iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 Issue: V Month of publication: May 2018

DOI: http://doi.org/10.22214/ijraset.2018.5339

Prioritization of Software Applications in Cloud Using Gwo Algorithm

Sunitha B[1], Y. S. S. R Murthy[2]
[1, 2]Computer Science Department, Computer Science Engineering Department

***Abstract:** In any Software development one of the prominent activities is testing. In software development testing any defects which are introduced or unsolved as a result of the changes is defined as Regression testing. New algorithms and techniques are developed from time to time with intent to generate new design patterns in code and hence the risky areas of the program are more effective than earlier design patterns. One of the challenging activity which overcomes elimination of cost, pay per use in capital expenditures and hence reduces the hardware/software cost. To overcome the drawbacks of various algorithms, and to find an accurate optimization of existing software applications proposed technique use optimal test case prioritization in Cloud using KFCM technique. Technique consists of 2 basic steps. Firstly, for organizing the test cases in software application KFCM technique is applied based on a similarity feature. In each cluster the test cases should be relatively independent. Secondly, Grey Wolf Optimization algorithm is applied to achieve an optimal solution. KFCM uses an objective function, centroid to compute the distance between any two test cases in the cloud. Later test cases are grouped or clustered. This proposed method will improve the performance of clustering test cases in cloud. Thus we will obtain effective prioritized test cases.*
***Keywords:** Cloud testing, prioritization, Kernel fuzzy C means, GWO*

I. INTRODUCTION

Given a set of system requirements the Software development process aims at evaluating a software item(s) which are studied from the initial phase of SDLC to its retirement phase. In any Software development process, testing being a comprehensive set of activities that are applied to find bugs or errors in the software.[1] Regression testing explores on the modified software versions to check whether it behaves according to the specifications [2]. The technique used in the regression testing ensures that the recently done changes do not exist anymore [3]. The traditional approaches to software testing are becoming very costly in terms of software development cost and effort. With the evolution of cloud-based testing, organizations are becoming much easier and don't have to worry about finding servers; procuring testing tools and installing them .Testers often are provided with various services to access ready to use virtual labs that are online. With cloud testing the test management and execution tools, middleware and storage necessary for creating a test environment closely mirrors the real environment. Grey Wolf Optimizer (GWO) algorithm is a new optimization method which is employed to solve optimization problems of different varies (S. Mirjalili et al, 2014). As other heuristic algorithms in the area of evolutionary computation, in optimization processes GWO has not required to the gradient of the function. GWO a mathematical model and the computer simulation which mimics the similar occurrences which match with a hunting mechanism of grey wolves in nature. This paper proposes a new method which utilizes GWO (Grey Wolf Optimizer) algorithm with an aid of KFCM (Kernel Fuzzy C Means) technique in cloud testing.

II. LITERATURE SURVEY

Jyoti, Kamna Solanki Suggested "A Comparative Study of Regression Testing Techniques" in their paper, the results of a comparative study on five different regression test optimization techniques is represented. The comparison is based on different criteria such as number of test cases chosen, their test execution time, accuracy, user parameters, global variables handling, type of testing etc. The above specified algorithms are found to be suitable for different requirements/environments of regression testing[4]
Krishnapuram and Keller changed the constraint of memberships in FCM and proposed the possibilistic c-means (PCM) algorithm [5]. The advantages of PCM are, it overcomes the need to specify the number of clusters and it is highly robust in a noisy environment. However, still exist some weaknesses in the PCM, i.e., it highly depends on a good initialization and has the poor performance to identify similar clusters [5][6]. Usually, the FCM can provide a reasonable initialization and an approximate scale that determines the degree to which the objective function is important when compared with the first.
Cloud testing is one amongst the emerging areas of testing in which web applications make use of cloud computing environment and infrastructure as a service (IaaS) to simulate real world scenarios on how to control the user traffic by using cloud applications

across the globe. The work on this paper by Dr. Rahul Malhotra & Prince Jain [7] is explanation on various cloud computing testing techniques and challenges.

Chitra Jain, Gaurav Srivastava proposed "Designing a Classifier with KFCM technique to Achieve Optimization of Clustering" suggested that KFCM is based on the idea of minimizing the objective function by and reducing the number of iterations required to gather a feasible cluster center. Initially, the cluster centers are scattered and then based on the concept of empirical formula

The distance in which distance of test cases (samples) from the center of the clusters is calculated and the centroid cluster thus obtained, its cluster membership degree is calculated. The Cluster center is updated iteratively and the difference between the objective function and cluster center get minimized [8].

Er. Tamanna Narula, Geetika Sharma proposed a paper " Framework for Analyzing and Testing Cloud based applications". This paper explained that Cloud based testing is new paradigm in software testing method where in this system, the use Cloud computing use web applications environments ("cloud") and gather to simulate Real-world user traffic as a means of evaluating stress testing and load testing web sites. Paper emphasizes on how automatic test case generation for unit tests [9] and concept of Symbolic execution are migrated to cloud environment. The problem of regression testing case selection is solved by prioritizing test cases [10].

The work proposes a new meta-evolution called Grey Wolf Optimizer (GWO) inspired by grey wolves (Canis lupus). The GWO depicts a leadership hierarchy and mechanism of hunting grey wolves in nature. For simulating the leadership hierarchy four types of grey wolves parameters such as alpha (α), beta (β), delta (δ), and omega (ω) are considered. In addition, there are three main steps of hunting, searching for prey, encircling prey, and attacking prey, are implemented.

The results show that the GWO algorithm is able to provide very competitive results compared to these well-known meta-heuristics [11].

Fuzzy C-Means can determine, and updating the membership values iteratively using the test cases with already existing number of clusters i.e. K. Thus, every test case present in the test suite carries a membership value for entire cluster. FCM has been prominently used in various fields which is discussed in [12]-[15]. Many numbers of variants of the FCM algorithm had been discussed. Sikka et al. [16] pointed out some of these algorithms. In this paper, they implemented FCM using three basic options an compared number of iterations, accuracy with K-Means algorithm.

In this paper, a kernel-based fuzzy c-means algorithm (KFCM+GWO) is proposed. KFCM uses a new kernel-induced metric in the cloud space to replace the original Euclidean norm metric in FCM. By replacing the inner product by an appropriate 'kernel' function, performing a nonlinear mapping to a high dimensional feature space without increasing the number of parameters.

Many learning systems have used this 'kernel method' and this motivated to find a minimal objective function GWO (Grey Wolf Optimizer). The rest of this paper is organized as following: In Section 2, GWO algorithm is introduced and Section 3 shows proposed method and conclusion.

III. SECTION II

A. Grey Wolf Optimizer (Gwo)

This section summarizes the main steps in grey wolf optimizer (GWO) to optimally run the number of test cases in the cloud. Grey wolf algorithm (Canis lupus) is a new population based algorithm which is introduced in 2014 by Mirjalili et al (Mirjalili,et al, 2014). Grey Wolf Optimizer algorithm inspired by grey wolves. Method mimics grey wolves the social hierarchy and hunting behavior. For simulating the leadership hierarchy in Grey Wolf Optimizer algorithm, four groups are defined: alpha, beta, delta, and omega. Furthermore, the three main steps of hunting, searching for prey, encircling prey, and attacking prey, are simulated. This GWO algorithm requires few parameters need to be set, namely initializing alpha, beta, and delta, number of iterations, and a stopping criterion. designing Grey.

Wolf Optimizer, the fittest solution is considered as the alpha (α) ,accordingly, the second and third best solutions are named as beta (β) and delta (δ) respectively. The rest of the remaining candidate's solutions are considered to be omega (ω) .The final position would be in a random position within a circle which is defined by the position of α, β, and δ in the search space. In other words alpha (α), beta (β), and delta (δ) estimate the exact victim's current position and other wolves update their positions randomly around the victim. Pseudo code of the algorithm is shown in its simplest form in Fig. 1.

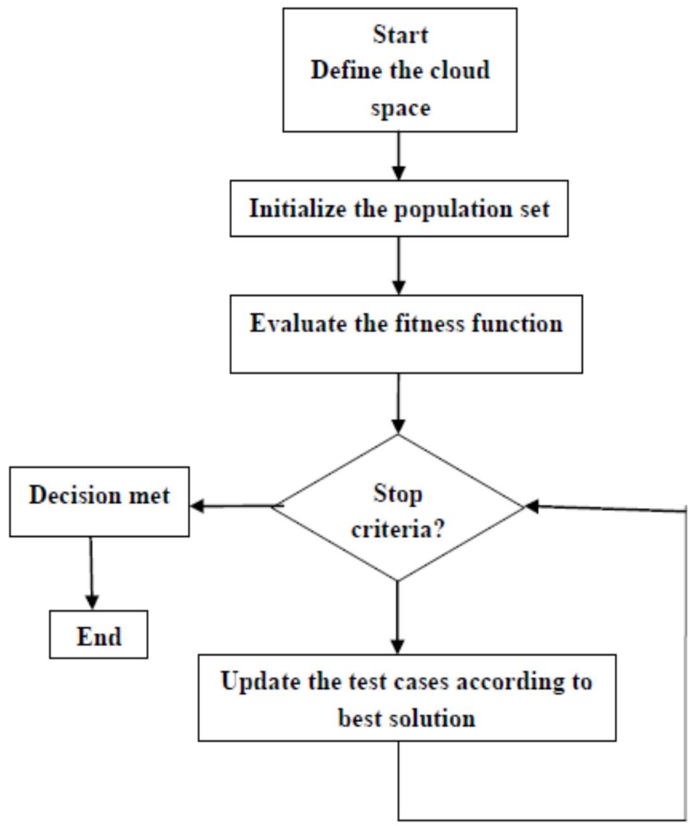


Fig.1 Flowchart for gwo technique.

IV. SECTION III

A. *Proposed Method*

1) First, based on similarity feature values, test cases are generated.
2) KFCM technique separates the test cases in cloud environment.
3) GWO is used to prioritize the test cases in cloud.
4) Optimal test cases in software application results.

Proposed technique KFCM is used to refine the software productivity and its competency in cloud. Initially software application is sent to the cloud and test case generation being one of the major steps in the study is to associate the test cases. As part of test case generation we take into account of considering basic feature values like statement coverage, line coverage, loop coverage etc. The test generation step uses KFCM to identify relevant and irrelevant test cases. Since our aim is to improve software efficiency viz performance only the relevant test cases are sent to cloud for prioritization. And GWO algorithm which is illustrated above is used here which will obtain the prioritized test cases.

B. *Analysis Process*

Prioritization of test cases in software applications using GWO algorithm is better than any legacy algorithms in cloud. Fig.2 illustrates GWO algorithm takes minimum execution time to comprehensive the prioritization procedure related to the available techniques.

While applying GWO algorithm every cluster is given for test case prioritization and for each cluster thus created is iterated using regression testing. Testing is conducted and at regular intervals at the end of every 5th iteration , 10th iteration and subsequent iteration the performance evaluation is carried out on each cluster i.e. cluster 1,2,3.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue V, May 2018- Available at www.ijraset.com

C. Analysis I

It is observed that at the end of 20th iteration and 25th iteration the objective function OF is attained with various values of cluster1 (0.0099) and cluster 2 (0.02) and cluster 3 (0.011).

Table 1. Fitness value using GWO for each cluster

Iteration	Fitness value using GWO for each cluster		
	Cluster 1	Cluster 2	Cluster 3
5	0.1077	0.478	0.06046
10	0.079	0.057	0.1923
15	0.088	0.2825	0.048
20	0.0199	0.6466	0.0479
25	0.0099	0.002	0.011

Table.1 shows the fitness value for each cluster using GWO. For 5 iterations the fitness value for cluster 1 using GWO is 0.1077% and for cluster 2 the fitness value is 0.478% and for cluster 3 the fitness value is 0.06046%. For 10 iterations the 0.079% is the value of fitness of cluster 1 using GWO and 0.057% is the value of fitness for cluster 3 and 0.1923% is the fitness value for cluster 3. The value of fitness for cluster 1 is 0.088% using GWO for 15 iterations and the value of fitness for cluster 2 is 0.2825% and the value of fitness for cluster 3 is 0.048%. For 20 iterations the value of fitness for cluster 1 is 0.0199% and the value of fitness for cluster 2 0.6466% and the value of fitness for cluster 3 is 0.0479%. The value of fitness for cluster 1 is 0.0099% using GWO for 25 iterations and the value of fitness for cluster 2 is 0.002% and the value of fitness for cluster 3 is 0.011%. From the solution we evidently understand that when the number iteration influences the maximum the projected method achieves the objective performance.

D. Analysis II

Table 2. The overall execution time of proposed technique

Iteration	Time (ms)
5	327
10	393
15	424
20	578
25	619

Table.2 characterizes the execution time by varying the number of iteration of the projected optimal test case prioritization. For 5 iterations the execution time is 327ms and the execution time is 393ms for 10 iterations. The execution time taken for 20 iterations is 578ms and for 25 iterations the execution time taken is 619ms. Hence from this it is clear that the execution time increases as the number of iterations gets increases.

E. Analysis III

Table 3. The overall memory of proposed technique

Iteration	Memory (bits)
5	5779872
10	5658728
15	6458456
20	8105880
25	4874008

The table 3 shows for 5 iterations the memory occupied using the proposed method is 5779872 bits. 5658728 bits of memory is occupied for 10 iterations. The value of memory occupied for 15 iterations using the proposed method is 6458456 bits. For 20

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue V, May 2018- Available at www.ijraset.com

iterations the memory occupied using the proposed method is 8105880 bits. The value of memory occupied using the proposed method for 25 iterations is 4874008 bits.

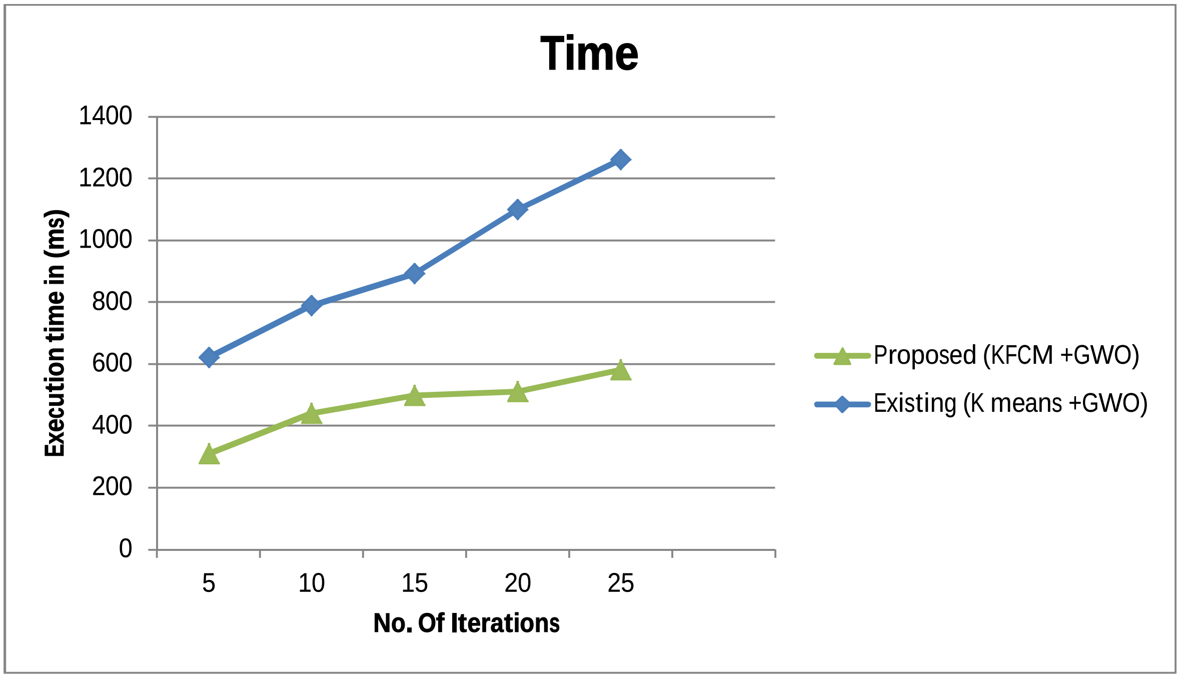


Fig.2 Execution time comparison for proposed Vs existing method

V. SECTION IV

A. *Conclusion*

In cloud based scenario, it is observed that GWO algorithm is better than any underlying algorithms. To find a best solution the distance between any two test cases is considered as a metric. Clustering technique using KFCM results are evaluated using the centroid function rather than the distance between the test cases. GWO is easy to implement due to simple structure. Less storage is required other existing techniques. Convergence is faster due to continuous reduction of search space and decision variables are very less. It avoids optima when applied to composite functions. Only two main parameters to be adjusted. Hence GWO when applied with suitable clustering algorithms can produce better results. And also since the applications are put to cloud there is user privacy in diverse web clients. Both virtual and real-time test cases can be applied by the users. Online validation and measurements is quite possible rather than fixed test environment. More performance evaluation metrics can be applied looking into the growing needs and security factors on the cloud.

REFERENCES

[1] Siripong Roongruangsuwan and Jirapun Daengdej,"Test Case Prioritization Techniques", In proceeding of the Journal of Theoretical and Applied Information Technology, pp. 45-60, 2010.

[2] Deepak Garg and Amitava Datta, "Test Case Prioritization due to Database Changes in Web Applications", In proceeding of IEEE International Conference on Software Testing, Verification and Validation, pp. 726-730, April 2012.

[3] Albert Pravin and Subramaniam Srinivasan, "Effective Test Case Selection and Prioritization in Regression Testing", In proceeding of the Journal of Computer Science, Vol. 9, No. 5, pp. 654-659, 2013.

[4] Jyoti, KAMNA "**A** Comparative Study of Five Regression Testing Techniques":A Survey International Journal of Scientific and Research Vol.3, Issue 8, August 2014

[5] S. Nachiyappan and S. Justus, "Cloud Testing Tools and Its Challenges: A Comparative Study", International symposium on big data and cloud computing, Vol.50, pp. 482-489, 2015.

[6] Shraddha D. Mungekar and Dhanashree K. Toradmalle, "Cloud Testing- An Advancement in Software Testing", International Journal of Application or Innovation in Engineering and Management, Vol.4, Issue 3, pp. 61-63, March 2015.

[7] Dr. Rahul Malhotra and Prince Jain "Testing techniques and its challenges in cloud" The SIJ Transactions on Computer Science Engineering & its Applications (CSEA), Vol. 1, No. 3

[8] Chitra Jain, Gaurav Srivastava **"**Designing a Classifier with KFCM Algorithm to Achieve Optimization of Clustering and Classification Simultaneously" International Journal of Emerging Technology and Advanced Engineering Volume 3, Issue 9

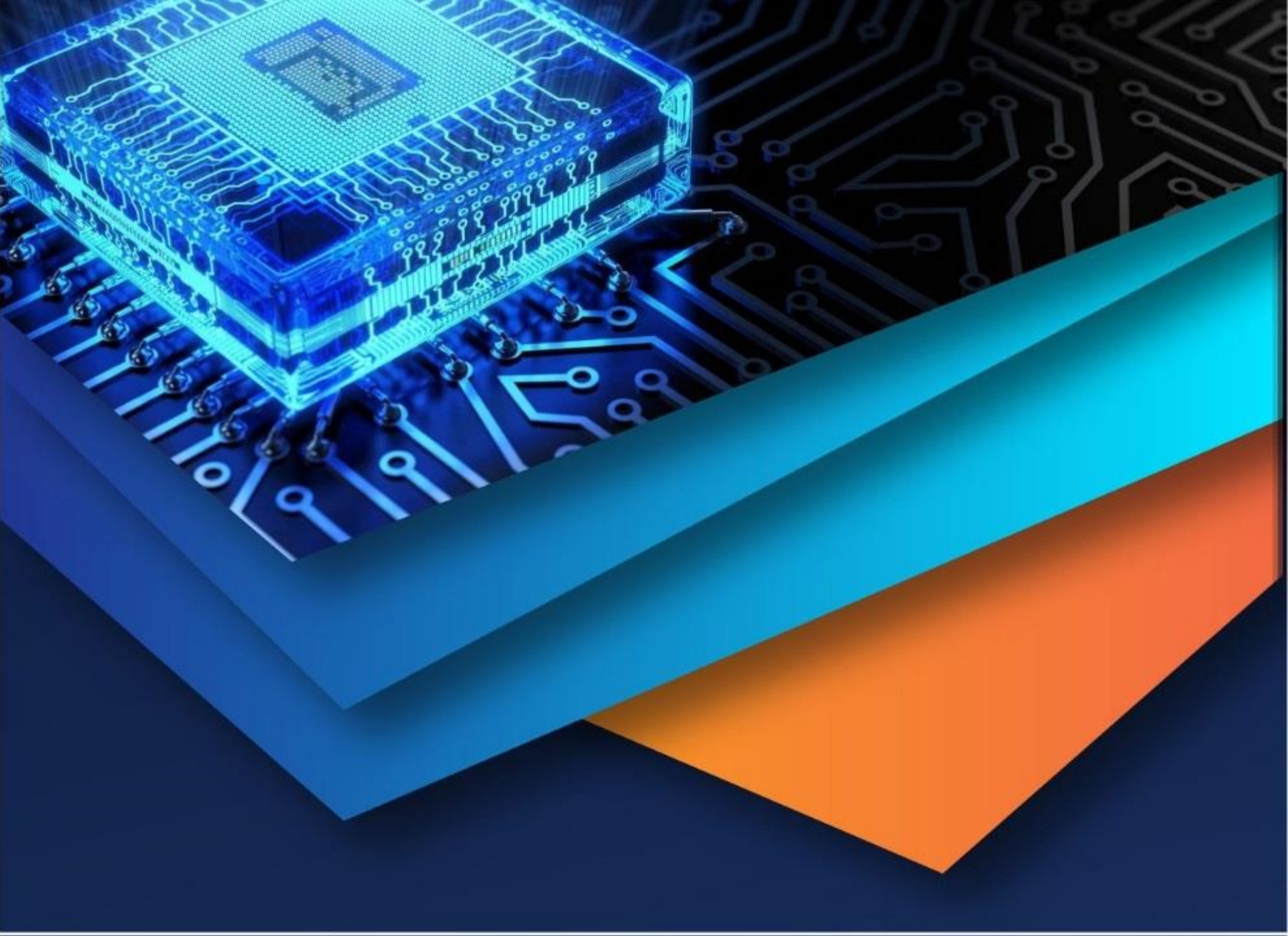
[9] Er. Tamanna Narula, Er. Geetika Sharma "Framework for Analyzing and Testing Cloud based Applications" International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 6, June 2014

[10] Khushwant Virdi , Radhika Kalyan, Navjot Kaur "Software Testing as a Service (STaaS) using Cloud Computing" Volume 4, Issue 1, January 2015

[11] sayedalili mirjalili "Grey wolf Optimization" Advances in Engineering Software Volume 69, March 2014, Pages 46–61

[12] D. Q. Zhang and S. C. Chen, "A novel kernelized fuzzy C-Means algorithm with application in medical image segmentation," artif. Intel. Med, vol. 32, pp. 37–50, 2004.

[13] S. Migaly, J. Abonyi, and F. Szeifert, "Fuzzy Self-Organizing Map Based on Regularized Fuzzy C-Means Clustering," in Advances in Soft Computing, Engineering Design and Manufacturing, J. M. Benitez, O. Cordon, F. Hoffmann et al. (Eds.), Springer Engineering Series, (Revised papers of the 7th On-line World Conference on Soft Computing in Industrial Applications (WSC7)), 2002, pp. 99–108.



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY