Use of Genetic Algorithms for Finding Roots of Algebraic Equations

Harsh Bhasin[#], Surbhi Bhatia[*]
[#]*www.computergrad.com*
Faridabad, Haryana, India
[*]*Amity University, Noida, Uttar Pradesh, India*

***Abstract*— An idea of the values between which the roots of the algebraic equation lie is needed in both bisection method and other numerical methods for finding out the roots. Therefore, a method to suggest them is required. Finding them is a search process, so Genetic Algorithms (GA) can be used for the above task as GA's are theoretically and empirically proven to provide robust search in complex spaces. (1.)**

After finding out these values, the population obtained in the initial step needs to be converted into population having equivalent values lying between the numbers .This increases the plausibility of the root of the equation lying near to; if not equal; to the values so obtained. Crossover and mutation increases the chances of finding out the roots .The work explores finding out the roots of an algebraic equation with the help of GA.

***Keywords*—Root Finding, Genetic Algorithms, Evolutionary Algorithms, Theory of natural selection**

I. INTRODUCTION

Algorithms for solving root problems numerically can be divided into two main groups: direct methods and iterative methods. Direct methods are those which can be completed in a predetermined finite number of steps. Iterative methods are methods which converge to the solution over time. An iteration method or an approximation method is a method in which we start from an initial guess x0 and compute step-by-step approximations x1, x2, . . . of the unknown solution of equation f(x)=0(1).The bisection method is the simplest and most robust algorithm for finding the root of a one-dimensional continuous function on a closed interval. The basic idea is a follows. Suppose that f (¢) is a continuous function defined over an interval [a; b] and f(a) and f(b) have opposite signs. Then by the intermediate value theorem, there exists at least one r 2 [a; b] such that f(r) = 0. The method is iterative and each iteration starts by breaking the current interval bracketing the root(s) into two subintervals of equal length. One of the two sub intervals must have end points of different signs. This subinterval becomes the new interval and the next iteration begins. Thus we can define smaller and smaller intervals such that each interval contains r by looking at subintervals of the current interval and choosing the one in which f(¢) changes signs. This process continues until the width of the interval containing a root shrinks below some predetermined error tolerance. A weakness of bisection method is that it cannot be generalized to the general n dimensional

Case and cannot find the root of a function unless the function value takes opposite signs on either side of that root. So, Roots of this form are more difficult to find in general. The initial values needed in the above process re difficult to find, so the use of Genetic Algorithm is proposed to find them. The work explores hoe the evolutionary search suggested in Gas helps in finding out a and b such that f (a) and f (b) have opposite signs. Moreover, once this point has been reached, the use of GA can also help in reaching closer to the roots.

II. GENETIC ALGORITHMS

Genetic Algorithms are adaptive heuristic search algorithms which are based on Charles Darwin theory of the survival of the fittest. The main idea behind these algorithms was to replicate the randomness of the nature .This required that the algorithm proposed should behave like a natural system. GAs emulate the nature to large extent .GAs produce a population in such a way that the trait which is popular ,that is, has higher fitness value is replicated more, as is done by the nature. This is also the fundamental concept behind evolution. So, these algorithms are also referred as the evolutionary algorithms.

A. Steps In Genetic Algorithms

A brief overview of the steps involved in Genetic Algorithms is as follows.

Step 1: A population having P individuals are randomly generated by pseudo random generators whose individuals may represent a feasible solution. This is a representation of solution vector in a solution space and is called initial solution. This ensures the search to be unbiased, as it starts from wide range of points in the solution space.

Step 2: Individual members of the population are evaluated to find the objective function value.

Step 3: In the third step, the objective function is mapped into a fitness function that computes a fitness value for each member of the population. This is followed by the application of GA operators.

1) Reproduction Operator: Reproduction is done on the basis of Rowlett Wheel selection .It selects chromosomes from the initial population and enters them into the mating procedure.

2) Crossover Operator: Crossover Rate (0 to 1) determines the probability of producing a new chromosome form the parents. For example, the strings 10000100 to 11111111 could be crossed over after the third locus in each to produce the two offspring 10011111 to 11100100. The crossover operator roughly mimics biological recombination between two single-chromosomes (haploid) organisms.

3) Mutation Operator: It randomly changes its genetic makeup. This operator randomly flips some of the bits in a chromosome. For example, the string 00000100 might be mutated in its second position to yield 01000100. Mutation can occur at each bit position in a string with some probability, usually very small (e.g., 0.001)

To illustrate the working principle of GA, an example to maximize the optimization problem has been taken

F(x) = 3x2-3x+1; x varies from 0 to 31

Step 1: Coding binary string of length 5

Step 2: A population having 5 cells and number of chromosomes be zero is taken. The values are converted into binary and then to decimal

The corresponding value of F (i) is computed by substituting the corresponding value in F(x).

TABLE I: Illustration of GA

String numbers	Initial Population	x	F(x)	F(i)/Sum	F(i)/F	Actual count
1	01101	13	469	0.14	0.56	1
2	11000	24	1657	0.50	2.00	2
3	01000	8	169	0.05	0.20	0
4	10011	19	1027	0.31	1.23	1
	Summation		3322	1.00	4.06	4
	Maximum		1657	0.31	2.00	1
	Average		831	0.25	1.02	2

Step 3: Crossover

TABLE III: Illustration of Cross Over

Earlier Population	Mate	New Population	X F(x)	F(x)
0 01101 1 11000	2	01100	12 397	397

Steps of the above process

Step 1: The population of size 5 in binary with the result of crossover is generated with the values of f (i).

Step 2: The deviation is calculated with the difference of maximum f (i) and the corresponding values of f(x).

Step 3: The fitness function (delta) is computed.

Step 4: The fitness of each individual is calculated with the help of fitness function.

Step 5: The nearest integer count can be determined by multiplying the fitness by 100.

Step 6: The angle in the Roulette Wheel is calculated by dividing the total sum by 360 degree.

Step 7: A random number is generated and the region in which it lies is found out by the angle in the Roulette Wheel.

An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

1V. PROPOSED WORK

A. BLOCK DIAGRAM

The work is briefly described with the help of a block diagram.

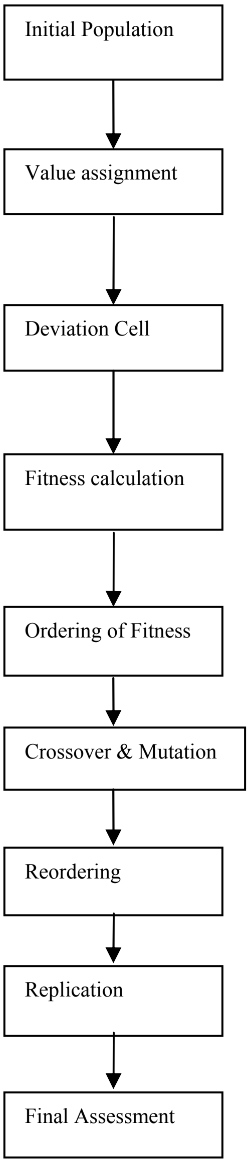


Fig. 1: Block Diagram of the work proposed

B. Explanation Of the Steps

Step 1: Initial Population Generation

To generate each cell of chromosome of 25 cells a random number is generated between 0 and 100 and if its value is greater than 50 the corresponding cell becomes 1 else it becomes 0.

Step 2: Value generation module

An effective value of chromosomes is computed by converting the chromosome into decimal numbers by multiplying each cell by $2^{(i-12)}$, where i is the cell number of the chromosome.

Step 3: To determine F(x)

For all xs, generation, the value of F(x) is computed, where F(x) is the function whose roots are desired.

Step 4: Deviation Calculation

The mod of the values of F(x) is taken this gives an idea of how far the value is from the root. Deviation is denoted by Δ.

Step 5: Fitness value of each x is calculated as follows
Fitness = 1/ $(1+e^{\Delta})$

Step 6: Reordering is done on the basis of fitness values this helps in Rowlett wheel selection.
The chromosomes are arranged according to the fitness value.

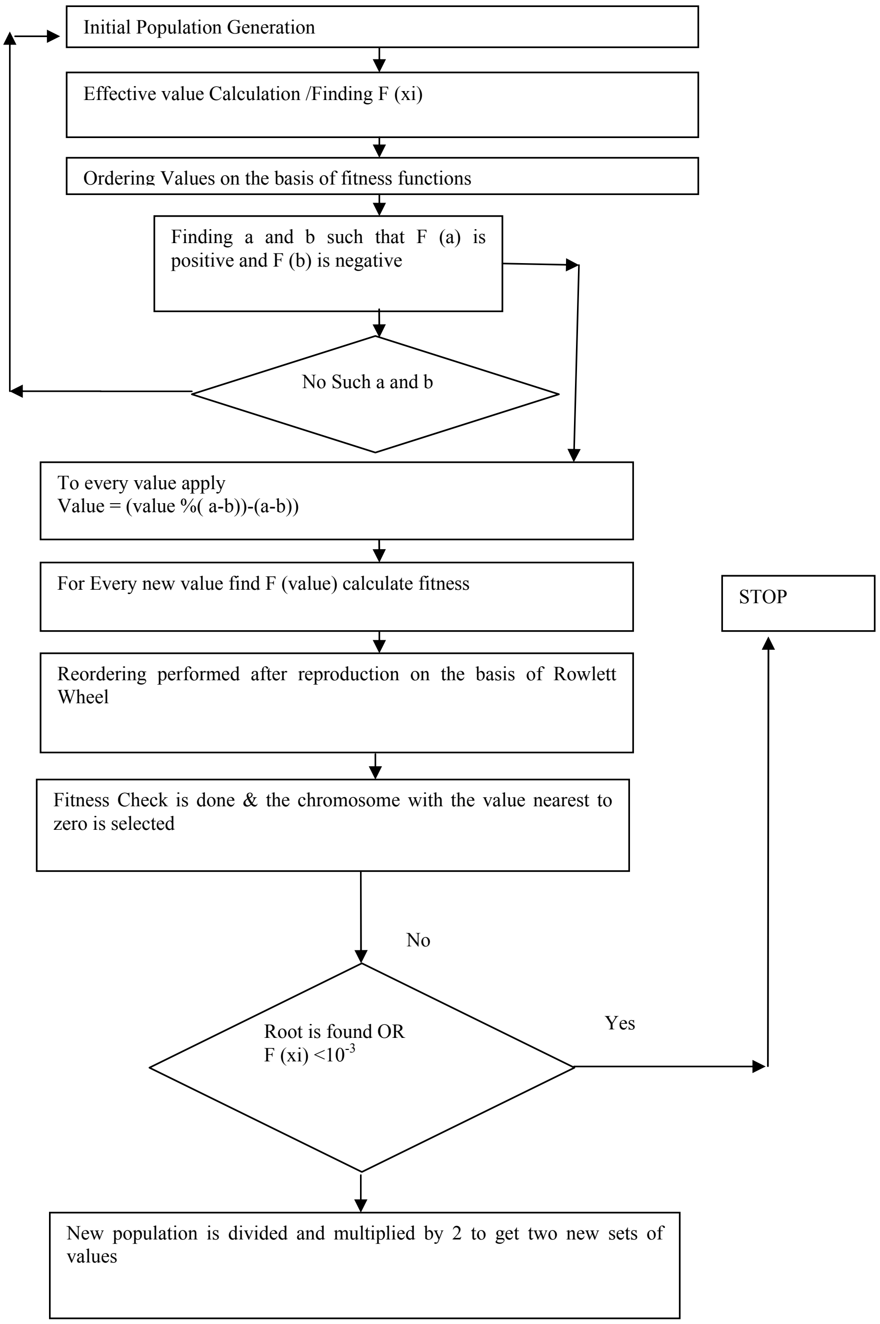
Step 7: Condition checking
The value of F(x) is checked, if it is found to be zero, then roots of the equation are found, else
The fitness values are carried to each chromosome to the top values, since they are more favourable. The lowest values are rejected.

Step 8: Crossover
Crossover is done to generate new set of values in between the a and b such that F (a) is positive and F(b) is negative.

Step 11: Again the fitness is determined of the new population.

Step 12: Reordering is done after the reproduction is completed.

Step 13: The fitness values are checked and the chromosome with the highest fitness value is selected. If the roots are found then values which are nearer to zero are multiplied and divided by 2, Else the process is repeated n times.


Initial Population Generation
Effective value Calculation /Finding F (xi)
Ordering Values on the basis of fitness functions
Finding a and b such that F (a) is positive and F (b) is negative
No Such a and b
To every value apply
Value = (value %(a-b))-(a-b))
For Every new value find F (value) calculate fitness
STOP
Reordering performed after reproduction on the basis of Rowlett Wheel
Fitness Check is done & the chromosome with the value nearest to zero is selected
No
Yes
Root is found OR
F (xi) <10-3
New population is divided and multiplied by 2 to get two new sets of values

V. CONCLUSIONS

A set of 50 chromosomes was taken and converted into values by the above process. The values were substituted into the function and hence, a & b such that f (b) as positive and f(a) as negative have been chosen. After this step, the whole population was converted into the population which has its value between a & b by the following formula written as:-

Value = (Value %(b-a) – (b-a)).

The new population obtained has its value between b & a and thus, the plausibility of root lying between them increases. The new value obtained was put again into function and the function was evaluated. The process stops when we get n such that f (n) is either 0 or less than 10-3.In our case, 500 values were taken and satisfactory results were obtained. In case of the equation, the roots were not found,. Later on it was discovered that those equation did not have any real roots or the roots were too large to be found .The work process can be enhanced by taking larger set of values and can be implemented to find out the roots in handheld devices.

REFERENCES

[1] Melanie Mitchell, “McGraw-Hill Dictionary of Scientific &Technical Terms”, 6E, Copyright @2003 by McGraw-Hill Companies, Inc. “An Introduction to Genetic Algorithms.”

[2] Karen A.Kopecky,”Root Finding Methods”, Eco 613/614, Fall 2007.E.

[3] Dr Conor Brennan,”Iterative method for finding Out Roots”, Course EE317 Computation and Simulation EE317, School Of Electronic Engineering.