

MODELING AND SIMULATION

LECTURE 5

CHAPTER 5

Monte Carlo Simulation

2

- We define Monte Carlo simulation as *a scheme (منظومة) employing random numbers, that is, $U(0, 1)$ random variates, which is used for solving certain stochastic or deterministic problem.* Thus, a stochastic discrete-event simulation is included in this definition. The name “**Monte Carlo**” simulation or method originated during World War II, when this approach was applied to problems related to the development of the *atomic bomb and other weapons*.

5.1 Introduction to Monte Carlo Simulation

5.1.1 Introduction

- Monte Carlo technique deals with taking random samples from distribution of a variable in order to supply a series of values for use in the model.
- This requires *a method of generating or obtaining random variates from the uniform distribution* on the interval $[0, 1]$ this distribution is denoted by $U(0, 1)$ and random variates generated from the $U(0, 1)$ distribution will be called random numbers.

Introduction Monte Carlo Simulation

- The probability of any random number in that domain $[0, 1]$ is equal. Random values can be conveniently and efficiently generated from a desired probability distribution by **the inverse transform method** for use in executing simulation models. Other methods of generating random numbers like **linear congruential and mixed generators** are explained in **chapter 3**.
- **Monte Carlo methods are stochastic and are typically simple to implement.**
- It is often used to solve **stochastic problems** to mimic randomness of physical behaviour.
- Monte Carlo can also be used to solve **deterministic problems** by using probability for estimation.

Introduction Monte Carlo Simulation

5.1.2 Requirements of Stochastic Simulation

- Knowledge of relevant probability distributions depend on the **theoretical or empirical information about physical system being simulated.**
- **Supply of random numbers for making random choices.** By simulating large number of trials, probability distribution of overall results can be approximated with accuracy attained depending on number of trials. ***As the number of iterations increases we come to actual value.***

Introduction Monte Carlo Simulation

Example 5.1

In a queueing system, the arrival rate and service rate in minutes with their probabilities are given as:

Arrival rate	Probability		Service rate	Probability
3	15 %		4	30 %
4	30 %		5	40 %
5	20 %		6	30 %
6	35 %			

As a conclusion from the given data, ***they are not following any typical distribution.***

To get help from random numbers in predicting the values of number of arrival and service rate, we find the ***cumulative frequency and distribute the uniformly distributed two digit random numbers (from 00 to 99) according to frequency as shown in the following table:***

Introduction Monte Carlo Simulation

6

Arrival rate	Frequency	Cumulative frequency	Two digit random numbers
3	15	15	00 – 14
4	30	45	15 – 44
5	20	65	45 – 64
6	35	100	65 -99

Service rate	Frequency	Cumulative frequency	Two digit random numbers
4	30	30	00 – 29
5	40	70	30 – 69
6	30	100	70 - 99

Generating a stream of random numbers (*from 00 to 99*): (78, 47, 13, 87, 89, 05 ...) and pick the first number **78** which corresponds to arrival rate of **6 minutes**, and similarly for the service rate. We can use *the random function software* to generate uniformly distributed random numbers from the *excel sheet* as shown in the following example.

Introduction Monte Carlo Simulation

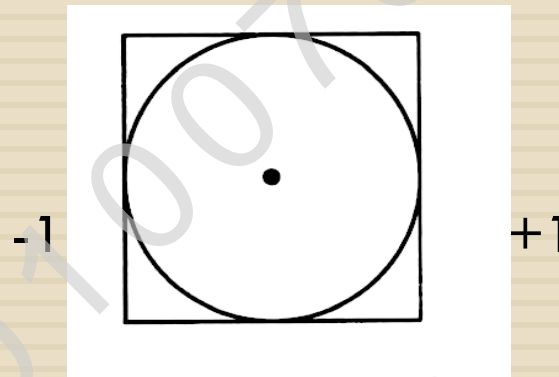
7

Example 5.2

Calculate the deterministic value of π by using random numbers.

Solution

Assume a circle of **radius=1** and its center is at the origin, then its area is π . The circle is surrounded by a square, the length of its **side=2**.



If we select a point (*from -1 to 1*) at random, it *may be located inside the circle or outside the circle but still inside the square*. If we repeat this selection for *10 times*, we may get *7 times the point is located inside the circle* and *3 times outside the circle but still inside the square*.

Introduction Monte Carlo Simulation

8

This means that: ***Probability of a point located inside the circle = 0.7***

Probability of a point located outside the circle = 0.3

Therefore: ***The area of the circle = 0.7 the area of the square***

The area of the circle = $0.7 \times (2 \times 2) = 2.8$

- ***Actually the area of the circle = $\pi \approx 3.14$, so the result of 2.8 is an approximation and as the number of iterations increases it will go closer to the actual value.***
- We can do that on the ***Excel sheet*** by generating a stream of random numbers using ***rand ()*** function. Let us first have the numbering (1, 2... to 500) in column A, then generate random numbers between [0, 1] by the command ***rand ()*** in column B.

Introduction Monte Carlo Simulation

9

	A	B	C	D	E	F	G	H	I	J	K
422	422	0.792474									
423	423	0.630393									
424	424	0.887268									
425	425	0.60742									
426	426	0.074592									
427	427	0.312088									
428	428	0.474509									
429	429	0.079151									
430	430	0.202532									
431	431	0.010498									
432	432	0.664024									
433	433	0.864519									
434	434	0.336462									
435	435	0.065581									
436	436	0.916816									
437	437	0.48908									
438	438	0.45364									
439	439	0.00226									
440	440	0.305381									
441	441	0.259473									
442	442	0.535631									
443	443	0.688949									

For the circle example, the random numbers should be in between $[-1, 1]$, so we use the function as: $1-2*\text{rand}()$ and the result will be in column C, then generating a similar stream in column D:

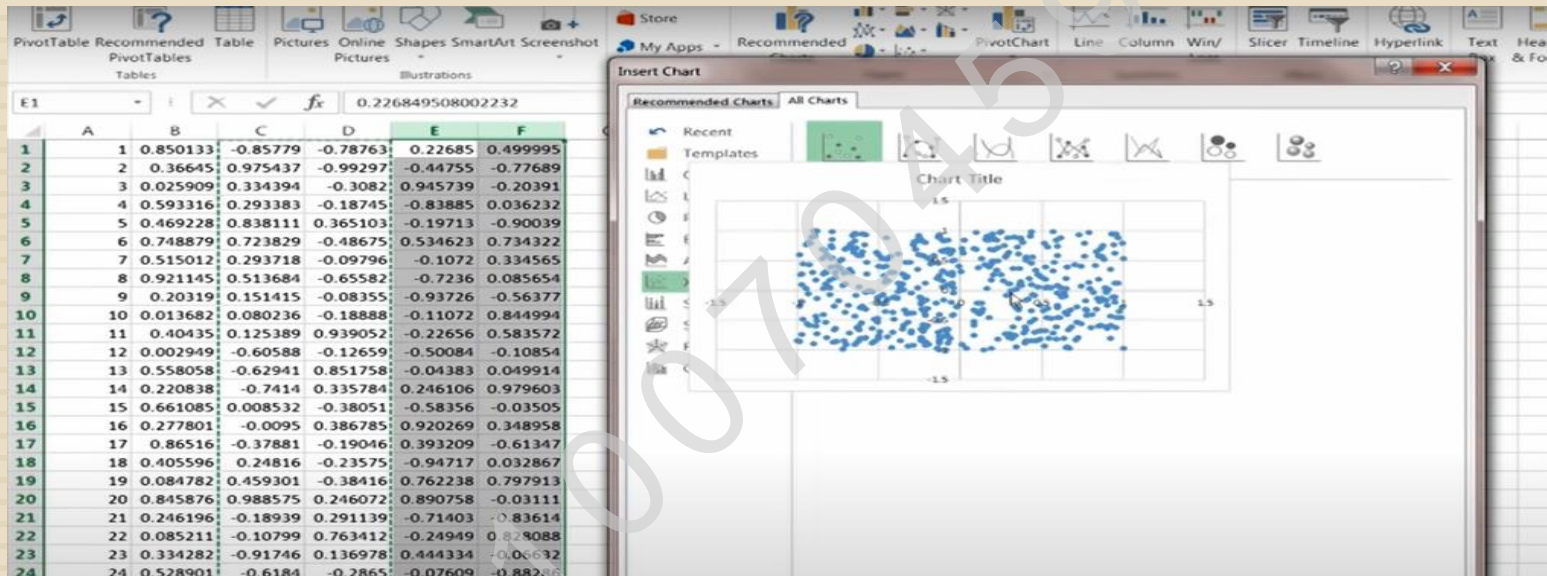
Introduction Monte Carlo Simulation

[illegible][illegible]

Introduction Monte Carlo Simulation

11

- If we plot these points on the scatter (from **Insert** menu on the excel sheet choose **scatter**) to notice that their uniform distribution:



To draw these random numbers in a circle of radius = 1, the circle equation is $x^2 + y^2 = 1$ with the center in the origin. On the Excel sheet in column G we have the condition: **IF (Column E)² + (column F)² < 1, then 1, else 0**
= if ((E1^2 + F1^2) < 1, 1, 0)

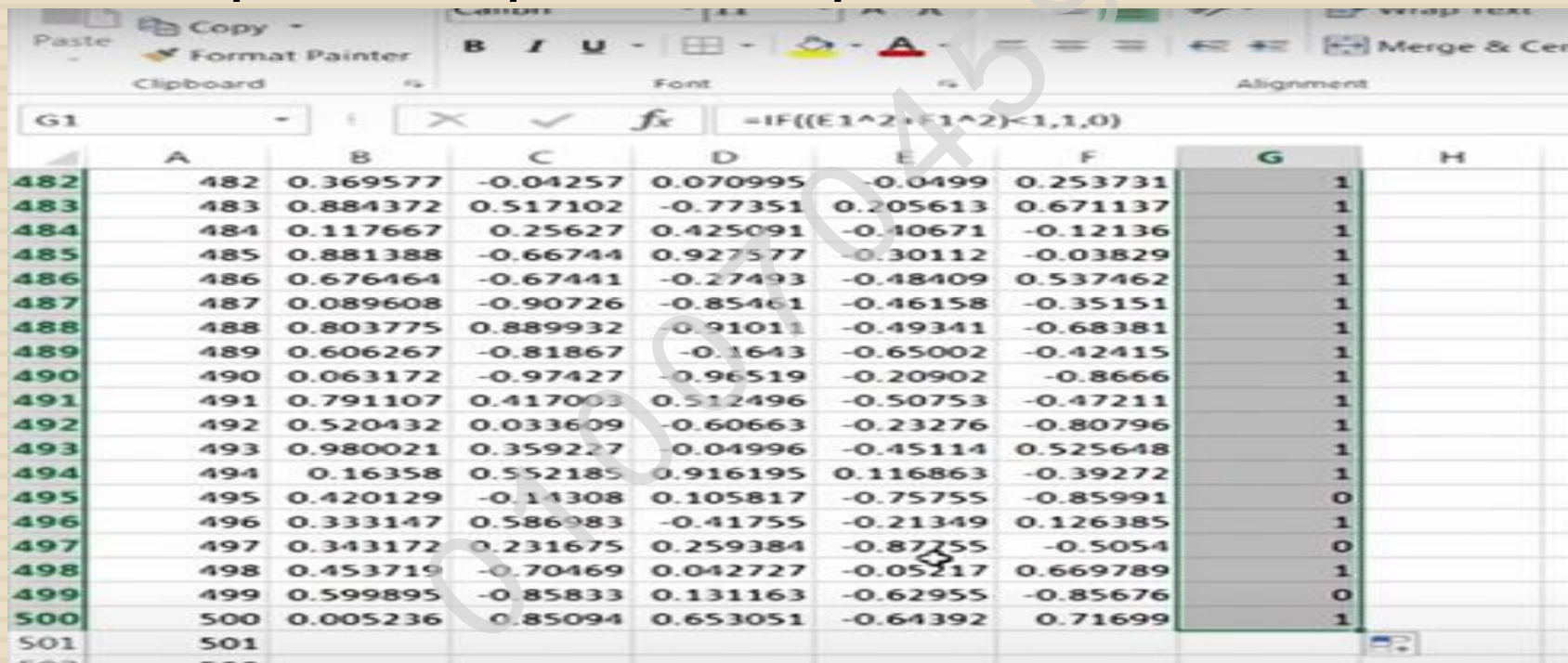
Gives the points *in the quarter of the circle*.

Introduction Monte Carlo Simulation

12

the condition: $(\text{Column E})^2 + (\text{column F})^2 \leq 1$, then 1, else 0

Gives the points in the quarter of the square.



	A	B	C	D	E	F	G	H
482	482	0.369577	-0.04257	0.070995	-0.0499	0.253731	1	
483	483	0.884372	0.517102	-0.77351	0.205613	0.671137	1	
484	484	0.117667	0.25627	0.425091	-0.10671	-0.12136	1	
485	485	0.881388	-0.66744	0.927577	-0.30112	-0.03829	1	
486	486	0.676464	-0.67441	-0.27493	-0.48409	0.537462	1	
487	487	0.089608	-0.90726	-0.85461	-0.46158	-0.35151	1	
488	488	0.803775	0.889932	-0.91011	-0.49341	-0.68381	1	
489	489	0.606267	-0.81867	-0.1643	-0.65002	-0.42415	1	
490	490	0.063172	-0.97427	-0.96519	-0.20902	-0.8666	1	
491	491	0.791107	0.417003	0.512496	-0.50753	-0.47211	1	
492	492	0.520432	0.033609	-0.60663	-0.23276	-0.80796	1	
493	493	0.980021	0.359227	-0.04996	-0.45114	0.525648	1	
494	494	0.16358	0.552185	0.916195	0.116863	-0.39272	1	
495	495	0.420129	-0.14308	0.105817	-0.75755	-0.85991	0	
496	496	0.333147	0.586983	-0.41755	-0.21349	0.126385	1	
497	497	0.343172	0.231675	0.259384	-0.87755	-0.5054	0	
498	498	0.453719	-0.70469	0.042727	-0.05217	0.669789	1	
499	499	0.599895	-0.85833	0.131163	-0.62955	-0.85676	0	
500	500	0.005236	0.85094	0.653051	-0.64392	0.71699	1	
501	501							

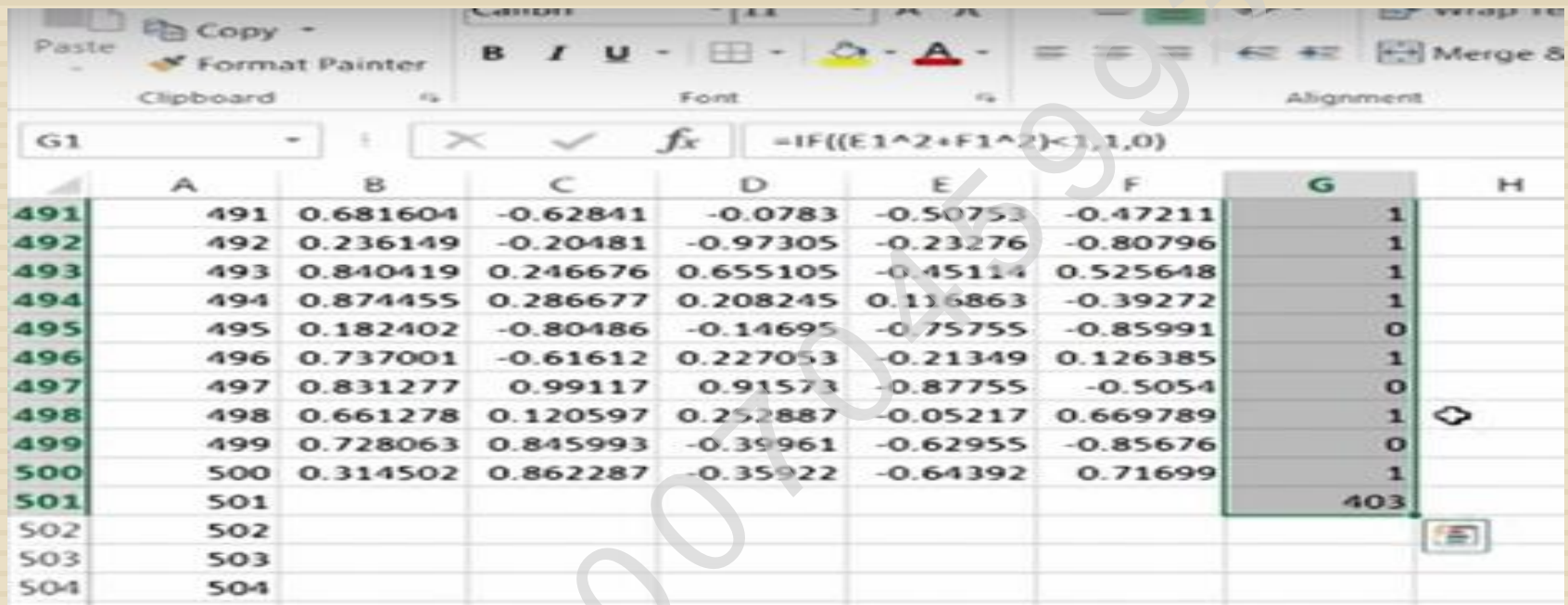
For example:

$(0.87)^2 + (0.505)^2 = 0.7569 + 0.255 = 1.0119 > 1 \rightarrow 0$ i.e. outside the circle

$(0.05)^2 + (0.66)^2 = 0.0025 + 0.4356 = 0.4381 < 1 \rightarrow 1$ i.e. inside the circle

Introduction Monte Carlo Simulation

13



	A	B	C	D	E	F	G	H
491	491	0.681604	-0.62841	-0.0783	-0.50753	-0.47211	1	
492	492	0.236149	-0.20481	-0.97305	-0.23276	-0.80796	1	
493	493	0.840419	0.246676	0.655105	-0.45114	0.525648	1	
494	494	0.874455	0.286677	0.208245	0.116863	-0.39272	1	
495	495	0.182402	-0.80486	-0.14695	-0.75755	-0.85991	0	
496	496	0.737001	-0.61612	0.227053	-0.21349	0.126385	1	
497	497	0.831277	0.99117	0.91573	-0.87755	-0.5054	0	
498	498	0.661278	0.120597	0.252887	-0.05217	0.669789	1	
499	499	0.728063	0.845993	-0.35961	-0.62955	-0.85676	0	
500	500	0.314502	0.862287	-0.35922	-0.64392	0.71699	1	
501	501						403	
502	502							
503	503							
504	504							

The sum of column G gives 403 which means 403 ones out of 500, i.e.

$1/4$ of the area = $403/500 = 0.806 \approx \pi/4$ which means $\pi \approx 3.224$

While the actual value of π is 3.14.

If we increase the number of points more than 500 the result will be closer to the actual value.

Introduction Monte Carlo Simulation

14

	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
9036		9035	0.111996	0.05393	9034	-0.73506	0.616849	-0.73506	0.616849	1	1	
9037		9036	-0.67621	-0.41704	9035	0.771019	-0.85972	0	0	0	1	
9038		9037	-0.91788	0.091685	9036	0.423377	-0.438	0.423377	-0.438	1	1	
9039		9038	-0.10909	-0.98994	9037	-0.6796	0.778415	0	0	0	1	
9040		9039	0.496315	-0.13704	9038	-0.70003	0.667369	-0.70003	0.667369	1	1	
9041		9040	0.943058	0.813325	9039	-0.75443	0.13484	-0.75443	0.13484	1	1	
9042		9041	-0.0847	0.161169	9040	0.930772	0.223361	0.930772	0.223361	1	1	
9043		9042	-0.08577	0.932784	9041	-0.27427	0.998539	0	0	0	1	
9044		9043	0.337188	-0.79446	9042	0.95344	-0.5435	0	0	0	1	
9045		9044	-0.65313	-0.73275	9043	-0.05055	0.191511	-0.05055	0.191511	1	1	
9046		9045	0.355877	-0.48675	9044	-0.576	-0.33419	-0.576	-0.33419	1	1	
9047		9046	-0.36553	0.839768	9045	0.065446	-0.0066	0.065446	-0.0066	1	1	
9048		9047	0.17097	0.398822	9046	0.258921	-0.59693	0.258921	-0.59693	1	1	
9049		9048	-0.4202	-0.29098	9047	0.596386	0.210163	0.596386	0.210163	1	1	
9050		9049	-0.71896	-0.72332	9048	-0.37848	-0.26891	-0.37848	-0.26891	1	1	
9051		9050	0.664953	-0.11874	9049	0.926885	-0.78109	0	0	0	1	
9052		9051	0.916883	-0.30728	9050	0.005152	-0.41738	0.005152	-0.41738	1	1	
9053		9052	0.5156	0.127699	9051	0.794619	0.042357	0.794619	0.042357	1	1	
9054		9053	-0.01183	0.214748	9052	-0.49249	-0.43334	-0.49249	-0.43334	1	1	
9055		9054	-0.81943	0.264617	9053	-0.17105	-0.53544	-0.17105	-0.53544	1	1	
9056		9055	-0.09319	0.01629	9054	-0.11091	0.050493	-0.11091	0.050493	1	1	
9057		9056	-0.83904	-0.78924	9055	-0.16286	0.61441	-0.16286	0.61441	1	1	
9058		9057	-0.16028	0.765966	9056	-0.47883	0.955191	0	0	0	1	
9059		9058	-0.8305	0.562403	9057	-0.34008	0.985192	0	0	0	1	

For a stream of 10000 random numbers we get:

7857 ones out of 10000, i.e. the area = $(7857/10000) \times 4 = 3.1428$

Inventory Control Simulation

15

5.2 Inventory Control Simulation using Monte Carlo Technique

- For inventory control problem we are interested in finding **when to reorder**, **what is the reorder level**, **what is the daily demand**, **what is the stock in hand**, **what is the total cost of holding material**, and **what is the shortage cost**. Those issues are to be modeled through this technique.

Example 5.3

Suppose you have certain **demand for the past 100 days** and the **lead time** in which the ordered quantity has arrived are given in the following table.

- Notice that for the **lead time** we may have a **standard frequency distribution** rather than the given frequencies to find the random numbers corresponding to a particular lead time.
- We need to know **the demand in the coming days** and the **possible lead time**, and **when to order**.

Inventory Control Simulation

16

22	44	33	57	41	45	49	47	44	44	Demand for past 100 days
48	45	33	55	41	42	50	72	46	43	
66	40	70	43	39	49	44	66	39	31	
52	42	44	31	42	34	38	38	48	31	
52	43	50	57	42	44	43	40	53	48	
52	49	50	37	40	28	46	47	31	38	Lead time
26	30	62	35	41	42	48	39	38	47	3 days
32	41	43	43	64	51	55	31	55	46	4 days
52	45	44	56	34	52	50	56	59	58	5 days
58	44	57	53	48	54	33	27	57	39	Frequency

Suppose that in the inventory control we have the following costs:

C_0 ... is the cost of placing an order = 100 pound/order

C_i ... is the cost of holding an item in stock = 0.1 pound/unit/day

C_s ... is the cost of shortage = 5 pounds/unit

Inventory Control Simulation

17

(when you have shortage you have to bring items from local market at higher price)

We will calculate what will be the reordered level and other quantities by statistical calculations. We have: **Mean demand μ_D**

$$\mu_D = \Sigma \text{ all demand in } (n = 100) \text{ days} / 100 = 45.23 \text{ unit/day}$$

Standard deviation of the demand $\sigma_D = \sqrt{[(\Sigma Di^2) - [(\Sigma Di)^2 / n]] / (n-1)}$

$$\sigma_D = \sqrt{[214077 - [(4523)^2 / 100]] / 99} = 9.8$$

Mean lead time $\mu_L = [(28 \times 3) + (44 \times 4) + (28 \times 5)] / 100 = 400 / 4 = 4 \text{ days}$

Standard deviation of the lead time σ_L

$$\sigma_L = \sqrt{\{[(28 \times 3)^2 + (44 \times 4)^2 + (28 \times 5)^2] - [(400)^2 / 100]\} / 99} = 0.75$$

All these calculations can be done from the available data.

Inventory Control Simulation

18

- Now we find the **histogram of the demand** which ranges from **20 to 70**, we calculate the **frequency, cumulative frequency**, and assign **two digit** random numbers (from **00 to 99**) for each interval of the demand according to its frequency as shown in the following table. **When we predict the demand we will take the average value of the demand in each interval.**

Demand interval	Average demand	frequency	Cumulative frequency	Random numbers
20 – 24	22	1	1	00
25 – 29	27	3	4	01 – 03
30 – 34	32	12	16	04 – 15
35 – 39	37	10	26	16 – 25
40 – 44	42	26	52	26 – 51
45 – 49	47	17	69	52 – 68
50 – 54	52	13	82	69 – 81
55 – 59	57	12	94	82 – 93
60 – 64	62	2	96	94 – 95
65 – 69	67	2	98	96 – 97
70 – 74	72	2	100	98 – 99

Inventory Control Simulation

19

Similarly we can have the calculations for *the lead time*:

Lead time	frequency	Cumulative frequency	Random numbers
3	28	28	00 – 27
4	44	72	28 – 71
5	28	100	72 – 99

economic order quantity (EOQ) is given as:

$$EOQ = \sqrt{2 \mu_D C_0 / C_i} = \sqrt{(2 \times 45.23 \times 100) / 0.1} \approx \mathbf{300 \text{ units}}$$

Whenever the stock quantity reaches below certain limit (the **reorder level**) we have to order this quantity (**300 units**) because we are going for fixed quantity system.

The reorder level is given as:

$$ROL = \mu_D \mu_L + K [\sqrt{\mu_L \sigma_D^2 + (\mu_D^2 \sigma_L^2)}]$$

Inventory Control Simulation

20

Where **K** is a constant that determines the **probability of shortage** of stock, e.g. for **$K = 1.65$** the chance of shortage of stock is **5 %** (i.e. **95%** sure that there will not be any shortage). If **$K = 2.33$** there will be only **1 %** chance of shortage.

For **$K = 1.65$** :

$$ROL = (45.23 \times 4) + 1.65 \times \sqrt{4 (95.98)^2 + (45.23)^2 \times 0.5657} = 245$$

Now we start the simulation with **EOQ = 300** and **ROL = 245 units**

- Assuming the **initial stock = 300 units** on **Friday** evening, so we will **start simulation from Monday** onwards.
- The random numbers can be generated from **(00 to 99)** if we put in the excel **100** into the **rand ()** function. We can also generate random numbers by using other generators.

Inventory Control Simulation

21

81	78	89	59	40	33	60	67	19	21
94	51	37	19	31	89	34	62	23	42
21	75	28	18	20	10	76	43	91	69
35	88	90	88	53	00	80	77	42	96
60	94	49	75	69	26	96	24	05	91
97	84	54	83	27	28	99	94	46	19
92	10	80	60	32	63	08	71	06	14
23	70	49	30	71	21	23	26	20	76
82	07	15	38	54	15	75	99	27	84
80	75	31	64	67	97	64	06	56	81
42	10	00	37	24	33	56	28	43	89
54	94	54	43	71	87	78	60	72	06
57	36	84	56	98	10	17	89	53	25
61	37	35	11	63	87	59	64	92	62
85	17	23	11	05	56	35	36	34	52
02	84	29	56	99	02	03	35	96	70
24	64	48	50	42	79	28	99	53	12
63	20	82	33	82	22	07	33	39	93
38	94	98	52	70	50	78	69	55	28
98	76	37	41	46	27	78	69	55	28

Random
numbers

So, suppose this is the random number which is to be used, so we have to use the first

We will use random numbers *for demand from the left side of the first row* and *for lead time from the right side of the last row (65, 67, 22 ..)*

Inventory Control Simulation

(**initial stock** = 300 units, **EOQ** = 300 units, and **ROL** = 245 units)

22

The simulation table for two working weeks (10 working days will be):

Day	Initial stock	R. No. for demand	Daily demand	Shortage	Quantity received	Final stock	Quantity in hand + order	Quantity ordered	R. No. for lead time	Lead time
M	300	81	52	-	-	248	248	0	-	-
T	248	78	52	-	-	196	196	300	65	4
W	196	89	57	-	-	139	439	0	-	-
Th	139	59	47	-	-	92	392	0	-	-
F	92	40	42	-	-	50	350	0	-	-
M	50	33	42	-	300	308	308	0	-	-
T	308	60	47	-	-	261	261	0	-	-
W	261	67	47	-	-	214	214	300	67	4
Th	214	19	37	-	-	177	477	0	-	-
F	177	21	37	-	-	140	440	0	-	-

Final stock = initial stock – daily demand + quantity received

On first Monday = ***300 – 52 = 248 units***, ***Initial stock = previous final stock*** →
on Tuesday = 248 and final stock = 248 – 52 = 196 → ***we have to reorder***

Inventory Control Simulation

Quantity in hand + order = final stock + quantity ordered

When the ***lead time is 4 working days*** from ***Tuesday***, the quantity ordered will be ***received on next Monday***, final stock will be:

Final stock on next Monday = (50 – 42) + 300 = 308 units

The second ordered quantity will be received on next Tuesday.

We can calculate parameters like cost in certain period:

Cost of holding inventory = holding cost per unit x final stock

Cost of ordering = number of orders x order cost

Monte Carlo Simulation of Queueing Problem

24

5.3 Monte Carlo Simulation of Queueing Problem

Monte Carlo simulation can be used in solving different kinds of problems like *inventory control, reliability, queueing* ... etc. In this section we discuss *Monte Carlo simulation of queueing problem*.

Example 5.4

At a *dentist clinic*, apply *Monte Carlo simulation* to find *the average waiting time of customers*. Simulation *starts at 8 a.m.* in the morning and the *arrival of customers is scheduled* (in other cases *it may be randomly depending on some probability distributions*). The *first customer* arrives at *8 a.m.* and every *30 minutes* one customer is coming for five different treatments to be carried out. The five treatment services are *filling, crown, cleaning, extraction, and checkup* with different probabilities and service times.

Monte Carlo Simulation of Queueing Problem

25

We do not know the treatment service for which a customer is coming and that guess will be done **by referring random numbers** generated from any generator as shown in the following table:

services	Service time	Probability of service	Cumulative probability	Random number intervals
Filling	40 min.	0.40	0.40	00 – 39
Crown	60 min.	0.15	0.55	40 – 54
Cleaning	15 min.	0.15	0.70	55 – 69
Extraction	45 min.	0.10	0.80	70 – 79
Checkup	15 min.	0.20	1.00	80 – 99

Then we can predict for what service a customer is coming.

Given the random numbers **40, 82, 11, 34, 25, 66, 17, 79** to be used for predicting the required service for **8 patients** by the dentist:

Monte Carlo Simulation of Queueing Problem

Patient	1	2	3	4	5	6	7	8
Arrival time	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30
Random number of service	40	82	11	34	25	66	17	79
Category of service	crown	checkup	filling	filling	filling	cleaning	filling	extraction
Service time	60	15	40	40	40	15	40	45

26

Then we have to find when a customer arrives and when he leaves to calculate the average waiting time:

patient	Arrival time	Service time	Service starts at:	Waiting time
1	8:00	60	8:00	0 minutes
2	8:30	15	9:00	30 minutes
3	9:00	40	9:15	15 minutes
4	9:30	40	9:55	25 minutes
5	10:00	40	10:35	35 minutes
6	10:30	15	11:15	45 minutes
7	11:00	40	11:30	30 minutes
8	11:30	45	12:10	40 minutes

Monte Carlo Simulation of Queueing Problem

27

The total waiting time = 220 minutes

Average waiting time = $220 / 8 = 27.5$ minutes

In many cases we have to generate the random numbers from certain distribution. ***Inter-arrival time and service time follow the exponential distribution in most cases.*** To generate random numbers using the exponential distribution we use ***the inverse transform method.***

EXAMPLE 5.5

Let X have the exponential distribution with **mean β** , the distribution function is:

$$F(x) = \begin{cases} 1 - e^{-x/\beta} & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

so to find F^{-1} , we set $u = F(x)$ and solve for x to obtain

$$F^{-1}(u) = -\beta \ln (1 - u)$$

Monte Carlo Simulation of Queueing Problem

28

- Thus, to generate the desired random variate, we first generate a $U \sim U(0,1)$ and then let $X = -\beta \ln U$. [It is possible in this case to use U instead of $(1-U)$ since $(1-U)$ and U have the same $U(0,1)$ distribution. [This saves a subtraction.]

Example 5.6

Suppose in **example 5.4** the *inter-arrival time is exponentially distributed* with mean value $\beta_1 = 2$ and *service time is also exponentially distributed* with mean value $\beta_2 = 1.33$. From the simulation table for the arrival, service, and departure times of 8 customers **calculate the average waiting time for a customer.**

Solution

We get the random variates representing the *inter-arrival time* X_i and *service time* Y_i from the inverse transform equations:

$$X_i = -\beta_1 \ln U_1 \quad \text{and} \quad Y_i = -\beta_2 \ln U_2$$

Where the random numbers $0 \leq (\text{both } U_1 \text{ and } U_2) \leq 1$. Given a stream of **two digit** random numbers u_1, u_2 [from 00 to 99], we can get U_1, U_2 by dividing each random number in the stream by 100, then we calculate X_i, Y_i from the above equations, we get the following results:

Monte Carlo Simulation of Queueing Problem

29

$u_1 \rightarrow 20, 23, 86, 09, 92, 35, 38, 01$ & $u_2 = 21, 44, 27, 70, 73, 36, 59, 42$

For $u_1 = 20 \rightarrow U_1 = 20/100 = 0.20 \rightarrow X_1 = -2 \ln(0.20) = 3.22$

For $u_2 = 21 \rightarrow U_2 = 21/100 = 0.21 \rightarrow Y_1 = -1.33 \ln(0.21) = 2.08$

Similarly for all values of u_1 and u_2 as shown in the simulation table:

u_1	Inter-arrival time X_i	Arrival time	u_2	Service time Y_i	Departure time	Waiting time
20	3.22	3.22	21	2.08	5.30	0
23	2.94	6.16	44	1.09	7.25	0
86	0.30	6.46	27	1.75	9.0	0.79
09	4.82	11.28	70	0.48	11.76	0
92	0.17	11.45	73	0.42	12.18	0.31
35	2.10	13.55	36	1.36	14.91	0
38	1.94	15.49	59	0.70	16.19	0
01	9.21	24.70	42	1.16	25.86	0

The average waiting time = $(0.79 + 0.31) / 8 = 0.14$ and we can calculate other measures of performance.

Monte Carlo Simulation of Queueing Problem

30

Example 5.7

Simulate the operation on a *teller window* for *one hour*, given the following data:

$u_1 \rightarrow 25, 37, 91, 00, 61, 62, 80, 15, 23$

$u_2 \rightarrow 84, 01, 59, 40, 03, 29, 50, 77, 32$

Inter-arrival time (minutes)	5	6	7	8		Service time (minutes)	5	6	7
Probability	0.15	0.35	0.35	0.15		Probability	0.25	0.50	0.25
Random No.	00-14	15-49	50-84	85-99		Random No.	00-24	25-74	75-99

Monte Carlo Simulation of Queueing Problem

31

The simulation table for one hour will be:

Random number u_1	Inter-arrival time	Arrival time	Random number u_2	Service time	Departure time	Waiting time	Server idle time
25	6	6	84	7	13	0	6
37	6	12	01	5	18	1	0
91	8	20	59	6	26	0	2
00	5	25	40	6	32	1	0
61	7	32	03	5	37	0	0
62	7	39	29	6	45	0	2
80	7	46	50	6	52	0	1
15	6	52	77	7	59	0	0
23	6	58	32	6	65 stop	1	0
						Total = 3	Total = 11

When departure time reaches beyond 60 minutes we have to stop.

Average waiting time per customer = $3/9 = 1/3$ minutes

Percentage utilization of server = $[(65-11)/65] \times 100 = 83 \%$

Monte Carlo Simulation in Manufacturing

32

5.4 Monte Carlo Simulation in Manufacturing

We can have the generated random numbers by suitable software programs or by using different kinds of random number generators like linear congruential generator or mixed generator as shown in **Ex. 5.8**

Example 5.8

Given the units produced in a manufacturing facility and the number of vehicles available for transportation at a particular station with the corresponding probabilities as shown in the following table:

Units produced X_i	Probability $P(x = X_i)$	Random number intervals u_i	Number of vehicles available Y_i	Probability $P(y$ $= Y_i)$	Random number intervals u_i
500	0.06	00 – 05	5	0.16	00 – 15
550	0.14	06 – 19	6	0.36	16 – 51
600	0.20	20 – 39	7	0.20	52 – 71
650	0.40	40 – 79	8	0.16	72 – 87
700	0.20	80 – 99	9	0.12	88 – 99

CHAPTER 8

Monte Carlo Simulation

33

Using *mixed congruential generator* of 3 parameters **a**, **c**, and **m** to generate two digit random numbers between 00 and 99, given:

For $u_1 \rightarrow a = 21, c = 3, m = 100$, seed $X_0 = 20$

For $u_2 \rightarrow a = 41, c = 7, m = 100$, seed $X_0 = 1$, where:

$$X_i = a X_{i-1} + c \bmod (m)$$

For $u_1: X_0 = 20$

$$X_1 = a X_0 + c \bmod (m) = (21 \times 20) + 3 \bmod (100) = 423 \bmod (100) = 23$$

$$X_2 = (21 \times 23) + 3 \bmod (100) = 486 \bmod (100) = 86 \text{ and so on to } X_9$$

Similarly for $u_2: X_0 = 1$

$$X_1 = a X_0 + c \bmod (m) = (41 \times 1) + 7 \bmod (100) = 48 \bmod (100) = 48$$

$$X_2 = (41 \times 48) + 7 \bmod (100) = 1975 \bmod (100) = 75 \text{ and so on to } X_9$$

CHAPTER 8

Monte Carlo Simulation

34

Now we can have the simulation table for **10 days**.

Day	R.No. for units produced	Units produced	R.No. for vehicles availability	Vehicles available
1	20	600	01	5
2	23	600	48	6
3	86	700	75	8
4	09	550	82	8
5	92	700	69	7
6	35	600	36	6
7	38	600	83	8
8	01	500	10	5
9	24	600	17	6
10	07	550	04	5
		Total = 6000		Total = 64

Average number of units produced per day = $6000 / 10 = 600$ units/day

Average number of available vehicles per day = $64/10 = 6.4$ vehicle/day

Simulation Modeling and Analysis

Averill M. Law