Module 1.2 - Simulation Examples

Simulation Examples

- Three steps to carry out Simulation:

1. *Determine the characteristics of each of the inputs to the simulation.* Often modeled as probability distributions - continuous or discrete
2. *Construct a simulation table* (provides a systematic method for tracking system state over time)

   Example: there are *p* inputs, x_{ij}, j = 1,2,..,p, and one response, y_i, for each of the repetitions i = 1,2,..,n. Initialize the table by filling in the data for repetition 1.
3. *For each repetition i, generate a value for each of the p inputs, and evaluate the function, calculating a value of the response* y_i. The input values may be computed by sampling values from the distributions chosen in step 1. A response typically depends on the inputs and one or more previous responses

Simulation Table

Repetitions	Inputs						Response y_i
	x_{i1}	x_{i2}	..	x_{ij}	..	x_{ip}	
1							
2							
3							
:							
n							

Simulation of Queuing Systems

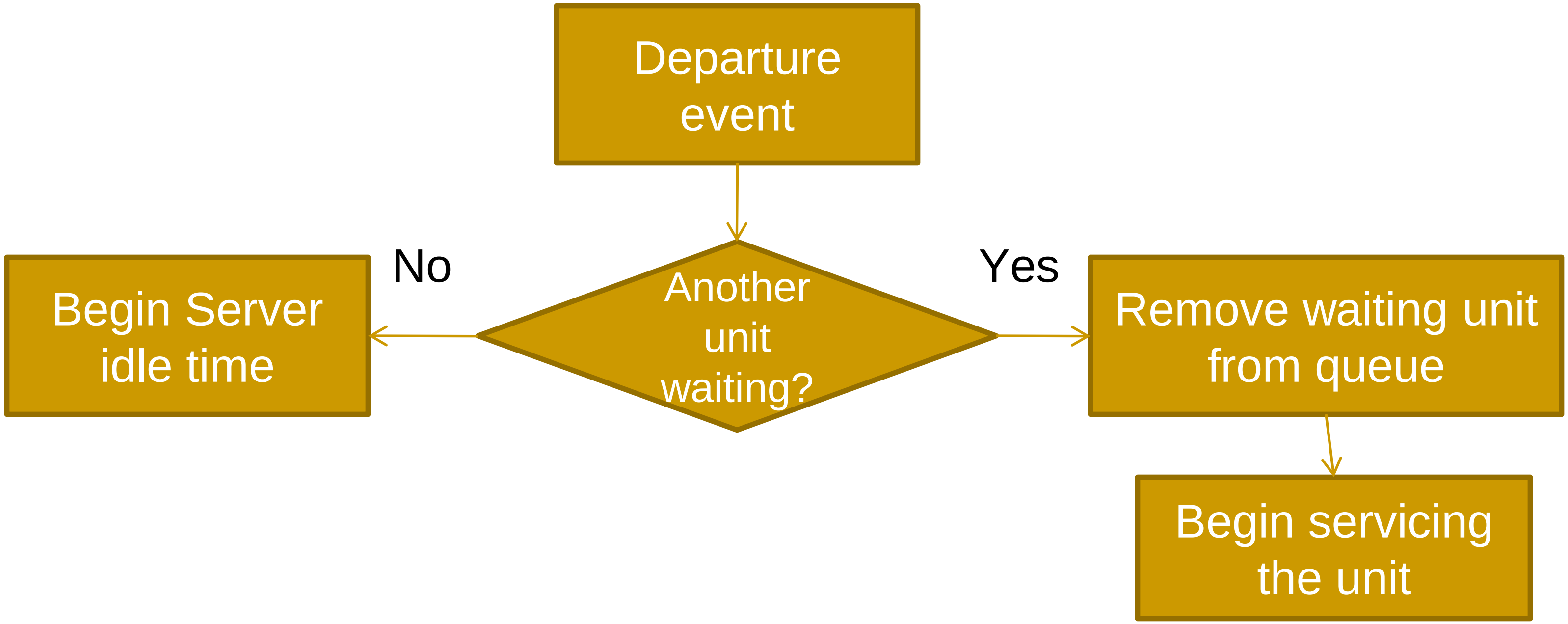
- A queuing system is described by its calling population, the nature of arrivals, the service mechanism, the system capacity, and the queuing discipline.
- In a single-channel queue, the calling population is infinite; that is, if a unit leaves the calling population and joins the waiting line or enters service, there is no change in the arrival rate of other units that could need service.
- Arrivals for service occur one at a time in a random fashion; once they join the waiting line, they are eventually served
- Service times are of some random length according to a probability distribution which does not change over time
- The system capacity has no limit, meaning that any number of units can wait in line
- Units are served in the order of their arrival by a single server or channel

Simulation of Queuing Systems

- Arrivals and services are defined by the distribution of the time between arrivals and distribution service times, resp.
- For any simple single- or multi-channel queue, the overall effective arrival rate must be less than the total service rate, or the waiting line will grow without bound (they are then termed ‘explosive’ or unstable)
- The *state* of the system is the number of units in the system and the status of the server (busy or idle)
- An *event* a set of circumstances that cause an instantaneous change in the state of the system
- In a *single-channel* queuing system, there are only two possible events that can affect the state of the system – entry of a unit (*arrival* event) and completion of service on a unit (*departure* event)
- *Queuing system* includes the server, unit being serviced, and units in queue; *simulation clock* is used to track simulated time

Simulation of Queuing Systems

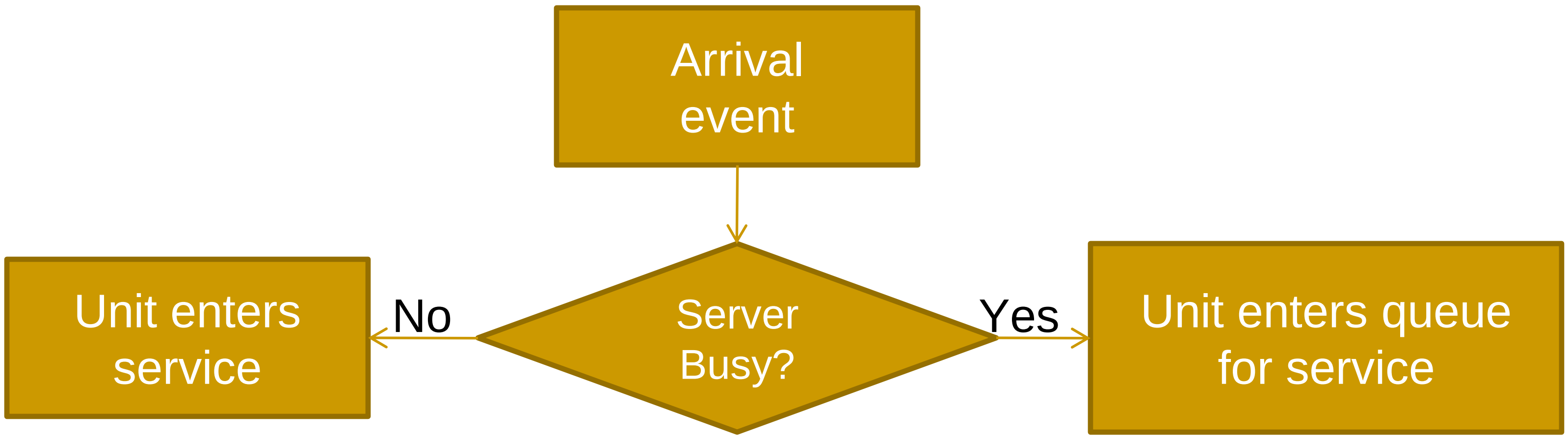
- *Service just completed* flow diagram



- If unit has just completed service, simulation proceeds as above: [Departure event occurs when unit completes service]
  - Server has only two possible states – busy or idle

Simulation of Queuing Systems

- *Unit entering system* flow diagram



- If unit has just entered the system, simulation proceeds as above: [Arrival event occurs when unit enters system]
  - Unit will find server either busy/idle; unit begins service immediately or enters queue for server
  - It is not possible for server to be idle while queue is nonempty

Simulation of Queuing Systems

- Potential unit actions upon arrival

		Queue status	
		Not empty	Empty
Server status	Busy	Enter queue	Enter queue
	Idle	Impossible	Enter service

- Server outcomes after completion of service

		Queue status	
		Not empty	Empty
Server outcomes	Busy	//////////////////	Impossible
	Idle	Impossible	//////////////////

Simulation of Queuing Systems

- Simulations of queuing systems require the maintenance of an *event list* for determining what happens next
- The event list tracks future times at which the different types of events occur
- Simulation clock times for arrivals and departures are computed in a simulation table customized for each problem
- In simulation, event usually occur at random times, the randomness imitating uncertainty in real life
- A statistical model of data is developed from data collected and analyzed or from subjective estimates and assumptions
- The randomness needed to imitate real life is made possible through the use of ‘random numbers’
- Random numbers are distributed uniformly and independently on the interval (0,1)

Simulation of Queuing Systems

- Random digits are uniformly distributed on the set {0,1,2,..,9} and can be used to form random numbers by selecting the proper number of digits for each random number and placing a decimal point to the left of the value selected
- Random numbers can also generated in simulation packages and in spreadsheets
- In a single-channel queuing simulation, interarrival times and service times are generated from the distributions of these random variables
- Example: the *interarrival* times are generated by rolling a die five times and recording them (x_1,x_2..x_5) and these five i.a.times are used to compute the arrival times of six customers
- The first customer is assumed to arrive at clock time 0 and this starts the clock in operation; the second customer arrives x_1 minutes after first customer, that is, at clock time x_1, and so on.

Simulation of Queuing Systems

Table 1: Interarrival and Clock Times

Customer	***Interarrival time***	***Arrival time on clock***
1	-	0
2	2	2
3	4	6
4	1	7
5	2	9
6	6	15

Simulation of Queuing Systems

- The second time that needs to be generated is the *service* time - generated at random and the only possible service times are 1,2,3, and 4 time units and all equally likely to occur
- The interarrival times and service times must be now meshed to simulate the single-channel queuing system. Ref table 4
- The first customer arrives at clock time 0 and immediately begins service, which requires two minutes; service is completed at clock time 2
- The second customer arrives at clock time 2 and is finished at clock time 3 (service time 1 minute)
- Third customer arrives at clock time 6 and finishes at c.t. 9
- Fourth customer arrives at 7, but service begins at 9 and finishes at clock time 11
- Fifth customer arrives at 15 and finishes at 19

Simulation of Queuing Systems

Table 2: Service Times

Customer	*Service time*
1	2
2	1
3	3
4	2
5	1
6	4

Simulation of Queuing Systems

Table 3: Simulation Table Emphasizing Clock Times

***A* Customer Number**	***B* Arrival time (Clock)**	***C* Time Service Begins (Clock)**	***D* Service Time (Duration)**	***E* Time Service Ends (Clock)**
1	0	0	2	2
2	2	2	1	3
3	6	6	3	9
4	7	9	2	11
5	9	11	1	12
6	15	15	4	19

Simulation of Queuing Systems

Table 4: Chronological Ordering of Events

Event Type	*Customer number*	*Clock Time*
Arrival	1	0
Departure	1	2
Arrival	2	2
Departure	2	3
Arrival	3	6
Arrival	4	7
Departure	3	9
Arrival	5	9
Departure	4	11
Departure	5	12
Arrival	6	15
Departure	6	19

Simulation of Queuing Systems

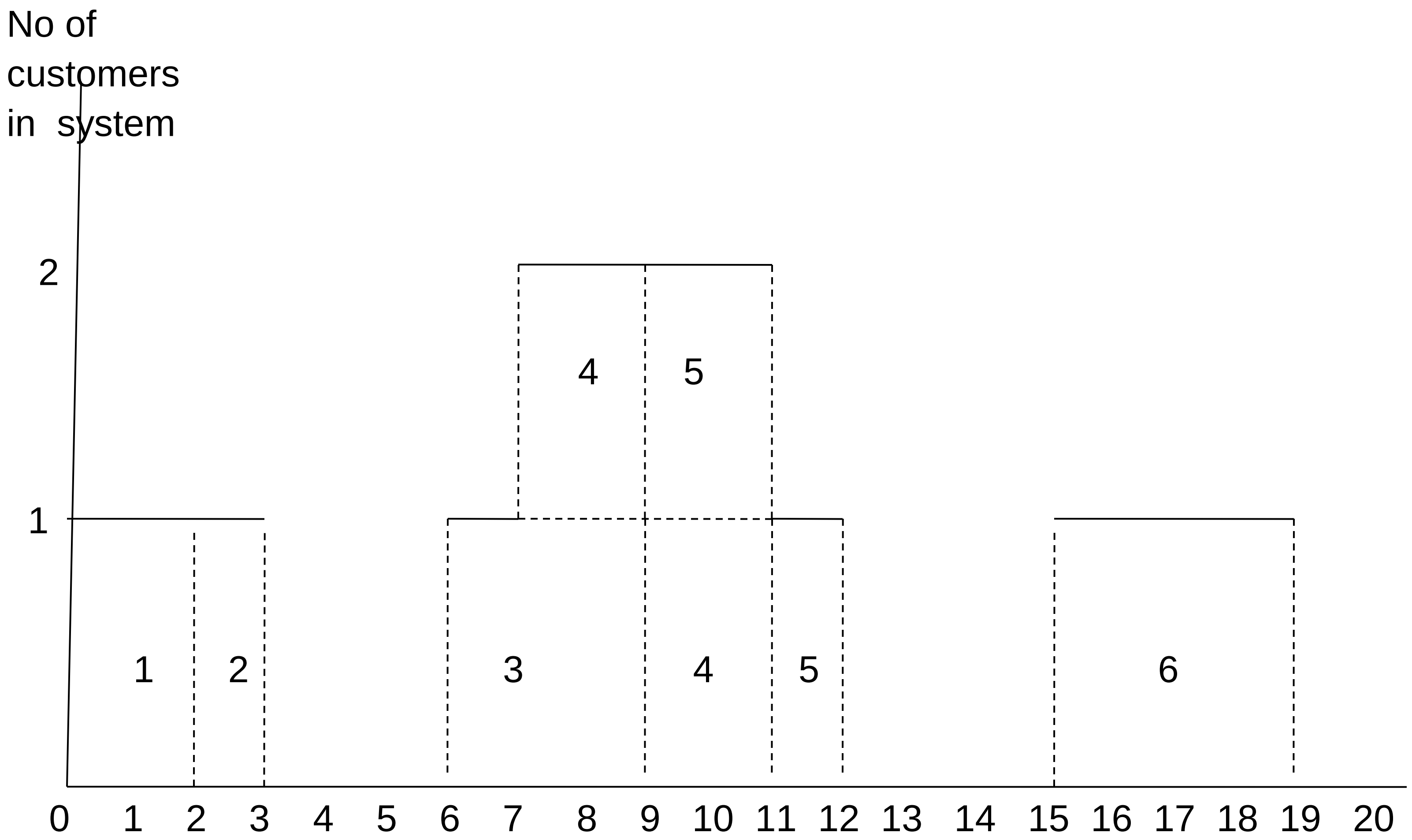


Figure 1: Clock Time

Example 1: Single-Channel Queue

- A small grocery store has one checkout counter
- Customers arrive at this checkout counter at random time fro 1 to 8 minutes apart
- Each possible value of interarrival time has the same probability of occurrence. Ref. table 5
- Service time varies from 1 to 6 minutes, with probabilities shown in table 6
- Problem is to analyze the system by simulating the arrival and service of 100 customers
- A set of uniformly distributed random numbers is needed to generate the arrivals at the checkout counter
- These random numbers have the following properties:
  - It is uniformly distributed between 0 and 1
  - Successive random numbers are independent

Example 1: Single-Channel Queue

Table 5: Distribution of time Between Arrivals

Time between Arrivals (minutes)	***Probability***	***Cumulative Probability***	***Random digit Assignment***
1	0.125	0.125	001 - 125
2	0.125	0.250	126 - 250
3	0.125	0.375	251 - 375
4	0.125	0.500	376 - 500
5	0.125	0.625	501 - 625
6	0.125	0.750	626 - 750
7	0.125	0.875	751 - 875
8	0.125	1.000	876 - 000

Example 1: Single-Channel Queue

Table 6: Service-Time Distribution

Service Time (minutes)	*Probability*	*Cumulative Probability*	*Random digit Assignment*
1	0.10	0.10	01 - 10
2	0.20	0.30	11 - 30
3	0.30	0.60	31 - 60
4	0.25	0.85	61 - 85
5	0.10	0.95	86 - 95
6	0.05	1.00	96 - 00

Example 1: Single-Channel Queue

- We have to list 99 random numbers (say from table A.1) to generate the time between arrivals, accurate to 3 decimal places (since the probabilities in table 5 are accurate to 3 significant digits) for the 100 customers
- Rightmost two columns of table 5 and 6 are used to generate random arrivals and random service times
- The rightmost column contains random digit assignment; 001 – 0125 represents 1 minute and 126 – 250 represents 2 minutes, and so on, for up to 8 minutes interarrival times
- The time-between-arrival determination is shown in table 7
- The service times generated are shown in table 8
- The first customer's service time is 4 minutes, because random digits 84 fall in bracket 61-85
- The simulation table for the single-channel queue is shown in table 9

Example 1: Single-Channel Queue

Table 7: Time-Between-Arrival Determination

Customer Number	*Random Digits*	*Time between Arrivals (Minutes)*	*Customer Number*	*Random Digits*	*Time between Arrivals (Minutes)*
1	-	-	11	413	4
2	064	1	12	462	4
3	112	1	13	843	7
4	678	6	14	738	6
5	289	3	15	359	3
6	871	7	16	888	8
7	583	5	17	902	8
8	139	2	18	212	2
9	423	4	:	:	:
10	039	1	100	538	5

Example 1: Single-Channel Queue

Table 8: Service Times Generated

Customer	*Random Digits*	*Service Time (Minutes)*	*Customer*	*Random Digits*	*Service Time (Minutes)*
1	84	4	11	94	5
2	18	2	12	32	3
3	87	5	13	79	4
4	81	4	14	92	5
5	06	1	15	46	3
6	91	5	16	21	2
7	79	4	17	73	4
8	09	1	18	55	3
9	64	4	:	:	:
10	38	3	100	26	2

Example 1: Single-Channel Queue

- The first step to construct table 9 is to initialize table by filling cells for the first customer
- First customer is assumed to arrive at time 0; service begins immediately and finishes at time 4; customer was in the system for 4 minutes
- Subsequent rows in table are based on the random numbers for interarrival time, service time, and the completion time of the previous customer
- Example: The second customer arrives at time 1; service could not begin until time 4; customer waited in queue for 3 minutes; and was in the system for 5 minutes
- The rightmost two columns were added to collect statistical measures of performance
- Totals are calculated to compute summary statistics

Simulation Table for Single-Channel Queuing Problem

Customer	*Interarrival time (minutes)*	*Arrival Time*	*Service Time (minutes)*	*Time Service begins*	*Waiting time in Queue*	*Time Service ends*	*Time Customer in system*	*Idle Time of Server*

Example 1: Single-Channel Queue

Table 9: Simulation Table for Single-Channel Queuing Problem

Customer	*Interarrival time (minutes)*	*Arrival Time*	*Service Time (minutes)*	*Time Service begins*	*Waiting time in Queue*	*Time Service ends*	*Time Customer in system*	*Idle Time of Server*
1	-	0	4	0	0	4	4	-
2	1	1	2	4	3	6	5	0
3	1	2	5	6	4	11	9	0
4	6	8	4	11	3	15	7	0
5	3	11	1	15	4	16	5	0
6	7	18	5	18	0	23	5	2
7	5	23	4	23	0	27	4	0
8	2	25	1	27	2	28	3	0
9	4	29	4	29	0	33	4	1
10	1	30	3	33	3	36	6	0
11	4	34	5	36	2	41	7	0
12	4	38	3	41	3	44	6	0
13	7	45	4	45	0	49	4	1
14	6	51	5	51	0	56	5	2
15	3	54	3	56	2	59	5	0
16	8	62	2	62	0	64	2	3
17	8	70	4	70	0	74	4	6
18	2	72	3	74	2	77	5	0
19	7	79	1	79	0	80	1	2
20	4	83	2	83	0	85	2	3
:	:	:	:	:	:	:	:	:
100	5	415	2	416	1	418	3	0
Total	415		317		174		491	101

Example 1: Single-Channel Queue

- Some of the findings from the simulation table 9 are:

  [All time figures are in minutes]

  - Average waiting time = $\frac{\text{total time customers wait in queue}}{\text{total number of customers}} = \frac{174}{100} = 1.74$
  - Probability that customer has to wait in queue

    $= \frac{\text{number of customers who wait}}{\text{total number of customers}} = \frac{46}{100} = 0.46$
  - Proportion of idle time of server = $\frac{\text{total idle time of server}}{\text{total run time of simulation}} = \frac{101}{418} = 0.24$
  - Average service time = $\frac{\text{total service time}}{\text{total no. of customers}} = \frac{317}{100} = 3.17$

  [compare to expected service time given by equation

  $E(s) = \Sigma sp(s), \quad s = 0 \text{ to } \infty$

  $= 1(0.1) + 2(0.2) + 3(0.3) + 4(0.25) + 5(0.1) + 6(0.05) = 3.2$]

Example 1: Single-Channel Queue

- Some of the findings from the simulation table 9 are:
  - Average time between arrivals = <u>sum of all times between arrivals</u>

    number of arrivals - 1

    = 415 / 99 = 4.19

[compare the above time with expected time between arrivals by finding mean of discrete uniform distribution whose endpoints are a = 1 & b = 8

Mean E(A) = (a + b) / 2 = (1 + 8)/2 = 4.5]

  - Average waiting time of those who wait = <u>total time customers wait in queue</u>

    total number of customers that wait

    = 174 / 54 = 3.22

  - Average time customer spends in system = <u>total time customers spend in system</u>

    total number of customers

    = 491 / 100 = 4.91

[another way to find the same is to add average time customers spends waiting in queue and average time customers spends in service

= 1.74 + 3.17 = 4.91]

Example 1: Single-Channel Queue

- Some of the conclusions from the simulation table 9 are:
  - A longer simulation would increase the accuracy of the findings
  - About half of the customers had to wait; however, average waiting time not excessive
  - Server does not have undue amount of idle time

Example 2: The Able-Baker Call Center Problem

- This example illustrates the simulation problem when there is more than one service channel
- Consider a computer technical support center where personnel takes calls and provide service
- The time between calls range from 1 to 4 minutes. Ref table 10
- There are two technical support people – Abel and Baker
- Able is more experienced and can provide faster service than Baker
- Distribution of their service times are shown in table 11 and 12
- Simple rule: Able gets the call if both of them are idle
- *Problem: To find out how well the arrangement is working*
- To estimate the system measures of performance, a simulation of the first 100 callers is made

Example 2: The Able-Baker Call Center Problem

Table 10: Interarrival Distribution of Calls for Technical Support

Time between Arrivals (minutes)	***Probability***	***Cumulative Probability***	***Random digit Assignment***
1	0.25	0.25	01 - 25
2	0.40	0.65	26 - 65
3	0.20	0.85	66 - 85
4	0.15	1.00	86 - 00

Example 2: The Able-Baker Call Center Problem

Table 11: Service Distribution of Able

Service Time (minutes)	*Probability*	*Cumulative Probability*	*Random digit Assignment*
2	0.30	0.30	01 - 30
3	0.28	0.58	31 - 58
4	0.25	0.83	59 - 83
5	0.17	1.00	84 - 00

Table 12: Service Distribution of Baker

Service Time (minutes)	*Probability*	*Cumulative Probability*	*Random digit Assignment*
3	0.35	0.35	01 - 35
4	0.25	0.60	36 - 60
5	0.20	0.80	61 - 80
6	0.20	1.00	81 - 00

Example 2: The Able-Baker Call Center Problem

- The simulation proceeds in accordance with the following steps
  - *<u>Step 1:</u>* For caller *k*, generate an interarrival time A_k; Add it to previous arrival time T_{k-1} to get the arrival time of Caller *k* as
    $T_k = T_{k-1} + A_k$
  - *<u>Step 2:</u>* If Able is idle, Caller *k* begins service with Able at the current time T_{now}
  - Able's service completion time, $T_{fin,A}$ is given by $T_{fin,A} = T_{now} + T_{svc,A}$, where $T_{svc,A}$ is the service time generated from Able's Service Time Distribution
  - Caller *k's* time in system, T_{sys}, is given by $T_{sys} = T_{fin,A} – T_k$
  - Because Able was idle, Caller *k's* delay T_{wait}, is given by $T_{wait} = 0$
  - If Able is busy, but Baker is idle, Caller *k* begins service with Baker at the current time T_{now}; Baker's service completion time, *T* is given by $T_{fin,B} = T_{now} + T_{svc,B}$ where $T_{svc,B}$ is the service time generated from Baker's Service Time Distribution
  - Caller *k's* time in system, T_{sys}, is given by $T_{sys} = T_{fin,B} – T_k$
  - Because Baker was idle, Caller *k's* delay T_{wait}, is given by $T_{wait} = 0$

Example 2: The Able-Baker Call Center Problem

- *Step 3:* If Able and Baker are both busy, then calculate the time at which the first one becomes available, as follows:

$$T_{beg} = min(T_{fin,A}\ ,\ T_{fin,B})$$

- Caller *k* begins service at T_{beg}; When service for Caller *k* begins, set $T_{now} = T_{beg}$
- Then compute $T_{fin,A}$ or $T_{fin,B}$ as in Step 2
- Caller *k's* time in system, T_{sys}, is given by

  $T_{sys} = T_{fin,A} – T_k$ or $T_{sys} = T_{fin,B} – T_k$, as appropriate
- Ref Table 13 - Caller 1 arrives at clock time 0 to get simulation started; Able is idle, so Caller 1 begins service with Able at clock time 0.
- The service time, 2 minutes, is generated from information given in table 11 by following the procedure in Example 1. Thus, Caller 1 completes service at clock time 2 minutes and was not delayed
- An interarrival time of 2 minutes is generated from table 11 by following earlier procedure. So, the arrival of Caller 2 is at clock time 2 minutes

Example 2: The Able-Baker Call Center Problem

- Able is idle at the time, having just completed service on Caller 1, so Caller 2 is served by Able
- Caller 4 is serviced by Able from clock time 8 minutes to clock time 12 minutes
- Caller 5 arrives at clock time 9 minutes; Because Able is busy with caller 4 at that time, and baker is available, Baker services Caller 5, completing service at clock time 12 minutes

Example 2: The Able-Baker Call Center Problem

Table 13: Simulation Table for Call-Center Example

Caller No.	*Interarrival time (minutes)*	*Arrival Time (clock)*	*When Able Avail (clock)*	*When Baker Avail (clock)*	*Server chosen*	*Service Time (minutes)*	*Time Service begins (clock)*	*Able's Svc Comp Time (clock)*	*Baker's Svc Comp time (clock)*	*Caller delay (minutes)*	*Time in Sys (minutes)*
1	-	0	0	0	Able	2	0	2		0	2
2	2	2	2	0	Able	2	2	4		0	2
3	4	6	4	0	Able	2	6	8		0	2
4	2	8	8	0	Able	4	8	12		0	4
5	1	9	12	0	Baker	3	9		12	0	3
:	:	:	:	:	:	:	:	:	:	:	:
100	1	219	221	219	Baker	4	219		223	<u>0</u>	<u>4</u>
Total										211	564

Notes:

- Total customer delay 211 minutes or about 2.1 minutes per caller
- Total time in system 564 minutes or 5.6 minutes per caller
- One server cannot handle all callers, and three servers would be more than necessary; Adding addl. server reduces waiting time; but cost of waiting would have to be quite high to justify an addl. server.

Simulation of Inventory Systems

- Inventory systems are an important class of simulation problems
- The inventory system in fig.2 has a periodic review of length N, at which time the inventory level is checked
- An order is made to bring the inventory up to the level M
- At the end of 1st review period, an order quantity, Q1, is placed
- The lead time is zero for this inventory system
- Demands are not usually known with certainty, so the order quantities are probabilistic
- Demand is shown as being uniform over the time period in fig.2
- In reality, demands are not usually uniform and do fluctuate over time and most demands all occur at beginning of the cycle and that lead time is random of some positive length
- Note: amount in inventory drops below zero in second cycle, indicating a shortage. These units are backordered and when order arrives, demand for backordered items is satisfied first

Simulation of Inventory Systems

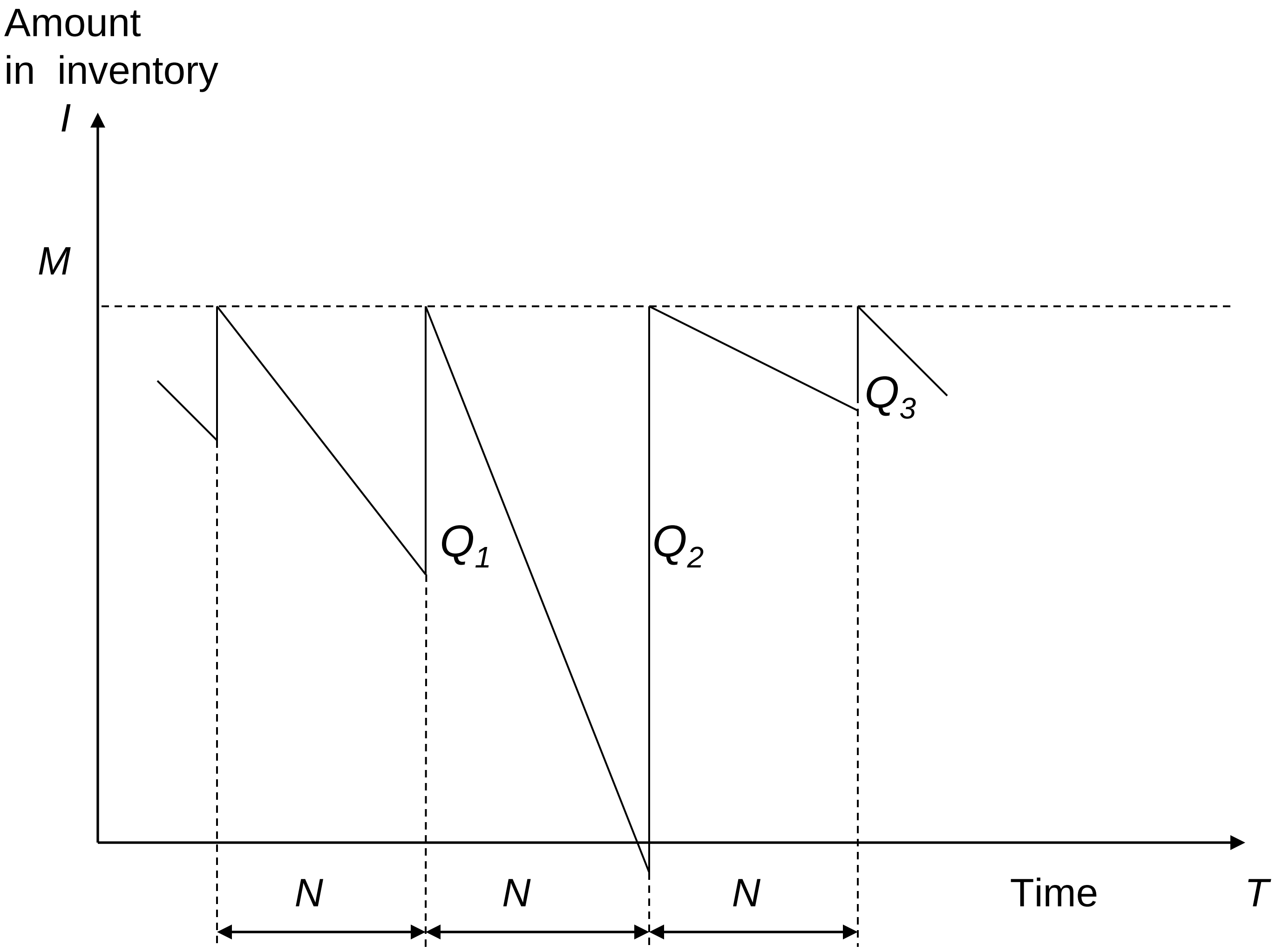


Figure 2: Probabilistic order-level inventory systems

Simulation of Inventory Systems

- Carrying stock in inventory has an associated cost attributed to the interest paid on the funds borrowed to buy the items; Other costs are carrying or holding cost, renting of storage space, security guards, etc
- Alternative to holding high inventory is to make frequent reviews and, consequently, more purchases or replenishments; this has associated costs also – the ordering cost, goodwill costs when customers get angry when shortages exist, etc
- The above two costs should be traded off to minimize total cost of an inventory system
- The total cost of an inventory system is the measure of performance; this can be affected by policy alternatives
- For example, decision makers can control the maximum inventory level, M, and the length of the cycle, N,

Example 3: The News Dealer's Problem

- This inventory problem concerns the purchase and sale of newspapers
- The newsstand buys the papers for 33 paise each and sells them for 50 paise each; Newspaper not sold at the end of the day are sold as scrap for 5 paise each
- Newspaper can be purchased in bundles of 10; thus the newsstand can buy 50, 60, and so on
- There are three types of newsdays: 'good', 'fair' and 'poor' having probabilities 0.35, 0.45 and 0.20, respectively
- The distribution of newspapers demanded on each of these days is given in table 14
- *Problem: Compute the optimal number of papers the newsstand should purchase*
- This is accomplished by simulating demands for 20 days and recording profits from sales each day

Example 3: The News Dealer's Problem

Table 14: Distribution of Newspapers Demanded Per Day

Demand	***Demand Probability Distribution***		
	Good	***Fair***	***Poor***
40	0.03	0.10	0.44
50	0.05	0.18	0.22
60	0.15	0.40	0.16
70	0.20	0.20	0.12
80	0.35	0.08	0.06
90	0.15	0.04	0.00
100	0.07	0.00	0.00

Example 3: The News Dealer's Problem

- The profits are given by the following relationship:

profit	=	*revenue from sales*	–	*cost of newspapers*	–	*lost profit from excess demand*	+	*salvage from sale of scrap papers*

- The revenue from sales is 50 paise for each paper sold; the cost of newspapers is 33 paise for each paper purchased; the lost profit from excess demand is 17 paise for each paper demanded that could not be provided; salvage value of scrap papers is 5 paise each
- Tables 15 and 16 provide the random digit assignments for the types of newsdays and the demands for those newsdays
- To solve this problem by simulation requires setting a policy of buying a certain number of papers each day, then simulating the demands for papers over the 20-day time period to determine the total profit; The policy (number of newspapers purchased) is changed to other values and the simulation repeated until the best value is found

Example 3: The News Dealer's Problem

Table 15: Random digit Assignment for type of Newsday

Type of Newsday	***Probability***	***Cumulative Probability***	***Random digit Assignment***
Good	0.35	0.35	01 - 35
Fair	0.45	0.80	36 - 80
Poor	0.20	1.00	81 - 00

Example 3: The News Dealer's Problem

Table 16: Random Digit Assignments for Newspapers Demanded

Demand	*Cumulative Distribution*			*Random Digit Assignments*		
	Good	*Fair*	*Poor*	*Good*	*Fair*	*Poor*
40	0.03	0.10	0.44	01 - 03	01 - 10	01 - 44
50	0.08	0.28	0.66	04 - 08	11 - 28	45 - 66
60	0.23	0.68	0.82	09 - 23	29 - 68	67 - 82
70	0.43	0.88	0.94	24 - 43	69 - 88	83 - 94
80	0.78	0.96	1.00	44 - 78	89- 96	95 - 00
90	0.93	1.00	1.00	79 - 93	97 - 00	
100	1.00	1.00	1.00	94 - 00		

Example 3: The News Dealer's Problem

Table 17: Simulation Table for Purchase of 70 Newspapers

Day	Random Digits for Type of Newsday	Type of Newsday	Random digits for Demand	Demand	Revenue from Sales (Rupees)	Lost Profit from Excess Demand	Salvage from Sale of Scrap	Daily profit
1	58	Fair	93	80	35	1.70	-	10.20
2	17	Good	63	80	35	1.70	-	10.20
3	21	Good	31	70	35	-	-	11.90
4	45	Fair	19	50	25	-	1.00	2.90
5	43	Fair	91	80	35	1.70	-	10.20
6	36	Fair	75	70	35	-	-	11.90
7	27	Good	84	90	35	3.40	-	8.50
8	73	Fair	37	60	30	-	0.50	7.40
9	86	Poor	23	40	20	-	1.50	-1.60
10	19	Good	02	40	20	-	1.50	-1.60
11	93	Poor	53	50	25	-	1.00	3.90
12	45	Fair	96	80	35	1.70	-	10.20
13	47	Fair	33	60	30	-	0.50	7.40
14	30	Good	86	90	35	3.40	-	8.50
15	12	Good	16	60	30	-	0.50	7.40
16	41	Fair	07	40	20	-	1.50	-1.60
17	65	Fair	64	60	30	-	0.50	7.40
18	57	Fair	94	80	35	1.70	-	10.20
19	18	Good	55	80	35	1.70	-	10.20
20	98	Poor	13	40	20	-	1.50	-1.60
					600	17.00	10.00	131.00

Example 3: The News Dealer's Problem

- The Simulation table for the decision purchase 70 newspapers is shown in Table 17
- On day 1, the demand is for 80 newspapers, but only 70 newspapers are available; revenue from sales is Rs.35.00
- The lost profit for the excess demand for 10 newspapers is Rs.1.70; The profit for the first day is computed as follows:

  profit = 35.00 – 23.10 – 1.70 + 0 = Rs.10.20

- On the 4th day, the demand is less than the supply. The revenue from sales of 50 newspapers is Rs.25.00
- Twenty newspapers are sold for scrap at Rs.0.05 each yielding Rs.1.00; The daily profit is determined as follows:

  profit = 25.00 – 23.10 – 0 + 1.00 = Rs.2.90

Example 3: The News Dealer's Problem

- The profit for 20-day period is the sum of daily profits, Rs.131.00
- Also computed from total for 20 days of simulation

  Total profit = 600 - 462 + 17 - 10 = Rs.131.00

  [462 = cost of newspaper = 20 x 0.33 x 70]

Example 4: Order-Up-To-Level Inventory System

- Consider a company which sells refrigerators; System used to maintain inventory is to review the situation after a fixed number of days (N) and make a decision about what is to be done
- Policy is to order up to a level (M), using following relationship:
Order quantity = (Order-up-to-level) - (Ending inventory) + (Shortage quantity)
- Suppose, M is 11 and ending inventory is 3 and review period N is 5 days
- Thus, on the 5th day of cycle, 8 refrigerators will be ordered from the supplier; If there is shortage of 2, then 13 will be ordered
- Note: there cannot be both ending inventory and shortage qty at the same time
- If there were shortage, then the required shortage will be provided to customers first when the order arrives. This is called ‘making up backorders’

Example 4: Order-Up-To-Level Inventory System

- *Lost sales* case occurs when customer demand is lost if inventory is not available
- The number of refrigerators ordered each day is randomly distributed as shown in table 18
- Another source of randomness is the number of days after the order is placed with the supplier before arrival or *lead time*
- The distribution of lead time is shown in table 19
- Assume that orders are placed at the end of the day
- If lead time is zero, the order from supplier will arrive next morning, and the refrigerators will be available for distribution the next day
- If the lead time is one day, the order from the supplier arrives the second morning after, and will be available for distribution that second day

Example 4: Order-Up-To-Level Inventory System

- Simulation has been started with the inventory level at 3 refrigerators and an order for 8 refrigerators to arrive in 2 days time; the simulation is shown in table 20
- Order for 8 refrigerators is available on the morning of the third day of the first cycle, raising the inventory level from zero refrigerators to 8 refrigerators
- Demand during the remainder of the first cycle reduced the ending inventory level to 2 refrigerators on the fifth day
- Thus, an order for 9 refrigerators was placed; the lead time for this order was 2 days
- The order for 9 refrigerators was added to inventory on the morning of day 3 of cycle 2
- Note: beginning inventory of fifth day of fourth cycle was 2; an order for 3 refrigerators on that day led to a shortage condition; one refrigerator was backordered on that day;

Example 4: Order-Up-To-Level Inventory System

- 12 refrigerators were ordered (11 + 1), and they had a lead time of one day
- On the next day, the demand was two, so additional shortages resulted
- At the beginning of the next day, the order had arrived; Three refrigerators were used to make up the backorders and there was a demand for one refrigerator, so ending inventory was 8
- From five cycles of simulation, the average ending inventory is approximately 2.72 (68/25) units
- On 5 of 25 days, a shortage condition existed

Example 4: Order-Up-To-Level Inventory System

Table 18: Random Digit Assignment for Daily Demand

Demand	***Probability***	***Cumulative Probability***	***Random digit Assignment***
0	0.10	0.10	01 - 10
1	0.25	0.35	11 - 35
2	0.35	0.70	36 - 70
3	0.21	0.91	71 - 91
4	0.09	1.00	92 - 00

Example 4: Order-Up-To-Level Inventory System

Table 19: Random Digit Assignment for Lead Time

Lead Time (Days)	***Probability***	***Cumulative Probability***	***Random digit Assignment***
1	0.6	0.6	1 - 6
2	0.3	0.9	7 - 9
3	0.1	1.0	0

Table 20: Simulation Table for [M,N] Inventory System

Day	Cycle	Day within Cycle	Beginning Inventory	Random Digits for Demand	Demand	Ending Inventory	Shortage Quantity	Order Quantity	Random Digits for Demand	Lead Time (days)	Days until Order Arrives
1	1	1	3	26	1	2	0	-	-	-	1
2	1	2	2	68	2	0	0	-	-	-	-
3	1	3	8	33	1	7	0	-	-	-	-
4	1	4	7	39	2	5	0	-	-	-	-
5	1	5	5	86	3	2	0	9	8	2	2
6	2	1	2	18	1	1	0	-	-	-	1
7	2	2	1	64	2	0	1	-	-	-	-
8	2	3	9	79	3	5	0	-	-	-	-
9	2	4	5	55	2	3	0	-	-	-	-
10	2	5	3	74	3	0	0	11	7	2	2
11	3	1	0	21	1	0	1	-	-	-	1
12	3	2	0	43	2	0	3	-	-	-	-
13	3	3	11	49	2	6	0	-	-	-	-
14	3	4	6	90	3	3	0	-	-	-	-
15	3	5	3	35	1	2	0	9	2	1	1
16	4	1	2	08	0	2	0	-	-	-	-
17	4	2	11	98	4	7	0	-	-	-	-
18	4	3	7	61	2	5	0	-	-	-	-
19	4	4	5	85	3	2	0	-	-	-	-
20	4	5	2	81	3	0	1	12	3	1	1
21	5	1	0	53	2	0	3	-	-	-	-
22	5	2	12	15	1	8	0	-	-	-	-
23	5	3	8	94	4	4	0	-	-	-	-
24	5	4	4	19	1	3	0	-	-	-	-
25	5	5	3	44	2	1	0	10	1	1	1
Total						68	9				

Example 5: A Reliability Problem

- A milling machine has 3 different bearings that fail in service.
- The distribution of the life of each bearing is identical, as shown in Table 21
- When a bearing fails, the mill stops, a repairperson is called, and a new bearing is installed
- The delay time of the repairperson's arriving at the milling machine is also a random variable having distribution given in Table 22
- Downtime for the mill is estimated at Rs.10 per minute
- The direct on-site cost of repairperson is Rs.30 per hour; It takes 20 minutes to change one bearing, 30 minutes to change 2 bearings, and 40 minutes to change 3 bearings; cost per bearing Rs.32
- *A proposal has been made to replace all 3 bearings whenever a bearing fails*
- Management needs an *evaluation of the proposal*; the total cost per 10,000 bearing-hours will be used as the measure of performance

Example 5: A Reliability Problem

Table 21: Bearing-Life Distribution

Bearing Life (Hours)	*Probability*	*Cumulative Probability*	*Random digit Assignment*
1000	0.10	0.10	01 - 10
1100	0.13	0.23	11 - 23
1200	0.25	0.48	24 - 48
1300	0.13	0.61	49 - 61
1400	0.09	0.70	62 - 70
1500	0.12	0.82	71 - 82
1600	0.02	0.84	83 - 84
1700	0.06	0.90	85 - 90
1800	0.05	0.95	91 - 95
1900	0.05	1.00	96 - 00

Example 5: A Reliability Problem

Table 22: Delay-Time Distribution

Delay Time (Minutes)	***Probability***	***Cumulative Probability***	***Random digit Assignment***
5	0.6	0.6	1 - 6
10	0.3	0.9	7 - 9
15	0.1	1.0	0

Example 5: A Reliability Problem

Table 23: Bearing Replacement under Current Method

	Bearing 1				Bearing 2				Bearing 3			
	Random Digits	***Life (Hours)***	***Random Digits***	***Delay (Mins)***	***Random Digits***	***Life (Hours)***	***Random Digits***	***Delay (Mins)***	***Random Digits***	***Life (Hours)***	***Random Digits***	***Delay (Mins)***
1	67	1400	7	10	71	1500	8	10	18	1100	6	5
2	55	1300	3	5	21	1100	3	5	17	1100	2	5
3	98	1900	1	5	79	1500	3	5	65	1400	2	5
4	76	1500	6	5	88	1700	1	5	03	1000	9	10
5	53	1300	4	5	93	1800	0	15	54	1300	8	10
6	69	1400	8	10	77	1500	6	5	17	1100	3	5
7	80	1500	5	5	08	1000	9	10	19	1100	6	5
8	93	1800	7	10	21	1100	8	10	09	1000	7	10
9	35	1200	0	15	13	1100	3	5	61	1300	1	5
10	02	1000	5	5	03	1100	2	5	84	1600	0	15
11	99	1900	9	10	14	1000	1	5	11	1100	5	5
12	65	1400	4	5	5	1000	0	15	25	1200	2	5
13	53	1300	7	10	29	1200	2	5	86	1700	8	10
14	87	1700	1	5	07	1000	4	5	65	1400	3	5
15	90	1700	2	5	20	1100	3	5	44	1200	4	5
Total		22300		110		18700		110		18600		105

Example 5: A Reliability Problem

- Table 23 represents a simulation of 15 bearing changes under the current method of operation
- It is assumed that the times when more than one bearing failing are never exactly the same and thus no more than one bearing is changed at any breakdown
- The cost of the current system is estimated as follows:

Cost of bearing = 45 bearings x Rs.32/bearing = Rs.1440

Cost of delay time = (110 + 110 + 105) minutes x Rs.10/minute = Rs.3250

Cost of downtime during repair = 45 bearings x 20 minutes/bearing x Rs.10/minute = Rs.9000

Cost of repairpersons = 45 bearings x 20 minutes/bearing x Rs.30/60 minutes = Rs.450

- Total cost = 1440 + 3250 + 9000 + 450 = Rs.14,140
- Total life of bearings = 22300 + 18700 + 18600 = 59600 hours
- Total cost per 10000 bearing-hours = 14140/5.96 = Rs.2372

Example 5: A Reliability Problem

- Table 24 is a simulation of the proposed method
- For the first set of bearings, the earliest failure is at 1000 hours
- All 3 bearings are replaced at that time even though the remaining bearings had more life in them
- The cost of the *proposed system* is estimated as follows:

Cost of bearings = 45 bearings x Rs.32/bearing = Rs.1440

Cost of delay time = 110 minutes x Rs.10/minute = Rs.1100

Cost of downtime = 15 sets x 40 minutes/set x Rs.10/minute
during repairs = Rs.6000

Cost of repairpersons = 15 sets x 40 minutes/set x Rs.30/60 mins
= Rs.300

Total cost = 1440 + 1100 + 6000 + 300 = Rs.8840

Total life of bearings = 17000 x 3 = 51000 hours

Total cost per 10000 bearing-hours = 8840/5.1 = Rs.1733

Example 5: A Reliability Problem

- The new policy generates a savings of Rs.634 per 10000 hours of bearing-life; If machine runs continuously, the savings are about Rs.556 per year
- In the examples, user can change the distribution of bearing life (making sure cumulative probability is exactly 1.0).
- The distribution of delay time can be changed
- Also, the parameters of the problem can be changed (bearing cost per unit, etc)
- The number of trails can be varied from 1 to 400
- Endpoints of the bins can be changed for observing the frequency of total cost for 10000 hours of bearing life

Example 6: Random Normal Numbers

- Consider a bomber attempting to destroy an ammunition depot
- If a bomb falls anywhere on target, a hit is scored; otherwise, the bomb is a miss
- The bomber flies in the horizontal direction and carries 10 bombs
- The aiming point is (0,0)
- The point of impact is assumed to be normally distributed around the aiming point with a standard deviation of 400 meters in the direction of flight and 200 meters in the perpendicular direction
- *Problem: Simulate the operation and make statements about the number of bombs on target*
- Standard normal variate, Z, having mean 0 and standard deviation 1, is distributed as

$$Z = (X - \mu) / \sigma$$

where X is a normal variate, μ is the mean distribution of X, and σ is the standard deviation of X

Example 6: Random Normal Numbers

- Then,

$$X = Z\sigma_x$$
$$Y = Z\sigma_y$$

  where (X, Y) are the simulated coordinates of the bomb after it has fallen.
- With $\sigma_x = 400$ and $\sigma_y = 200$, we have

$$X = 400\, Z_i$$
$$Y = 200\, Z_j$$

  The i and j subscripts have been added to indicate that the values of Z should be different; values of Z are random normal numbers
- Table 25 shows the result of a simulated run; the results of a simulated run; random normal numbers are shown to 4 decimal place accuracy
- RNN_x stands for 'Random Normal Number to compute x coordinate' and corresponds to Z_i; We multiply this number by 400 to get X value and similarly we multiply RNN_y (corresponds to Z_j) by 200 to get the Y value

Example 6: Random Normal Numbers

Table 25: Simulated Bombing Run

Bomb	***RNN_X***	***X Coordinate (400 RNN_X)***	***RNN_Y***	***Y Coordinate (200 RNN_Y)***	***Results***
1	2.2296	891.8	-0.1932	-38.6	Miss
2	-2.0035	-801.4	1.3034	260.7	Miss
3	-3.1432	-1257.3	0.3286	65.7	Miss
4	-0.7968	-318.7	-1.1417	-228.3	Miss
5	1.0741	429.6	0.7612	152.2	Hit
6	0.1265	50.6	-0.3098	-62.0	Hit
7	0.0611	24.5	-1.1066	-221.3	Hit
8	1.2182	487.3	0.2487	49.7	Hit
9	-0.8026	-321.0	-1.0098	-202.0	Miss
10	0.7324	293.0	0.2552	-51.0	Hit

Example 7: Lead-Time Demand

- Lead-time demand occurs in an inventory system when the lead time is not instantaneous
- The lead time is the time from placement of an order until the order is received
- Assume that lead time is a random variable
- Demand also occurs at random during the lead time
- Lead-time demand is thus a random variable defined as the sum of the demands over the lead time, or

  $\sum_{i=0}^{T} D_i$ where i is the time period of the lead time, i =0,1,2….

  D_i is the demand during the i^{th} time period

  T is the lead time

- The distribution of lead-time demand is found by simulating many cycles of lead time and building a histogram based on the results

Example 7: Lead-Time Demand

- A firm sells bulk roll of newsprint
- Daily demand is given by the following probability distribution:

Daily Demand (Rolls)	3	4	5	6
Probability	0.20	0.35	0.30	0.15

- Lead time is a random variable given by the following distribution:

Lead Time (Days)	1	2	3
Probability	0.36	0.42	0.22

- Table 26 shows the random digit assignment for demand and Table 27 for the lead time; Table 28 shows the incomplete simulation
- The random digits for the first cycle was 57 which generated a lead time of 2 days; Two pairs of random digits must be generated for the daily demand; First pair 11 leads to demand 3 and second pair 64 leads to demand 5; Thus, total demand for 1st cycle is 8

Example 7: Lead-Time Demand

Table 26: Random Digit Assignment for Demand

Demand	*Probability*	*Cumulative Probability*	*Random digit Assignment*
3	0.20	0.20	01 - 20
4	0.35	0.55	21 - 55
5	0.30	0.85	36 - 85
6	0.15	1.00	86 - 00

Example 7: Lead-Time Demand

Table 27: Random Digit Assignment for Lead Time

Lead Time (Days)	*Probability*	*Cumulative Probability*	*Random digit Assignment*
1	0.36	0.36	01 - 36
2	0.42	0.78	37 - 78
3	0.22	1.00	79 - 00

Example 7: Lead-Time Demand

Table 28: Simulation Table for Lead-time Demand

Cycle	Random Digits for Lead Time	Lead Time (Days)	Random Digits for Demand	Demand	Lead-Time Demand
1	57	2	11	3	
			64	5	8
2	33	1	37	4	4
3	46	2	13	3	
			80	5	8
4	91	3	27	4	
			66	5	
			47	4	13
:		:	:	:	:
:		:	:	:	:

- *Note: After many cycles are simulated, a histogram can be generated*

Example 8: Project Simulation

- Suppose a project requires the completion of a number of activities
- Some activities must be carried out sequentially and others can be done in parallel
- The project can be represented by a network of activities
- In fig.3 (Activity network), there are three paths through the network, each path representing a sequence of activities that must be completed in order
- The activities on two different paths can be carried out in parallel
- In the activity network, the arcs represent activities and the nodes represent the start or end of an activity
- The time to complete all activities on a path is the sum of the activity times along the path
- To complete the entire project, all activities must be completed; therefore, project completion time is the maximum over all path completion times

Example 8: Project Simulation

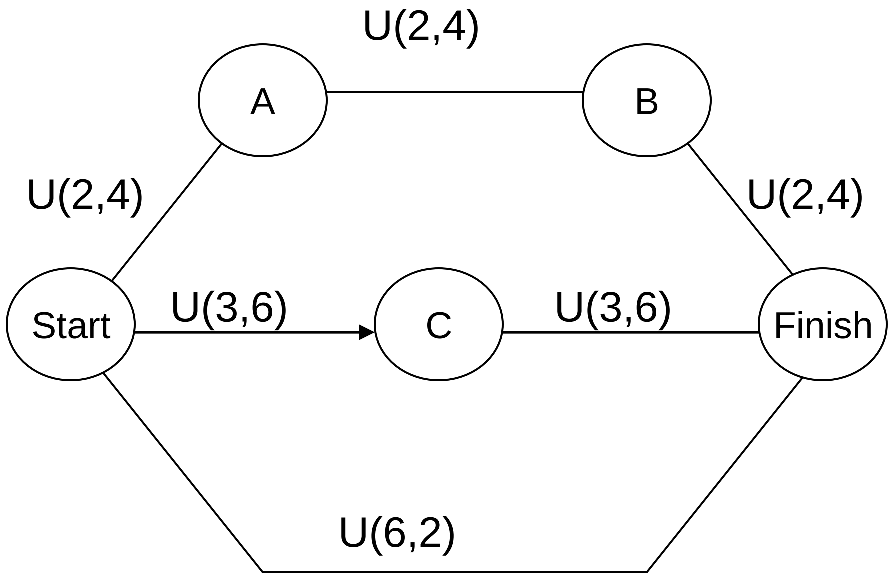


Fig. 3 Activity Network

Example 8: Project Simulation

- Topmost path is along the path Start → A → B → Finish
- Middle path is along path Start → C → Finish
- Bottom path is along path Start → Finish
- Example: Three friends wanted to cook bacon, eggs, and toast for breakfast for some visitors; Each friend was going to prepare one of the three items; Activities might be as follows:

Top path:	Start → A	Crack eggs
	A → B	Scramble eggs
	B → Finish	Cook eggs
Middle path:	Start → C	Make toast
	C → Finish	Butter toast
Bottom path:	Start → Finish	Fry bacon

- The times to accomplish each of the activities in preparing this breakfast are variable represented by a <u>uniform distribution</u> between a lower and upper limit as per fig. 3

Example 8: Project Simulation

- *Problem: 1. To find the preparation time so that the visitors can be informed what time to be present for breakfast*
- (or) 2. *To estimate the probability of preparing breakfast within a specified amount of time*
- The activity times are shown on the arcs of the activity network
- The activity time from Start → A (crack the eggs) is assumed to be uniformly distributed between 2 and 4 minutes; That means that all times between 2 and 4 minutes are equally likely to occur
- Expected or mean time for this activity is the midpoint, 3 minutes
- Expected value along topmost path is 9 minutes, determined by adding the three expected values (3 + 3 + 3)
- Shortest possible completion time, determined by adding the smallest values, is 6 minutes (2 + 2 + 2)
- Largest possible time along top path is 12 minutes (4 + 4 + 4)
- Similarly the expected, shortest and longest paths for the middle and bottom paths are 9, 6 and 12 minutes

Example 8: Project Simulation

- The time that the project (breakfast of eggs, toast and bacon) will be completed is the maximum time through any of the paths
- But since activities are assumed to be some random variability, the time through the paths are not constant
- For a uniform distribution, a simulated activity is given by [Pritsker]:

*Simulated Activity Time = Lower limit + (Upper limit - Lower limit) * Random number*

- The time for each simulated activity can be computed as follows:

Example: for activity Start → A, if random number is 0.7943, the simulated activity time is 2 + (4 - 2) * 0.7943 = 3.59 minutes

- Simulate using Experiment worksheet (downloaded from www.bcnn.net) in the Excel workbook for this example and compute the average, median, minimum and maximum values. With 400 trials using default seed, results were as follows:

    Mean 10.12, Min 6.85, and *Max 12.00* minutes

Example 8: Project Simulation

- The *critical path* is the path that takes the longest time for completion; that is, its time is the project completion time
- For each of the 400 trials, the experiment determines at which path was critical, with these results:

  Top path 30.00% of trials

  Middle path 31.25%

  Bottom path 38.75%

- Conclusion is that the chance of the bacon being the last item ready is 38.75%. [Question: *Why aren't the paths each represented about 1/3 of the time?]*
- The project completion times were placed in a frequency chart; These differ each time that spreadsheet is recalculated, but, in any large number of trials, the basic shape of the chart (or histogram) will remain roughly the same; Inferences drawn from fig. is that:

  13.5% of time (54/400), breakfast will be ready in <=9 minutes

  20.5% of time (82/400), it will take 11 to 12 minutes