ABES Engineering College, Ghaziabad
B. Tech Odd Semester Sessional Test-3

Printed Pages: 2
Session: 2024-25

Semester: V
Course Code: BCS056
Course Name: Application of Soft Computing
Maximum Marks: 40

Roll No.:
Time: 1Hr 30 Min.

Instructions:
1. **Attempt all sections.**
2. **If require any missing data, then choose suitably.**

Q. No.	Question	Marks	CO	KL	PI
Section-A		**Total Marks : 20**			
1	**Attempt ANY ONE part from the following**				
a)	Define Membership Function. Explain its features.	**2+3**	**CO4**	K2	1.3.1

Sol

Membership Function definition

Membership function defines the fuzziness in a fuzzy set whether discrete or continuous via mathematical model. Generally represented by a graph.

Let us consider fuzzy set A, A = {(x, μA(x))| x ∈ X} where μA(x) is called the membership function for the fuzzy set A. X is referred to as the universe of discourse. The membership function associates each element x ∈ X with a value in the interval [0, 1].

In fuzzy sets, each elements is mapped to [0,1] by membership function. That is, $\mu_A : X \in [0, 1]$, where [0,1] means real numbers between 0 and 1 (including 0,1). Consequently, fuzzy set is with 'vague boundary set' comparing with crisp set.

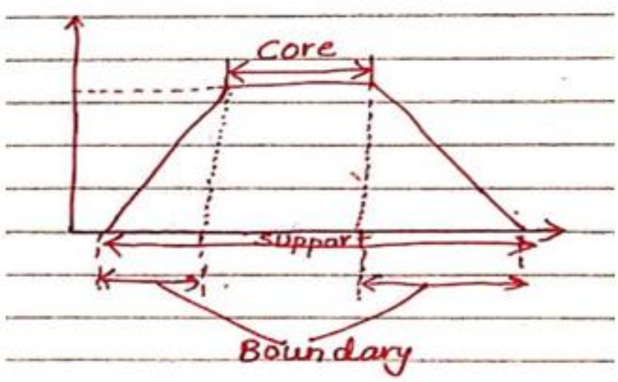
The fuzzy set A can be alternatively denoted as follows:

If X is discrete then $A = \Sigma\, \mu_A(xi) / xi$

If X is continuous then $A = \int \mu_A(x) / x$

Features of Membership Function

- Core
- Support
- Boundary
- Crossover point(x |μ(x)=0.5)
- Height (max value of μ)



b)	What do you understand by Fuzzification? Enlist methods of membership value assignment. Explain any one with example.	**1.5+1.5+2**	**CO4**	K2	1.3.1

Concept of Fuzzification

For a fuzzy set $\underset{\sim}{A} = \{\mu_i/x_i | x_i \in X\}$, a common fuzzification algorithm is performed by keeping μ_i constant and x_i being transformed to a fuzzy set $Q(x_i)$ depicting the expression about x_i. The fuzzy set $Q(x_i)$ is referred to as the *kernel of fuzzification*. The fuzzified set $\underset{\sim}{A}$ can be expressed as

$$\underset{\sim}{A} = \mu_1 Q(x_1) + \mu_2 Q(x_2) + \cdots + \mu_n Q(x_n)$$

where the symbol ~ means fuzzified. This process of fuzzification is called support fuzzification (s-fuzzification). There is another method of fuzzification called *grade fuzzification* (g-fuzzification) where x_i is kept constant and μ_i is expressed as a fuzzy set. Thus, using these methods, fuzzification is carried out.

Methods of Membership Value Assignments

- Intuition
- Inference
- Rank ordering
- Angular fuzzy set
- Neural network
- Genetic algorithm
- Inductive reasoning

Intuition

1. Using your own intuition and definitions of the universe of discourse, plot fuzzy membership functions for "weight of people".

Solution: The universe of discourse is weight of people. Let the weights be in kg, i.e., kilogram. Let the linguistic variables be the following:

$$\begin{aligned}
\text{Very thin } (VT) &: W \le 25 \\
\text{Thin } (T) &: 25 < W \le 45 \\
\text{Average } (AV) &: 45 < W \le 60 \\
\text{Stout } (S) &: 60 < W \le 75 \\
\text{Very stout } (VS) &: W > 75
\end{aligned}$$

Now plotting the defined linguistic variables using triangular membership functions, we obtain Figure 1.

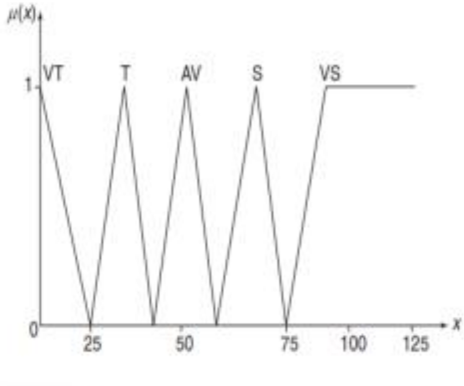


Figure 1 Membership function for weight of people.

2	**Attempt ANY ONE part from the following**				
a)	Explain the rule-base and membership functions used in Greg Viot's Fuzzy Cruise Controller.	**5**	**CO4**	K2	1.2.1

Sol

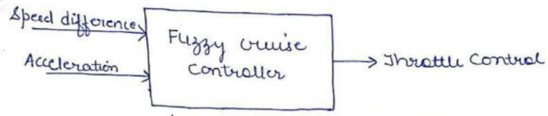


fig: Fuzzy cruise Controller

Fuzzy rule base

A sample fuzzy rule base R governing the cruise Control.

Table: Sample cruise control rule base

Rule 1: If (Speed difference is NL) and (acceleration is ZE) then (throttle control is PL).

Rule 2: If (speed difference is ZE) and (acceleration is NL) then (throttle control is PL)

Rule 3: If (speed difference is NM) and (acceleration is ZE) then (throttle control is PM)

Rule 4: If (speed difference is NS) and (acceleration is PS) then (throttle control is PS)

Rule 5: If (speed difference is PS) and (acceleration is NS) then (throttle control is NS)

Rule 6: If (speed difference is PL) and (acceleration is ZE) then (throttle control is NL)

Rule 7: If (speed difference is ZE) and (acceleration is NS) then (throttle control is PS)

Rule 8: If (speed difference is ZE) and (acceleration is NM) then (throttle control is PM).

NL - Negative Large
ZE - Zero
PL - Positive Large
NM - Negative Medium
PM - Positive Medium
NS - Negative Small
PS - Positive Small

∴ The fuzzy sets which characterize the inputs and outputs are given as (14)

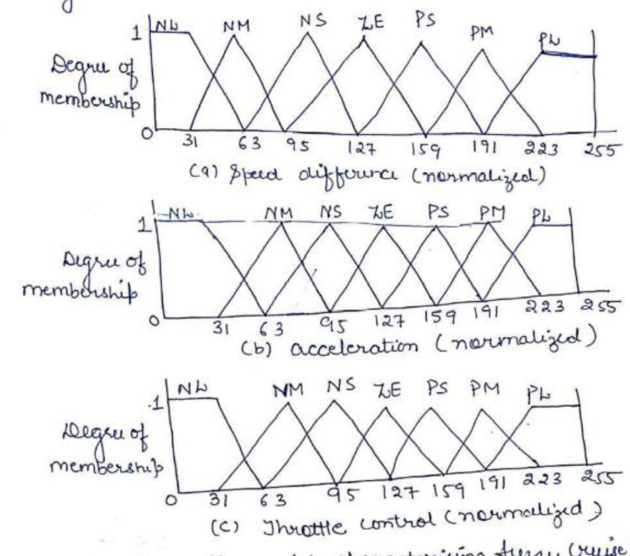


fig: Fuzzy sets characterising fuzzy cruise Control

Fuzzification of inputs

for the fuzzification of inputs, that is, to compute the membership for the antecedents, the formula is used.

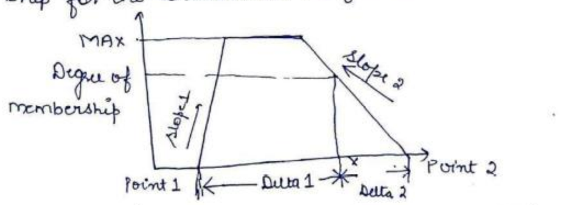
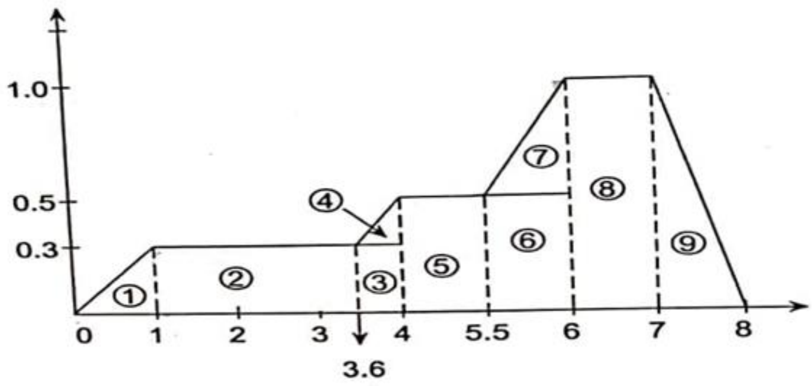


fig: Computation of fuzzy membership value

b)	Using the inference approach, obtain the membership values for the triangular shapes ($\tilde{I}$,$\tilde{R}$, $\tilde{T}$) for a triangle with angles 40°, 60° and 80°.	**5**	**CO4**	K2	1.2.1
Sol	Using the inference approach, obtain the membership values for the triangular shapes ($\underset{\sim}{I}, \underset{\sim}{R}, \underset{\sim}{T}$) for a triangle with angles 40°, 60° and 80°. **Solution:** Let the universe of discourse be $U = \{(X,Y,Z): X = 80° \geq Y = 60° \geq Z = 40°$ and $X+Y+Z = 80°+60°+40° = 180°\}$ • Membership value of isosceles triangle, $\underset{\sim}{I}$: $\mu_{\underset{\sim}{I}} = 1 - \frac{1}{60°}\min(X-Y, Y-Z)$ $= 1 - \frac{1}{60°}\min(80°-60°, 60°-40°)$ $= 1 - \frac{1}{60°}\min(20°, 20°)$ $= 1 - \frac{1}{60°}\times 20° = 0.667$ • Membership value of right-angle triangle, $\underset{\sim}{R}$: $\mu_{\underset{\sim}{R}} = 1 - \frac{1}{90°}\lvert X - 90°\rvert = 1 - \frac{1}{90°}\lvert 80° - 90°\rvert$ $= 1 - \frac{1}{90°}\times 10° = 0.889$ • Membership value of other triangles, $\underset{\sim}{T}$: $\mu_{\underset{\sim}{T}} = \min[1-\mu_{\underset{\sim}{I}}, 1-\mu_{\underset{\sim}{R}}]$ $= \min[1-0.667, 1-0.889]$ $= \min[0.333, 0.111] = 0.111$ Thus the membership values for isosceles, right-angle triangle and other triangles are calculated.				
3	**Attempt ANY ONE part from the following**				
a)	For the figure given below representing union of fuzzy sets, calculate the defuzzified value using: i) Centroid method ii) First of Maxima method iii) Last of Maxima method iv) Mean of Maxima method 	**6+1+1 +2**	**CO4**	K3	2.4.1

Sol

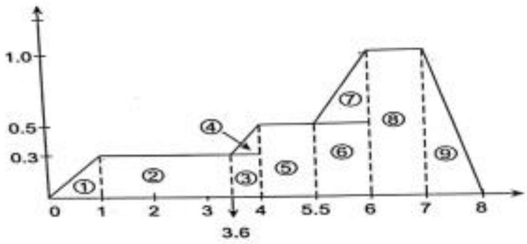


Fig. 7.3 Aggregated fuzzy set of $\tilde{A}_1$, $\tilde{A}_2$, and $\tilde{A}_3$ viewed as area segments.

Table 7.4 illustrates the computations for obtaining x^*.

Table 7.4 Computation of x^*

Area segment no.	Area (A)	$\overline{x}$	$A\overline{x}$
1	$\frac{1}{2} \times 0.3 \times 1 = 0.15$	0.67	0.1005
2	$2.6 \times 0.3 = 0.78$	2.3	1.794
3	$0.3 \times 0.4 = 0.12$	3.8	0.456
4	$\frac{1}{2} \times 0.4 \times 0.2 = 0.04$	3.8667	0.1546
5	$1.5 \times 0.5 = 0.75$	4.75	3.5625
6	$0.5 \times 0.5 = 0.25$	5.75	1.4375
7	$\frac{1}{2} \times 0.5 \times 0.5 = 0.125$	5.833	0.729
8	$1 \times 1 = 1$	6.5	6.5
9	$\frac{1}{2} \times 1 \times 1 = 0.5$	7.33	3.665

i) Centroid method = 4.9
ii) First of Maxima method = 6
iii) Last of Maxima method = 7
iv) Mean of Maxima method = 6.5

b)	Apply the fuzzy modus ponens rule to deduce rotation is quite slow. i) If the temp is high then the rotation is slow. ii) The temperature is very high. Where $\widetilde{H}$ (High), $\widetilde{VH}$ (Very High), $\tilde{S}$ (Slow) and $\widetilde{QS}$ (Quite Slow) indicate the associated fuzzy sets described as below: $\widetilde{H}$ = {(70,1), (80, 1), (90,0.3)} $\widetilde{VH}$ = { (90,0.7), (100,1)} $\widetilde{QS}$ = {(10,1), (20, 0.8)} $\tilde{S}$ = {(30,0.8), (40, 1), (50, 0.6)} Universal set for temperature is X = {30,40,50,60,70,80,90,100}, for rotation, Y = {10,20,30,40,50,60}.	**10**	**CO4**	K3	2.4.1

Solution:

$$\tilde{R}(x, y) = \max(\tilde{H} \times \tilde{S}, \bar{\tilde{H}} \times Y)$$

$$\tilde{H} \times \tilde{S} = \begin{array}{c} \\ 30 \\ 40 \\ 50 \\ 60 \\ 70 \\ 80 \\ 90 \\ 100 \end{array} \begin{array}{c} \begin{array}{cccccc} 10 & 20 & 30 & 40 & 50 & 60 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0 & 0 & 0.3 & 0.3 & 0.3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$\bar{\tilde{H}} \times Y = \begin{array}{c} \\ 30 \\ 40 \\ 50 \\ 60 \\ 70 \\ 80 \\ 90 \\ 100 \end{array} \begin{array}{c} \begin{array}{cccccc} 10 & 20 & 30 & 40 & 50 & 60 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0.7 & 0.7 & 0.7 & 0.7 & 0.7 & 0.7 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \end{array}$$

$$\tilde{R}(x, y) = \begin{array}{c} \\ 30 \\ 40 \\ 50 \\ 60 \\ 70 \\ 80 \\ 90 \\ 100 \end{array} \begin{array}{c} \begin{array}{cccccc} 10 & 20 & 30 & 40 & 50 & 60 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0.7 & 0.7 & 0.7 & 0.7 & 0.7 & 0.7 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \end{array}$$

To deduce Rotation is quite slow we make use of the composition rule

$$\tilde{QS} = \tilde{VH} \circ \tilde{R}(x, y)$$

$$= [0\ 0\ 0\ 0\ 0\ 0\ 0.9\ 1] \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0 & 0 & 0.8 & 1 & 0.6 & 0 \\ 0.7 & 0.7 & 0.7 & 0.7 & 0.7 & 0.7 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$= [1\ 1\ 1\ 1\ 1\ 1]$$

	Section-B	**Total Marks : 20**			
4	**Attempt ANY ONE part from the following**				
a)	What are the advantages of genetic Algorithm over Conventional Algorithm?	**5**	**CO5**	K2	1.3.1

<table>
<tr>
<td>Sol</td>
<td>

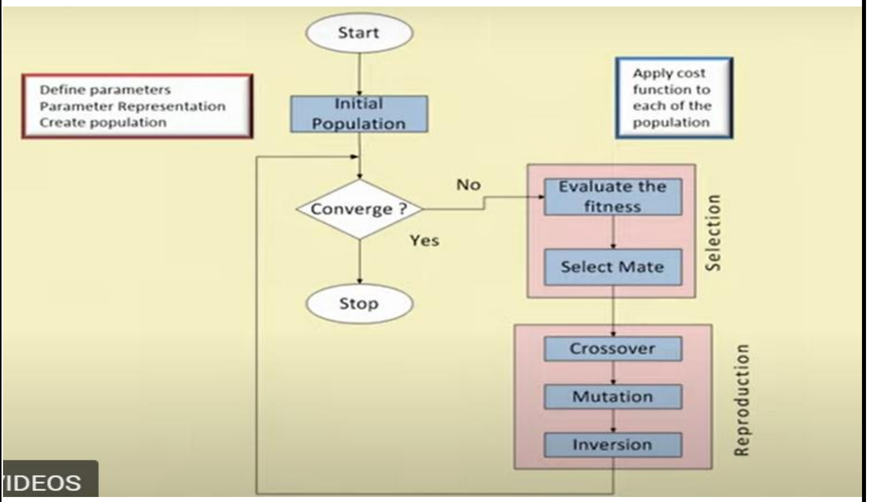
Advantages of GA

1. Parallelism.
2. Liability.
3. Solution space is wider.
4. The fitness landscape is complex.
5. Easy to discover global optimum.
6. The problem has multiobjective function.

</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b)</td>
<td>Explain Generational cycle in Genetic algorithm with diagram.</td>
<td>5</td>
<td>CO5</td>
<td>K2</td>
<td>1.3.1</td>
</tr>
<tr>
<td></td>
<td>

One complete cycle of GA is known as Generational cycle

- Step 1: Definition of objective function (maximize f(X) is generally taken. If we want to minimize f(X), we generally take maximize (1/(1+f(X)))
- Step 2: Definition and implementation of genetic representation
  - **ENCODING** Coding of inputs/variables (population/search space). Advantage: Discretizes search space even if function is continuous.
- Step 3: Definition and Implementation of genetic operators:
  - **SELECTION** : Set of variables are selected to generate new set of I/p (population).
  - **CROSSOVER:** Generation of new population by recombination of selected I/ps
  - **MUTATION and INVERSION:** Modification in individual element.



</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</table>

5	**Attempt ANY ONE part from the following**				
a)	Explain the Roulette Wheel Selection method with example.	**5**	**CO5**	K2	1.3.1

Sol

Roulette-Wheel selection : Implementation

Input: A Population of size N with their fitness values
Output: A mating pool of size N_p

Steps:

1) Compute $p_i = \frac{f_i}{\sum_{i=1}^{N} f_i}, \forall i = 1,2 \cdots N$
2) Calculate the cumulative probability for each of the individual starting from the top of the list, that is
$$P_i = \sum_{j=1}^{i} p_j, \text{ for all } j = 1,2 \cdots N$$
3) Generate a random number say r between 0 and 1.
4) Select the $j-th$ individual such that $P_{j-1} < r \leq P_j$
5) Repeat Step 3-4 to select N_p individuals.
6) End

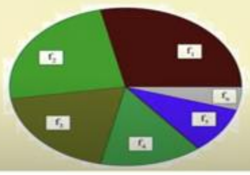
Roulette Wheel Selection

- In this method we use a wheel to represent the fitness values and do selection with the help of this wheel.

$i \rightarrow f_i$

$j \rightarrow f_j$

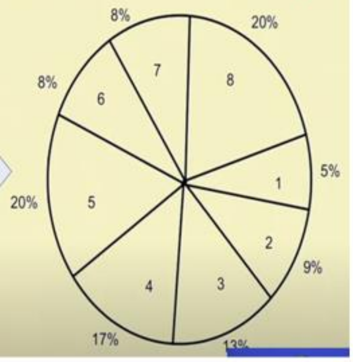
$f_i > f_j \Rightarrow i$ (Preferable than j)



[Wheel Game (Rotate and See the Pointer)]

Example

Individual	Fitness value	p_i
1	1.01	0.05
2	2.11	0.09
3	3.11	0.13
4	4.01	0.17
5	4.66	0.20
6	1.91	0.08
7	1.93	0.08
8	4.51	0.20



Roulette-Wheel selection: Example

The probability that $i-th$ individual will be pointed is $p_i = \frac{f_i}{\sum_{i=1}^{N} f_i}$

Example:

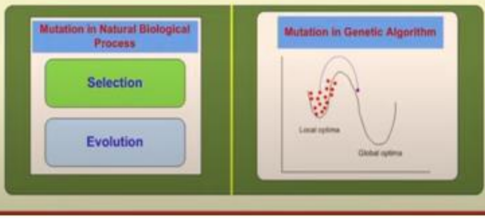
Individual	p_i	P_i	r	T
1	0.05	0.05	0.26	I
2	0.09	0.14	0.04	I
3	0.13	0.27	0.48	II
4	0.17	0.44	0.43	I
5	0.20	0.64	0.09	II
6	0.08	0.72	0.30	
7	0.08	0.80	0.61	
8	0.20	1.0	0.89	I

p_i = Probability of an individual

P_i = Cumulative Probability

r = Random Number between 0..1

T=Tally count of selection

b)	Explain Mutation and Mutation rate with example.	**3+2**	**CO5**	K2	1.3.1
	MUTATION 1) In genetic algorithm, the mutation is a genetic operator used to maintain genetic diversity from one generation of a population (of chromosomes) to the next. 2) It is analogues to biological mutation. 3) In GA, the concept of biological mutation is modelled artificially to bring a local change over the current solutions.  Mutation Operation in Binary coded GA 1) In binary-coded GA, the mutation operator is simple and straight forward. 2) In this case, one (or a few) $1(s)$ is(are) to be converted to $0(s)$ and vice-versa. 3) A common method of implementing the mutation operator involves generating a random variable called mutation probability (μ_p) for each bit in a sequence. 4) This mutation probability tells us whether or not a particular bit will be mutated (i.e., modified). **Note:** 1) To avoid a large deflection, μ_p is generally kept to a low value. 2) It is varied generally in the range of $\frac{0.1}{L}$ to $\frac{1.0}{L}$, where L is the string length.				
6	**Attempt ANY ONE part from the following**				
a)	How crossover is used in Genetic Algorithm? Enlist the types of crossover. Perform crossover with at-least 5 methods on below mentioned strings: Parent 1: 1010100110 Parent 2: 0101100011.	**3+2+5**	**CO5**	K3	1.4.1

Crossover (recombination)

- Operation to be performed after Selection
- Consumes most of the time of Genetic Algorithm
- Output of Selection: Mating Pool => I/P of Crossover
- Mating pool: Consists of best-fit individuals (let us say N)
- The number of combining strings(mating pairs)=N/2
- These pairs/strings will participate in reproduction process as per the crossover probability (which we keep generally as 1)
- NOTE: Depending upon type of encoding schemes we have different crossover mechanisms.

Crossover Techniques in Binary coded GA

There exists a large number of crossover schemes, few important of them are listed in the following.

- Single point crossover
- Two-point crossover
- Multi-point crossover (also called n-point crossover)
- Uniform crossover (UX)
- Half-uniform crossover (HUX)
- Shuffle crossover
- Matrix crossover (Tow-dimensional crossover)
- Three-parent crossover

i) Single point crossover: Let k=8

Parent 1: 1010100110
Parent 2: 0101100011.
Offspring 1: 1010100111
Offspring 2: 0101100010

ii) Two point crossover: Let k1=2, k2=8

Parent 1: 1010100110
Parent 2: 0101100011.
Offspring 1: 0110100111
Offspring 2: 1001100010

iii) Uniform crossover:

Parent 1: 1010100110
Parent 2: 0101100011
Coin tossing: 1100100011
Offspring 1: 0110100111
Offspring 2: 1001100010

iv) Shuffle crossover: Let k1=2

Parent 1: 1010100110
Parent 2: 0101100011.
Parent 1’: 1010011001
Parent 2’:0110011001
Offspring 1: 0110011001
Offspring 2: 1010011001

v) Half Uniform crossover

	Parent 1: 1010100110 Parent 2: 0101100011 Hamming distance = 6. So only 3 bits will be swapped but probabilistically. Coin tossing: 1100100000 Offspring 1: 0110100110 Offspring 2: 1001100011				
b)	Maximize the function f(x) = x^2 , with x in the integer interval [0,31] with the help of Genetic Algorithm. Show one generational cycle.	**10**	**CO5**	K3	1.4.1

Step 1: For using GA approach, one must first code the decision variable "x" into a finite length string. Using a five bit (binary integer) unsigned integer, numbers between 0(00000) and 31(11111) can be obtained.

The objective function here is $f(x) = x^2$ which is to be maximized. A single generation of a GA is performed here with encoding, selection, crossover and mutation. To start with, select initial population at random. Here initial population of size 4 is chosen, but any number of populations can be selected based on the requirement and application. Table 21-4 shows an initial population randomly selected.

Step 2: Obtain the decoded x values for the initial population generated. Consider string 1,

$$\begin{aligned} 01100 &= 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 0*2^0 \\ &= 0+8+4+0+0 \\ &= 12 \end{aligned}$$

Thus for all the four strings the decoded values are obtained.

TABLE 21-4 SELECTION

String no.	Initial population (randomly selected)	x value	Fitness $f(x) = x^2$	Prob_i	Percentage probability (%)	Expected count	Actual count
1	0 1 1 0 0	12	144	0.1247	12.47	0.4987	1
2	1 1 0 0 1	25	625	0.5411	54.11	2.1645	2
3	0 0 1 0 1	5	25	0.0216	2.16	0.0866	0
4	1 0 0 1 1	19	361	0.3126	31.26	1.2502	1
Sum			1155	1.0000	100	4.0000	4
Average			288.75	0.2500	25	1.0000	1
Maximum			625	0.5411	54.11	2.1645	2

Step 3: Calculate the fitness or objective function. This is obtained by simply squaring the "x" value, since the given function is $f(x) = x^2$. When $x = 12$, the fitness value is

$$f(x) = x^2 = (12)^2 = 144$$

$$For\ x = 25,\quad f(x) = x^2 = (25)^2 = 625$$

and so on, until the entire population is computed.

Step 4: Compute the probability of selection,

$$\text{Prob}_i = \frac{f(x)_i}{\sum_{i=1}^{n} f(x)_i} \tag{21.15}$$

where n is the number of populations; $f(x)$ is the fitness value corresponding to a particular individual in the population;

$\Sigma f(x)$ is the summation of all the fitness value of the entire population.

Considering string 1,

$$\text{Fitness } f(x) = 144$$

$$\Sigma f(x) = 1155$$

The probability that string 1 occurs is given by

$$P_1 = 144/1155 = 0.1247$$

The percentage probability is obtained as

$$0.1247 * 100 = 12.47\%$$

The same operation is done for all the strings. It should be noted that summation of probability select is 1.

Step 5: The next step is to calculate the expected count, which is calculated as

$$\text{Expected count} = \frac{f(x)_i}{[\text{Avg } f(x)]_i} \tag{21.16}$$

where

$$(Avg\ f(x))_i = \left[\frac{\sum_{i=1}^{n} f(x)_i}{n}\right]$$

For string 1,

Expected count = Fitness/Average = 144 / 288.75 = 0.4987

We then compute the expected count for the entire population. The expected count gives an idea of which population can be selected for further processing in the mating pool.

Step 6: Now the actual count is to be obtained to select the individuals who would participate in the crossover cycle using Roulette wheel selection. The Roulette wheel is formed as shown in Figure 21-33.

The entire Roulette wheel covers 100% and the probabilities of selection as calculated in step 4 for the entire populations are used as indicators to fit into the Roulette wheel. Now the wheel may be spun and the number of occurrences of population is noted to get actual count.

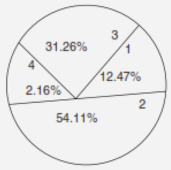


Figure 21-33 Selection using Roulette wheel.

1. String 1 occupies 12.47%, so there is a chance for it to occur at least once. Hence its actual count may be 1.
2. With string 2 occupying 54.11% of the Roulette wheel, it has a fair chance of being selected twice. Thus its actual count can be considered as 2.
3. On the other hand, string 3 has the least probability percentage of 2.16%, so their occurrence for next cycle is very poor. As a result, it actual count is 0.
4. String 4 with 31.26% has at least one chance for occurring while Roulette wheel is spun, thus its actual count is 1.

The above values of actual count are tabulated as shown is Table 21-5.

Step 7: Now, write the mating pool based upon the actual count as shown in Table 21-5.

TABLE 21-5 CROSSOVER

String no.	Mating Pool	Crossover point	Offspring after crossover	x value	Fitness value $f(x) = x^2$
1	0 1 1 0 0	4	0 1 1 0 1	13	169
2	1 1 0 0 1	4	1 1 0 0 0	24	576
3	1 1 0 0 1	2	1 1 0 1 1	27	729
4	1 0 0 1 1	2	1 0 0 0 1	17	289
Sum					1763
Average					440.75
Maximum					729

The actual count of string no. 1 is 1, hence it occurs once in the mating pool. The actual count of string no. 2 is 2, hence it occurs twice in the mating pool. Since the actual count of string no. 3 is 0, it does not occur in the mating pool. Similarly, the actual count of string no. 4 being 1, it occurs once in the mating pool. Based on this, the mating pool is formed.

Step 8: Crossover operation is performed to produce new offspring (children). The crossover point is specified and based on the crossover point, single-point crossover is performed and new offspring is produced. The parents are

Parent 1	0 1 1 0 0
Parent 2	1 1 0 0 1

The offspring is produced as

Offspring 1	0 1 1 0 1
Offspring 2	1 1 0 0 0

In a similar manner, crossover is performed for the next strings.

Step 9: After crossover operations, new offspring are produced and "x" values are decoded and fitness is calculated.

Step 10: In this step, mutation operation is performed to produce new offspring after crossover operation. As discussed in Section 21.9.4.1 mutation-flipping operation is performed and new offspring are produced. Table 21-6 shows the new offspring after mutation. Once the offspring are obtained after mutation, they are decoded to x value and the fitness values are computed.

TABLE 21-6 MUTATION

String no.	Offspring after crossover	Mutation chromosomes for flipping	Offspring after mutation	x value	Fitness $f(x) = x^2$
1	0 1 1 0 1	1 0 0 0 0	1 1 1 0 1	29	841
2	1 1 0 0 0	0 0 0 0 0	1 1 0 0 0	24	576
3	1 1 0 1 1	0 0 0 0 0	1 1 0 1 1	27	729
4	1 0 0 0 1	0 0 1 0 0	1 0 1 0 0	20	400
Sum					2546
Average					636.5
Maximum					841

CO Course Outcomes mapped with respective question

KL Bloom's knowledge Level (K1, K2, K3, K4, K5, K6)

K1-Remember, K2-Understand, K3-Apply, K4-Analyze, K5: Evaluate, K6-Create