite often involves some degree of abstraction about system operations, or some amount of simplification of actual operations.

Many common-sense suggestions can be given for use in the verification process:-

1. Have the computerized representation checked by someone other than its developer.
2. Make a flow diagram which includes each logically possible action a system can take when an event occurs, and follow the model logic for each a for each action for each event type
3. Closely examine the model output for reasonableness under a variety of settings of input parameters.
4. Have the computerized representation print the input parameters at the end of the simulation to be sure that these parameter values have not been changed inadvertently.
5. Make the computerized representation of self-documenting as possible.
6. If the computerized representation is animated, verify that what is seen in the animation imitates the actual system.
7. The interactive run controller (IRC) or debugger is an essential component of successful simulation model building. Even the best of simulation analysts makes mistakes or commits logical errors when building a model. The IRC assists in finding and correcting those errors in the following ways:
   (a) The simulation can be monitored as it progresses.
   (b) Attention can be focused on a particular line of logic or multiple lines of logic that constitute a procedure or a particular entity
   (c) Values of selected model components can be observed. When the simulation has paused, the current value or status of variables, attributes, queues, resources, counters, etc. can be observed.
   (d) The simulation can be temporarily suspended, or paused, not only to view information but also to reassign values or redirect entities

Calibration and Validation of Models

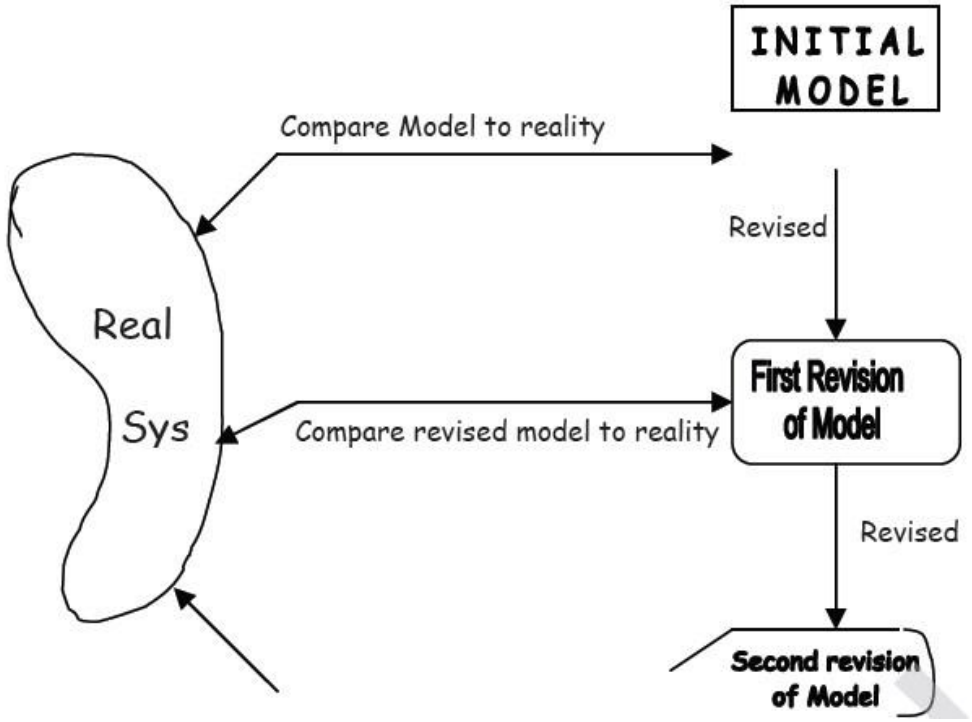


Figure: the calibration of model

- Verification and validation although are conceptually distinct, usually are conducted simultaneously by the modeler. · Validation is the overall process of comparing the model and its behavior to the real system and its behavior.

- Calibration is the iterative process of comparing the model to the real system, making adjustments to the model, comparing again and so on.

- The above figure shows the relationship of the model calibration to the overall validation process. The comparison of the model to reality is carried out by variety of test.

- Tests are subjective and objective. Subjective test usually involve people, who are knowledgeable about one or more aspects of the system, making judgments about the model and its output.

- Objective tests always require data on the system's behavior plus the corresponding data produced by the model.

- A possible criticism of the calibration phase, were it to stop at point, ie., the model has been validated only for the one data set used; that is, the model has been "fit" to one data set

- Validation is not an either/or proposition—no model is ever totally representative of the system under study. In addition, each revision of the model, as in the Figure above involves some cost, time, and effort.

Validation Process

As an aid in the validation process, Naylor and Finger [1967] formulated a three step approach which has been widely followed:-

1. Build a model that has high face validity.
2. Validate model assumptions.
3. Compare the model input-output transformations to corresponding input-output transformations for the real system.

1. Face Validity

- The first goal of the simulation modeler is to construct a model that appears reasonable on its face to model users and others who are knowledgeable about the real system being simulated.
- The users of a model should be involved in model construction from its conceptualization to its implementation to ensure that a high degree of realism is built into the model through reasonable assumptions regarding system structure, and reliable data.
- Another advantage of user involvement is the increase in the models perceived validity or credibility without which manager will not be willing to trust simulation results as the basis for decision making.
- · Sensitivity analysis can also be used to check model's face validity.
- The model user is asked if the model behaves in the expected way when one or more input variables is changed.
- Based on experience and observations on the real system the model user and model builder would probably have some notion at least of the direction of change in model output when an input variable is increased or decreased.
- The model builder must attempt to choose the most critical input variables for testing if it is too expensive or time consuming to: vary all input variables.

2. Validation of Model Assumptions

Model assumptions fall into two general classes: structural assumptions and data assumptions.

a) Structural assumptions involve questions of how the system operates and usually involve simplification and abstractions of reality.

   For example, consider the customer queuing and service facility in a bank. Customers may form one line, or there may be an individual line for each teller. If there are many lines, customers may be served strictly on a first-come, first-served basis, or some customers may change lines if one is moving faster. The number of tellers may be fixed or variable. These structural assumptions should be verified by actual observation during appropriate time periods together with discussions with managers and tellers regarding bank policies and actual implementation of these policies.

b) Data assumptions should be based on the collection of reliable data and correct statistical analysis of the data.data were collected on.
   1) Inter arrival times of customers during several 2-hour periods of peak loading ("rush-hour" traffic)
   2) Inter arrival times during a slack period.
   3) Service times for commercial accounts.
   4) Service times for personal accounts.

The procedure for analyzing input data consist of three steps:- 1:
Identifying the appropriate probability distribution.
2: Estimating the parameters of the hypothesized distribution .
3: Validating the assumed statistical model by goodness – of – fit test such as the chi-square test, K-S test and by graphical methods.

3. Validating Input-Output Transformation:-

- In this phase of validation process the model is viewed as input – output transformation.
- That is, the model accepts the values of input parameters and transforms these inputs into output measure of performance. It is this correspondence that is being validated.
- Instead of validating the model input-output transformation by predicting the future ,the modeler may use past historical data which has been served for validation purposes that is, if one set has been used to develop calibrate the model, its recommended that a separate data test be used as final validation test.
- Thus accurate " prediction of the past" may replace prediction of the future for purpose of validating the future.
- A necessary condition for input-output transformation is that some version of the system under study exists so that the system data under at least one set of input condition can be collected to compare to model prediction.

- If the system is in planning stage and no system operating data can be collected, complete input-output validation is not possible.
- Validation increases modeler's confidence that the model of existing system is accurate.
- Changes in the computerized representation of the system, ranging from relatively minor to relatively major include :

1.Minor changes of single numerical parameters such as speed of the machine, arrival rate of the customer etc.

2: Minor changes of the form of a statistical distribution such as distribution of service time or a time to failure of a machine.

3: Major changes in the logical structure of a subsystem such as change in queue discipline for waiting-line model, or a change in the scheduling rule for a job shop model.

4: Major changes involving a different design for the new system such as computerized inventory control system replacing a non computerized system.

- If the change to the computerized representation of the system is minor such as in items one or two these change can be carefully verified and output from new model can be accepted with considerable confidence. Partial validation of substantial model changes in item three and four may be possible.

3.1 Input-Output Validation: Using Historical Input Data

- When using artificially generated data as input data the modeler expects the model produce event patterns that are compatible with, but not identical to, the event patterns that occurred in the real system during the period of data collection.
- Thus, in the bank model, artificial input data {X\n, X2n, n = 1,2, , .} for inter arrival and service times were generated and replicates of the output data Y2 were compared to what was observed in the real system · An alternative to generating input data is to use the actual historical record, {An, Sn, n = 1,2,...}, to drive simulation model and then to compare model output to system data.
- To implement this technique for the bank model, the data Ai, A2,..., S1 S2 would have to be entered into the model into arrays, or stored on a file to be read as the need arose.
- To conduct a validation test using historical input data, it is important that all input data (An, Sn,...) and all the system response data, such as average delay(Z2), be collected during the same time period.
- Otherwise, comparison of model responses to system responses, such as the comparison of average delay in the model (Y2) to that in the system (Z2), could be misleading.
- responses (Y2 and 22) depend on the inputs (An and Sn) as well as on the structure of the system, or model.

- Implementation of this technique could be difficult for a large system because of the need for simultaneous data collection of all input variables and those response variables of primary interest.

3.2 Input-Output Validation: Using a Turing Test

- In addition to statistical tests, or when no statistical test is readily applicable persons knowledgeable about system behavior can be used to compare model output to system output. For example, suppose that five reports of system performance over five different days are prepared, and simulation outputs are used to produce five "fake" reports. The 10 reports should all be in exactly in the same format and should contain information of the type that manager and engineer have previously seen on the system.

- The ten reports are randomly shuffled and given to the engineers , who is asked to decide which report are fake and which are real.

- If engineer identifies substantial number of fake reports the model builder questions the engineer and uses the information gained to improve the model.

- If the engineer cannot distinguish between fake and real reports with any consistency, the modeler will conclude that this test provides no evidence of model inadequacy.

- This type of validation test is called as TURING TEST.

Exercise

1. Discuss the process of verification and validation.
2. What do you mean by calibration of model? Explain with example.
3. What do you mean by face validity? Why is it important?
4. What is the difference between verification and validation?
5. what are the different technique for input output validation?

Chapter 5

Analysis of Simulation Output

Nature of the problem

Once a stochastic variable has been introduced into simulation model, almost all the system variables describing the system behavior also become stochastic. Hence it needs some statistical method to analyze the simulation output. A large body of statistical methods has been developed over the years to analyze results in science, engineering and other fields.

It seem natural to attempt applying these methods to analyze the simulation output but most of them pre-suppose that the results are mutually independent (IID) and the simulation process almost never produce raw output that is IID. For example: customer waiting times from queuing system are not IID.

Thus it is difficult to apply classical statistical techniques to analysis of simulation model.

Some definitions

a) Independently and identically distributed random variables: Usually a random variable is drawn from an infinite population that has probability distribution with finite mean u and finite varianceσ^2. This mean that the population distribution is not affected by the number of sample already made or doee it change with time. Further the value of sample is not affected in anyway by value of another sample.

   Random variables that meet all these conditions are said to be independently and identically distributed.

b) **Central Limit theorem**: the theorem state that the sum of n iid variables drawn form population that has a mean of mu and a variance of sigma2, is approximately distributed as a normal variable with a mean of µ and a variance of nσ^2.

1. Type of simulation on the basis of output

a) Termination simulation or finite simulation: the termination of a finite simulation takes place at a specified time or is caused by some specific events.

For example: Banks opens at 8:30am with no customers present and 8 of 11 teller working and closed at 4:30 pm.
The terminating simulation runs for some specified duration of time T_E where E is a specified event that stops the simulation. It starts at time 0 under well specified initial condition and ends at the stooping time T_E.

For the banking system, The simulation analyst choose to consider a terminating system because object of interest is one days operations on the bank.

b) Non terminating Simulation (Steady state simulation): the propose of steady study simulation is the study of long run behavior of system. Performance measure is called a steady state parameter if it is a characteristic of the equilibrium distribution of an output stochastic process.

Examples are: Continuously operating communication system where the objective of computation of mean delay of packet in the long run.

Here the non-terminating simulation has:

1. Runs continuously or at least over a very long period of time.
2. Initial conditions defined by analyst.
3. Runs for some analyst specified period of time T_E
4. Study the steady state (long run) properties of the system, properties that are not influenced by the initial condition of model.

(Note: whether a simulation is considered to be terminating or non-terminating depends on both the objective of study and nature of the system)

Estimation Methods

Any normal distribution can be transform into a standard distribution that has a mean of 0 and variance of 1. Let xi (i=1,2,3,……….n) be n iid random variables. Using central limit theorem, we have normal variate

$$z = \frac{\sum_{i=1}^{n} x_i - n\mu}{6\sqrt{n}}$$

Dividing by n in both denominator and numerator, we get

$$z = \frac{\bar{x} - \mu}{6/\sqrt{n}}$$

Where

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

The variable $\bar{x}$ is the sample mean and it can be shown to be a consistent estimator for mean of the population form which sample is drawn.

a) **Point estimation (For discrete data):**

Let discrete time data y1, y2,…..yn with ordinary mean θ, then point estimator

$$\hat{\theta} = \frac{1}{n}\sum_{t=1}^{n} y_t$$

Is unbiased if its expected value is θ i.e. E($\hat{\theta}$)=θ and is biased if . $E(\hat{\theta}) \neq \theta$ and the difference $E(\hat{\theta}) - \theta$ is called bias of $\hat{\theta}$.

For the continuous data { $y(t): \theta < t < T$} with mean ϕ then the point estimator is given by

$$\hat{\phi} = \frac{1}{T_E}\int_{0}^{T_E} y(t)dt$$

b) Interval Estimation

Suppose the model is the normal distribution with mean θ, varience σ^2 (both unknown) and we have a sample of n size then the variance of sample data is

$$S^2 = \frac{1}{(n-1)}\sum_{t=1}^{n}(y_t - \bar{y})^2$$

Where $\bar{y}$ is the sample mean.

Simulation Run statistics

In the estimation method, it is assumed that the observations are mutually independent and the distribution from which they are draws is stationary. Unfortunately many statistics of interest in simulation do not meet these conditions.

For example: consider a single server system in which the arrival occurs with poison distribution and service time has an exponential and queue discipline is FIFO. Suppose the study objective is to measure the mean waiting rime. In simulation run, the simplest approach to estimate the mean waiting time by accumulating the waiting time of n successive entities and dividing by n. this is the sample mean, denoted by $\bar{x}(n)\ or\ \bar{x}$

$$\bar{x}(n) = \frac{\sum_{i=1}^{n} x_i}{n}$$

Waiting time measured in this way is not independent. Whenever a waiting line forms, the waiting time of each entity on the line clearly depends upon the waiting time of its predecessors. Such data are called auto-correlated.

Another problem that must be faced is that distribution is not stationary. Te early arrivals get the service quickly, so a sample mean that include early arrivals is biased. The following figure show the mean waiting time for different sample sizes.

Fig

Replications of Runs

One way of obtaining independent result is to repeat simulation. Repeating the experiment with different random numbers for the sample size n gives a set of independent determination of sample mean$\bar{x}(n)$.

Suppose the experiment is repeated p times with independent random values of n sample sizes. Let X_{ij} be the i^{th} observation in j^{th} run and let the sample mean and the variance for the jth run is denoted by $\bar{x}_j(n)$ and $s_j^2(n)$ respectively. Then for j^{th} run, the estimates are

$$\bar{x}_j(n) = \frac{1}{n}\sum_{i}^{n} x_{ij}$$

$$s_j^2(n) = \frac{1}{n-1}\sum_{i=1}^{n}\left[x_{ij} - \bar{x}_j(n)\right]^2$$

Combining the result of p independent measurement gives the following estimate for the mean $\bar{x}$ and variance s^2 of the populations as:

$$\bar{x} = \frac{1}{p}\sum_{j=1}^{p} \bar{x}_j$$

$$s^2 = \frac{1}{p}\sum_{j=1}^{p} s_j^2$$

Elimination of initial bias

The initial bias as shown in figure below needs to be removed.

Two general approaches can be taken into remove the bias.

1) The system can be started in a more representative then an empty state. 2) The first part of the simulation can be removed.

The some simulation studies where the information about expected value is available, it is feasible to select better initial conditions. The ideal solution is to know the steady state distribution for the system and select the initial conditions from that distribution.

The more common approach to remove initial bias is to illuminate an initial section of run. The run is started form an idle state and stopped after a certain period of time. The entities existing in that system at that time are left as they are. The run is then restarted with the statistics being gathered from the point of restart.

No simple rules can be given to decide how long an interval would be eliminated. It is advisable to use some pilot runs starting form idle state to judge how long the initial bias remains.

Exercise

1. Write short notes on Replication of Runs.
2. Write short notes on simulation statistics.
3. Discuss the different method of estimation technique.
4. What do you mean by central value theorem? Write its applications.

Why the output analysis of simulation methods is different from traditional statistical methods of analysis.

Chapter 6

Simulation Language

Basic concept of simulation Software

Simulation software is a program that allows the user to observe an operation through simulation without actually performing that operation. Simulation software is used widely to design equipment so that the final product will be as close to design specs as possible without expensive in process modification.

There are essentially three types of languages one can use for simulation:.

a) General purpose programming language
b) Simulation programming language
c) Simulation tools or environment or software

General purpose programming language such as Java, C,++, C can be used to developed the simulation model of real system as a computer program and can be used for the experimentation purpose. It may be the difficult and time consuming task since the general purpose programming language may not support directly the tools and function required by a particular simulation model. In such a case, we have to use specially designed language for simulation.

These are languages used for many applications. There are a number of advantages to implementing simulations in general languages : they tend to be very fast, there are few limitations on what can be done, programmers are easily found, and simulations can be done on a number of different computers (due to standardization of language). The disadvantage of this approach is cost. Simulations based on general languages take knowledge of simulation implementation and are generally very large, intricate programs. The time needed to design, code, and verify such a system may be overwhelming.

The second class of languages is simulation specific, and includes GPSS, SIMAN, etc. These languages take most of the work out of creating a simulation. All these systems can do all the queue manipulation needed in a single line. The main disadvantages of these languages are in their limited domain. It is difficult, but not impossible, to create a financial portfolio planner in GPSS. It is just not designed for it. On the other hand, another specialized simulation language can easily handle it within a 123 spreadsheet. It is also difficult to use other programs in the simulation. Imagine trying to put in a network optimizer in a transportation simulator if you are required to use SIMAN. Although there is the ability to link in your own FORTRAN programs, the methods are cumbersome and inefficient.

Third, there are simulation environment or IDE such as **Arena, AutoMod** etc. these products includes common characteristics such as graphical user interface and an environment that supports all aspects of a simulation study.

Some other examples are :

ACSL, APROS, ARTIFEX, C++ SIM, CSIM, CallSim, FluidFlow, GPSS, Gepasi, JavSim, MJX, MedModel, Mesquite, Multiverse, NETWORK, OPNET Modeler, POSES++, Simulat8, Powersim, QUEST, REAL, SHIFT, SIMPLE++, SIMSCRIPT, SLAM, SMPL, SimBank, SimPlusPlus, TIERRA, Witness, SIMNON, VISSIM, and Javasim.

Specific Simulation Programming Language

The overall simulation language can be categorized into two broad classes: discrete system simulation language and continuous system simulation language.

Continuous system simulation language:

These languages use the familiar statement type of input for digital computer allowing a program to be programmed directly from the equation of mathematical model, rather than requiring the equations to be broken into o functional element.

CSSL includes a variety of algebraic and logical expressions to describe the relation between variables. Several implementation of CSSL have been published. One particular CSSL that illustrate the nature of these language is the Continuous system modeling program (CSMP, version 3) i.e. CSMP III

Discrete simulation language

A number of programming languages have been developed to simplify the task of writing discrete system simulation program. These programs include a language with which to describe the system and a programming system that will establish a system image and execute a simulation algorithm. Each language is based on set of concept used of describing system. The word world-view has come to be used to describe this aspect of simulation program. The user of the program must learn the world view of particular language he is using and be able to describe the system in those terms.

Here in our course we will discuss the GPSS : A discrete system simulation language.

Introduction to GPSS:

GPSS is one of the earliest discrete simulation languages, it was developed by Geaffrey Gordon(Author of our Text Book) and presented in two papers in 1961 &1962

GPSS was designed specially for analysis who were not necessarily computer programmer. It is particularly suited for modeling traffic and queuing systems.

The general propose simulation language has been developed over many years by IBM Corporation. It has been implemented on several different manufacturer machines. The version GPSS/360 and GPSS V are most popular.

General Description of GPSS:

The system to be simulated in GPSS is described as block diagram in which the blocks represents the activities and line joining the block indicated the sequence in which the activities can be executed.

The use of block to describe the system being simulated of course, make easy and familiar to understand what kind of system we are going to simulate and what are the essential or important part of the system are included in our simulation model.

The GPSS defines a set of 48 specific block types, each of which represents a characteristics action of a system. The program user must draw a block diagram of the system using only these blocks and then write the corresponding code for each block.

GPSS block

Each block type is given a name that is descriptive of block action and is represented by a particular symbol.

A GPSS block diagram can consists of many blocks up to some limit prescribed by the program.

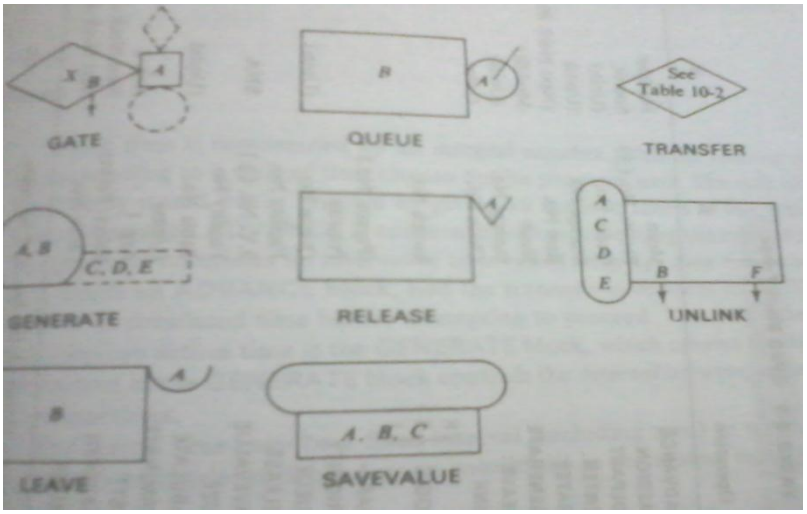
An identification number called location is given to each block.

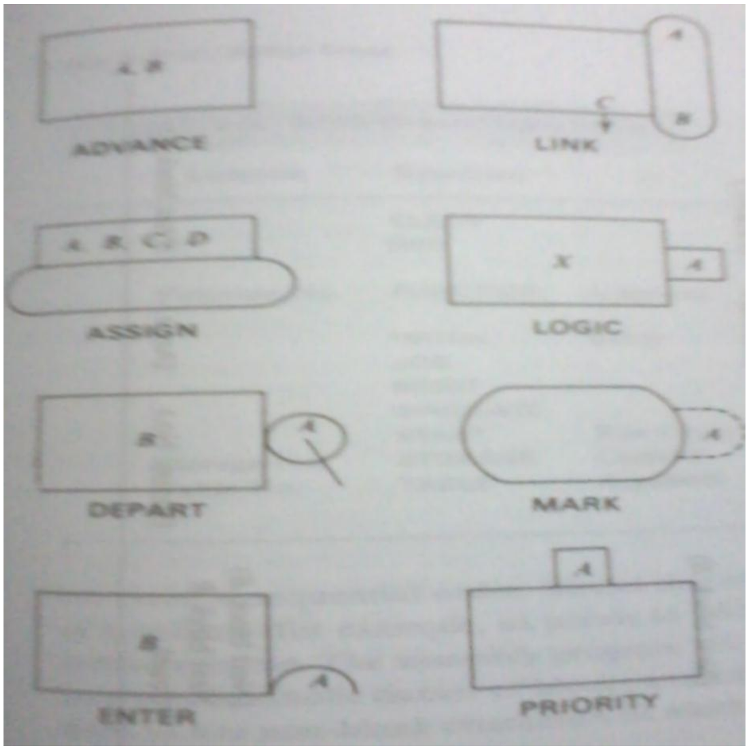
Transections

There are entities moving throughout the system. For example, a communication system is concerned with the movement massage, a road transportation system with motor vehicles. In simulation, these entities are called transactions.

The location is assigned automatically by an assembly program within GPSS so that when a problem is coded the blocks are listed in sequential orders. The assembly program will associate with the name with appropriate location. The symbolic names of blocks and other entities of the program must be from 3 to 5 non blank characters of which first three must be letters.

Pictorial representation of Block





Action Times

Clock time is represented by an integer number. The unit of time is not specifically stated but all times are expressed in terms of same unit.

The program computes an interval of time called an action time for each transection as it enters an ADVANCE block and the transaction remains at the block for this interval of time before attempting to proceed.

An action time is defined by giving a mean and modifier as the A and B fields for the block. The C field however can be used to specify an offset time as the time when the first transaction will arrive. The field D is used to specify a limit to the total no of transaction that will come from the block.

The E field determines the priority of the transactions.

Succession of Events

The program maintains records of when each transaction is due to move. It proceeds by completing all movements that are scheduled for execution ata a particular time.

Normally a transaction spends no time at a block other than at an ADVANCE block.

Choice of path

The transfer block allows some location other than the next sequential location to be selected. The choice is normally between two clocks referred to as next blocks A and B. the method used for choosing is indicated by a selection factor in field of the TRANSFER block. Next block A and B are placed in field B and C respectively.

Example Program: Simulation of manufacturing shop

To illustrate feature of the GPSS described so far, we consider a simple example here. A machine tool in manufacturing shop is turning out parts at a rate of one every 5 minutes. As they are finished, the parts go to an inspection, who takes 4+-3 minutes to examine each one and reject about 10% of the parts.

Here each parts will be represented by one transition and he time unit selected fo the problem will be 1 minutes.

The block diagram representing the system is shown in figure.

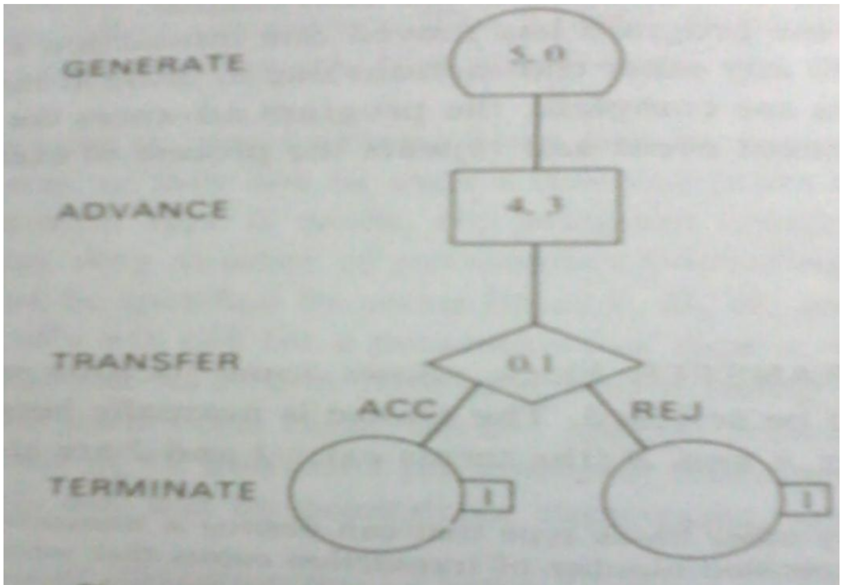


Figure 9-2. Manufacturing shop - model 1.

Here a GENERATE block is used to represent the output of the machine by creating one transaction.

An ADVANCE block with a mean of 4 and modifier of 3 is used to represent inspector. The time spent on inspection will be any one of the values 1, 2, 3, 4, 5, 6 or 7. With equal probability given t each value.

Upon completion of the inspection, transactions go to TRANSFER block with selection factor of 0.1 so that 90% of the part of to next location called ACC, to represent acceptance part ad 10% got to another location called REJ to represent rejected parts.

Since there is no further interest in the history of parts in this simulation, both location reached from the TRANSFER blocks to TERMINATE blocks.

The equivalent program code (in GPSS source code)

manufacturing shop simulation

```
                GENERATE    5                   ; create parts
                ADVANCE     4,3                 ;inspects
                TRANSFER    .1, ACC, REJ        ; Selects or rejects
ACC             TERMINATE   1                   ; Accepted
REJ             TERMINATE   1                   ; Rejected
                START 1000
```

Fig: coding of manufacturing shop model in GPSS

Introduction to CSMP III

A CSMP III program is constructed form three general types of statement.

1) Structural statements which defines the model. They consist of FORTARN like statement and functional block designed for operations that frequently occurs in a model definitions.
2) Data statements which assign numerical value to parameters, constant and intial conditions.
3) Control statements which specify options in the assembly and execution of program and choice of inputs.

Structural Statements

Structural statement can make use of the operation of addition, subtraction, multiplication, division and exponentiation, using the same notation and rule as are used in FORTRAN. If the model include the equation

$$X = \frac{6Y}{W} + (Z - 2)^2$$

Then the following statement would be used

```
X=6.0*Y/W+(Z-2)**2.0
```

There are many functional block which in addition to provide operation specific to simulation. Some of them are the exponential function, trigonometric function and function for taking maximum values.

Functional Block in CSMP III

The following is the list of functional blocks available in CSMP III

S.N	General form	Functions
1	Y=INTGRL(IC,X) Y(0)=IC (*Name: Integrator)*	$Y = \int_0^T Xdt + IC$

2	Y=LIMIT(P1,P2,X)
	(*Name: limiter)*
		Y=P1, X<P1
		Y=P2, X>P2
		Y=X, P1<=X<=P2

3	Y=STEP(P) *(Name: step function)*	Y=0, T<P Y=1, T>=P
4	Y=EXP(X) (*Name: Exponential)*	$Y=e^X$
5	Y=ALOG(X) (*Name: natural logarithm)*	Y=log(X)
6	Y=SIN(X) *(Name: Trigonometric Sine)*	Y=sinX
7	Y=COS(X) (Name: Trigonometric Cosine)	Y=cosX
8	Y=SQRT(X) *(Name: Square root of X)*	$Y=X_{1/2}$
9	Y=ABS(X) *(Name: Absolute value of X)*	Y=|X|
10	Y=AMAX1(X1,X2,……..Xn) (*Name: Largest value among N real values)*	Y=max(x1,x2,…..xn)
11	Y=AMIN(X1,X2,…………..Xn)\ (*Name: Minimum value among N real values)*	Y=min(x1,x2,…..xn)

Data statements:

Data statements are used to set the initial values to the model parameter. For example one data statement called INCON can be used to set the initial value of integration function block.

Control Statements

Among the control statement, TIMER is one of the control statements which specify certain time interval. For adequate accuracy, it should be small in relation to the rate at which variable change values. The following is an example

TIMER DELT=0.005, FINTIM=1.5, PRDEL=0.1

The item specified are

DELT-→ integration interval

FINTIM→ Finish time

PRDEL→ Interval, at which to print result.

If printed output is required, control statements with PRINT and PRTPLT are used followed by the names of variables to form the outputs.

The set of structural, data and control statements for a problem can be assembled in any order but they must be end with control statement END.

Example Program 1 in CSMP

The following code shows a CSMP III program for the automobile wheel suspension problem represented by

$$M\ddot{x} + \dot{D}x+Kx \dot{=} KF(t)$$

It has been coded for the case where M=2.0, F=1 and K=400. And there will be defferent runs with different value of D as specified

```
TITILE AUTOMOBILE SUSPENSION SYSTEM

    *

    PARAM D=(5.656,16.968,39.592,56.56)

    *



    X2DOT=(1.0/M)*(K*F-K*X-D*XDOT)
    XDOT=INTGRL(0.0,X2DOT)

    X=INTGRL(0.0,XDOT)

    *

    CONST M=2.0,F=1.0,K=400.0
    TIMER DELT=0.005,FINTIM=1.5, PRDEL=0.05

    PRINT X, XDOT, X2DOT

    END

    STOP
```

Example program 2 in CSMP III

Let we have a model represented by differential equation

$$\frac{dc}{dt} = \frac{c_0 - c}{\theta} - kc$$

In CSMP

Solution

```
TITLE SIMPLE CSTR REACTION SIMLUATION

*

PARAM ICC=0, THETA=2.0, K=1.0, C0=1.0,V=1.0

*

DCDT=(C0-C)/THETA-K*C

C=INTGRL(ICC,DCDT)

*

TIMER FINTIM=15.0, DELT=0.05, PRDEL=0.5

PRINT C, DCDT

END
STOP
```

Hybrid simulation

For the most studies, the model is clearly either of continuous or discrete nature; an that is determining factor in deciding whether to use an analog or digital computer for system simulation.

However, there are times when an analog and digital computer are combined to provide a hybrid simulation. The form taken by hybrid simulation depends upon the application. One computer may be simulating the system being studied while other is providing a simulation of the environment in to which the system is to operate. It is also possible that the system being simulated is an interconnection of continuous and discrete subsystem, which can best be modeled by an analog and digital computer being linked together..

The introductions of hybrid simulation require certain technological development for its use. High speed converters are needed to transform signal form one form of representation to another. As a practical matter, the availability of minicomputer has made hybrid simulation easier, by lowering cost and allowing computer to be dedicated to an application.

The term hybrid simulation is used generally of the case in which functionally distinct analog and digital computer are linked together for proposes of simulation