International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 8 (2015) pp. 20571-20580

http://www.ripublication.com

Comparision Between Genetic algorithm and Genetic Programming Solving a Quadratic Equatition

Sumit choudhary and Shailee Choudhary

Mtech Scholar, Computers Science Department
Manav Rachna college of Engineering, Faridabad, Haryana
Sumit_choudhary47@yahoo.com
Faculty, Computer science Department
Manav Rachna college of Engineering Faridabad, Haryana
shailee.mrce@mrei.ac.in

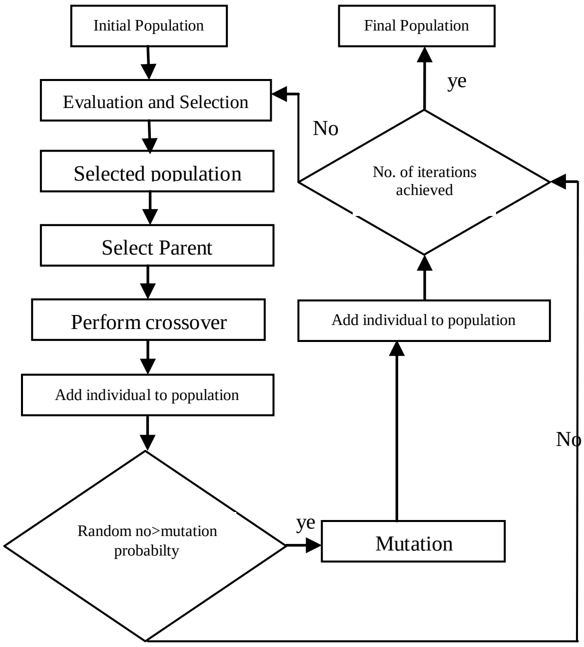
Abstract

This paper gives brief overview of application of genetic algorithm and genetic programming in solving a given mathematical equation. Genetic algorithm deals with optimization of a given equation for a set of input variable in a given domain, while genetic programming deals with symbols and operators. The symbols may be variables or constants while the operators are basic mathematical operators. Both the approaches follow a set same operation namely crossover, selection, mutation etc but the difference lies in the structure with genetic algorithm giving optimized solution for a given equation, while genetic programming along with optimization giving an alternative mathematical expression which would produce same or near about solution. The paper broadly divided into four section. Section I is the introduction the genetic algorithm and genetic programming, section II consisting of Genetic algorithm being used in solving equation section III correspond to Genetic programming based implication, while section IV is Experimentation and section V shows the result obtained

Keywords—Genetic Algorithm; genetic Programming;

I. INTRODUCTION

Genetic algorithm (GA) and Genetic Programming (GP) are special optimization techniques of evolutionary algorithm[1]. Both the approaches start with initialization of some random population (input variables set in case of GA or a tree with operator and variables in case GP). The population initialized is then evaluated based on certain on a fitness function[2] (function to be optimized) and selection is made for the fittest individual (Those with nearest solution to required solution). The individuals selected are then paired up for cross over[3]. There is a chance mutation on any of the member population with a certain probabilistic value. The above operation are inspired from darwins theory of survival of fittest where the population at any instance of time if evaluated on a fitness test which at instance may be climatic condition or level in food chain etc. the one which survives this fitness test reproduce to produce new generation, which is again tested for same or adverse fitness test and the process continues till all the individual in population are fit. The population set which does not lead to a fit offspring moves toward extinction with passage of time. There is also a probabilistic mutation[4] for example modifying organism modifying its part to adapt to given circumstance, but since it is a chance modification, mutation may always not lead to a fit individual.The equation taken for the purpose is $6 - x_1^2 - x_2^2 - x_3^2$.The detailed procedure and representations are covered in further section.



II. GENETIC ALGORITHM APPROACH

Like biological organism that is made up cells , composed of nucleus enclosing the genetic material ‘gene’ in a unit called ‘Chromosomes’, our problem too consist of chromosomes[5], that are input variables to the problem. The Input variable can be represented as a sequence of binary string where each bit corresponds to biological gene. The sequence of genes or bits is called genotype and the value it corresponds to is its phenotype, that is what actually solution it actually leads to. The following are the steps the problem

A. *Population initialisation*

For our given problem the individual X= $\{x_1, x_{2,} x_3\}$ is the encoded using a 8 bits for each $x_1, x_{2,} x_3$ making it total of 24 bits string. For example $x_1 = 32, x_2 = 8, x_3 = 2$ can be represented as 000100000000100000000010, these 24 bit string is called chromosome. The population is initialized by creating randomly such 24 bit string of the required population size.

B. ***Population evaluation***

Bounds define the value between which the values of x_1, x_2 and x_3 lie so we use the formula {(decimal form of above input variable)*(upper bound-lower bound)+lower bound}for x_1, x_2 and x_3 and pass the value to the function we are using for optimization and thus calculate fitness of the individual made from x_1, x_2 and x_3.

C. ***Cross over***

Cross Over is done by randomly selecting a point of crossover, however there may be multipoint crossover we can have multipoint crossover but in our study we used a single point crossover. Selection of parents for crossover[6] is random. The result of cross over is a population which is double the size of original population as each pair of parent produces two Childs. Each chromosome or binary string of input variable undergoes a cross over. For example parent1 x_1 = 00110011 And parent 2 x_1= 11001100 and cross over point is 3 from right would generate a binary strings child1 x_1 = 00110100 child2 x_1 = 11001011 in addition to original parent.

D. ***Mutation***

Mutation unlike cross over occur in any one of the individual of population. Mutation occurs when a randomly generated number is greater than a mutation probability provided, which in our case is 0.05.A mutation is a flip of bit in our case for example x_1 = 00110011 has a mutation at 5^{th} point from right than x_1 = 00100011.Mutation may or may not produce a fitter individual and does not add a single individual to population.

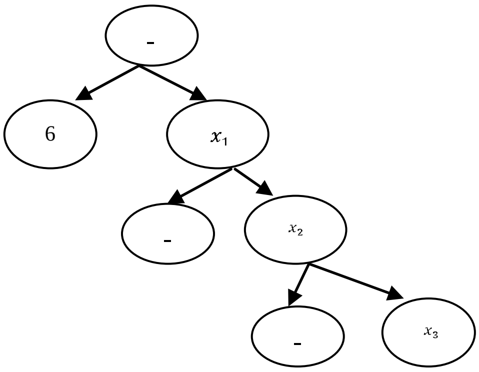
E. ***Selection***

Once the crossover and/or mutation are done each of the individual is evaluated again based on Fitness function. The selection is done to reduce the population to the required sample size population. We have followed a roulette selection[7]method where the fitness function assigns fitness to possible solutions or chromosomes. This fitness level is used to associate a probability of selection with each individual chromosome. Other selection methods are tournament, uniform distribution selection method.

The steps B, C, D, E continues for N no. of iteration.

III. GENETIC PROGRAMMING APPROACH

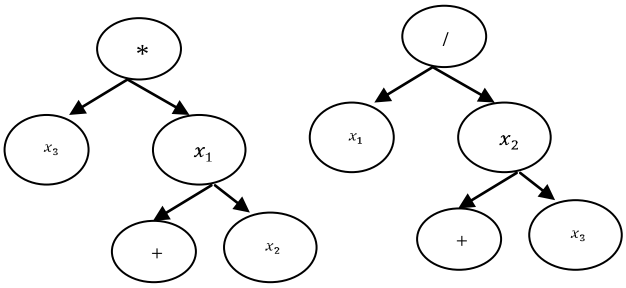
Like Genetic algorithm, chromosome do occur in occur in Genetic Programming too, with a difference that chromosome is a tree[8] consisting of symbols which may be terminals/constants and operators. The genotype is the here is the tree while phenotype is the solution that correspond to the expression generated after parsing the tree. For our example $6 - x_1^2 - x_2^2 - x_3^2$ the tree which could be formed is



The symbols that are taken for our problem are operator set {'+','-','*','/'} and terminals $\{x_1, x_{2,} x_3\}$,thus these are used to generate a tree. The following are the steps in genetic Programming:

A. ***Population Initialisation***

The population is initialized with random generation of tree of depth provided with maximum of 2^{depth} -1 node. The sample individual consists of a tree and fitness value assigned, which is 0 for during initialization. Some of the random individual tree may be

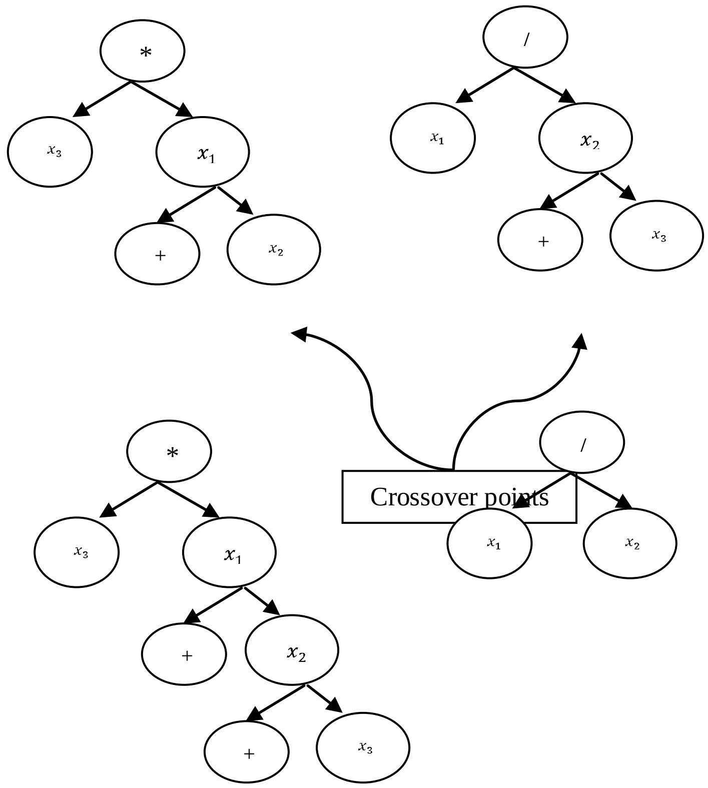


B. ***Population evaluation***

The Population is then evaluated by parsing the tree. During parsing the generates an expression which is evaluated for the given set of input domain which results in the fitness value for each individual of population .For example above tree generates expression $x_3 * (x_1 + x_2)$ and $x_3 * (x_1 + x_2)$.

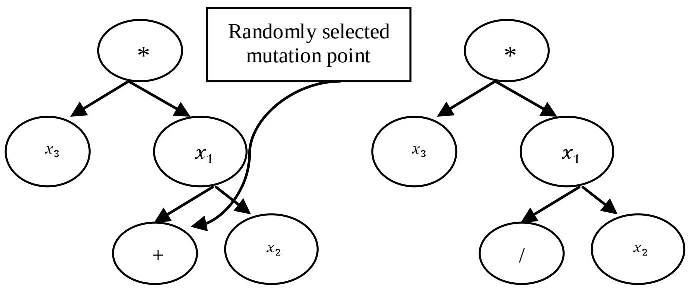
C. ***Cross Over***

Cross over[9] in a tree could a single point or multipoint as case of genetic algorithm, difference lies in the chromosome length, which is variable in case of Genetic Program.



D. ***Mutation***

Mutation is a chance change of a operator/terminal in the individual chromosome. The node modified is replaced with the symbol of similar kind i.e. operator is replaced with a element from operator set and terminal node from a terminal set.



E. ***Selection***

Selection is based on fitness value which is reevaluated once a crossover and/or a mutation is made to the individual. The selection method varied from roulette, tournament, stochastic uniform, uniform. The experimentation performed uses roulette selection where fitness value is associated with a probability factor.

The above steps B, C, D, E are repeated for number for iteration.

IV. EXPERIMENT

The Experimentation was done using MATLAB where Genetic algorithm was coded fully, while genetic Programming used GPols tool of MATLAB. There were certain changes made in the m-files of tool to fit into the requirement to solve the quadratic equation. Bound is being used to set the range of the input variables $x_1 = [-2, 3]$, $x_2 = [-2, 4]$ and $x_3 = [-1, 1]$ for both the approaches.

A. *Genetic Algorithm*

The experimentation started with generation of initial population in a given range and converting in to a binary string of length 24 bit, each 8 bit corresponding to one of the three input variable. The requirement of binary string has importance being complimentary to human genetic layout. Like each gene plays a vital role in human gene sequence, every bit has a value for the position its hold. More over they become important to make it easier genetic operation like cross over and mutation

Once the population is initialized each and every individual is evaluated for the first time. During evaluation the string is decoded to decimal number and getting the actual value of $x_1, x_{2,} and\ x_3$, which are then fed to the quadratic equation $6 - x_1^2 - x_2^2 - x_3^2$ .The solution obtained is fitness value of that individual. The fitness obtained is used to make a selection for the crossover or mutation using the selection operation of GA. During selection we have chosen a roulette wheel method where cumulative sum of fitness and their probability in total fitness is considered, that is the higher the probability the higher the chance of its selection. The selected individual are participate in crossover or mutation as discussed in previous section. The processes of selection, crossover and /or mutation occur iteratively for the required number of generation.

B. *Genetic Programming*

Gpols tool is a Genetic programming tool with some inbuilt m files for handling tree which is the basic requirement to perform genetic programming. However some m-files were modified in order to evaluate a quadratic equation and to produce MSE. Mean square error (MSE) is used as a approximation for correctness of the result produced. MSE is average of the square of the difference between the desired response and the actual system output (the error).The lesser the value of MSE the more the result is appropriate.

The experiment here started with generation of random value set for X $(x_1, x_{2,} x_3)$.The values set produced for $X(x_1, x_{2,} x_3)$ are in range as discussed before, which are input to the equation $6 - x_1^2 - x_2^2 - x_3^2$ thus giving a set of desired output.

Once done with it operator set {'+','-','*','/'} and terminal set $\{x_1, x_{2,} x_3\}$ are defined. The above set is used to generate tree for initial population. The tree produced has above symbols as node in some random manner, with mean square error and fitness value equal to 0.The evaluation starts with parsing of tree and generating an expression which is evaluated. The evaluated expression is fed with the same set of input variable set $(x_1, x_{2,} x_3)$ produced at the starting of experimentation and actual result is produced.

MSE is calculated for each individual of the population. Fitness value is calculated on the basis of correlation coefficients[10], which demonstrate how closely two values are related. MATLAB provide a inbuilt function **corrcoef(actual value,desired value)**,whose value lies between 0 and 1 where 1 says they match exactly and 0 vice versa.

Once we have fitness value for each individual in the population, section(roulette in our case) can be applied to get the best individual for cross over and/or mutation. Crossover and mutation phenomenon occurs in accordance to concept discussed earlier. The process of selection, cross over and/or mutation repeats

V. RESULTS

The following is the table showing various parameters used, which are common in both our approach

Table I

Entity	**Value**
population size	10
Selection Mode	Roulette
No. of iteration	20
Cross over	Single point
Mutation rate	0.05

The following is the table for best values of X $(x_1, x_{2,} x_3)$ along with the fitness function

Table II

Iteration	x_1	x_2	x_3	Fitness Value
1.	0.447059	0.196078	0.003922	4.282061
2.	0.290196	0.215686	0.019608	4.277201
3.	0.447059	0.196078	0.082353	4.56872
4.	0.321569	0.196078	0.082353	4.470296
5.	0.447059	0.211765	0.082353	4.714879
6.	0.439216	0.133333	0.333333	4.410442
7.	0.478431	0.211765	0.082353	4.616455
8.	0.415686	0.211765	0.090196	4.79005
9.	0.415686	0.211765	0.082353	4.764091
10.	0.415686	0.211765	0.082353	4.764091
11.	0.415686	0.211765	0.090196	4.79005
12.	0.431373	0.243137	0.078431	4.971642
13.	0.431373	0.243137	0.090196	5.010765

14.	0.415686	0.211765	0.082353	4.764091
15.	0.415686	0.243137	0.333333	5.589865
16.	0.415686	0.243137	0.333333	5.589865
17.	0.415686	0.243137	0.333333	5.589865
18.	0.447059	0.227451	0.333333	5.429927
19.	0.384314	0.243137	0.341176	5.600077
20.	0.384314	0.243137	0.341176	5.600077

The above result is being demonstrated with the help of graph. The graph consists of fitness value that is the obtained result of equation bieng and iterations. It can be observed that with the course of time the fitness value tend to stabilize, that is a point after which there is no change is optimized value. This point is called saturation and population is fittest at his point and after that point till fitness function to be maximized is not changed.

The following are the graphs generated using genetic programming.

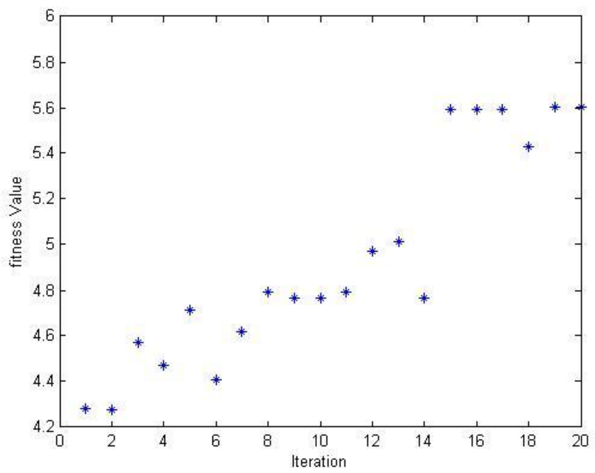


Figure I

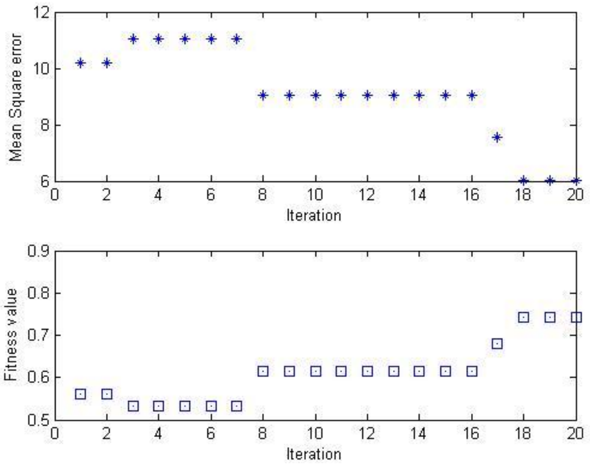


Figure II

It is clearly visible from the graph as with each iteration correlation between the desired value and actual values increases which demonstrates the fit individual with time. It may also be noticed that mean square error which is the measure of closeness to the result is just compliment of fitness value/iteration graph, which make it clear that lesser the MSE value ,the more optimized the result is

CONCLUSION

It can be concluded that the Genetic algorithm is an optimization process which tend to maximize or minimize the solution to the quadratic equation, while GP works the same goal with extra functionality of generating an quadratic equation which could be substituted for the given equation to be optimized. It may also be nice to know that if dataset is available to make a correlation with the solutions of sample equations produced as individual in Genetic programming, a standard equation can be derived for the same dataset.

REFERENCES

[1] J. R. Koza, "Genetic Programming: On the Programming of Computer by Means of Natural Selection."MIT Press:Cambridge, MA, 1992.

[2] D. E. Goldberg, "Genetic algorithms in search, Optimization and Machine Learning," Addison Wesley, 1989. D. E. Goldberg, "Genetic algorithms in search, Optimization and

[3] J. Koza , "Genetic Programming II," MIT Press, Cambridge, MA, 1994.

[4] D. E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning,Addison Wesley, 1989

[5] Generic Chromosome Representation and Evaluation for Genetic Algorithms Kristian Guillaumier Department of Computer Science and AI,University of Malta

[6] R. Thamilselvan and P.Balasubramanie, “A Genetic Algorithm with a Tabu Search (GTA) for Travelling Salesman Problem”, International Journal of Recent Trends in Engineering, Issue I, Vol I, pp 607 – 610, 2009.

[7] Khalid Jebari, Mohammed Madiafi, Selection Methods for Genetic Algorithms Int. J. Emerg. Sci., 3(4), 333-344, December 2013 ISSN: 2222-4254

[8] General Schema Theory for Genetic Programming with Subtree-Swapping Crossoverby Riccardo Poli - *In Genetic Programming, Proceedings of EuroGP 2001, LNCS , 2001*

[9] Patrik D’haeseleer. Context preserving crossover in genetic programming. In Proceedings of the 1994 IEEE World Congress on Computational Intelligence, volume 1, pages 256–261, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[10] Enrique F Schisterman, Kirsten B Moysich, Lucinda J England and Malla Rao Estimation of the correlation coefficient using the Bayesian Approach and its applications for epidemiologic research, *BMC Medical Research Methodology* 2003, **3**:5 doi:10.1186/1471-2288-3-5.