MODULE 1: Introduction to Modeling and Simulation

- A simulation is the imitation of the operation of real-world processes over time.
- A model is a construct that describes the system.

Types of simulation

1. Continuous simulation
    a. Refers to a system that tracks system changes continuously over a period of time.
    b. A continuous simulation model is in which the state variable changes continuously.
    c. Eg. Flight simulators, advanced engineering design tools, etc

2. Discrete event simulation
    a. Refers to the simulation where the state variable changes at discrete time.
    b. Useful when simulation is a queue of events
    c. The system can change a countable number of times in time.
    d. Eg. Stress testing, financial investments, etc

Single server queue system

- Calling population
    - Infinite
    - Meaning if a unit leaves the calling population and joins waiting line, there is no change in the calling population
- Arrival
    - Defines the way customers enter the system
    - Most arrivals are random with random intervals between two adjacent arrivals.
    - Arrival is described by a random distribution of intervals aka Arrival Pattern

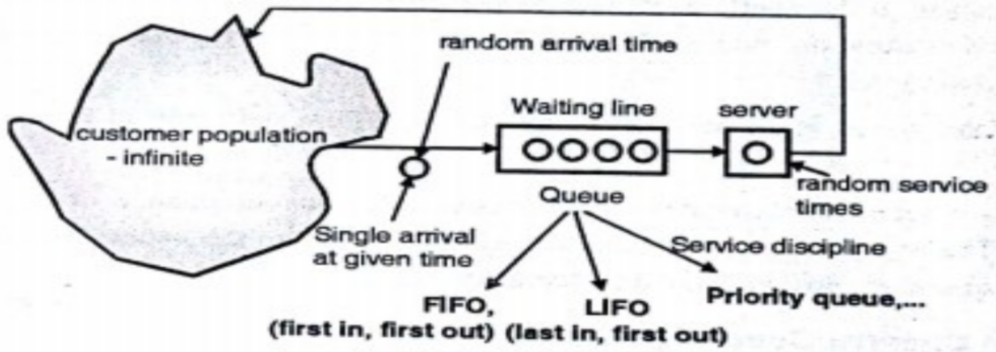

Fig. 2.2.2 : Single Server Queue

- Service mechanism
    - FIFO is used by the server
    - Service mechanism is according to the distribution of service times
    - Service times are randomly distributed according to some distribution
- System capacity
    - Represents the wait queue length accepted by the system
    - If system capacity is limited, some customers may have to leave without being served
    - Hence, the system capacity is assumed to be infinite.

- Arrival rate
    - Arrival rate must be less than service rate for stable systems. Otherwise, the system will be unbounded and the queue will grow endlessly
- Queue disciplines
    - FIFO
    - LIFO
    - SIRO (Serve in random order)
    - Priority queue
- State
    - Represents the number of units in the system and the state of system (busy/idle)
- Event
    - Something that causes the state of the system to change.

Performance measures

1. Average waiting time
   Total waiting time / Total number of customers
2. Probability of customers waiting
   Number of customers waiting / Total number of customers
3. Probability of idle server
   Idle time of server / total runtime of simulation
4. Average time between arrival
   Total time between arrivals / number of arrivals **- 1**
5. Average waiting time for those who wait
   Total time customer waits in queue / total number of customers who wait
6. Average time spent in system
   Total time spent in system / total number of customers

MODULE 2: Random Number Generation

Need for random numbers

1. Random numbers are essential for mathematical modeling
2. Used in simulation of all discrete systems
3. Helps demonstrate the uncertainty of simulation models using random inputs

Properties Random Numbers

Uniformity: If the interval (0, 1) is divided into ‘n’ classes or subintervals of equal length, the expected number of observations in each interval is N/n, where N is the total number of observations.

Independence: The probability of observing a value in a particular interval is independent of the previous drawn value.

Problems faced in generating random numbers

1. Numbers may not be uniformly distributed
2. Numbers may not be independent
3. Mean of the numbers may be too high or low
4. Variance of the numbers may be too high or low

5. Insufficient period between random numbers. (Period: number of elements until numbers start to repeat with a given sequence)

Criteria for random number generator

1. Routine should be fast
2. Routine should be portable
3. Should have a sufficiently large period
4. Routine should be replicable
5. Generated random numbers should closely replicate the ideal properties of a random number: Uniformity and Independence

Linear Congruential Method

Generate a sequence of random integers X1, X2, … between 0 and m-1

$X_{i+1} = (aX_i + c) \bmod m$, i=0,1,2,...

X_0 : seed a : multiplier c : increment m : modulus

If c=0 →Multiplicative generator c!=0 →Mixed generator

Multiple Linear Congruential Method

Given will be:
K, m_1, a_1, m_2, a_2

To do:
1. Select X_0 in range $[1, m_1-1]$
   Select X_1 in range $[1, m_2-1]$
   Set j=0
2. Evaluate for each generator
   $X_{1,j+1} = a_1X_{1,j} \bmod m_1$
   $X_{2,j+1} = a_2X_{2,j} \bmod m_2$
3. Set $X_{j+1} = (X_{1,j+1} - X_{2,j+1}) \bmod m_1-1$
4. Return
   $R_{j+1} = X_{j+1}/m_1$ if $X_i > 0$
   $= m_1-1/m_1$ if $X_i = 0$
5. Set j = j+1 and go to step 2

Monte Carlo Simulation

- Simulation that allows taking risk into consideration for decision making
- AKA Static simulation model. Represents the model at a particular point in time
- Eg. Simulating probabilities of rolling a dice

Tests on Random Numbers

TESTS FOR UNIFORMITY → KS TEST; CHI SQUARE TEST

TESTS FOR INDEPENDENCE → RUNS TEST; AUTOCORRELATION

KS TEST

[II] Algorithm

The test consists of the following steps

(i) Define the hypothesis for testing the uniformity as:

$H_0 : R_i \sim U[0,1]$

$H_1 : R_i \nsim U[0,1]$

(ii) Rank the data from smallest to largest

$R_{(1)} \leq R_{(2)} \leq \ldots \leq R_{(N)}$

(iii) Compute D^+ and D^-

$$D^+ = \max_{1 \leq i \leq N} \left\{ \frac{i}{N} - R_{(i)} \right\}$$

$$D^- = \max_{1 \leq i \leq N} \left\{ R_{(i)} - \frac{i-1}{N} \right\}$$

(iv) Compute $D = \max(D^+, D^-)$

(v) Determine the critical value, D_α for the significance level and the sample size N given.

(vi) If $D \leq D_\alpha$ accept, otherwise reject H_0.

Draw table

i	R_i	$D^+ = i/N - R_i$	$D^- = R_i - (i-1)/N$
		Max D^+ here	Max D^- here

MARK NEGATIVE VALUES AS – IN TABLE

Example:

Numbers : 0.44, 0.81, 0.14, 0.05, 0.93

$\alpha = 0.05 \Rightarrow D_\alpha = \cancel{0.26}\ 0.565$ $N = 5$

① Sort numbers

0.05, 0.14, 0.44, 0.81, 0.93

② Table.

i		R_i	$i/N - R_i$	$R_i - (i-1)/N$
1	0	0.05	0.15	0.05
2	1	0.14	0.26	–
3	2	0.44	0.16	0.04
4	3	0.81	–	0.21
5	4	0.93	0.07	0.13
			$\max = D^+ = 0.26$	$\max = D^- = 0.21$

$\therefore D = \max(D^+, D^-)$
$= 0.26$

Since $D < D_\alpha$ $(0.26 < 0.565)$

H_0 is accepted.

$\boxed{\begin{array}{l} H_0 : R_i \sim U[0,1] \\ H_1 : R_i \nsim U[0,1] \end{array}}$

CHI SQUARE TEST

7.5.2.2 Chi-Square Test

- It is used to test the uniformity within a data set or to determine the probability of a dependency relationship between two or more distinct data set.
- It tests a null hypothesis that the relative frequencies of occurrence of observed events follow a specified frequency distribution.
- It is valid only for large samples say $N \geq 50$.

Algorithm :

(i) Define the hypothesis for testing the uniformity as :

$H_0 : R_i \sim U[0,1]$

$H_1 : R_i \nsim U[0,1]$

(ii) Divide the total number of observation (N) into equally numbered classes (n), $[a_0, a_1)\ldots[a_{n-1}, a_n)$. n should be chosen in such a way that $E_i(= N/n) \geq 5$.

(iii) Compute the sample test statistics

$$\chi_0^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

where O_i is the observed number in the i^{th} class

E_i is the expected number in the i^{th} class

n is the number of classes.