SIMULATION OF RAPID TRANSPORT SYSTEM FOR IRON ORE MINE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

MINING ENGINEERING

DEPARTMENT OF MINING ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA – 769 008
April 2010

SIMULATION OF RAPID TRANSPORT SYSTEM FOR IRON ORE MINE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

MINING ENGINEERING

Under the Guidance of

Dr. B.K PAL

DEPARTMENT OF MINING ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA – 769 008

April 2010

National Institute of Technology
Rourkela

C E R T I F I C A T E

This is to certify that the thesis entitled **“SIMULATION OF RAPID LOADING SYSTEM FOR IRON ORE MINE”** submitted by **Sri Sandeep R Manakeshwar & Sri Sumeet Verma** in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Mining Engineering at the National Institute of Technology, Rourkela is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/Institute for the award of any Degree or Diploma.

Date:

Dr B.K.PAL

Dept. of Mining Engineering

National Institute of Technology

Rourkela, 769008

ACKNOWLEDGEMENT

I record my sincere gratitude to **Dr. B.K PAL**, Professor, (Department of Mining Engineering) for assigning me the project "SIMULATION OF RAPID TRANSPORT SYSTEM FOR IRON ORE MINE". It is not possible to acknowledge sufficiently his important contribution of talent and time given unselfishly in proceeding with this work. His overall constructive criticism has helped me to present my work in the present form.

I would also like to convey our sincere gratitude and indebt ness to the faculty and staff members of Department of Mining Engineering, NIT Rourkela, for their help at different times.

I would also like to extend our sincere thanks to the Mines Manager of TATA STEEL, Mr Mudit Tandon (Planning) and other officials, who helped me during my sample collection in their respective regions.

Last but not the least; we would like to thank all our friends who have been a constant source of help to us.

SANDEEP R MANAKESHWAR &

SUMEET VERMA

B.Tech

MINING ENGINEERING

ABSTRACT

In the mining industry, it is observed that sudden break-down of any equipment may stop the entire system, resulting in drastic production losses and enhancing the cost production. In this paper, the probability of sudden break down of each equipment are individually analyzed from their previous performances where the frequency of occurrences, duration and the time-interval of each breakdown has given an additional stress and the non-availability of that equipment on the entire system is discussed. Computerized best fit matching is found out for preventive maintenance of this equipment by developing different sub-routines and simulation models. Optimum utilization of those equipment shows a particular steady- state production from the mine.

Advanced technology is used for the operation of the open-pit mining. Hazard due to open cast mining is less than that of underground mines and the recent trend is to adopt the former one. For the mechanization, different types of machineries are used, such as shovel, dumper, dozer, drill machines, etc. use of more machineries increases the complexity of the operation and as a result it is very difficult to the proper matching of these equipment's. As these machineries are very costly so unless they are properly matched, reduction in production cost is not possible. Sudden breakdown of one equipment may stop the production from whole mine. So it is needed to analyze the breakdown data by statistical approach to find out the possibility of breakdown of that particular equipment and ask for preventive maintenance. This analysis needed the help of computer for simulating the whole mining system to judge the performance of each equipment. This paper deals with the computerized best fit matching, for preventive maintenance of equipment.

CONTENTS

INTRODUCTION

Simulation is a powerful technique for solving a wide variety of problems some of the basic ideas in simulation can be best understood by performing actual simulations. Let us, therefore, consider the following example and see how simulation is done.

1.1 Simulation of an Inventory Problem

Suppose you work in a retail store and it is your responsibility to keep replenishing a certain item (say, automobile tyres) in the store by ordering it from the wholesaler. You want to adopt a simple policy for ordering new supplies:

When my stock goes down to P items (called reorder point), I will order Q more items (called reorder quantity) from the wholesaler.

Assume the following conditions:

- There is a three day lag between the order and arrival. The merchandise is ordered at the end of the day and is received at the beginning of the fourth day. That is ,merchandise ordered on the evening of the ith day is received on the morning of the (i+3)rd day
- For each unit of inventory the carrying cost for each night is Re 0.75
- Placement of each order costs Rs 75.00 regardless of the number of units ordered.
- The demand in a day can be for any number of units between 0 and 99, each equiprobable.
- Each unit out of stock when ordered results into a loss of goodwill worth Rs 2.00 per unit +loss of Rs 16.00 net income, which would result in a total loss of Rs 18.00.
- Initially we have 115 units on hand and no reorder outstanding.

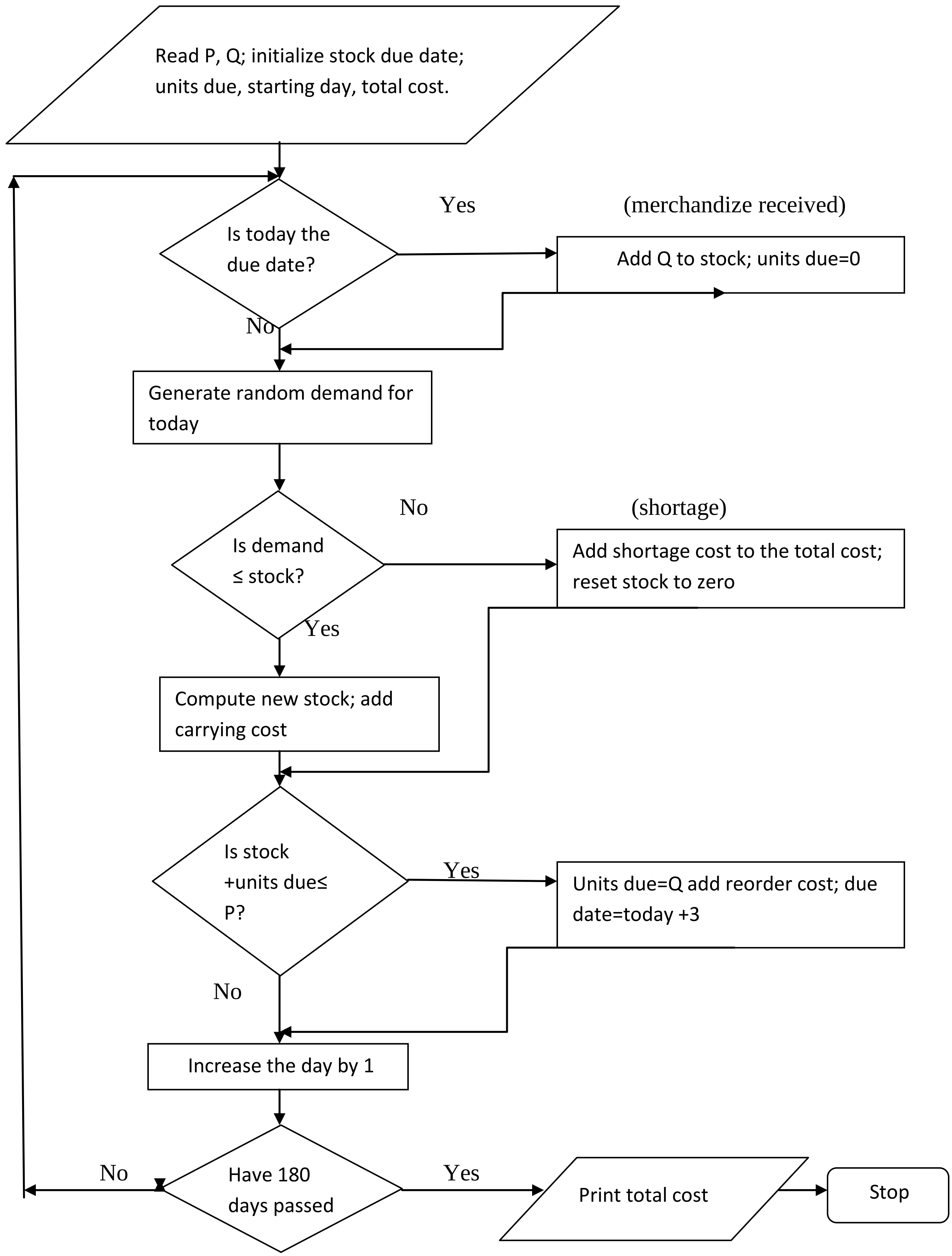
With these conditions in task you have been asked to compare the following replenishment policies

	P(reorder point)	Q(reorder quantity)
Policy I	125	150
Policy II	125	250
Policy III	150	250
Policy IV	175	250

The problem does not easily lend itself to an analytic solution; it is best therefore to solve it by simulation. let us simulate the running of the stores for about six months(180 days) under each of the four policies and then compare their costs.

A simulation model of this inventory system can be easily constructed by stepping time forward in the fixed increment of the day, starting with day I, and continuing up to day 180.on a particular day, day '*i*', first we check to see if the merchandise is due to arrive today. if yes, then the existing stock *S* is increased by Q(the quantity that was ordered).if DEM is the demand for today, and DEM ≤ S, our new stock at the end of the day will be (S-DEM) units. if DEM > S, then our new stock will be zero. in either case , we calculate the total cost from today's transactions, and add it to the total cost C incurred till yesterday. Then we determine if the inventory on hand plus units on order is greater than P, the reorder point. if not, place an order (to be delivered),by stating the amount ordered and the day it is due to be received.we repeat this procedure for 180 days.

Initially we set day number i=1, stock=115,number of units due UD=0(because there is no outstanding order), and the they are due DD=0


Read P, Q; initialize stock due date; units due, starting day, total cost.
Is today the due date?
Yes
(merchandize received)
Add Q to stock; units due=0
No
Generate random demand for today
Is demand ≤ stock?
No
(shortage)
Add shortage cost to the total cost; reset stock to zero
Yes
Compute new stock; add carrying cost
Is stock +units due≤ P?
Yes
Units due=Q add reorder cost; due date=today +3
No
Increase the day by 1
Have 180 days passed
No
Yes
Print total cost
Stop

Cost of inventory policy (P, Q)

The program yielded the following cost figures for the inventory policies:

P	Q	cost in Rs
125	150	38769.75
125	250	31268.25
150	250	23699.25
175	250	26094.00

Thus, policy IV (P=175, Q=250) is the best among the four considered.

1.2When to Simulate

All of us in our daily lives encounter problems, which although mathematical in nature ,are too complex to lend themselves to exact mathematical analysis. the performance of such a system(say, weather or traffic jam) may be difficult to predict, either the system is itself complex or the theory is not yet sufficiently developed.

Simulation in science and engineering research

Simulation has changed, in a very fundamental sense, the way in which research is conducted today. Thousands and millions are spent on physical models and expensive experiments. Today a majority of these experiments are simulated on a computer.

Simulation in business executive

There are many problems faced by the management that cannot be solved by standard operations research tools like linear and dynamic programming, inventory and queueing theory. Business executive can use simulation to make better and more meaningful decisions.

2 INDIAN IRON ORE MINE

Haemetite and magnetite are the most important of iron ores found in India. of these Haemetite is considered to be the most important because of its high quality grade ore which is consumed in both sponge and steel industries. the most important grade of ore are found in the states of Jharkhand ,Orissa, Karnataka ,Goa etc.

Iron Ore Reserves/Resources and Distribution in India

India has large reserves of iron ore reserves which can meet the growing demand of domestic iron and steel industries and can also sustain considerable external trade .with a total reserves of around 25.25 billion tonnes, India is one of the leading producers as well as exporters in the world.

About 60% of Haemetite iron ore deposits are found in the eastern sector and about, 80% of magnetite deposits are found in southern sector, especially in Karnataka

Sl .no	State	Reserves	Resources	Total
1	Andhra Pradesh	-	1463541	1463541
2	Karnataka	148437	7663437	7811784
3	Goa	50112	164057	214169
4	Rajasthan	4225	522652	526877
5	Jharkhand	3391	6879	10269

Source: *Indian bureau of mines, Nagpur*

2.1 Grade wise iron ore resources as on 01.04.2005 (provisional)

(Unit: million tonnes)

Ore type	Grade	resources as on 01.04.2005 (provisional)
Iron ore (Haemetite)	1)(+)65% Fe 2)(+)62-65% Fe Below 62% Fe(including all other grades) Total	2132 6694 5804 14630
Iron ore (magnetite)	Metallurgical Coal washery Foundry Others	2186 8 1 25

Source: *Indian bureau of mines, Nagpur*

3 SIMULATION OF CONTINUOUS SYSTEMS

From the viewpoint of simulation there are two fundamentally different types of systems:

1) Systems in which the state changes smoothly or continuously with time (continuous systems).

2) Systems in which the state changes abruptly at discrete points in time (discrete systems).

Usually, the simulation of most systems in engineering and physical sciences turns out to be continuous, whereas most systems encountered in operations research and management sciences are discrete. The methodologies of discrete and continuous simulations are inherently different.

Continuous dynamic systems, those systems in which the state or the variables vary continuously with time, can generally be described by means of differential equations. If the set of (simultaneous) differential equations describing a system are ordinary, linear, and time –invariant (i.e. have constant coefficients), an analytic solution is usually easy to obtain. In general differential equations of a more difficult nature can only be solved numerically. Simulating the system often gives added insight into the problem besides giving the required numerical solution.

4 SELECTING SIMULATION SOFTWARE

Overview of the Steps Involved in Selecting Simulation Software

The steps for selecting simulation software are outlined below (and detailed in subsequent sections):

1. Establish the commitment to invest in simulation software to solve your problem.
2. Clearly state the problem (or class of problems) that you would like to solve.
3. Determine the general type of simulation tool required to solve the problem.
4. Carry out an initial survey of potential solutions.
5. Develop a list of functional requirements.
6. Select the subset of tools that appear to best meet the functional requirements.
7. Carry out a detailed evaluation of the screened tools and select a solution.

Step 1: Establish the Commitment to Invest in Simulation Software

Before spending any effort to research simulation tools, the organization should establish the commitment to invest both the necessary money and staff time into purchasing and learning how to use a simulation software program. Depending on the type of simulation tool selected, the price for a single license is likely to be no less than $2000, and could be as much as ten times higher than that.

Note that it is important for the organization to understand that the cost of a simulation tool is not just the cost of the software itself, but the cost to become a fluent user of the software (since staff time has an inherent cost). In fact, given the complexity of the more powerful simulation tools, the investment in time is likely to be greater than the investment in the software itself.

If the issue you are trying to address represents a one-time need, it may be more cost-efficient to hire a consultant to do the work (so that the organization does not need to purchase and learn the software at all). However, if the issue is recurring or ongoing such that the model will need frequent refinement, or if for some other reason it is important to the organization that the work be done internally, it will be necessary to purchase a simulation software too and train individuals in its use.

Step 2: Clearly State the Problem You Wish to Address

Perhaps the most important step in selecting simulation software is to clearly state the problem (or class of problems) that you would like to address. This must include a general statement of what you would like the simulation tool to do.

Without doing so, it will be impossible to determine, first, the type of simulation tool you should look for, and subsequently, to list the functional requirements and desired attributes of the tool.

To illustrate what is required, several examples of simulation problem statements are listed below:

> **Managing the water supply for a city:** Managing a water supply is difficult due to the dynamic (and naturally unpredictable) nature of the problem (resulting from uncertainties in both weather and demand). The simulation tool must be able to predict the movement of water through a system (e.g., reservoirs, distribution systems) tracking the quantities and flow rates at various locations. It must be able to quantitatively represent the inherent uncertainty in the system (due to the uncertainty in the weather and demand), and represent various management options (e.g., rules for allocating flows under specified

conditions). The output of the simulation will consist of probabilistic predictions of daily water levels and flow rates over time given a specified management alternative.

Carrying out a risk analysis for a complex mission (i.e., a machine and/or persons performing a specified task or set or tasks) : Carrying out a risk analysis for a complex mission is difficult due to the complex interactions and dependencies of the various components, and the fact that the environment may evolve dynamically during the mission. The simulation tool must be able to simulate the operation of the machine throughout the mission, explicitly modeling component interactions, dependencies and failures. It must also be able to represent the impact of a changing environment on the components. The output of the simulation will consist of probabilities of failure (and success) for a mission of specified length, and identification of key failure mechanisms.

Modeling the financial outcome of several alternative projects: When selecting or ranking various alternative projects or undertakings, it is necessary to quantitatively evaluate both the costs and revenues associated with each project. The simulation tool must be able to simulate the future costs and revenues associated with alternative projects, explicitly accounting for the uncertainty in costs, durations and revenues. The simulation must be able to represent disruptive events (e.g. strikes, price changes) and resulting contingency plans that allow a simulated project to respond to new developments in a realistic way. The output of the simulation will consist of probabilistic predictions of the NPV and IRR for each alternative.

Note that these statements are not extremely detailed, but provide a clear statement of the problem, a general statement of what processes and features must be included, and what the output of the simulation will be. This provides enough information to direct a survey of potential solutions and carry out an initial screening. In a later step in the process, more detailed requirements will need to be defined in order to differentiate between the available options.

Step 3: Determine the General Type of Simulation Tool Required

Because simulation is such a powerful tool to assist in understanding complex systems and to support decision-making, a wide variety of approaches and tools exist. Before trying to survey all available tools, you must first decide upon the general type of tool that you require.

There are a variety of simulation frameworks, each tailored for a specific type of problem. What they all have in common, however, is that they allow the user to model how a system might evolve or change over time. Such frameworks can be thought of as high-level programming languages that allow the user to simulate many different kinds of systems in a flexible way.

Perhaps the simplest and most broadly used general purpose simulator is the spreadsheet. Although spreadsheets are inherently limited in many ways by their structure (e.g., representing complex dynamic processes is difficult, they cannot display the model structure graphically, and they require special add-ins to represent uncertainty), because of the ubiquity of spreadsheets, they are very widely used for simple simulation projects (particularly in the business world).

Other general purpose tools exist that are better able to represent complex dynamics, as well as provide a graphical mechanism for viewing the model structure (e.g., an influence diagram or flow chart of some type). Although these tools are generally harder to learn to use than spreadsheets (and are typically more expensive), these advantages allow them to realistically simulate larger and more complex systems.
The general purpose tools can be broadly categorized as follows:

Discrete Event Simulators: These tools rely on a transaction-flow approach to modeling systems. Models consist of entities (units of traffic), resources (elements that service entities), and control elements (elements that determine the states of the entities and resources). Discrete event simulators are generally designed for simulating processes such as call centers, factory operations, and shipping facilities in which the material or information that is being simulated can be described as moving in discrete steps or packets. They are not meant to model the movement of continuous material (e.g., water) or represent continuous systems that are represented by differential equations.

Agent-Based Simulators: This is a special class of discrete event simulator in which the mobile entities are known as agents. Whereas in a traditional discrete event model the entities only have attributes (properties that may control how they interact with various resources or control elements), agents have both attributes and methods (e.g., rules for interacting with other agents). An agent-based model could, for example, simulate the behavior of a population of animals that are moving around and interacting with each other.

Continuous Simulators: This class of tools solves differential equations that describe the evolution of a system using continuous equations. Although these tools usually have some mechanism to represent discrete events, they are most appropriate if the material or information that is being simulated can be described as evolving or moving smoothly and continuously, rather than in infrequent discrete steps or packets. For example, simulation of the movement of water through a series of reservoirs and pipes can most appropriately be represented using a continuous simulator. Continuous simulators can also be used to simulate systems consisting of discrete entities if the number of entities is so large that the movement can be treated as a flow.

Hybrid Simulators: These tools combine the features of continuous simulators and discrete simulators. That is, they solve differential equations, but can superimpose discrete events on the continuously varying system. This can be useful, for example, in business simulations, in which information and material can be modeled as moving continuously, but discrete financial transactions also need to be represented

Before starting your search for a simulation tool, you should first determine which of these types of tools is required to solve your problem. In most cases, this can be determined from the problem statement. If you are unsure, you should seek input from someone who is familiar with simulation modeling (e.g., a consultant). One of the worst mistakes you can make is to select the wrong type of tool (e.g., to select a continuous simulator, when what you really need is a discrete event simulator).

Step 4: Carry Out an Initial Survey of Potential Solutions

Once you have selected the general type of tool you will need, you can then carry out an initial survey to try to identify the possible options. Note that this process does not involve actively evaluating any software tools. It is simply a survey to see what options are available. The only screening that should be carried out should be based on general type. For example, if you have determined that a continuous simulation tool is required, you should screen out pure discrete event simulators.

This initial list of candidate tools can be generated from a variety of sources, including web searches, peer recommendations, advertisements in trade magazines, and vendor lists from trade-shows.

Step 5: Develop a List of Functional Requirements

Step 5 involves developing a set of functional requirements that you would like the software tool to have. This list will then be used in a subsequent step to compare and contrast the candidate solutions and filter out all but the most promising candidates.
A functional requirement is a necessary feature or attribute of the simulation software solution. Note that requirements specify what the simulation software will do, not how. They should be as concise as possible. You should also note whether a requirement is mandatory or simply desired (e.g., "must have" in a requirement could indicate mandatory; "should have" could indicate desired, but not mandatory).

In order to develop a list of requirements, you generally start with your problem statement, and describe the minimum set of functionality that will be necessary in order for the software to solve your problem. The actual users of the software will be the primary developers of the requirements list, but other stakeholders should also be involved, such as the ultimate client for the model (e.g., a manager) and IT personnel, as they may have their own requirements.

To illustrate what is meant by a functional requirement, let's consider the first example problem statement listed in the description of Step 2 above:

> Managing the water supply for a city: Managing a water supply is difficult due to the dynamic (and naturally unpredictable) nature of the problem (resulting from uncertainties in both weather and demand). The simulation tool must be able to predict the movement of water through a system (e.g., reservoirs, distribution systems) tracking the quantities and flow rates at various locations. It must be able to quantitatively represent the inherent uncertainty in the system (due to the uncertainty in the weather and demand), and represent various management options (e.g., rules for allocating flows under specified conditions). The output of the simulation will consist of probabilistic predictions of daily water levels and flow rates over time given a specified management alternative.

The list of functional requirements for this problem statement would likely include the following mandatory requirements:

- Must be able to track and conserve the continuous movement of material through a system (in this case water).
- Must be able to represent random discrete changes to the system (e.g., pump failures)
- Must be able to represent stochastic processes (e.g., rainfall).
- Must be able to represent rules for allocating and splitting flows.
- Must be able to enter time series inputs.
- Must be able to import time series inputs and other data from spreadsheets.
- Must support Monte Carlo simulation.
- Must have a user interface that supports creation of transparent, well-documented models.

Desired (but perhaps not mandatory) requirements might include:

- Should be able to easily handle unit conversions
- Should be able to support distributed processing (for Monte Carlo simulation).
- Should support optimization.
- Should provide tools for sensitivity analysis.

Step 6: Select the Subset of Tools that Appear to Best Meet the Functional Requirements

Once you have defined your functional requirements, the next step is to apply the requirements, to the candidate solutions, identifying and eliminating candidates that do not meet the mandatory requirements.

Note that this step should not require downloading and running the candidate software. Instead, the reviewer should be able to gather sufficient information to develop informed yes/no answers to the requirements based on the vendor's web pages, quick tours, animated demos, white papers, case studies, recorded webinars, and in some cases, phone calls with technical sales representatives. If you cannot easily gather information about a software product, it is recommended that you eliminate that product from consideration (as this is generally an indication that the quality of the product and/or the level of support is likely to be poor).

The output of this step is a list of viable solutions, each one of which will then be evaluated in greater detail in the next step.

Step 7: Carry out a Detailed Evaluation of the Screened Tools

The final step in the process involves carrying out a detailed evaluation of the tools screened in Step 6 and selecting the most appropriate tool.

To do so, you should obtain an evaluation version of each product and experiment with the software yourself. Although this is necessary, it can be time-consuming, since each product will have a learning curve.

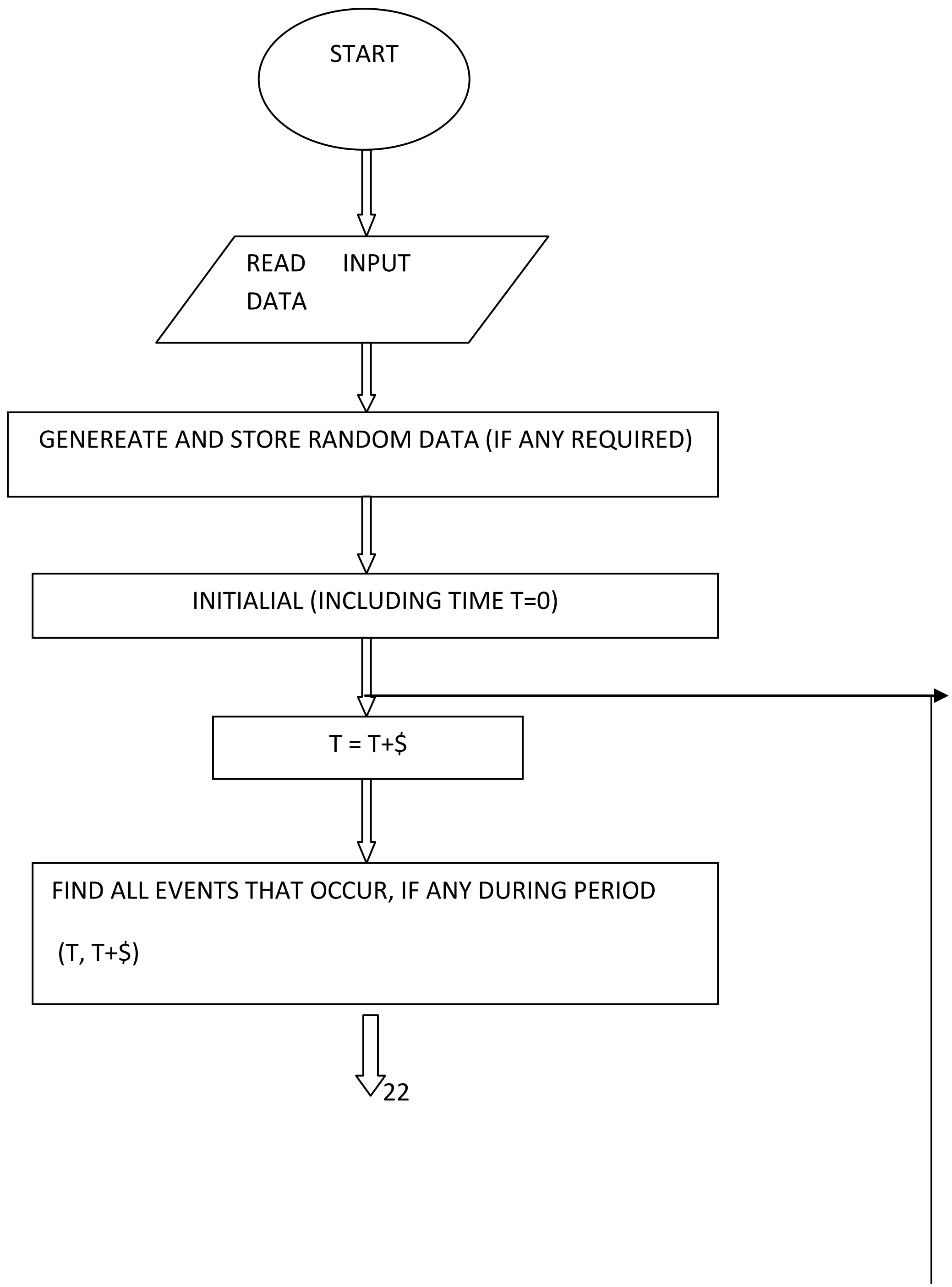
5 DISCRETE SYSTEM SIMULATIONS:

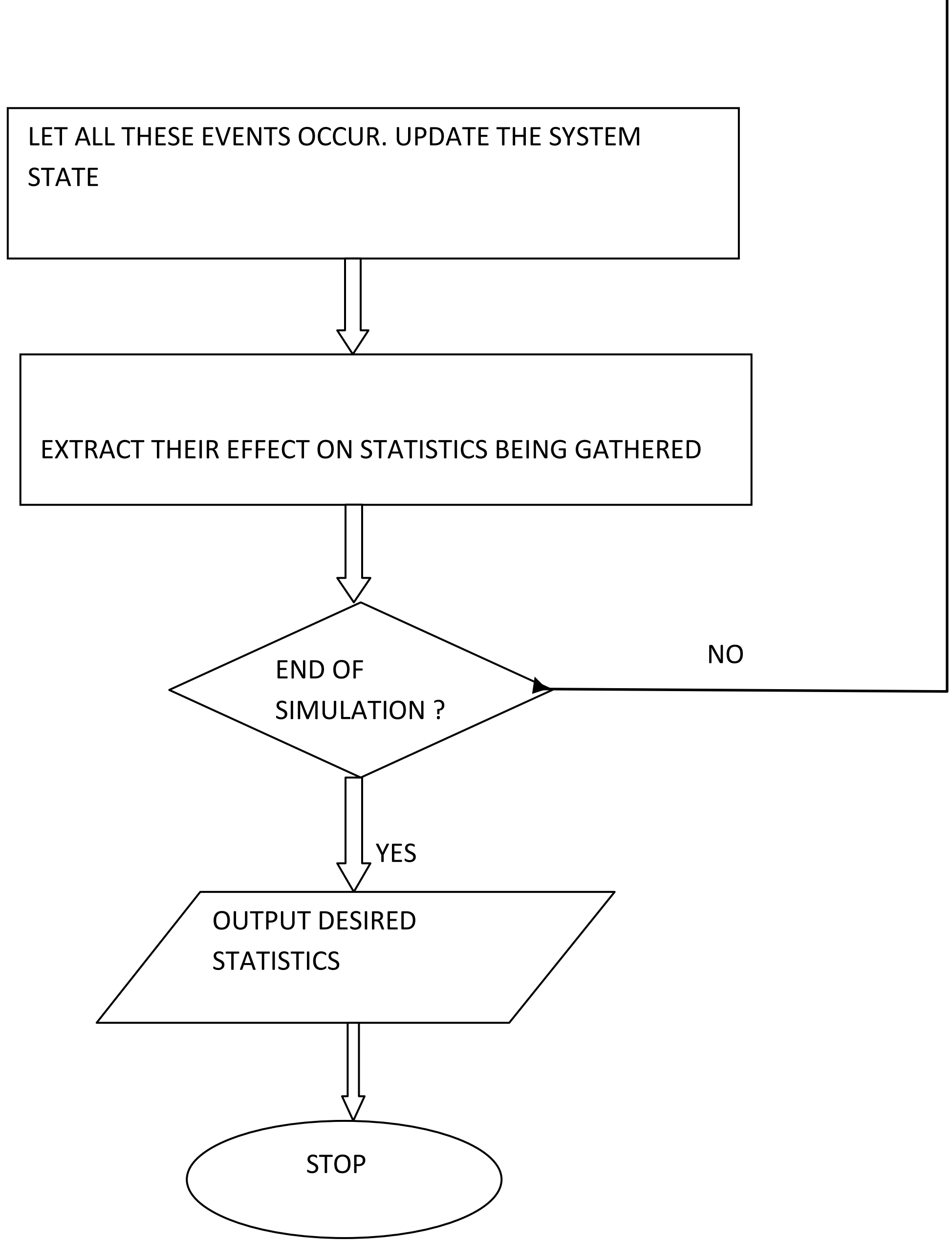
In this type of system the changes are discontinuous. Each change in the state of system is called an event. For example, arrival or departure of a customer in a queue is an event. Likewise, sale of an item from the stock or arrival of an order to replenish the stock is an inventory system. Arrival of a car at an intersection is an event if we are simulating street traffic. Therefore, the simulation of a discrete system is often referred to as discrete event simulation.

It is commonly used by operations research workers to study large, complex systems which do not lend themselves to a conventional analytic approach. Some other examples are the study of sea and air ports, steel melting shops, telephone exchanges, production line, stock of goods scheduling of projects, to name a few. Discrete system simulation is more diverse and has less of a theory than continuous system simulation. There are no overall sets of equation to be solved in discrete – event simulation.

FIXED TIME STEP VS EVENT-TO-EVENT MODEL:

In simulating any dynamic system – continuous or discrete – there must be a mechanism for the flow of time. For we must advance time, keep track of the total elapsed time, determine the state of the system at the new point in time, and terminate the simulation when the total elapsed time equals or exceeds the simulation period. In simulation of discrete systems, there are two fundamentally different models for moving a system through time: the fixed time step model and the event-to-event (or next event) model. In a fixed time-step model a “timer” or “clock” is simulated by the computer. This clock is up-dated by a fixed time interval, and the system is examined to see if any event has taken place during this time interval (minutes, hours, days , whatever.). All events that take place during this period are treated as if they occurred simultaneously al the tail end of this interval. In a next event simulation model the computer advances time to the occurrence of the next event. It shifts from event to event. The system state does not change in between. Only those points in time are kept track of when something of interest happens to the system.

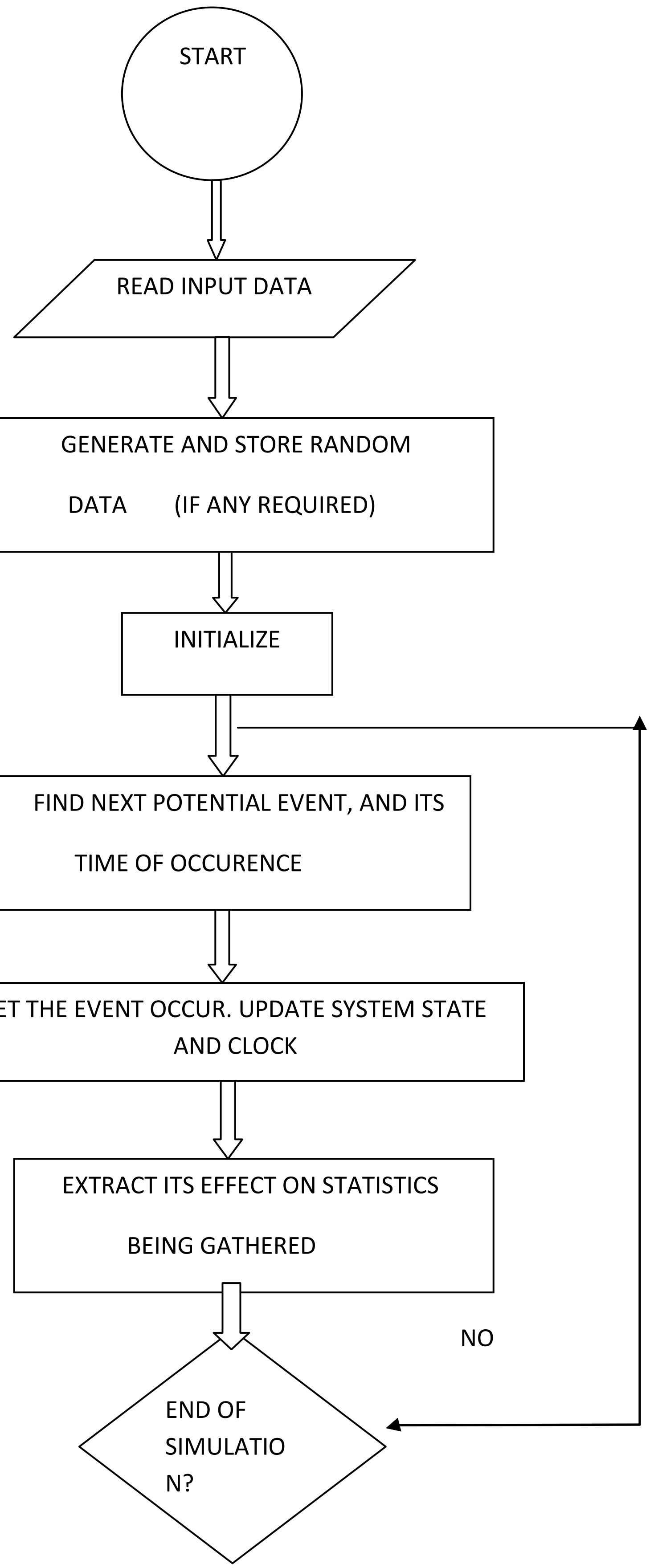


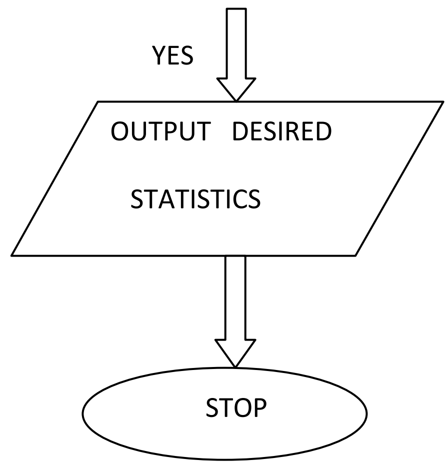


To illustrate the difference between the two models, let us assume that we are simulating the dynamics of the population in a fish bowl, starting with, say, 10 fish. If we used the fixed time – step model with, say, t=1 day, then we would scan the fish bowl once every 24 hrs. and any births and deaths that takes place are presumed to be during the last moment of this period. On the other hand, if we use a next-event model then we will first find out when the next-event (birth or death) is to take place and then advance the clock exactly to that time.

In general, the next event model is preferred,(except when we may be forced to use the fixed time-step model) because we do not any computer time in scanning those points in time when nothing takes place. The only drawback of the next-event model is that usually its implementation turns out to be more complicated than the fixed time- step model.

NEXT EVENT SIMULATION





NEXT EVENT SIMULATION

To illustrate the difference between the two models, let us assume that we are simulating the dynamics of the population in a fish bowl, starting with, say 10 fish. If we used the fixed time-step model with, say $ = 1 day, then we would scan the fish bowl(figuratively speaking) once every 24 hours, and any births deaths that take place are presumed to be during the last moment of this period. On the other hand, if we use a next event model then we will first find out when the next event (birth or death) is to take place and then advance the clock exactly to that time.

In general, the next-event model is preferred, (except when we may be forced to use the fixed time-step model) because we do not waste any computer time in scanning those points in time when nothing takes place. This waste is bound to occur if we pick a reasonably small value of $. On the other hand, if $ is so large that one or more events must take place during each interval then our model becomes unrealistic and may not yield meaningful results. Therefore in most simulations of discrete systems the next-event model is used. The only drawback of the next-event model is that usually its implementation (programming for it) turns out to be more complicated than the fixed time step model.

MONTE CARLO SIMULATION

A **Monte Carlo method** is a technique that involves using random numbers and probability to solve problems.

The idea behind Monte-Carlo simulations gained its name and its first major use in 1944 [Pllana, 2000], in there search work to develop the first atomic bomb. The scientists working on the Manhattan Project had intractably difficult equations to solve in order to calculate the probability with which a neutron from one fissioning Uranium1atom would cause another to fission. The equations were complicated because they had to mirror the complicated geometry of the actual bomb, and the answer had to be right because, if the first test failed, it would be months before there was enough Uranium for another attempt

2 Simple Example:

2.1 Birthday Problem - Classical Approach

Simple examples of Monte-Carlo simulation are almost embarrassingly simple. Suppose we want to find out the probability that, out of a group of thirty people, two people share a birthday. It's a classic problem in probability, with a surprisingly large answer.

Classically, you approach it like this: Pick people (and their birthdays) randomly, one at a time. We will keep track of the probability that there are no shared birthdays.

• The first person can have any birthday, and there is still a 100% chance of no shared birthdays.

• The second person has one chance of overlapping with the first person, so there is a 364/365 chance of placing him/her without an overlap. The probability of no shared birthdays is 364/365 The third person has two chances of overlapping with the first two people, so there is a 363/365 chanceof placing him/her without overlaps (two days are taken). The probability of no shared birthdays is now (364/365) · (363/365).

• The fourth person has three chances of overlapping with the first three people, so there is a 362/365 chance of placing him/her without overlaps. The probability of no shared birthdays is now (364/365) · (363/365) · (362/365).

• The thirtieth person has 29 chances of overlapping with the first three people, so there is a 336/365chance of placing him/her without overlaps. The probability of having no shared birthdays is now (364/365) · (363/365) · (362/365) . . . (336/365).

The overall probability of no overlapping birthdays is then 0.294, giving a 71% chance that at least one pair of people have overlapping birthdays. It's not too complex if you see the trick of keeping track of the probability of zero overlaps, rather than trying to add up the probability of one or more overlaps. It also takes some thought to realize that the probabilities are conditioned properly, so that multiplying together all the various P (Nth person doesn't overlap |first N − 1 people don't overlap) factors

2.2 **Birthday Problem – Monte-Carlo Approach**

The solution here is conceptually very simple:

1. Pick 30 random numbers in the range [1,365]. Each number represents one day of the year.
2. Check to see if any of the thirty are equal.
3. Go back to step 1 and repeat 10,000 times.
4. Report the fraction of trials that have matching birthdays.

A computer program in Python to do this calculation is quite simple:

```
#!/usr/bin/env python
import random                         # Get a random number generator.
NTRIALS = 10000                  # Enough trials to get an reasonably accurate answer.
NPEOPLE = 30                          # How many people in the group?
matches = 0                      # Keep track of how many trials have matching birthdays.
for trial in range(NTRIALS):              # Do a bunch of trials...
taken = {}                            # A place to keep track of which birthdays
# are already taken on this trial.
for person in range(NPEOPLE):      # Put the people's birthdays down, one at a time...
day = random.randint(0, 365)        # On a randomly chosen day.
if day in taken:
matches += 1                        # A match!
break                               # No need to look for more than one.
```

```
taken[day] = 1                          # Mark the day as taken.
print 'The fraction of trials that have matching birthdays is', float(matches)/NTRIALS
```

And the answer is:

The fraction of trials that have matching birthdays is 0.7129

8 ON SIMULATING RANDOMNESS:

There are numerous as well as man-made systems where chance plays some part. These are called stochastic systems. There is inherent randomness or unpredictability in their behavior. Some other examples of randomness that are frequently simulated are: arrival of customers in a store, arrival of vehicles at a traffic light, request for telephone lines at a telephone exchange, births and deaths in a population, collision of particles in a nuclear reactor, arrival of an elevator on a given floor, etc.

Discrete dynamic systems could be classified as deterministic or stochastic. The former are less demanding computationally than the latter and are frequently solved analytically. Hence simulation in the study of discrete dynamic systems is used almost exclusively for stochastic systems- systems in which at least one of the variables is given by a probability function. Complex discrete, dynamic, stochastic systems often defy an analytic solution and are therefore studied through simulation.

To stimulate such random variables, we require a source of randomness. In simulation experiments, this is achieved through a source of uniformly distributed random numbers. These numbers are samples from a uniformly distributed random variable between some specified interval, and they have equal probability of occurrence in the same manner as all six faces of an unbiased die have equal chance of occurrence. A random number generator and its appropriate use form the heart of any simulation experiment involving a stochastic system.

8.1 GENERATION OF RANDOM NUMBERS:

Random numbers could be obtained from a sack of numbered beads as in bingo; or from rotations of a roulette wheel; or from any randomizing device. However, such physical generators of random numbers are not suitable for simulation experiments on computers because of the following reasons:

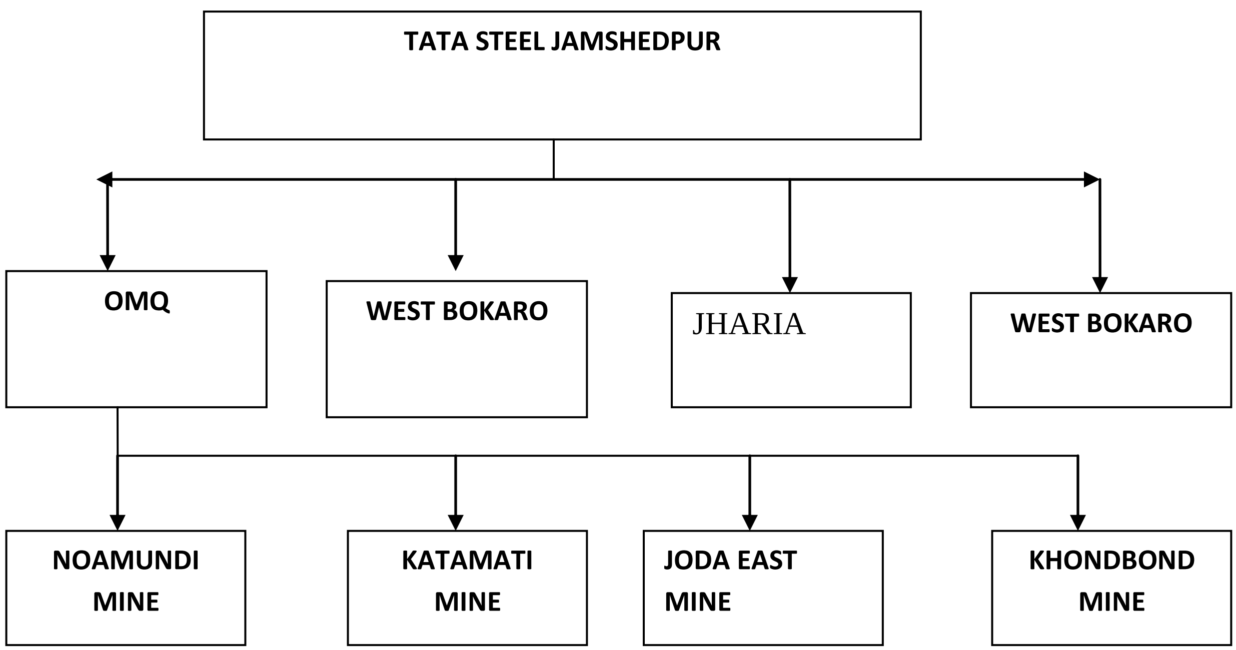
1) The generation and feeding into the computer of thousands of such numbers is excessively laborious and time consuming.

2) A sequence of numbers generated cannot be reproduced at a later time or by another person for repeating a simulation run. Such repetitive runs are required for debugging computer programs as well as for studying the effect of changes in the model.

9 Case study: NOAMUNDI IRON ORE MINE

INTRODUCTION

The process of steel making requires different types of raw materials which include iron ore, coal, manganese and chromites to name a few. The raw materials to Tata steel plant at Jamshedpur is supplied from 4 different divisions – Ore Mines and Quarries (OMQ), West bokaro, Jharia and Ferro Alloys Metal Division (FAMD). Both west bokaro and jharia deal with coal production whereas the OMQ division deals with the supply of raw iron ore. Under the OMQ division are 4 mines – Noamundi iron mine, Katamati iron mine, Joda east iron mine and Khondbond mine. Of these the last three mines lie in Orissa and Noamundi iron mine lies in Jharkhand. The flowchart below describes the whole process:



Noamundi iron mine is an open cast heavily mechanized mine producing iron ore as fines and finished size. It is located on the interstate border of Orissa and Jharkhand. The same deposit extends upto the Joda east iron mine. In operation since 1925, the Noamundi Iron Mine (NIM) is a fully mechanised mine. The NIM supplies the principal raw materials for iron and steel making to the Company's steel works and other steel industries. Systematic mining and scientific processing of the ore enables it to conform to consistent physical and quality norms. The mine has belt conveyors, and loading onto railway wagons is fully mechanised. It produces sized ore (-40mm to + 10mm), LD ore (-40mm to +20mm) and blended fines (-10mm). The mining operations are carried out in series of 12 meter high benches 150mm diameter holes are drilled and blasted with explosives, the ore is then shovelled and trucked. The mine has the capability for dry processing of rich grade fine ore.

The NIM also processes and enriches the quality of the ore mined from the company's Katamati Iron Mine. The ore from this mine is transported to NIM and processed along with NIM ore at its Wet Processing and Dry Processing plants.

The total lease area of the mine is 1160.06 ha and the lease was obtained in the year 1923. Mining of the iron ore started in the year 1926 and the lease is valid till 31.12.2011. of the lease area about 762.43 ha of land is forest area and about 397.63 ha of land is non forest area.

To describe the deposit there are three essential features namely – topographic data, geological data and location data.

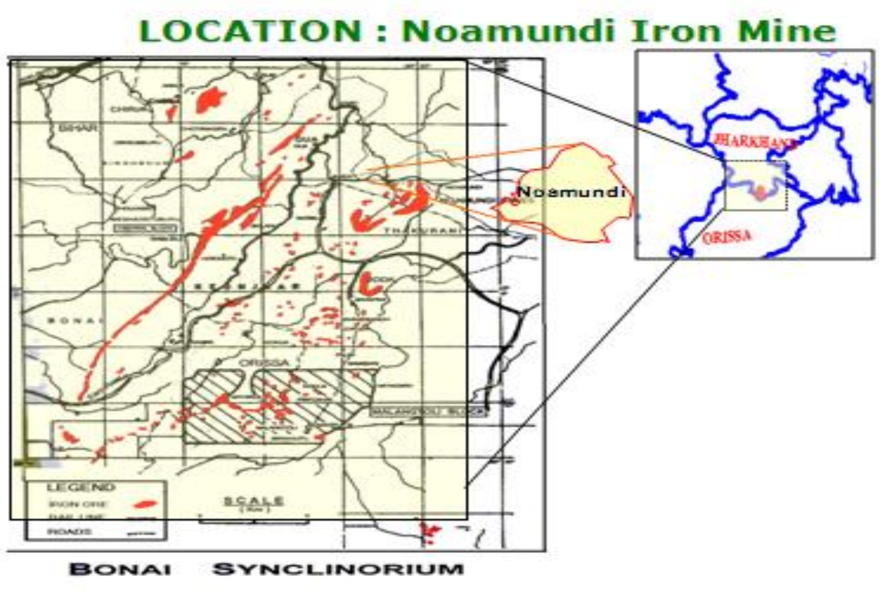
In any ore body there are three basic data sets that are required to establish a mining operation. They are

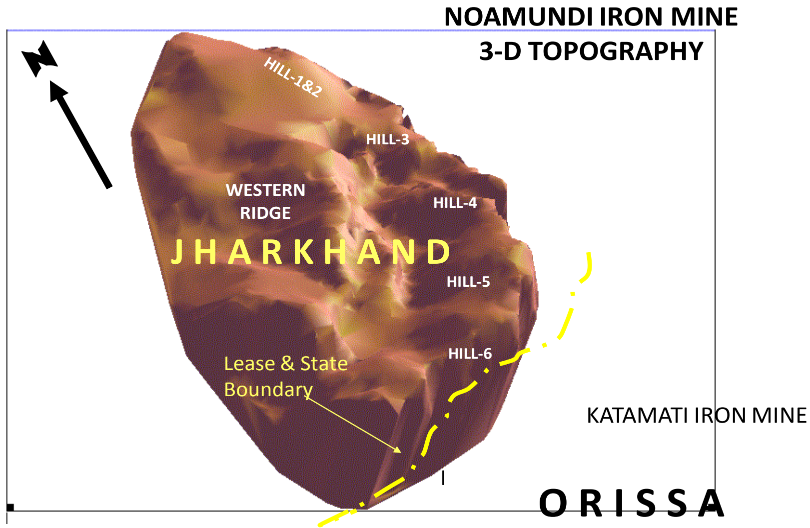
- TOPOGRAPHY DATA – it is an essential component as it gives an idea about the surrounding environment of the deposit. At Noamundi iron mine the entire area can be broadly classified as eastern ridge and western ridge that are separated by a small stream. The eastern ridge comprises of 6 distinctly visible hills while there are no such prominent hills in the western ridge. It is shown below in the diagram

- GEOLOGICAL DATA – it gives an idea of the kind of the deposit that is available and the nature of OB on the area and also faults or discontinuities if present any. In the Noamundi iron mine the eastern ridge has a strike of NNE-SSW and a dip of 20 to 40^0 west. The rock types of this area are quartzite’s, banded Haemetite jasper, iron ore, shale and lava. The ore found can be broadly classified into following types

1. HARD ORE – it is steel gray in color, fine grained, massive and is of homogeneous variety.
2. SOFT ORE – it is soft, spongy, laminated and often porous.
3. FRIABLE ORE – it is brownish to steel gray in color and contains kaolinous and shaly material.
4. BLUE DUST – these are natural fines containing powdery haematite.

9.1 LOCATION DATA – it is basically required to locate the area where the orebody is present and the state under which it falls. Noamundi is located at 22.15°N 85.53°E. It has an average elevation of 487 meters (1597 feet). Noamundi is located in the heart of Saranda forest, which is the densest deciduous forest of Asia. 'Saranda' in the local tribal language means 'The land of 700 hills'.





9.2 Mining Method In Noamundi Iron Ore Mine

- Noamundi Iron Mine is a fully mechanized Open Cast Mine.
- Production rate : 7.6 MTPA to 8.5 MTPA (During 2007 to 2011)
- The ROM from mine is processed in beneficiation plant and finished product (Sized Ore & Fines) is dispatched to Steel Plant.
- The mining operations are accomplished with the help of shovel - dumper combination. The bench height is kept at 12m and drilling is done by 150/165 mm dia drills with 10% sub-grade drilling.
- Blasting is done by adopting the state of art technology by using mostly SME (Site Mixed Emulsion Explosives) with the Nonel system of initiation so as to minimize adverse effect on environment such as ground vibration, noise and fly rock.
- The blasted material at the mining faces is loaded by shovels of different capacities into 50 / 60 tons dumpers.
- The ROM ore is hauled by the dumpers from different mining faces and dumped in the primary crusher in the pre-determined proportions for blending different qualities of ores.

9.3 *MACHINERY DEPLOYED*

The detail of the HEMM's used at the Noamundi iron mine are given below. Earlier 50 – 60 T dumpers were used but last year 4 new 90 T dumpers were ordered as the production was increased. The drills used are electrically operated while the shovels are diesel operated.

	CAPACITY OF EACH UNIT	**NUMBER OF UNITS**
Shovels	5.5 – 5.9 cu m	6
Drills	150 – 165 mm	7
Mining loaders	9 cu m	1
Dumpers	Rear dump truck (BEML / CAT, 50 / 60 T), Komatsu(90 T)	15+4
Dozers	D-155, CAT-D9R, Wheel Dozer, Komatsu	5
Graders	BEML , Komatsu	2
Loader	Front-End-Loader, 5.75 Cu.M.	3
Water sprinkler	28 KL	3
Trucks	10 T	6

10 DATA COLLECTION AND INTERPRETATION

Transport Systems in Iron Ore Mine

Time study of transport systems

Production loss due to break down of different equipment's in systems.

In dumper transport, the output from the dumper in an hour

Sl no	Equipment type	Equipment ID	Time taken	Distance travelled	Capacity
1	Rear Dumper	O389	4"30'	2	50 Tons.
2	Rear Dumper	O391	6"	2	50 Tons.
3	Rear Dumper	O392	break down		50 Tons.
4	Rear Dumper	O400	8"	3.5	50 Tons.
5	Rear Dumper	O401	9"10'	4	50 Tons.
6	Rear Dumper	O402	9"	3.5	50 Tons.
7	Rear Dumper	O403	11"	4	50 Tons.
8	Rear Dumper	O404	10"	3.5	50 Tons.
9	Rear Dumper	O405	2"	0.5	50 Tons.
10	Rear Dumper	O407	break down		60 Tons.
11	Rear Dumper	O409	break down		60 Tons.
12	Rear Dumper	O412	10"	3.5	60 Tons.
13	Rear Dumper	O414	9"20'	3.5	60 Tons.
14	Rear Dumper	O415	3"	1	60 Tons.
15	Rear Dumper	O416 (wb526)	3"30'	1	50 Tons.

q= (60.c.f)/T and T= $t_1+t_f+t_b+t_d+t_s$

- t_1=dumper loading time in minutes =3min
- t_f =forward haul time for dumper in minutes=4”30’for a distance travelled =2km
- t_b =backward haul time for dumper in minutes=5” ‘ ‘ ‘ ‘ ‘.
- t_d=time required for dumping and turning for the dumper near the primary crusher in min=3 min
- t_s=spotting time for dumper near shovel in min=2min

- f=fill factor for dumper=80%
- c= pay load capacity of the dumper in tons=50

So T=17”30’min

q= (60*50*80)/17”30’*100

q=141.2 tons

The face output per hour (Q) = (k*q*n)

Output from the mine per day =Q*2*6 considering 2 shifts production and 6 hrs. effective working in each shift.

For k=2; coefficient for truck utilization

n= no of dumper employed per shovel=3

Output from the mine per day=847.2 tons

Average utilization = availability /scheduled hour

=Scheduled hour –breakdown period /scheduled hour

=1-breakdown period/scheduled hour

Also net utilization=utilization hour/scheduled hours

=1-(breakdown period/scheduled hour)-(idle time/scheduled hour)

10.1 BREAK DOWN OF A DUMPER

Date	Time needed (hrs)	Frequency	Period Of Existence In (Hrs)	Idle time (hrs)
1.04.2009	-	-	-	

13.04.2009	222	1	3.416	0.5
14.04.2009	24	1	15.216	0.25
30.04.2009	56	1	0.514	0.3
25.05.2009	262	1	0.555	0.4
30.05.2009	40	1	4.716	0.2
15.06.2009	35	1	2.993	0.11
25.06.2009	168	1	6.533	0.6
10.07.2009	16	1	5.5	0.65
15.08.2009	12	1	3.083	0.49
15.09.2009	340	1	1.556	0.25

Date	Average utilization (%)	Net utilization (%)	
1.04.2009	-	-	
13.04.2009	98.416	97.55	
14.04.2009	36.6	35.89	
30.04.2009	99.08	97.52	
25.05.2009	97.86	96.55	
30.05.2009	88.21	86.32	
15.06.2009	91.44	90.65	

25.06.2009	96.11	95.23	
10.07.2009	65.62	64.98	
15.08.2009	74.29	73.22	
15.09.2009	99.54	98.64	

Conversion of Interval Between dumper Breakdowns to Cumulative Random Numbers

Interval between dumper breakdowns	**Frequency of occurrence**	**Frequency of occurrence (%)**	**Cumulative random number**
0-10	1	1000	0-1000
20-30	3	3000	1000-4000
30-40	3	3000	4000-7000
40-50	1	1000	7000-8000
50-60	1	1000	8000-9000
60-70	1	1000	9000-10000
	10	10000	

Conversion of Existency of Dumper Breakdowns to Cumulative Random Numbers

Existency of dumper breakdowns	**Frequency of occurrence**	**Frequency of occurrence (%)**	**Cumulative random number**
0-1	2	2000	0-2000
1-2	1	1000	2000-3000
2-3	1	1000	3000-4000
3-4	2	2000	4000-6000
4-5	1	1000	6000-7000
15-16	1	1000	7000-8000
	8	8000	

10.2 BREAK DOWN OF SHOVEL

Date	Time needed (hrs)	Frequency	Period Of Existence In (Hrs)	Idle time (hrs)
1.04.2009	-	-	-	
13.04.2009	25	1	7.216	0.55
14.04.2009	18	1	14.116	1.23
30.04.2009	56	1	8.514	2.35
25.05.2009	22	1	5.555	1.6

30.05.2009	72	1	5.716	2.54
15.06.2009	15	1	9.993	3.31
25.06.2009	36	1	7.533	0.87
10.07.2009	29	1	7.5	2.89
15.08.2009	55	1	6.44	0.64
15.09.2009	48	1	8.45	0.89

Date	Average utilization (%)	Net utilization (%)
1.04.2009	-	-
13.04.2009	71.11	70.23
14.04.2009	21.57	20.23
30.04.2009	84.79	83.56
25.05.2009	74.75	74.09
30.05.2009	92.06	91.87
15.06.2009	33.38	32.15
25.06.2009	79.07	78.23
10.07.2009	74.13	73.59
15.08.2009	88.29	87.98
15.09.2009	82.39	81.68

Conversion of Interval between Shovel Breakdowns to Cumulative Random Numbers

Interval between shovel breakdowns	Frequency of occurrence	Frequency of occurrence (%)	Cumulative random number
10-20	1	1000	0-1000
20-30	3	3000	1000-4000
30-40	2	2000	4000-6000
40-50	1	1000	6000-7000
50-60	2	2000	7000-9000
70-80	1	1000	9000-10000
Total	10	10000	

10.3 BREAK DOWN OF TRUCK

Date	Time Needed In Hrs	Frequency	Period Of Existence In (Hrs)	Idle Time (hrs)
1.04.2009	-	-	-	-
13.04.2009	36	1	2.558	1.5
14.04.2009	14	1	3.99	3.1
30.04.2009	34	1	5.7	2.5

25.05.2009	19	1	7.99	0.38
30.05.2009	56	1	5.4	1.67
15.06.2009	71	1	2.14	2.44
25.06.2009	18	1	3.66	0.97
10.07.2009	39	1	8.9	6.5
15.08.2009	85	1	7.66	4.21
15.09.2009	21	1	1.98	8.4

Date	Average utilization (%)	Net utilization (%)
1.04.2009	-	-
13.04.2009	92.89	90.56
14.04.2009	71.5	70.89
30.04.2009	83.23	81.21
25.05.2009	57.94	56.89
30.05.2009	90.35	88.25
15.06.2009	96.98	94.68
25.06.2009	79.66	74.65
10.07.2009	77.17	75.64
15.08.2009	90.98	89.78
15.09.2009	91.57	89.22

Conversion of Interval between Truck Breakdowns to Cumulative Random Numbers

Interval between truck breakdowns	Frequency of occurrence	Frequency of occurrence (%)	Cumulative random number
10-20	2	2000	0-2000
20-30	2	2000	2000-4000
30-40	3	3000	4000-7000
50-60	1	1000	7000-8000
70-80	1	1000	8000-9000
80-90	1	1000	9000-10000
Total	10	10000	

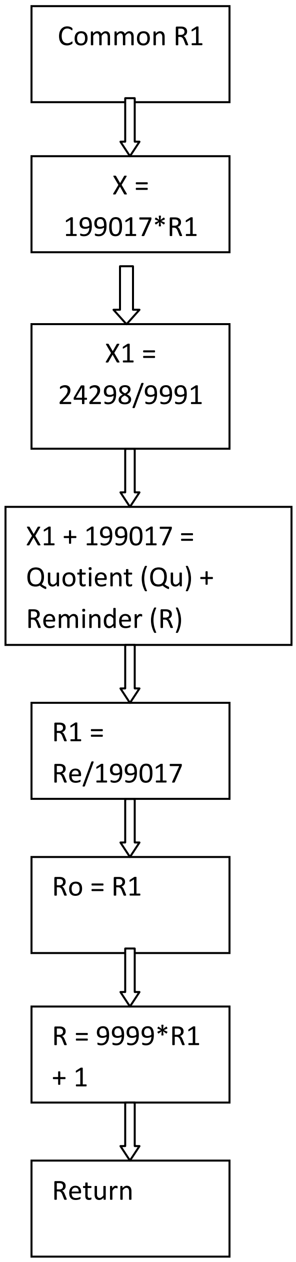
10.4 BREAK DOWN OF A LOADER

Date	Time needed (hrs)	Frequency	Period Of Existence In (Hrs)	Idle time (hrs)
1.03.2009	-	-	-	-
13.03.2009	28	1	5.324	0.98
14.03.2009	20	1	8.116	1.55
30.03.2009	49	1	12.514	2.3
25.04.2009	37	1	6.54	4.15
30.04.2009	59	1	8.79	2.68

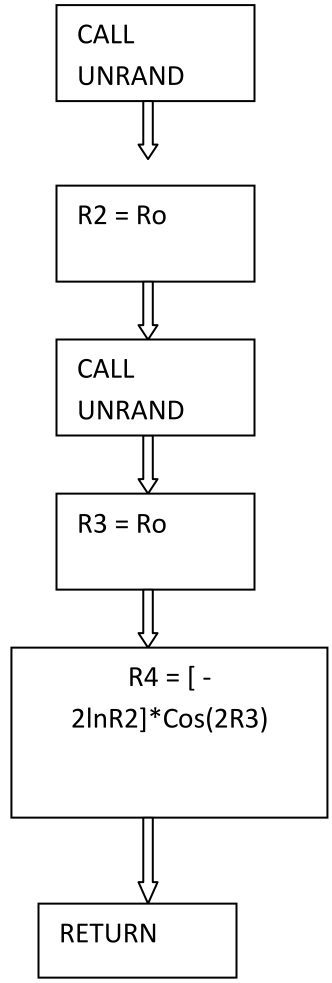
15.07.2009	64	1	3.87	0.87
25.07.2009	27	1	9.65	1.89
10.08.2009	18	1	12.46	5.64
15.09.2009	44	1	1.26	3.24
15.10.2009	38	1	8.46	5.23

Date	Average utilization (%)	Net utilization (%)
1.03.2009	-	-
13.03.2009	80.98	79.87
14.03.2009	59.42	58.66
30.03.2009	74.46	74.13
25.04.2009	82.32	81.65
30.04.2009	85.10	84.79
15.07.2009	93.95	92.89
25.07.2009	64.25	63.85
10.08.2009	30.77	29.65
15.09.2009	97.13	96.22

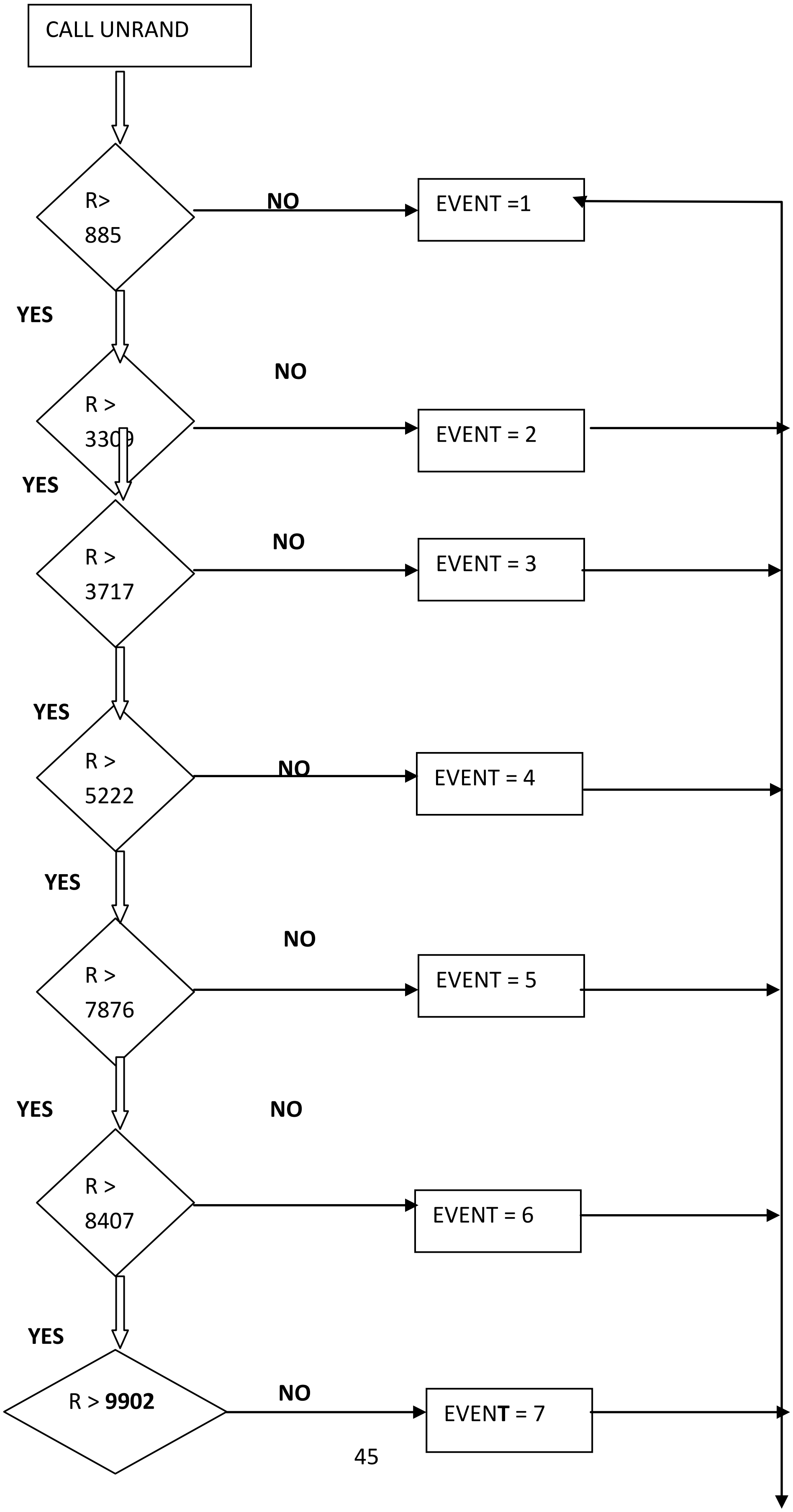
11 FLOWCHARTS



UNRAND (Uniformly Distributed Random Number)

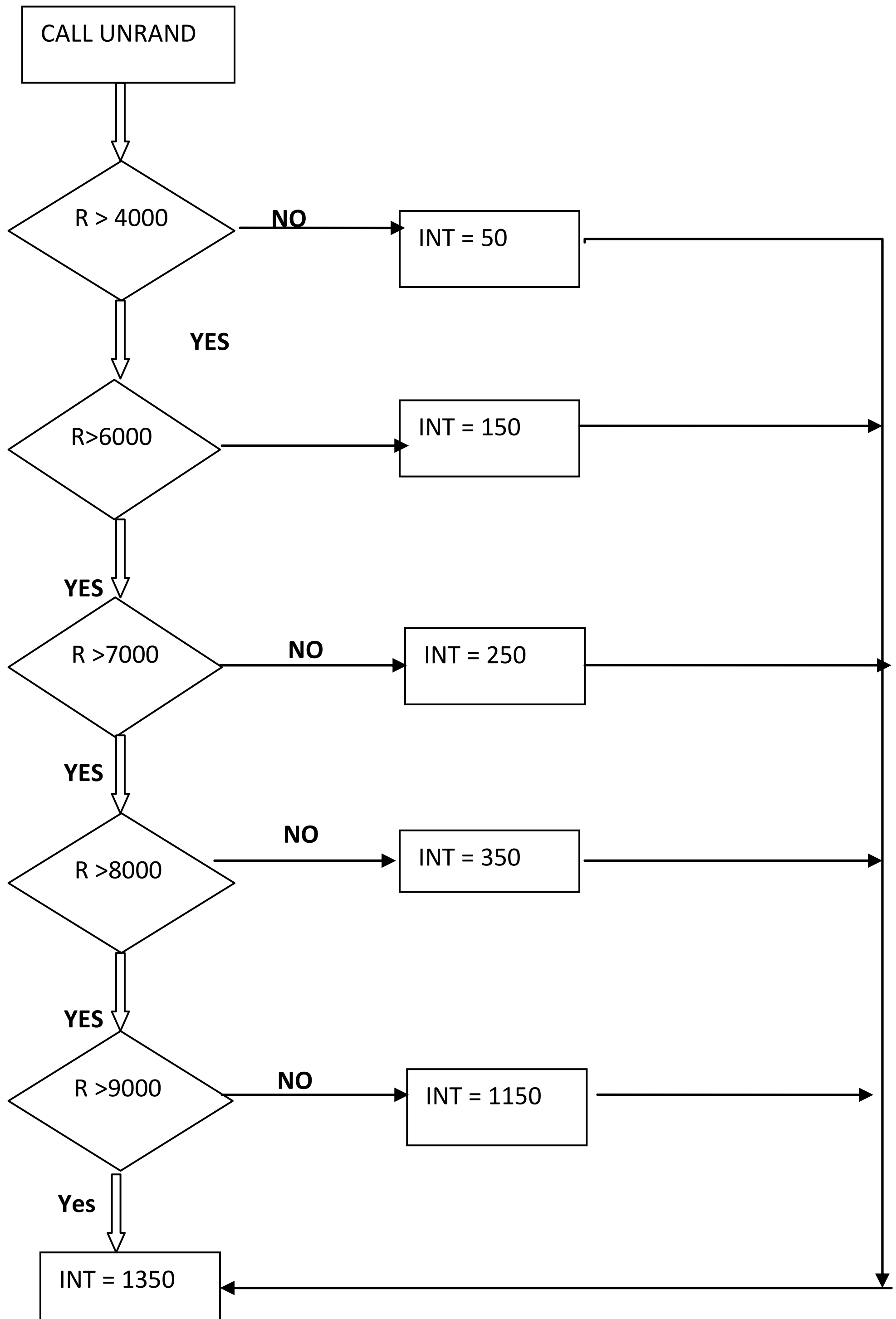


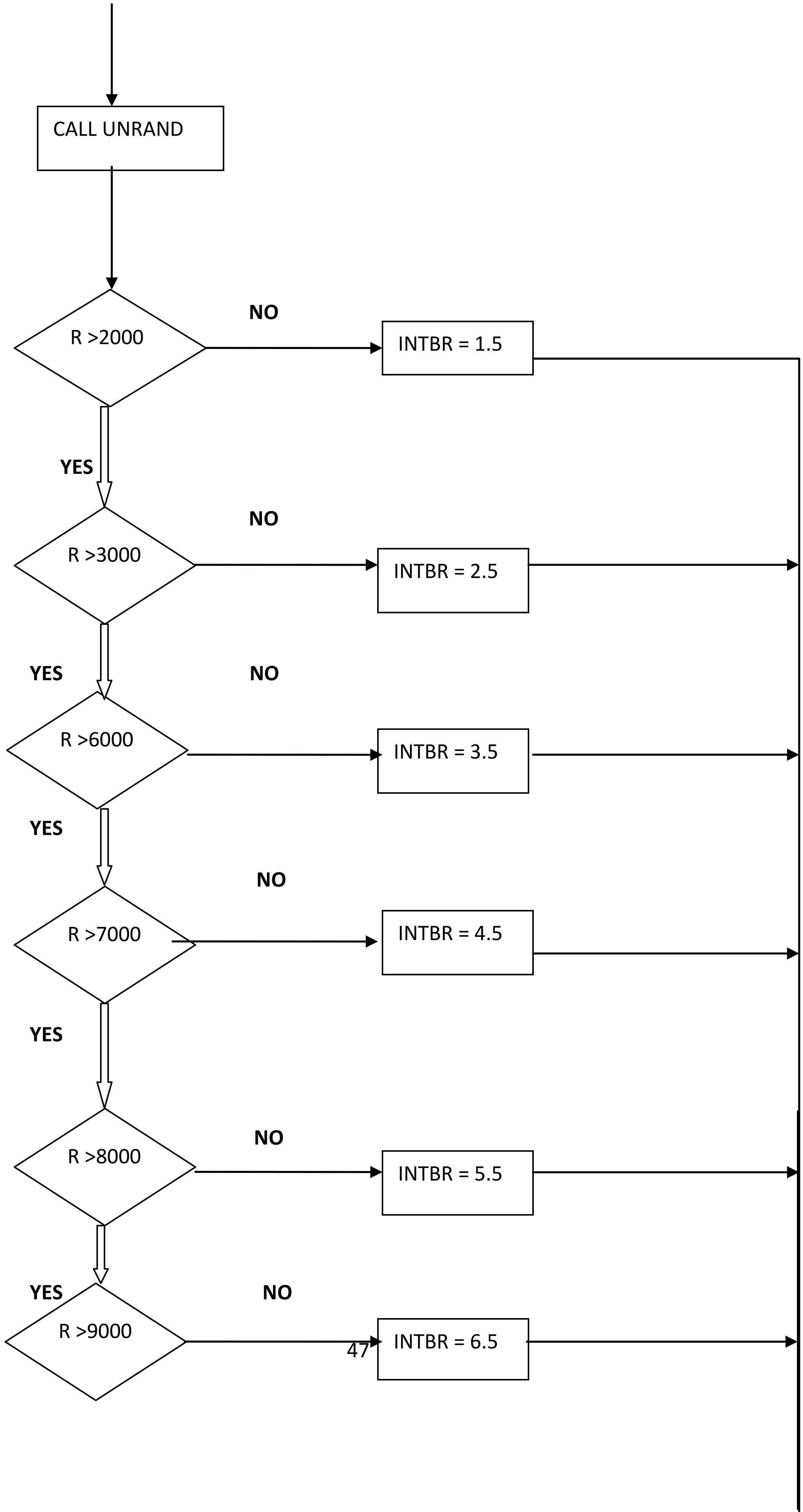
NORAND (Normally Distributed Random Numbe



CALL UNRAND
R> 885
NO
EVENT =1
YES
R > 3309
NO
EVENT = 2
YES
R > 3717
NO
EVENT = 3
YES
R > 5222
NO
EVENT = 4
YES
R > 7876
NO
EVENT = 5
YES
R > 8407
NO
EVENT = 6
YES
R > **9902**
NO
EVEN**T** = 7

Flowchart showing sub-routine event




CALL UNRAND
R >2000
NO
INTBR = 1.5
YES
R >3000
NO
INTBR = 2.5
YES
NO
R >6000
INTBR = 3.5
YES
NO
R >7000
INTBR = 4.5
YES
NO
R >8000
INTBR = 5.5
YES
NO
R >9000
INTBR = 6.5

Flowchart Showing Sub-Routine Shovel Break-Down

12 PROGRAMMING FOR SIMULATION

```
#include<iostream.h>

#include<conio.h>

 #include <stdlib.h>

  #include <time.h>

  int UNRAND();

  int EVENT_IDENTIFICATION();

void main()

{

clrscr();

int R;

randomize ();

R=random (300);

cout <<"Uniform Random Number R="<<R<<"\n";

cout<<"For this R the event generated is : \n ";

if (R<200)

cout<<"Event 1 : Breakdown of Water Sprinklers";
```

```
else if(R>=200 && R<228)
cout<<"Event 2: Breakdown of Dozers";
else if(R>=228  && R<285)
cout<<"Event 3: Breakdown of Dumpers";
else if(R>=228 && R<285)
cout<<"Event 4: Breakdown of shovels";
else if(R>=228 && R<285)
cout<<"Event 5: Breakdown of Loaders";
getch();
 }
```

13 SUMMARY AND CONCLUSION

Simulation is a very powerful, problem technique. Its applicability is so general that it would be hard to point out disciplines or systems to which simulation has not been applied. The basic idea behind simulation is simple, namely, model the given systems by means of some equations and then determine its time dependent behavior. The simplicity of the approach when combined with computational power of the high speed digital computer makes simulation a powerful tool. Normally, simulation is used when either an exact analytic expression for the behavior of the system under investigation is not available, or the analytic solution is too time consuming or expensive

From the above mentioned analysis it is observed that the random number distribution can give some indication about the breakdown of different events like:

Sl no	Events	Distribution

1	Break down of water sprinklers	0-200
2	Break down of dozers	200-228
3	Break down of dumpers	228-285
4	Break down of shovels	228-285
5	Break down of loaders	228-285

CONCLUSION

- For the highly mechanized mines, minor decrease in net utilization, increase the production cost drastically and sudden breakdown of equipment welcomes production loss.
- This simulation and identification of possibility of break-down of each equipment (event) will help the preventive maintenance work.
- Computerized best-fit matching will definitely increase the net utilization as well as reduction of capital investment for purchasing equipment
- As a result, particular steady-state production from the mine will be possible with optimum solution

14 REFERENCES:

- http://www.cpcb.nic.in/upload/NewItems/NewItem_105_iron_ore_mining_31.07.08.pdf
- Deo, Narsingh, System Simulation with Digital computer, Prentice - Hall India, 1979
- Hammersely, J.M. and Handcomb, D.C., Monte-carlo Methods, Jhon wiley and Sons, Newyork, 1964
- Haunser Bemard, Harry M. Marcowitz, Herbert W. Karr, Simscript a Simulation Programming Language < Prenctice- hall India (1963)
- Pal, B.K., Simulation Of the rapid loading system of an Iron ore processing plant, M.Tech. thesis, I.I.T.,Kharagpur, 1984.
- Radhakrishnan, V.R. and Sen, P., Optimum number of storage bins for Monte carlo Simulation, Indian Chemical engineer, 23(1981), 16.
- Sentiappan, R and Roy, S.C., Application of queuing theory in oprn pit equipment selection, Tr. Min, Geol. Met. Inst. India, 80(1983).
- KLEINROCK,L., Queuing Systems, Volume 1 : Theory, Wiley Interscience, New York, 1975.
- COX,D.R. and W.L.SMITH, Queus, Jhon Wiley and Sons, New York, 1961.
- KOSTEN,L., Stochastic Theory of Simulating Systems, Pergamon Press, Oxford, 1973.
- LEE,A.M., Applied Queuing Theory, Macmillan and co., London,1967
- MORSE, P.M., Queues, Inventories and Maintenance, Jhon Wiley and Sons , New York, 1958
- NEWELLG.F, Applications Of Queuing Theory, Chapman and Hall, London,1971.
- PRABHU,N.U., QUEUES and INVENTORIES, Jhon Wiley and Sons, New York,1965.
- EMSHOFF,J.R. and R.L.SISSON, Design and Use of Computer Simulation Models, Macmillan Co., New York, 1970.
- GORDON,G., System Simulation, Prentice Hall, Englewood Cliffs, N.J.1969.
- MARTIN,F.F., Computer Modeling and Simulation, John wiley and Sons, New York, 1968.
- MIZE,J.H. and J.G.COX, Essentials of Simulation, Prentice-Hall, Englewood Cliffs, N.J.,1968.

- NAYLOR,T.H.,J.L.BALINTFY, D.S.BURDICK, and K.CHU, Computer Simulation Techniques, John Wiley and Sons, New York,1966.
- SHANNON,R.E., Systems Simulation : the Art and Science, Prentice-Hall, Englewood Cliffs, N.Y.,1975.
- FISHMAN,G.S., Concepts and Methods in Discrete Event Digital Simulation, John Wiley and Sons, New York, 1973.
- ZEIGLER,B.P. Theory of Modelling and Simulation, Wiley-Interscience, New York, 1976.
- MAISEL,H. and GNUGNOLI, Simulation of Discrete Stochastic Systems, Science Research Associates, Chicago,1975.
- CHU,Y., Digital Simulation of Continuous Systems, Mcgraw Hill, New York, 1969.
- ORD-SMITH,R.J. and J.STEPHENSON, Computer simulation of continuous Systems, Cambridge University Press, Cambridge, U.K.,1975.
- STPHENSON,R.E., Computer Simulation for engineers, Harcourt,Brace,Jovanovich,New York,1971.
- SHAH,M.J., Engineering Simulation Using Small Scientific Computers, Prentice-Hall, Englewood Cliffs,N.J.,1976.
- McLEOD,J.(ed.).,SIMULATION : The dynamic Modeling if Ideas and Systems with Computers, Mcgraw-Hill, New York,1968.
- HUFSHMIDT,M. and M.B.FIERING, Simulation Techniques for the Design of Water Resources Systems, Harvard University Press, Cambridge, Mass.,1986.
- LUYBEn,W.L., Process Modeling, Simulation and control for chemical engineers, McGraw-Hill, New York 1973.
- ACTON,F.S., Numerical Methods that Work, Harper and Row, New York. 1970.
- CONTE,S.D. and C.DeBOOR, Elementary Numerical Analysis: An Algorithmic Approach,(2ndEd.), Mcgraw Hill, New York,1968.
- DAHLQUIST G. and A.BJORCK, Numerical Methods, Prentice-Hall, Englewood CLIFFS,N.J.,1974.
- RALSTON,A.,A first Course in Numerical Analysis, Mcgraw Hill, New York, 1965.
- RAJARAMAN,V., Computer Oriented numercal Methods, Prentice-Hall of India, New Delhi,1971.