International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.1 (Oct.-2021)

http://dx.doi.org/10.12785/ijcds/100186

Measure the Software Quality based on Grasshopper Optimization Algorithm

Ibrahim Ahmed Saleh[1], Asmaa Hadi Albayati[2], Kifaa Hadi Thanoon[3]

[1] Software engineering Dept., college of computer and Math., university of Mosul, Nineveh, IRAQ
[2] Software engineering Dept., college of computer and Math., university of Mosul, Nineveh, IRAQ
[3]Computer systems Techniques Dept. The Technical Institute/Mosul, Northern Technical University, IRAQ

Received 14 Mar. 2021, Revised 12 Aug. 2021, Accepted 5 Sept. 2021, Published 28 Oct. 2021

Abstract: Software quality is very essential function from development during the early life of software engineering. Software Quality helps to detect errors and potential errors during initial stage of design and software development process. In this paper, Grasshopper Optimization Algorithm (GOA) is used to improve software quality. Where multiple quality measures were used to calculate the quality standards that were used in testing the approved software. As software testing focuses on software defect. In addition this paper presents GOA to extract the best features of extraction to testing and evaluation of a set of software applications. The paper depended on NASA standards data. The result and experiment show that improved performance quality for all classification methods applied in the research grasshopper optimization algorithm based on feature selection and bagging for Software defect prediction.

Keywords: Grasshopper Optimization Algorithm ;Software quality; Software testing;Software engineering.

1. Introduction

Software Quality (SQ) is nature attributed of software develop projects according to well-defined software processes and templates far outweigh the costs. But software developers often tend to ignore or deviate from established process without external monitoring, thus giving software organization caused significant losses [1] so it is urgent to ensure that the process is performed through independent SQA activities such as project review and review by SQA personnel [2] .

Software quality is most useful tools in software optimization in organizations, the purpose of counting and offering optimization, efficiency and satisfaction of needs, which is why the software must have criteria that guarantee its quality. In accordance with this need, different entities or researchers have proposed model strategies, methodologies, guides, including quality standards and standards that provide support for the development and / or use of a software product and allow to assess whether it actually has a quality level during its life cycle, and in this way foster a quality environment, based on the proper administration of information [1, 2].

This document is initially contextualized in terms of software quality terms, after this a classification of the models is made according to the approach presented (process, product and use) and the time of appearance; This in order to make known those models that are considered pioneers or the basis of the development of recent ones, in the same way a description is made of the most relevant characteristics of some models, their structure and objective, finally there are cases of application of some models in the business sector[3].

It is important to know the concepts and characteristics about what is software quality? , terms of software quality models, their structure and approach. Term Software quality “indicates to degree of performance of original features which a computer system must respond during its life cycle" these features must ensures has reliable system for customer which increment their satisfaction with effectiveness and functionality of system [4].

As known, the benefits of software projects developing according to well-defined software processes and templates far reduce outweigh of project costs. Most of software developers often ignore or deviate from the established process without external monitoring thus

E-mail: i.hadedi@gmail.com, asmashade77@uomosul.edu.iq, kh.thanoon@ntu.edu.iq

caused significant losses in organization [3], so it is urgent to ensure that the process is performed through independent software quality assurance (SQA) activities such as project review by SQA personnel.

The SQA is established to ensure the smooth progress of work, improve the fault free codes in optimal time and cost., the basic of quality assurance are also determine whether the work can be implemented accuracy and higher detection rates [4]. At present, the academic and business organization have made a lot of discussions on the theory and practice of software quality assurance and set forward many processes and methods to create software quality rules [5, 6].

The most validation process is used Artificial intelligence (AI) methods in software engineering that merge ideas across two domains. The swarm optimization one of AI fields is better alternative to find optimal solution in research space, full research capability and significantly increase the ability to find high quality solutions within a reasonable period of time [7] . In this paper used new optimization method called grasshopper optimization algorithm (GOA) nature-Inspired by grasshopper when searching for their food, it is used to examine quality and accuracy of the software to effectiveness in dealing and evaluating the proposed method using the latest public datasets from the NASA database, it that stores problem, metrics data and product. The main goal of this data is to supply project data to the software applications [8]. NASA contain 1109 standards units of metrics of data program, generates metrics, and then product results without cost. [8, 9].

Reminder of paper is constructing as fellow, section 2 given related work. In section 3 introduced the software quality. In section 5 the proposed algorithm and describe the grasshopper optimization, and final section 5 describe expression result and conclusion.

2. RELATED WORK

Software quality techniques have been wide used in emerging area of research to improve software testing problems; it has concentrate mostly of them to optimization methods in the domain of software quality.

In [2019] Mehdi Gheisari et al [10] have proposed mathematical based on factors (Q) model to prediction of degree of customer's satisfaction. They used mathematical optimization models validate actual data using relationship impacts of deference features. The models based on maximum and minimum values for Q. Constraints constitutes for software quality features.

In [2018] Bestoun S. Ahmed et al [11] have presented them model to generate mannered linked interacted suites testing. Researchers used multi- objective particle swarm optimization (PSO) to find optimal test suites; the strategy takes input parameters and features of each configuration. The output of this strategy is given an optimal combinatorial test suite that satisfies constraints between input features. The method is gives speed mechanism but still find more than one solutions in critical area.

In [2017] K. Senthil Kumar et al [12], researchers proposed a method for estimation of software quality based on software metrics. They used two optimization methods PSO and cuckoo search algorithm (ICSA) to find optimization result to test software so as to reduce running time, effort and cost and thus it increases the quality and productivity of the project.

In [2016] Rajesh Kumar Sahoo1 et al [13], proposed method to find optimal test case of data by combine three artificial methods the result of heuristic algorithms harmony search (HS), particle swarm optimization (PSO) and bee colony optimization (BCO) . The researchers applied these methods to generate optimized test cases by taking an example of retreated for ATM machine and use random test case to find optimal solution.

In [2015] Deepti Arora et al [14], paper is combine two artificial methods (genetic and particle swarm optimization PSO) for software testing by find the most error path in the program. The researcher comparison between these two methods , Generally this paper don't enable to test all the possible input and not enable to find error because not enable test entire program.

In [2013] Romi Satria Wahono et al [15] paper also apply NASA metric for evaluate quality using two artificial algorithms particle swarm optimization and bagging method algorithm to enhance accuracy of software fault prediction. PSO is used to selected features while bagging technique is utilized to solve imbalance problem.

In [2012] Bestoun S. Ahmed et al [16] are produce them method to evaluated software quality way strategy for software testing generated called " Particle Swarm Test Generator PSTG" use suitable case with PSO to obtain better test sizes for most of configurations. Also the paper evaluates and demonstrates the capability of PSTG in generate efficient test suites.

3. PROPOSED ALGORITHM

A .Software Quality (SQ)

Software quality (SQ) "refers to the degree of performance for main features that a computer system must comply with during its life cycle", these features ensure that customer has a reliable system that increases their satisfaction with functionality and efficiency of system design**.** The information is mainly used for costs estimating, quality assurance, risk evaluation, and decision making in development and maintenance of software project [17].

Quality improvements are effected to software performance in different methods such as improve productivity, increase revenue and reduce costs. Quality is regarded as one of main drivers of competitive strategy in every manufacture [10]. However, as Reeves and Bednar stated that "no universal, Parsimonious, or all-encompassing definition or model of quality exists". ANSI and American Society for Quality (ANQ) define quality as: "The totality of features and characteristics of a product or service that impact its ability to satisfy given needs." [18].

The improving quality of software process leads to cost effectiveness, minimum re-job, and suitable up schedules, that leads to improve in capability measurement.◌ Software system is reviewed sequential process dynamically. The review must be conducted at every stage of the development process in order for transition to the test phase.

Software design and code review is a phase in the software development process in which programmers (authors of codes), quality assurance (QA) team, and peer reviewers get together to review design and code [6]. the main phase of Software development is design and code in which programmer (authors of codes), term of quality assurance (QA) and reviewers are working together to review design and code ,also they produce him advise suggestion[6] at the early stage of a software development process is relatively inexpensive such as :

- Finding and correcting errors.
- Minimize more expensive of management process.
- Determine and fixing faults during last stages of development or after delivered to users [3].

B. Grasshopper Optimization Algorithm

Grasshoppers are insects which damage to crop production and agriculture. Grasshopper has lived largest swarm of all creatures. The grasshopper behavior is depended to adults and nymph. Nymph is rolled on ground and feed on succulents and soft plants, while an adult grasshopper can search in a larger explore area for food [19].

when at grasshopper search for food source , at first time for lifecycle phase the swarm has very slow movement ; while in adulthood is moving long range and abrupt . The main two features of grasshopper swarming are exploration and exploitation when seeking regard to food, the grasshopper over with target seeking to perform these both inclinations naturally that move abruptly as well as locally in small areas. Mathematically express to behavior of grasshoppers for the design of the natural-inspired algorithm.

$$X_i = S_i + G_i + A_i \qquad (1)$$

Where: X_i represents i^{th} grasshopper position, S_i social interaction, G_i denoted to i^{th} grasshopper gravity force and A_i express wind advection. Note that to prepare random behavior, the equation written as:

$$X_i = r_1 S_i + r_2 G_i + r_3 A_i \qquad (2)$$

The r_1, r_2 and r_3 are random numbers that interval rang [0, 1] for social interaction as:

$$S_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} S(d_{ij}) \overrightarrow{d_{ij}} \qquad (3)$$

Such d_{ij} denoted to distance between i^{th} and j^{th} grasshopper is calculated

$$d_{ij} = |x_j - x_i| \qquad (4)$$

The function(S) is defined social force as shown in Equation (5) and d_{ij} is a unit vector from the i^{th} to jth grasshopper can be written as:

$$d_{ij} = \frac{x_j - x_i}{d_{ij}}$$

The *s* function is calculated as follows:

$$s(r) = f e^{\frac{-r}{l}} - e^{-r} \qquad (5)$$

Where: *f* represent attraction intensity and l is attractive of length scale [19, 20]. Generally, should be noted the regions of attraction or repulsion are very small values such as (l=1.0 or f=1.0 for instance). In this paper all values is chosen l=1.5 and f=0.5. The social interaction is interesting force in locust swarming models. The grasshopper space is divided into three areas (comfort zone, attraction region and repulsion region) is used to help function *S*.

- If distances between two grasshoppers are greater than 10, the return value of functions is near to zero.
- If distance is large value between any two grasshoppers, it cannot generate strong forces by this function.

To overcome this question the distance of grasshoppers is kept and mapped in the interval (1- 4). It can calculate the G component in equation (6) as follows:

$$G_i = -g \, \overrightarrow{e_g} \qquad (6)$$

Where: *g* gravitational constant and e_g unity vector to earth center. To calculate *A* as the following equation:

$$A_i = u \, \overrightarrow{e_w} \qquad (7)$$

Where *u* is a constant driftage and e_w is vector in the wind direction. The nymph movement is correlated with direction of wind because nymph doesn't have wings. For this reason substituting S, G and ***A*** in equation (8), it expanded as follows [20,21]:

$$X_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} S(d_{ij})\overrightarrow{d_{ij}} - g\overrightarrow{e_g} + u\,\overrightarrow{e_w} \tag{8}$$

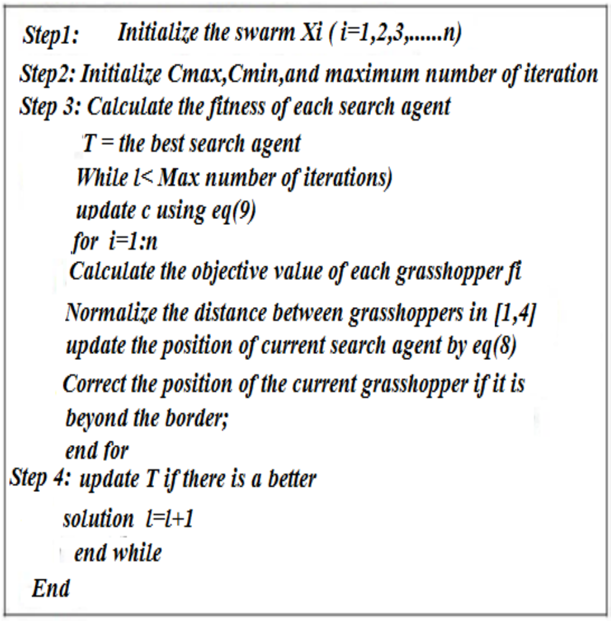
This equation is used to imitate interaction among swarm elements of grasshoppers, however, it cannot solve ideal problem directly because all swarm elements reach comfort zone quickly and grasshoppers not converge to a specified point. A adjust version of equation is proposed for give optimization solution to problems illustrate in following equation:

$$X_i^d = c\left(\sum_{\substack{j=1 \\ j\neq i}}^{N} c\,\frac{ub_d - lb_d}{2}\, s\left(\left|x_j^d - x_i^d\right|\right)\frac{x_j - x_i}{d_{ij}}\right) + \widehat{T_d} \tag{9}$$

c is a decrease coefficient to reduce comfort zone, repulsion zone, and attraction zone. The equation illustrates definition to next position of swarm element. It decreases with the number of iterations and can calculate it:

$$c = c_{max} - t\,\frac{c_{max} - c_{min}}{t_{max}} \tag{10}$$

The pseudo code of GOA is shown in figure (1)



```
Step1:   Initialize the swarm Xi ( i=1,2,3,......n)
Step2: Initialize Cmax,Cmin,and maximum number of iteration
Step 3: Calculate the fitness of each search agent
        T = the best search agent
        While l< Max number of iterations)
        update c using eq(9)
        for  i=1:n
        Calculate the objective value of each grasshopper fi
        Normalize the distance between grasshoppers in [1,4]
        update the position of current search agent by eq(8)
        Correct the position of the current grasshopper if it is
        beyond the border;
        end for
Step 4: update T if there is a better
        solution  l=l+1
        end while
  End
```

Figure 1. The pseudo code of GOA

B. Proposed Method

GOA is a proposed algorithm based on swarm elements was used to measure quality of software; It takes a random sample of the training group to form models based on the measurement of performance quality data taken from NASA contains 22 standards and 1109 standards units are used to describe the quality of software in the development stage and software testing; 19 metrics, including input properties and one metric representing actual output. The steps to represent the search algorithm are summarized as follows:

Step 1: Set full rules data from NASA dataset, it is divided two groups (80% training data and 20% test data).

Step 2: Set initial parameters of GOA algorithm (ub(upper bound) = 100, lb(lower bound) = -100, flag=1, cMax=1, cMin=0.00004, max_ alteration =500, f(intensity of attraction)= 0.5 , l(attractive length scale)=1.5) and implementation of algorithm on data based on 16 attributes at choice = 1 out of 10 options identified in the previous .

Step 3: Determine matrix that is used to make the specification [19 * 10] row equal 10 represents number of selections from which one will be chosen, and column equal 19 represents the number of specifications to be chosen from 19 attributes.

Step 4: Set locations with initial values where the speed is zero at the beginning and the sites represent the 16 attributes that are predetermined. Large on their parameters set as well as inputs to this model.

Step 5: Calculate the fitness function of values for each individual initially based on the sphere function, as the function represents the following equation:

$$f(x) = \sum_{i=1}^{n} x_2 \tag{11}$$

Step 6: Calculate the value of fitness function with using equation (11)

Step 7: Update the new current search agent position by equation 9.

Step 8: Fetch current search agent back if it goes calculate values of fitness function for each swarm element for new values.

Step 9: In the end, choose the best specifications based on the best value.

Step 10: Evaluate the proposed algorithm based on quality criteria.

Changing the location of each individual swarm means changing scale in order to reduce the error rate for each testing cycle. In each cycle that members of the swarm updated their positions and then move to new location which represents a new scale. It depends on the assessment of the fitness function; this process is returned best 16 values. These values determine the 16 best scale of accuracy for the classification of software quality standards. The steps of the search algorithm can be represented using an activity diagram as shown in the figure (2):

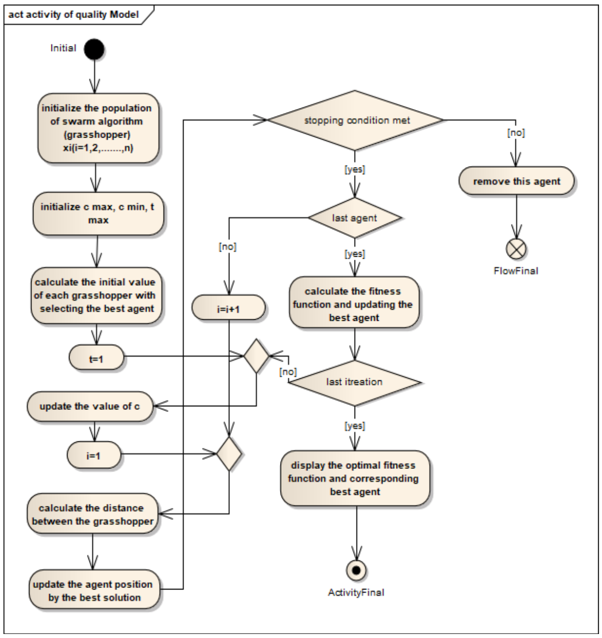

act activity of quality Model
Initial
initialize the population of swarm algorithm (grasshopper) xi(i=1,2,.......,n)
initialize c max, c min, t max
calculate the initial value of each grasshopper with selecting the best agent
t=1
update the value of c
i=1
calculate the distance between the grasshopper
update the agent position by the best solution
stopping condition met
[no]
remove this agent
FlowFinal
[yes]
last agent
[no]
[yes]
i=i+1
calculate the fitness function and updating the best agent
[no]
last itreation
[yes]
display the optimal fitness function and corresponding best agent
ActivityFinal

Figure 2. Activity diagram of system

3. EXPERIMENT AND RESULT

The data used NASA database that used 19 attributed to metrics the quality of software, the proposed algorithm was applied using Matlab R2017a to calculate quality criteria including accuracy, Error Rate (ER), and Mean Square Error (MSR).

Accuracy ratio of percent increase the chance of reaching the optimal solution that represents the difference between the actual value and the expected value shown in equation (12).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \qquad (12)$$

Where: TP, represent number of true positives ,FN false negatives, FP false positives and TN, true negatives, the optimal value of accuracy equal 99.2% as shown in figure (3).

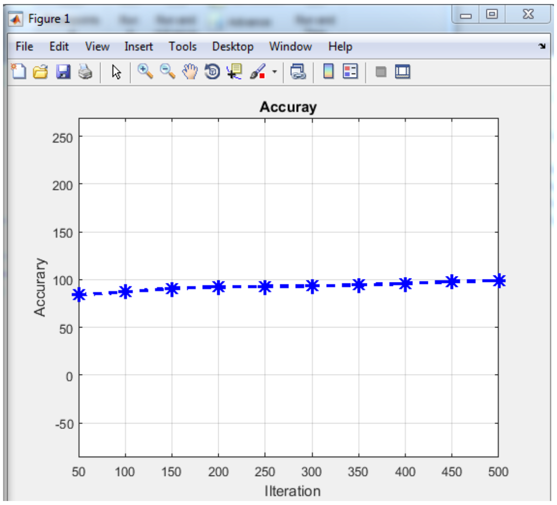


Figure 3. Accuracy of system

while the error rate[22] is calculated based on the percentage of accuracy, the average square error calculate the sum of the square of the difference between the actual value and result value illustrated in equation(13).

$$\text{ER} = \frac{FP+FN}{TP+TN+FP+FN} \qquad (13)$$

The evaluation result of error rate is shown in figure (4)

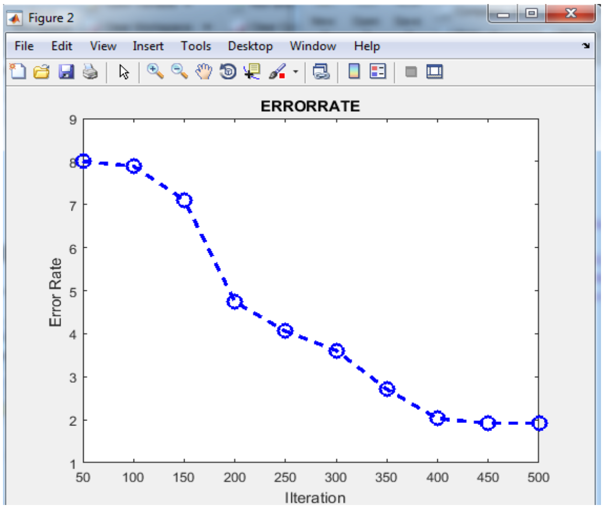


Figure 4. Error rate of system

Figure 5 represents the results of the mean square error (MSE) that illustrate in this equation (14)

$$MSE = \frac{1}{n}\sum(y-\hat{y})^2 \qquad (14)$$

GOA is shown MSE of software quality measurement decrement for iteration increase, the optimal solution equal (0.182) as illustrated in figure (5)

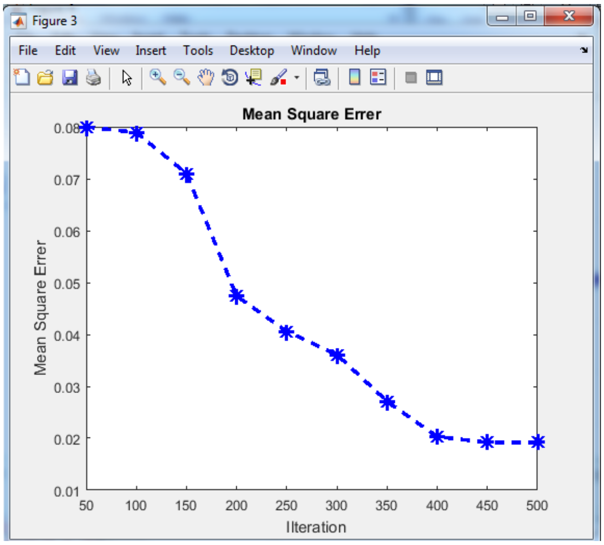


Figure 5.   Mean square error of system

4. CONCLUSION

Quality method is very important to reduce the adverse effects of defects in software engineering. This paper present framework depended to behavior of grasshopper called grasshopper optimization algorithm. The algorithm was implemented to nineteen metrics NASA dataset for projects measurement. The experiment result has shown the efficiency accuracy rate is equal 99% which given more trust over project. Also the results indicates that GOA is outperforms for checking examine optimal comprehensibility execution time and the fault detection rate.

In the future, this work can be expanded to verify software quality by combining two algorithms from metaheuristic optimization such as salp swarm optimization with sine-cosine optimization algorithm to reduce Mean square error.

REFERENCES

[1] Fatih Gurcan and Cemal Köse, “Analysis of software engineering industry needs and trends”, Implications for education International Journal of Engineering Education 33(4): 2017, 1361-1368.

[2] Giuliano C., Cristina C., Peter D. et al, “Current and Future Challenges of Software Engineering for Services and Applications ", Elsevier B.V., Procedia Computer Science 97 (2016) 34 – 42 , CF2016, 18-20 October 2016, Madrid.

[3] Inibehe Emmanuel Akpannah, “Optimization of Software Quality using Management and Technical Review Techniques”, International Journal of Computer Trends and Technology (IJCTT) – volume 17 Number 6 *Nov 2014* ISSN: 2231-5381.

[4] Daniel Galin,”Software Quality Assurance From theory to implementation Galan Daniel” Software quality assurance / Daniel Galin. p. cm. index. ISBN 0-201-70945-7.A.

[5] Kumar Jakhar and K. Rajnish, “Software Fault Prediction with Data Mining Techniques by Using Feature Selection Based Models”, International Journal on Electrical Engineering and Informatics - Volume 10, Number 3,2018.

[6] Ljubomir Lazic, Nikos E. Mastorakis, “Optimal SQM:Integrated and Optimized Software Quality Management” sweat transactions on information science and applications. 2015.

[7] Saleh Ibrahim Ahmed, Alsaif Omar Ibrahim , Thanoon Kifaa H.,” Deep Coverage Strategy for Private Wireless Network Power Using Hybrid (Salp Optimization – Genetic) Algorithms” TRKU Volume 62, Issue 03, April, 2020

[8] Alain Abran, ," Software Metrics and Software Metrology, 2010,ISBN: 9780470597200.

[9] Claude Y. Laporte and Alain April, " Software Quality Assurance ", Hoboken, NJ : Wiley-IEEE Computer Society, Inc., , LCCN 2017036440 (print) , 2018.

[10] Mehdi Gheisari, Deepak Panwar, Pradeep Tomar, Harshwardhan Harsh, Xiaobo Zhang, Arun Solanki, Anand Nayyar, Jafar A. Alzubi, “An Optimization Model for Software Quality Prediction with-Case Study Analysis Using MATLAB” DOI 10.1109 /ACCESS .2019. 2920879, IEEE.

[11] Bestoun S. A., Luca M. G., Wasif A.; and Kamal Z. Z, , “Handling Constraints in Combinatorial Interaction Testing in the presence of Multi Objective Particle Swarm and Multithreading”, Information and Software Technology Journal, arXiv:1804.07693v1 [cs.SE] 20 Apr 2018.

[12] K. Senthil Kumar and Dr. A. Muthukumaravel , “Optimal Test Suite Selection using Improved Cuckoo Search Algorithm Based on Extensive Testing Constraints”, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 9 (2017) pp. 1920-1928.

[13] R. Kumar Sahoo, D. Ojha, D. Prasad Mohapatra and M. Ranjan Patra , “AUTOMATED TEST CASE GENERATION AND OPTIMIZATION: A COMPARATIVE REVIEW”, International Journal of Computer Science & Information Technology (IJCSIT) Vol 8, No 5, October 2016

[14] Deepti Arora and Anurag Singh Baghel , “Application of Genetic Algorithm and Particle Swarm Optimization in Software Testing”, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 17, Issue 1, Ver. II (Jan – Feb. 2015), PP 75-78.

[15] Romi Satria Wahono and Nanna Suryana , “Combining Particle Swarm Optimization based Feature Selection and Bagging Technique for Software Defect Prediction”, International Journal of Software Engineering and Its Applications, Vol.7, No.5 (2013), pp.153-166.

[16] Bestoun S. Ahmed, Kamal Z. Zamli and Chee Peng Lim,, “Constructing A T-Way Interaction Test Suite Using The Particle Swarm Optimization Approach”, International Journal of Innovative Computing, Information and Control, ISSN 1349-4198, Volume 8, Number 1(A), pp. 431-451. 2012.

[17] B. Dhanalaxmi, Dr. G. Apparao Naidu, and Dr.K.Anuradha, “Practical Guidelines to Improve Defect Prediction Model – A Review “, International Journal of Engineering Science Invention ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726, Volume 5 Issue 9, September 2016, PP. 57-61.

[18] Yas A. Alsultanny, Ahmed M. Wohaishi, “Requirements of Software Quality Assurance Model”, Second International Conference on Environmental and Computer Science, DOI 10.1109/ICECS.2009.43.

[19] Muhammad Suleiman, Masihullah, Zubair Hussain, Implementation of improved grasshopper optimization algorithm to solve economic load dispatch problems, Hacettepe Journal of Mathematics & Statistics, Hacet. J. Math. Stat. Volume XX(x) (2019), 1 – 21 DOI : 10.15672/HJMS.xx

[20] Shahrzad S ,Seyedali M. ,b ,Andrew Lewis, ” Grasshopper Optimization Algorithm : Theory and application” Advances in Engineering Software 105(2017)30–47.

[21] Saleh Ibrahim. Ahmed, Alsaif Omar Ibrahim, Yahea Maan A.,” Optimal distributed decision in wireless sensor network using gray wolf optimization” IAES International Journal of Artificial Intelligence (IJ-AI) Vol. 9, No. 3, September 2020, pp. 535~543

[22] Bakota T, Hegedus P, Kortvelyesi P, et al. “A probabilistic software quality model” In: Proceedings of the 27th IEEE International Conference on Software Maintenance, 2011. 243–252.

Ibrahim Ahmed Saleh: born in Mosul-Iraq in 1963. He received his MSc. degree (in signal and image processing from the University of Mosul, Iraq in 2003 and in 2013 he received his PhD in using artificial techniques in Networking from Mosul University. Assistant Professor in 2016. From 1997 to 2005, he worked at computer Center in Mosul University/ Iraq. Currently he is lecturer at the Dept. of Software Engineering, College of Computer Sciences & Math, and University of Mosul, IRAQ. His interesting in Network and artificial intelligent simulation

Asmaa’ Hadi AL_Bayati: Born in Mosul/Iraq in 1977. She received her bachelor’s degree in chemistry in Iraq, Mosul University In 2000, and in 2007 She received her bachelor’s degree in software engineering sciences from **Mosul** University. Her MSc. degree (**in** software engineering) From the same university in 2013, Assistant lecturer in 2013. Since 2008 is lecturer at the Dept. of Software Engineering, College of Computer Sciences & Math, and University of Mosul, IRAQ. Her interesting in Software engineering tools and information technology.

Kifaa Hadi Thanoon: born in Mosul-Iraq in 1970. She received her MSc. degree (in image processing from the University of Mosul, Iraq. Lecturer in 2019. From 1998 to 2017, She worked at computer Center in Technical Institute Mosul / Iraq. Currently she is lecturer at the Dept. of Computer Systems Techniques, Technical Institute / Mosul, Northern Technical University, IRAQ. Her interesting in Image and signal processing, neural networks.