IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume4, Issue-8 E-ISSN: 2347-2693

A Firefly Algorithm Based Approach for Automated Generation and Optimization of Test Cases

Rajesh Kumar Sahoo[1*], Durga Prasad Mohapatra[2], Manas Ranjan Patra[3]

[1]Department of CSE,ABIT,CUTTACK,Orissa,India
[2]Department of CSE, NIT, ROURKELA, Orissa, India
[3] Department of CSE, Berhampur University, Berhampur, Orissa

Available online at: www.ijcseonline.org

Received: 22/Jun/2016 Revised: 10/Jul/2016 Accepted: 16/Aug/2016 Published: 31/Aug/2016

Abstract- Software testing requires functional and non functional test cases with the values of test data.Automated testing are a method to generate the test cases with test data automatically. Optimality of test case is required for fastest data generation. Test case optimization through search based techniques is used to optimize and generate optimal test cases from the set of data values. Firefly Algorithm (FA) is a bio-inspired, evolutionary, meta-heuristic algorithm based on mating or flashing behavior of fireflies. In this paper the role of Firefly meta-heuristic search technique which is analyzed to generate and optimize random test cases with test data by applying in a case study, i.e., a withdrawal method in Bank ATM and it is observed that this algorithm is able to generate suitable automated test cases as well as test data. In this case the test case generation is very efficient and effective. This paper further, gives the brief details about the Firefly method which is used for test case generation and optimization.

Keyword:-Software testing, test data generation, firefly algorithm, test case optimization.

1. INTRODUCTION

Software Testing is a technique to analyze the software by comparing the existing and evaluate the feature of the software with desired criteria. Software testing monitors the process which involves for the development of software. Testing software is used to verify validate and detection of error. In software development life cycle the testing takes around 60% of cost and time. Test case generation is a method to identify test data and satisfy the software testing criteria. Software testing is the primary phase, which is performed during software development and it is carried by a sequence of instructions of test inputs followed by expected output. Generation of test cases is based on the requirements detailing and ignores execution aspects of the system completely. The proposed approach focuses on redundancy issues and challenges of optimized test cases. It uses Firefly optimization algorithm to optimize the random test cases. Moreover, this proposed method inspires the developers to generate random test cases to improve the design quality. This paper is intended to present the result of the outcome of firefly algorithm (FA) to find optimal solution in the software construct. Optimization is a technique to generate the best solution under given circumstances. Generally optimization is applied to maximize or minimize the value of a fitness function, it may be local optimum or global optimum. In this paper, evolutionary Firefly algorithm is discussed. Firefly Algorithm is a meta-heuristic algorithm that was conceptualized using mating or flashing behavior of fireflies to get the best result. During mating, the fireflies are attracted to each other without considering their sex. Better the light intensity of fireflies, the better is the solution.

This paper is structured as follows, section 2 discusses about the basic concepts of automated test case generation and basics of Firefly algorithm. Section 3 represents the related work. Section 4 describes the proposed work, methodology and working principle of the proposed system. Section 5,6 explains about the simulation results, discussion and future scope of the proposed system. Finally section 7 concludes the paper.

2. BASIC CONCEPTS

2.1AUTOMATED TEST CASE GENERATION

Automation testing method is used to increase the efficiency and coverage of software testing. It also helps to improve the quality of software. Manual generation of test case takes lot of time and effort. Automatic generation of test case can be used to take the system in a safe state through desired test data. Automated test cases grouped into to form test suites which gives better consistency of the software. A test case is a technique to specify the input

values, expected outputs and preconditions for test execution.

2.2Overview of firefly algorithm

The firefly algorithm is a nature inspired algorithm which inspired by the flashing behavior of fireflies in continuous domain. This algorithm was developed by Xin She yang in 2007[17,19,20].It is a meta heuristic search based algorithm used to generate good solution by taking the current good solution and exploit it through new solution. According to this algorithm a bunch of random fireflies are used for specific problem domain. Attractions of two fire flies are gender independent. Every firefly is having light intensity. Fireflies are moving towards brighter fireflies.

3. LITERATURE SURVEY

Srivastava et. al[8] focused the importance and demand of structural testing in software testing for code-based criteria which generate all the paths possible in a Control Flow Graph required for Path Testing by using Cuckoo Search algorithm. According to Iqbal et. al [3] test case generation problem through multi objective optimization which are done simultaneously. In this case two objectives like Path Priority and Oracle Cost are used to generate the test cases which reduce overall efforts of testing. Multi Objective Firefly Algorithm (MOFA) is used to solve such kind of problems. Sudhir et. al[13] discussed how to reduce regression testing cost by scheduling the test cases through test case prioritization techniques with objective function which is maximized. The prioritizations of test cases are discussed with average percentage of faults Detection metric. Srivatsava et.al.[7] described the Firefly Algorithm which is a meta heuristic technique for the generation of optimal paths. The algorithm is modified by the different objective function with guidance matrix for traversal of path in the graph. According to X.S.Yang[16] the firefly algorithm is implemented to generate the optimal solutions through unconstrained test functions. It also explains the objective function and determines the brightness of fireflies. Ausiello et.al[1] described the swarm intelligence optimization problems having features like self-organization, derivative free and easier implementation which overcome the limitations of conventional methods. Xin-She Yang et.al[15] explained the optimization problems which belongs to a class of NP hard problem and the main objective is to find the minimize or maximize of objective function with number of variables [9]. Panthi et al.[14] described an approach which generates the test sequences using path coverage, node coverage and edge coverage automatically by analyzing the triangle problem. Malhotra et al.[9] focused on two aspects which generates adequate test cases and the test cases are generated through an application of genetic algorithm by integrating the mutation approach. Tahbildar et al.[5] introduced a hybrid approach for generation of automated test data by using static and dynamic method using the symbolic and actual value. Suresh et al.[18] represented that genetic algorithm(GA) is used to generate the test data automatically using basis path. According to this paper indicates that GA is more effective and efficient to generate the test data automatically. B.Korel [2] focused on the program to generate the software test data by using path coverage criteria with different possible test case values. Ojha et al.[6] explained how course time table is optimized by firefly algorithm.

4. PROPOSED SYSTEM

This paper proposed a methodology for generation of test cases for withdrawal method of an ATM machine and test cases are generated and optimized by firefly algorithm(FA).This method is used for evaluating the effectiveness and efficiency to generate the test cases to be maximized.

4.1 Necessity of Proposed System:

The proposed system is intended to generate an automatic and optimized test case with existing approaches of Firefly algorithm. Optimized test cases may not be helpful for the testing process. It may be required to differentiate among the various test cases. First of all the system may be initialized with firefly positions. Each firefly maintains its own prevailing location on the basis of which the test cases may be generated. This paper also finds out the effectiveness of the proposed approach in case of number of test case or test data generations.

4.2 Firefly Algorithm

The Firefly Algorithm (FA) is a population based stochastic and bio-inspired heuristic method. It is derived and motivated by the flashing or mating behavior of fireflies. The light intensity of firefly with a distance from the source of light which accepts inverse square law when light intensity is decreases the source of light increases. So intensity of light is directly proportional to brightness of fireflies. Light intensity of fireflies is calculated through the objective function of a particular problem. The position of all fireflies represents a possible set of solutions and their light intensities represent corresponding fitness values or quality of all solutions.

There are three idealized rules of firefly algorithm:

1. Fireflies are unisex in nature i.e. the firefly is attracted to other fireflies through their sex.
2. Brightness and attractiveness are proportional to each other that mean less brighter firefly is

attracted to brighter firefly. When the distance increases between fireflies, attractiveness and brightness decreases.

3. Brightness of firefly is analysed through the functional value of fitness function.

The flow chart of Firefly Algorithm is depicted in Figure 1[12]

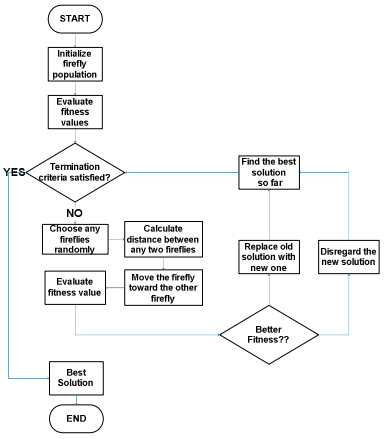

Figure1: Flow chart of Firefly Algorithm (FA)

4.3 Pseudo code of FIREFLY ALGORITHM (Test case/test data generation)

Initialize the number of generation.
Initialize fireflies' population size.
Generate initial solutions
Evaluate the initial Light Intensities value 'I'
$I=1/(abs((net\_bal-wd\_amt)-min\_bal)+\varepsilon)^2$
where ε varies from 0.1 to 0.9
Find the initial best solution
Define light absorbtion coefficient
$\gamma=1$
Initialize different Firefly parameters i.e.,
$\beta_0=1$
$\alpha=0.2$
While generation<500 do
For i=1 to n fireflies
For j=1 to n fireflies
If Intensity(j)>Intensity(i)
Evaluate distance r given by
$r=(x(i)-x(j))^2/(min\_val)^2$
Update new solution by the help of following equation
$xnew=x(i)+\beta_0*e^{-\gamma r}*(x(j)-x(i)+min\_val*\alpha*(rand-0.5)$
Check the boundary conditions
Evaluate the new fitness function
End if
End for
End for
Update current best solution
Generation (t) = Generation(t)+1
End While
Select the solution with best fitness function value

The distance between any two fireflies can be calculated as

$$r=(x(i)-x(j))^2/vmin \quad (1)$$

Where x(j)= candidate solution at j^{th} position
x(i)= candidate solution at i^{th} position
vmin= minimum balance

The movement of a firefly toward another is calculated as follows :

$$x1=x(i)+round((beta0*(exp(-(gamma*r)))*(x(j)-x(i))+vmin*alpha*(rand-0.5)) \quad (2)$$

where x1=new solution
rand= a random number in the range of [0,1]
beta0 = attractiveness at r=0
gamma and alpha = firefly parameters
where betao=1
gamma=1
alpha=0.2

4.4 Methodology:

The variables in the program code such as net-amt gives idea about the balance amount which is available in account of a customer, wd-amt explains the withdrawal amount, and min-bal represents the least possible balance in a customer's account after the withdrawal operation. Test-data and suc-bal represents the valid transactions with the data set which is used during the ATM withdrawal operation. In this case, it is intended to generate and optimized the automated test cases from firefly algorithm.

Initially each firefly is initialized with current position. Firefly will search for optimal solution. It will keep track of solution in the population and upgrade its position or location .Optimal solution is used to maximize the mathematical function $f(x)$ which may be implemented in Firefly Algorithm using MATLAB-7.0.as shown Table-1. In this table the Fireflies' movement to generate the best solution with the functional value of fitness function is primarily focused.

Table 1(Fitness Function Value for each test case or test data)

Iteration Number	Test cases/test datas	Fitness Function Value
1	2800	1.3516e-009
5	4000	1.4793e-009
10	5100	1.6129e-009

15	6300	1.7803e-009
20	7800	2.029e-009
25	9100	2.2893e-009
30	10300	2.5767e-009
40	12800	3.3802e-009
50	15500	4.7561e-009
70	20900	1.207e-008
80	23500	2.3668e-008
90	26200	6.9247e-008
102	29000	9.997e-007
125	29000	9.997e-007
150	29000	9.997e-007
160	29000	9.997e-007
180	29000	9.998e-007
200	29000	9.998e-007
250	29000	9.998e-007
300	29000	9.998e-007

In this case, 20 numbers of sample test cases are considered. The function value depends upon the parametric values of the input variables. It was found that the solution reaches its optimum value after 100 iterations

5. SIMULATION RESULTS

The proposed approach is used to generate the test cases with test data for bank ATM operation through firefly algorithm (FA). The figure 2 shows the relation between variables like test data and functional value of fitness function along one pair of axis represented in table-1.

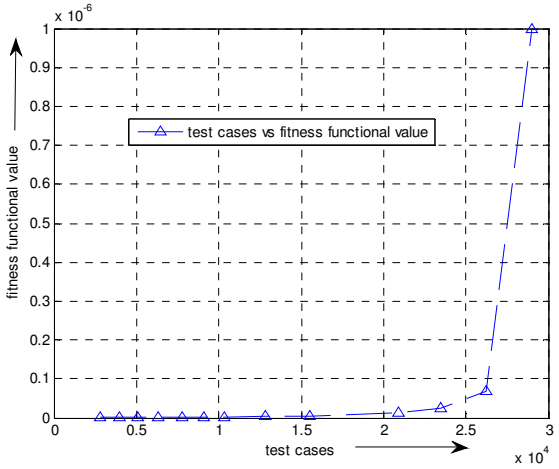

Figure 2**:** Graphical representation of test data and fitness function value for table-1

The proposed approach generates the test cases or test data for ATM's withdrawal operation using firefly algorithm (FA). Table-2 shows the value range of fitness functional value with test cases/test data. The functional value of fitness function is represented in terms of percentage.

Table 2: %of test cases/test data in terms of maximum fitness value

FITNESS VALUE RANGE	% OF TEST DATA
$0 \leq f(x) < 0.3$	45
$0.3 \leq f(x) < 0.7$	15
$0.7 \leq f(x) < 1.0$	40

According to table-2 about 40% of test data is having higher value of the fitness function f(x) which lies in between 0.7 and 1.0.The figure-2 represents the graphical view of the fitness functional value and test cases/test data in percentage.

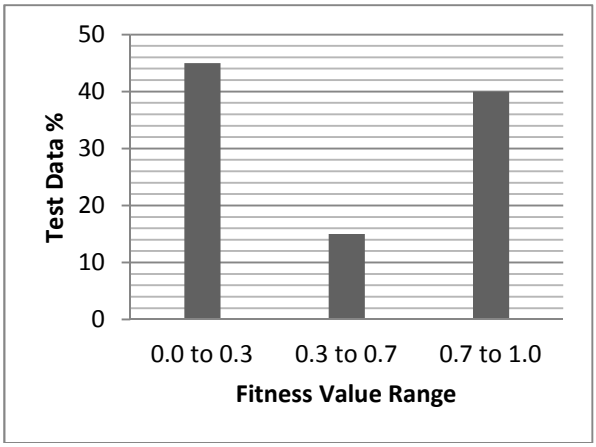

Figure 2: Graphical representation of % test cases/test data and fitness value range for table-2

6. DISCUSSION AND FUTURE SCOPE

While considering the mathematical function $fx=1/(abs(net\_bal-wd\_amt)+\varepsilon)^2$,where ε varies from 0.1 to 0.9, along with the each firefly initialized with current position and intensity values.It has been found that solution of optimality keeps track of best firefly position and updates its position according to the firefly . By considering some sample test cases it has been observed that the functional value of each firefly depends upon best solution or best firefly position so far. The future approach to this work could enhance the automated generation of test cases or test data for large programs .The different parameters could be added to the firefly technique which enhance the efficiency and producing the optimized test cases. Another perspective area could be the randomly generated test case or test data by using various paths according to the control flow graph (CFG).Test Cases can be generated by using various kinds of meta heuristic algorithms like BCO,GA and combination of different metaheuristic algorithms. The test cases or test data which is generated by firefly algorithm is compared with PSO, bat, harmony search, cuckoo search and it was found that firefly algorithm produces optimal result in very less time and with more accuracy. The future scope would be the test case generation and by using the fitness function

gives better code coverage, statement coverage and maximize path coverage by using hybrid Firefly Search algorithm.

7. CONCLUSION

In automated software testing the test cases with test data are very useful. Firefly Algorithm (FA) is an evolutionary meta-heuristic algorithm used to optimize the automated test cases with test data. Here Firefly algorithm has been discussed to generate the test cases which are optimized by taking an example of withdrawal operation by an ATM machine automatically. Test data values are selected based on the fitness function. This paper described the fundamental notions of FA, how the test cases are generated using Firefly algorithm and how they are useful in finding the optimal solution to maximize the problem. The result of Firefly algorithm is more accurate and this algorithm is generating automated test cases with test data efficiently.

REFERENCES

[1] Ausiello, Giorgio; et al., Complexity and Approximation (Corrected ed.), Springer, ISBN 978-3-540-65431-5,2003.

[2] B. Korel , "Automated software test generation", IEEE Trans. on Software Engineering,16(8): 870–879,1990.

[3]. Iqbal, Zafar, Zyad, "Multi-objective optimization of test sequence generation using multi-objective firefly algorithm (MOFA)", Robotics and Emerging Allied Technologies in Engineering (iCREATE), 2014.

[4] MA Sasa, Xue Jia, Fang Xingqiao, Liu Dongqing, "Research on Continuous Function Optimization Algorithm Based on Swarm Intelligence", 5th International Conference on Computation, pg no. 61-65,2009.

[5].Hitesh Tahbildar and Bichitra Kalita,"Automated software test data generation: Direction of Research",International Journal of Computer Science and Engineering Survey(IJCSES),Vol.2,No.1,2011.

[6]. Ojha.D,Sahoo.R.K.,Dash.S,"Automatic Generation Of Timetable Using Firefly Algorithm",International Journal of Advanced Research in Computer science and software engineering,Vol.6,Issue-4,pp.589-593,2016.

[7]. P. Srivatsava, B. Mallikarjun, X.Yang," Optimal test sequence generation using firefly algorithm"- Swarm and Evolutionary Computation, Volume 8, pp. 44-53, February 2013.

[8]. P. R. Srivastava, M. Chis, S.Deb, X.S. Yang, "An Efficient Optimization Algorithm for Structural Software Testing", International journal of artificial intelligence, 2012.

[9].R.Malhotra and M.Garg."An adequacy based test data generation technique using Genetic algorithm",Journal of Information Processing Systems,7(2),2011.

[10] Pei-Wei TSai, Jeng-Shyang Pan, Bin-Yih Liao, Shu-Chuan Chu, Enhanced Artificial Bee Colony Optimization , International Journal of Innovative Computing, Information and Control, Volume 5, Number 12, December 2009.

[11] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization: An overview (Springer Science and Business Media, LLC 2007).

[12]Sahoo.R.K,Ojha.D,Dash.S:Nature Inspired Metaheuristic Algorithms-A Comparative Review,International Journal of Development Research,Vol.06,Issue.07, pp.8427-8432, 2016,.

[13]. Sudhir, "Performance Evaluation of Regression Test Suite Prioritization Techniques", International Journal of Advanced Engineering and Global Technology Vol-2, Issue-10, October 2014.

[14].Vikas Panthi,D.P.Mohapatra."Test Scenarios Generation using Path Coverage",International Journal Of Computer Science and Informatics,Vol.3,Issue-2, pp.2231-5292, 2013.

[15] Xin-She Yang and Amir H. Gandomi, Bat Algorithm: A Novel Approach for Global Engineering Optimization, Engineering Computations, Vol. 29, Issue 5, pp. 464-483, 2012.

[16] X. S. Yang, Firefly Algorithm: Stochastic Test Functions and Design Optimisation, Int. J. Bio- Inspired Computation, Vol. 2, No. 2, pp.78–84, 2010.

[17] Yang, X. S., Nature-Inspired Metaheuristic Algorithms (Luniver Press),2008.

[18].Yeresime Suresh,Santanu Ku.Rath,"A genetic Algorithm based approach for test data generation in basis path Testing",International Journal of Soft computing and Software Engineering(JSCSE),Vol.3,No.3,2013.

[19] Sh. M. Farahani, A. A. Abshouri, B. Nasiri, and M. R. Meybodi, "A Gaussian Firefly Algorithm", International Journal of Machine Learning and Computing, Vol. 1, No. December 2011.

[20] Xin-She Yang, Chaos-Enhanced Firefly Algorithm with Automatic Parameter Tuning, International Journal of Swarm Intelligence Research, December 2011.

