Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DETERMINING MAXIMUM/MINIMUM VALUES FOR TWO-DIMENTIONAL MATHMATICLE FUNCTIONS USING RANDOM CREOSSOVER TECHNIQUES

SHIHADEH ALQRAINY·
Department of Software Engineering, Albalqa Applied University.
E-mail: sh_alqrainy@yahoo.com

ABSTRACT

This paper presents a solution to determining the maximum/minimum values for two-dimensional mathematical functions using the three most popular crossover techniques (Single point, Two point and cut and splice) randomly in genetic algorithm. A set of experiments were ran over 20 complex functions, the obtained results show that using random crossover techniques tends to be worst compared with traditional genetic algorithm that uses specific crossover method such as single point crossover.

Keywords: *Genetic Algorithm, Machine Learning, Heuristic Search. Crossover Methods*

1. INTRODUCTION

Genetic algorithms (GA) are computer based optimization methods that uses the Darwinian evolution of nature as a model [1]. A solution generated by genetic algorithm is called a chromosome, while collection of chromosome is referred as a population. A chromosome is composed from genes and its value can be either numerical, binary, symbols or characters depending on the problem want to be solved. These chromosomes will undergo a process called fitness function to measure the suitability of solution generated by GA with problem.

Some chromosomes in population will mate through process called crossover thus producing new chromosomes named offspring which its genes composition are the combination of their parent.

In a generation, a few chromosomes will also mutation in their gene. The number of chromosomes which will undergo crossover and mutation is controlled by crossover rate and mutation rate value. Chromosome in the population that will maintain for the next generation will be selected based on Darwinian evolution rule, the chromosome which has higher fitness value will have greater probability of being selected again in the next generation. After several generations, the chromosome value will converge to a certain value which is the best solution for the problem [2].

The next section presents the paper objective and the research contribution. The related works are described in section 3. The proposed algorithm is explained in section 4. Section 5 presented the analysis of the proposed algorithm. The experimental setup, comparative study and merits and demerits of the proposed algorithm are explained in section 6. Finally the paper conclusion is shown in section 7.

2. PAPER OBJECTIVE AND RESEARCH CONTRIBUTION

The current literatures in the field of using crossover techniques shows that very few attempts were made to handle the mathematical functions, these attempts were used one of the three most popular crossover techniques (Single point, two point and cut and splice) separately to determine the maximum/minimum values for only one-dimensional mathematical functions. However, section 3 described these attempts in more details.

The research contribution and the objective of this paper is to examine the three most popular crossover techniques together (Single point, Two point and cut and splice) randomly to present a solution to determining the maximum and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

minimum values for two-dimensional mathematical functions.

Furthermore, the ultimate goal and the main contribution of this paper represented in the proposed algorithm which described in section 4; this algorithm has been implemented based on selecting the above three techniques (Single point, Two point and cut and splice) together and randomly switched between these techniques during the iteration process to determining the maximum/ minimum values for two-dimensional mathematical functions.

3. RELATED WORK

Denny Hermawanto [3] used genetic algorithms to solve simple mathematical equality problem. The paper objective is minimizing the value of function f(x) where f (x) = ((a + 2b + 3c + 4d) - 30). The author described step by step numerical computation of genetic algorithm for solving the above function using single point crossover technique.
Tom V. Mathew [4] presented a simple two variable function $f(x_1,x_2)=(x_1^2+x_2-11)+(x_1+x_2^2-7)^2$ to illustrate how genetic algorithm works. The author described the detailed steps and solution obtained to solve the above minimization problem using single point crossover technique.

Pratibha Bajpai and Dr. Manoj Kumar [5] discussed an example that shows how to find the global minimum of Rastrigin's function in genetic algorithm: $Ras(x)=20+x_1^2+x_2^2-10(\cos 2\pi x_1+\cos 2\pi x_2)$ using single point crossover technique for two independent variables. The authors claimed that the final value of the fitness function (Rastrigin's function) when the algorithm terminated is 0.05531602101322264 (objective function value). They mentioned that the value obtained is very close to the actual minimum value of Rastrigin's function, which is 0.

4. THE PROPOSED ALGORITHM

The proposed algorithm using random crossover techniques can be summarized as follows:

Input: No. of generations, mathematical function, and function scope of every dimension
Output: Numbers that achieves the biggest fitness value.

Begin

1. *Let average of fitness value equal 0.*
2. *Create Initial generation **P (0)**. Let **g = 0**.*
3. *For each individual solution **i** ∈ **P(g)** , evaluate its fitness **f (i)** .*
4. *The algorithm automatically chooses one of the crossover methods [Single-Point, Two-Point, or cut splice] randomly.*
5. *Create generation **P (g+1)** by reproduction and crossover.*
6. *Calculate average of fitness value in every generation **g**.*
7. *Let **g = g + 1**, unless g equals the maximum number of generations OR the average of fitness value is steady, return to Step 3.*

End

The fitness function is defined over the genetic representation and measures the quality of the represented solution. The fitness function is always problem dependent.

Traditionally, solutions are represented in binary strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals. Genetic algorithms form one of the best ways to solve a problem for which a little is known. A genetic algorithm is able to create a high quality solution. When the average of fitness values of every generation is stable, that means in the proposed algorithm this is the best quality solution.

To explain how the proposed algorithm works, consider the following example:

$$f(x,y) = x^3 + y^2 + 20$$

where x belongs to the following range :
[1 , 100] and y [1 , 50].

Let's the number of generation to stop = 25

A population of candidate solutions includes 5000 solutions [100 solutions for x * 50 solutions for y] . a proportion of the existing population is selected to breed a new generation, first generation contains 4 random individual solutions selected from population .

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1. Selection

Table 1: Sample of (x,y) individual solutions obtained from first generation in population

Solutions (x,y)	Binary Encoding	Fitness Value
(39,12)	00100111.001100	59483
(42,9)	00101010.001001	74189
(49,2)	00110001.000010	117673
(72,35)	10010000.100011	374493

Every solution has a binary encoding contains two parts separated by **.** as shown in table 1. The first part present the value of x while the second present the value of y. The value that obtained from applying x and y in original mathematical function is called fitness value.

Let's assume we need to find the value of (x,y) that if applied in original mathematical function could produces the maximum value. Therefore, individual solutions are selected through a fitness-based process; highest fitness value is fitter than lowest

2. Crossover

Crossover is a process of generating new children, many crossover techniques exist for organisms that chosen by randomly and let's assume that one point crossover is selected.

Parent 1 (72,35) 10 | 010000.10 | 0011 374493
Parent 2 (49,2) 00 | 110001.00 | 0010 117673

Child 1 (81,34) 1010001.100010 532617
Child 2 (40,3) 0101000.000011 64029

As mentioned before, individual solutions are selected through a fitness-based process, the two parents for next generation are Parent 1 and Child 1. Let's assume that two-point crossover is selected.

Parent 1 (75,35) 10 | 010 | 11.10 | 00 | 11 374493
Parent 2 (81,34) 10 | 100 | 01.10 | 00 | 10 532617

Child 1 (73,34) 1001001.100010 390193
Child 2 (81,35) 1010001.100011 532686

Reproduce the parents until the average of fitness value of every generation leads to stable mode then the mutation is needed by flipping on gene (bit) and re-execute the previous steps until reach number of generations or the average of fitness value of every generation leads to stable mode. Figure 1 shows the snapshot of the implemented system.

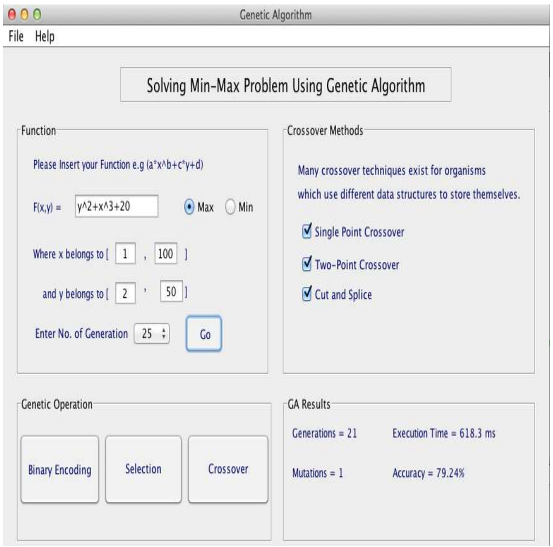

Figure 1: Snapshot Of The Implemented System

5. ANALYSIS OF THE PROPOSED ALGORITHM

The three cases of the proposed algorithm (Best, Average, and Worst) are calculated based on the experiments; these cases are shown below:

Case 1: **Best Case**

The best case of GA is getting optimum solution in first generation and when next generation is attained, the algorithm stops because it will be in steady mode. So, time complexity for best case is **Ω (1)**.

Case 2: **Average Case**

The average case has been calculated technically and theoretically. The results of the experiments show that the average case for **n*m** inputs is 1.397 measured by seconds; the proposed algorithm tends to be like **Θ ($\sqrt{nm}$)**, where n the size of population of one dimension and m to another dimension

Case 3: **Worst Case**

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Unfortunately, the proposed algorithm in worst case acts as brute-force algorithm. It will visit all populations of two dimensions **m** and **n**. So, Time complexity for worst case is **O (nm)**

6. EXPERIMENTAL SETUP AND COMPARTIVE STUDY

The proposed algorithm has been tested in machine has Mac OS Mountain Lion 8G RAM and 2.3GHz CPU. More than 20 different complex mathematical functions have been ran. The proposed algorithm achieves 81% accuracy while when we tested the same functions using only single-point crossover technique, the system achieved 86% accuracy. Table 2 shows the accuracy of the tested functions using the above three most popular crossover techniques.

Table 2: The Accuracy Of The Tested Functions

Mathematical Functions	Accuracy in Percentage	Generations
Polynomial Functions	82.63	21.62
Trigonometric Functions	79.25	14.9
Logarithmic Functions	81.24	17.2

The summary of the accuracy and the number of generation are shown in Figure 2 and Figure 3 respectively.

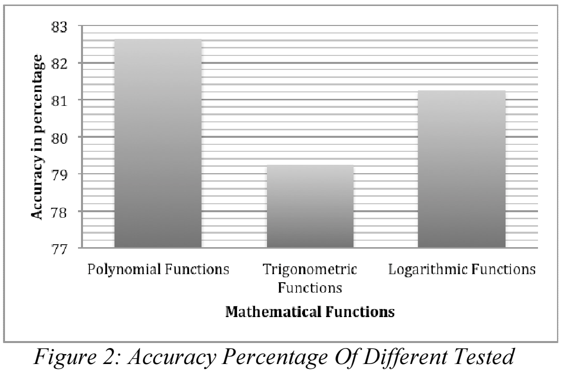

Figure 2: Accuracy Percentage Of Different Tested Functions

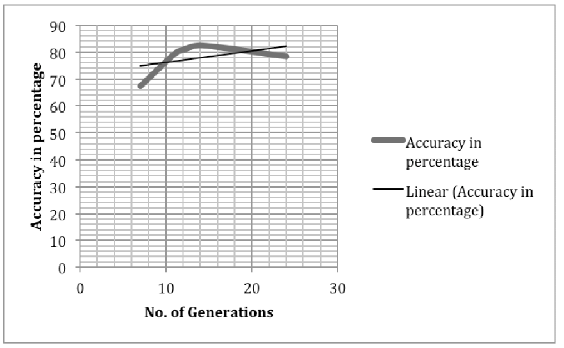

Figure 3: No Of Generations VS Accuracy Of Tested Functions

Regarding the number of generation versus input, the proposed algorithm creates more generation than the algorithm that uses single point crossover as shown in Figure 4. This means that the proposed algorithm will effect on time complexity.

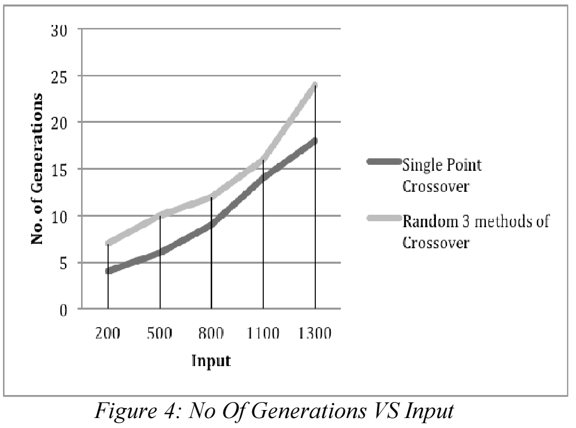

Figure 4: No Of Generations VS Input

This change of time complexity is covariant, due to the fact that when the number of generation increases, the time complexity increases too. Unfortunately, the proposed growths faster than the algorithms have single point crossover method. Thus, the accuracy of algorithm that uses random crossover method is low compared with the algorithm that uses single point crossover method as shown in Figure 5.

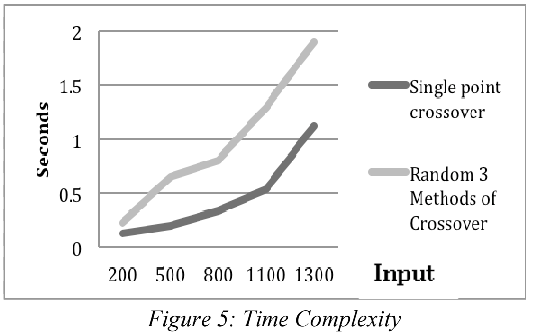

Figure 5: Time Complexity

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The improvement of the proposed algorithm comes from selecting the methods of crossover randomly. Thus, the time complexity is amortized. The proposed algorithm solves the polynomial functions with efficient way compared to trigonometric and logarithmic functions.

The traditional crossover method (Single point method) excels on the proposed algorithm, the reason stems from choosing cat-splice method which produces out of the range chromosomes. That's why the number of generation in the proposed algorithm is 1/3 times of number of generation in single point crossover method.

More generation leads to more executions, which are make the time complexity of the implemented system is double of the execution times of single point crossover method which has been used in most work described in section 3.

Furthermore, the accuracy of the implemented system described in this paper depends on the type of mathematical functions, where the polynomial functions achieved the higher accuracy due to the trends of these functions which is clear and can be easily expected while the case is different in other types of function (trigonometric and logarithmic) since these functions close to signal.

Unfortunately, since the current literatures shows that very few attempts were made to handle the mathematical functions, these attempts were used one of the three crossover techniques (Single point, two point and cut and splice) separately to determine the maximum/minimum values for only one-dimensional mathematical functions as described in section 3, this lack in literatures make the comparative study with other case is difficult.

7. CONCLUSION

The explanation of the Genetic Algorithm is simple intuitive, because it relates to every-day life in so many ways, and results are readily interpreted because the algorithm deals only with solutions that are implementable. These two points make Genetic Algorithms very attractive. Furthermore, Genetic Algorithms are robust, do not require gradient calculations, and remarkably efficient on very difficult problems. In genetic algorithm, all you need to know is what you need the solution to be able to do well, and a genetic algorithm will be able to create a high quality solution.

This paper examined the three most popular crossover techniques together (Single point, Two point and cut and splice) randomly to present a solution to determining the maximum and minimum values for two-dimensional mathematical functions through the proposed algorithm described in section 4. The current literatures were made to determine the maximum/minimum values for only one-dimensional mathematical functions not two-dimensional.

This paper illustrates that using random crossover techniques randomly tends to be worst compared with traditional genetic algorithm that uses specific crossover method such as single point crossover.

REFRENCES:

[1] Timo Mantere, "A Min-Max Genetic Algorithm with Alternating Multiple Sorting for Solving Constrained Problems", *Proceedings of the Ninth Scandinavian Conference on Artificial Intelligence.* pp.199–202Portugal, 2006.

[2] Mitsuo Gen, Runwei Cheng, "Genetic Algorithm and Engineering Design", *John Wiley*, 1997.

[3] Denny Hermawanto, "Genetic Algorithm for Solving Simple Mathematical Equality Problem" http://arxiv.org/ftp/arxiv/papers/1308/1308.4675.pdf

[4] Tom V. Mathew, "Genetic Algorithm". http://www.civil.iitb.ac.in/tvm/2701_dga/2701-ga-notes/gadoc/gadoc.html

[5] Pratibha Bajpai and Dr. Manoj Kumar, "Genetic Algorithm – an Approach to Solve Global Optimization Problems", *Indian Journal of Computer Science and Engineering.* Vol 1 No 3 199-206. 2010.