International Journal of Scientific Research & Engineering Trends
Volume 7, Issue 1, Jan-Feb-2021, ISSN (Online): 2395-566X

A Comparative Analysis of Optimization Techniques

M. Tech.Scholar Palac Gupta
Dept. of Computer Science and Engineering
Shaheed Udham Singh College of Engineering and Technology,
Tangori
palac1gupta12@gmail.com

***Abstract*-Regression testing is an important process during software development. For the purpose of reducing the number of test cases and detecting faults of programs early, this paper proposed to combine test case selection with test case prioritization. Regression testing process has been designed and optimization of testing scheme has been implemented. The criterion of test case selection is modify impact of programs, finding programs which are impacted by program modification according to modify information of programs and dependencies between programs. Test cases would be selected during test case selection. The criterion of test case prioritization is coverage ability and troubleshooting capabilities of test case. There are various optimization techniques available. This review explains about the different optimization techniques on the basis of their evolution, methodology, performance and applications.**

***Keywords*- Optimization techniques, evolution, applications, regression testing.**

I.INTRODUCTION

Software maintenance is defined as activity performed on a software product which includes enhancements, optimiz ation, error corrections, and deletion of inoperative func tionalities. Regression testing is conducted during maintenance phase of Software Development Life Cycle and it is defined as"the process of re-testing the modified components of the software and ensuring that no new errors have been encountered in previously tested code"[1].

The approach for performing regression testing is to again execute the test cases. Test cases are set of conditions under which a tester will determine whether an application or its features are functioning as it was originally established to do. Regression testing is an expensive activity, running all test cases requires large amount of time and cost. So, it is required to optimize test cases to reduce the efforts and cost.Optimization tends to provide the best output with the least investments[2]. The goal of optimization techniques for test cases is to minimize the number of test cases without affecting the fault coverage of the testing process. Component Based System(CBS) is the emerging technology in last few years. It is a more generalized approach for software develop ment.CBS are mainly developed using reusable components and Commercial-On-The-Shelf components [3].

Due to this,CBSs are developed with minimum engineering efforts and resource costs and time.CBS provide a new approach to construct,design and implement software applications. Software applications are foregathered from variety of components that may be written in different languages and may run on different platforms. That is, components are heterogeneous.
There are various advantages of using CBS for developing applications. Some of them are listed below:

1. **Reusability:**A code designed for a particular functionality may be used in a number of applications which need the same functionality,a programmer need not to write the same code multiple times.
2. **Interoperability**:Components have the ability to communicate, execute and exchange data among themselves without any need of knowing about underlying structure of the system.
3. **Upgradable:**The application can be upgraded easily if a new component has been introduced during the life of the application.
4. **Time and Cost Effective:**As in CBS, code for a component can be used again and again in similar type of applications. This makes it a time and cost efficient approach.

In this paper, a critical review of different test case optimization techniques is represented. Different techniques are compared with each other based on their evolution, methodology, performance and applications.

II. COMPARATIVE ANALYSIS

This section covers the comparative analysis of different optimization techniques based on the parameters like: Evolution, Methodology, Performance and Applications.

1. **Evolution:** It covers the evolution of the techniques all through their lifetime. It includes the growth of the technique from its development.

© 2021 IJSRET

2. **Methodology:** It describes the way in which the algorithm works and identifies the methods used in it. It shows how a result is to be calculated.
3. **Performance:** This section covers the overall performance of the techniques. It also covers how performance may be achieved by improvements in the technique and hybridization with other techniques.
4. **Applications:** Application is defined as the utility of a technique, technology or a system. It covers where these techniques may be applied to give a better result.

III. ANT COLONY OPTIMIZATION ALGORITHM

ACO,developed by Marco Dorigo in 1992 (Dorigo, 1992), was the first swarm intelligence-based algorithm. In essence, ACO mimics the foraging behavior of social ants in a colony, and pheromone is used for simulating the local interactions and communications among ants. ACO is inspired by the foraging behavior of ants. At the core of this behavior is the indirect communication between the ants with the help of chemical pheromone trails, which enables them to find short paths between their nest and food sources.

ACO optimizes a problem by having an updated pheromone trail and moving these ants around in the search space according to simple mathematical formulae over the transition probability and total pheromone in the region. At each iteration,ACO generates global ants and calculates their fitness. Update pheromone and edge of weak regions. If fitness is improved, then move local ants to better regions, otherwise select new random search direction. Update ant pheromone and evaporate ant pheromone.The continuous ACO is based on both local and global search. Localants have the capability to move toward latent region with best solution with respect to transition probability of region k,

$$\mathbf{Pk(t)}=\frac{\boldsymbol{tk(t)}}{\sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \boldsymbol{tj(t)}}$$

where tk(t) total pheromone at region k and n is number of global ants.

Pheromone is updated using the following equation:

(10.4)ti(t+1)=(1−r)ti(t)
where r is the pheromone evaporation rate.

The probability of selection of region for local ants is proportional to pheromone trail. The process flow for ACO is shown.

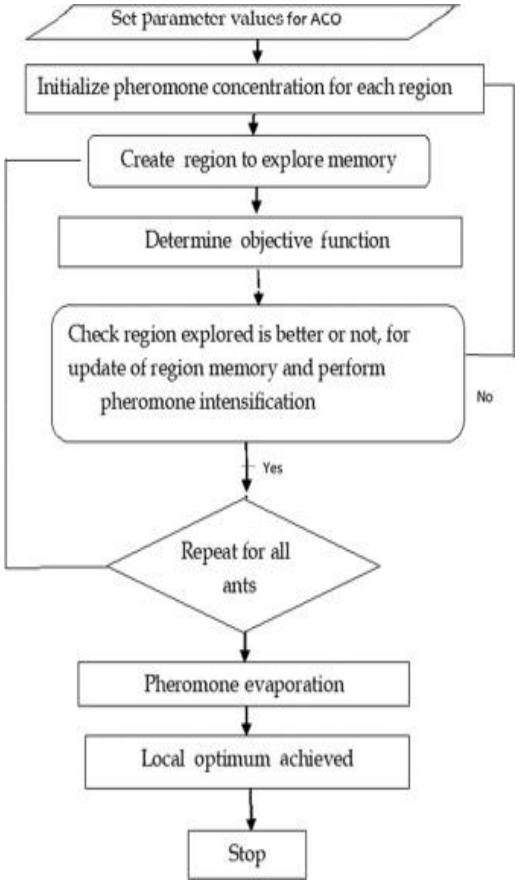

Fig 1: ACO Workflow Process

1. Performance:ACO has the ability to find optimal solution in less computational time. The performance of ACO may be improved by introducing approaches like modification of transition rule, parallel ACO [26]. It may be hybridized with other techniques for better results.

2. Applications: The first combinatorial problem solved using ACO was Travelling Salesman Problem (TSP) in 1991 by Dorigo in his Phd dissertation. The nest two applications of ACO were the Quadratic Assignment Problem (QAP) and Job Shop Scheduling 1994. Then it was applied to network routing applications, vehicle routing problems, sequential ordering, graph colouring problems and design of algorithms for knowledge representation structures.

3. Particle Swarm Optimization:
The PSO algorithm, a metaheuristic algorithm that is appropriate to optimize nonlinear continuous functions. The author derived the algorithm inspired by the concept of swarm intelligence, often seen in animal groups, such as flocks and shoals. A swarm of birds flying over a place must find a point to land and, in this case, the definition of which point the whole swarm should land is a complex problem,since it depends on several issues,that is, maximizing the availability of food and minimizing the risk of existence of predators. In this context, one can understand the movement of the birds as a choreography;

© 2021 IJSRET

the birds synchronically move for a period until the best place to land is defined and all the flock lands at once.

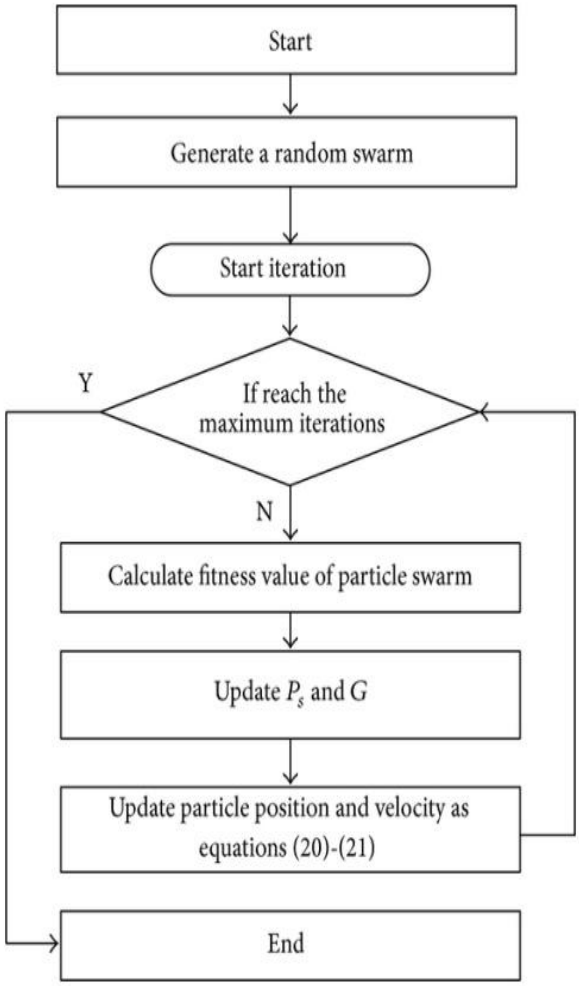

Fig 2: PSO Working Sphere

3.1 Evolution: The particle swarm optimization (PSO) algorithm, proposed by Kennedy and Eberhart, is a metaheuristic algorithm based on the concept of swarm intelligence capable of solving complex mathematics problems existing in engineering. It is of great importance noting that dealing with PSO has some advantages when compared with other optimization algorithms, once it has fewer parameters to adjust, and the ones that must be set are widely discussed in the literature.

3.2 Methodology: PSO is a global optimization algorithm which deals with problems which tends to find the best solution as a point in a D-dimensional space. In PSO, particles fly in the respective problem space by following the current optimum flying particles. Each of the particle keeps a track of its position in terms of coordinates in the problem space which are contributes to the best solution so far. Particles then move in solution space and they are evaluated after each iteration through some fitness functions.

3.3 Performance: PSO is a population based algorithm based on the cooperation of each particle. The convergence ability of PSO is faster than of other optimization techniques. It requires fewer parameters for the calculation of optimizing value.Number of particles may be decreased to increase the performance. It may be hybridized to further increase its performance.

3.4 Applications: PSO have a vast range of applications. It may be applied to optimize multi-objective problems, T-way software testing, mini-max problems , image classification, job shop scheduling, artificial neural networks, gesture recognition and many more.

4. Genetic Algorithm:

Genetic Algorithm is an adaptive heuristic search method based on the population genetics. It is probabilistic search method inspired from the process of natural selection and reproduction. It is used to generate solutions of optimization search problems. It belongs to the class of evolutionary algorithms which uses initialization, selection, crossover and mutation.

4.1 Basic Terms Used:

4.1.1Population− It is a subset of all the possible (encoded) solutions to the given problem. The population for a GA is analogous to the population for human beings except that instead of human beings, we have Candidate Solutions representing human beings.

4.1.2Chromosomes− A chromosome is one such solution to the given problem.

4.1.3Gene−A gene is one element position of a chromosome.

4.1.4Allele− It is the value a gene takes for a particular chromosome.

4.1.5Genotype−Genotype is the population in the computation space. In the computation space, the solutions are represented in a way which can be easily understood and manipulated using a computing system.

4.1.6Phenotype− Phenotype is the population in the actual real world solution space in which solutions are represented in a way they are represented in real world situations.

4.1.7Decoding and Encoding− Decoding is a process of transforming a solution from the genotype to the phenotype space, while encoding is a process of transforming from the phenotype to genotype space. Decoding should be fast as it is carried out repeatedly in a GA during the fitness value calculation.

© 2021 IJSRET

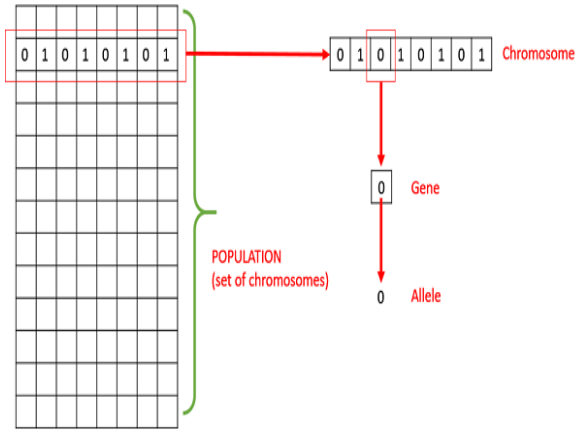

Fig 3: Gene, Chromosome,Allele and Population

4.2Evolution: John Holland introduced Genetic Algorithm in his book named „Adaptation in Natural and Artificial Systems‟ in 1975 [31]. Then John Koza has used genetic algorithm in 1992 to evolve programs to perform certain tasks. He called his method as "Genetic Programming" (GP). Some improvements in GA are listed below in Table 4.

4.3 Methodology: A GA starts with a set of solutions called as population. It consists of four operators:(1) Initialization(2) Selection,(3) Crossover and(4) Mutation. Initialization operator creates the initial population and assignsa fitness function which evaluates the fitness value. The selection operator chooses the chromosomes from population for mating. Crossover operator is used for sharing the information between two chromosomes. Mutation operator alters one or more gene values of a chromosome from its initial state. The process of evolution is repeated until end condition of the problem is satisfied.

4.4 Performance: GA is a search technique used to find the exact or approximate solutions to an optimization problem.GA exhibits “inherent parallelism” as the evaluation of individuals within a population is conducted simultaneously. It generally finds the global optima in complex spaces, hence GA is fast[32]. GA may be hybridized with any other search method to achieve an optimization goal.

SUMMARY

All the algorithms discussed in the above section are optimization problems. These are compared with each other and every algorithm has its own advantages. Genetic Algorithm is much more popular because of its parallel computation. One more advantage of Genetic Algorithm is that it may handle both continuous and discrete variable without any gradient information where as all other techniques may give best performance for continuous problems but need slightmodifications to handle discrete variables36]. The variations used are shown in the Table 5.

CONCLUSION

In this research paper, we examined various optimization techniques to find a best solution by a comparative analysis based on evolution, methodology, performance and applications.In this paper,we find that these algorithms may be applied in various domains whether using as a direct approach or in any modified version.In future, new improved algorithms can be found for different optimization algorithms in different approach areas.

REFERENCES

[1] K. K. Aggarwal & Yogesh Singh,“Software Engineering Programs Documentation, Operating Procedures,” New Age International Publishers, Revised Second Edition – 2005.
[2] Chen, Jun, and Mahdi Mahfouf. "Artificial immune systems as a bio-inspired optimization technique and its engineering applications." Handbook of Research onArtificial Immune Systems and Natural Computing: Applying Complex Adaptive Technologies: Applying Complex Adaptive Technologies (2009): 22.
[3] Philip TCox and Baoming Song, “A formal Model for Component-BasedSoftware”, IEEE Computer Society Document number 07695-474-4/01,2001, pp.304-310.
[4] Karaboga, Dervis. An idea based on honey bee swarm for numerical optimization.Vol.200. Technical report-tr06,Erciyes university, engineering faculty, computer engineering department, 2005.
[5] Alaya,I.,C.Solnon,andK.Ghedira."AntColonyOptimiz ation for Multi-Objective Optimization Problems." In Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on, vol.1, pp. 450-457. IEEE, 2007.
[6] Kulkarni, Nandakishore J.,K. Venkat Naveen, Puneet Singh,and Praveen Ranjan Srivastava."Test Case Optimization Using Artificial Bee Colony Algorithm."Advances in Computing and Communications(2011): 570-579.
[7] Karaboga, Dervis,and Bahriye Akay. "A comparative study of artificial bee colony algorithm." Applied Mathematics andComputation214,no.1(2009):108132.
[8] Zhu, Guopu, and Sam Kwong."Gbest-guided artificial bee colony algorithm for numerical function optimization." Applied Mathematics and Computation 217, no.7 (2010): 3166-3173.
[9] Karaboga, Dervis, and Beyza Gorkemli. "A combinatorial artificial bee colony algorithm for traveling salesman problem."In Innovations in Intelligent Systems and ApplicationsINISTA,2011 International Symposium on, pp. 50-53. IEEE, 2011.
[10] Li,JunQing,QuanKePan,andKaiZhouGao."Paretobase d discrete artificial bee colony algorithm for multi objective flexible job shop scheduling problems."The

International Journal of Advanced Manufacturing Technology55, no.9-12 (2011): 1159- 1169.

[11] Li, Junqing, Quanke Pan, and Shengxian Xie. "Flexible job shop scheduling problems by a hybrid arti ficial bee colony algorithm."In Evolutionary Computation (CEC), 2011 IEEE Congress on, pp. 78-83. IEEE, 2011.

[12] Karaboga, Dervis, and Celal Ozturk."Fuzzy clustering with artificial bee colony algorithm." Scientific research and Essays5, no.14 (2010): 1899-1902.

[13] Akay,Bahriye, and Dervis Karaboga. "Solving integer programming problems by using artificial bee colony algorithm." In AI* IA 2009: Emergent Perspectives in Artificial Intelligence, pp.355-364.

[14] Springer Berlin Heidelberg,2009.Jiao, Jian, Shan Yao, and Chunehe Xia."Application for artificial bee colony algorithm in migration of mobile agent." In Advanced Intelligent Computing Theories and Applications, pp.232-238.

[15] Springer Berlin Heidelberg, 2010.C. Vargas Ben´ıtez and H. Lopes. Parallel artificial bee colony algorithm approaches for protein structure prediction using the 3dhp-sc model. Intelligent Distributed Computing IV, pages 255-264, 2010.

[16] J. Kennedy and R.C.Eberhart. Particle swarm optimization.In Proceedings of the IEEE International Conference on Neural Networks, volume4, pages 1942–1948 vol.4.IEEEPress,1995.

[17] Windisch, Andreas, Stefan Wappler, and Joachim Wegener."Applying particle swarm optimization to software testing." In Proceedings of the 9th annual conference on Genetic and evolutionary computation, pp. 1121-1128. ACM, 2007.

[18] Hemlata SUrade and Prof. Rahila Patel. Article: Study and Analysis of Particle Swarm Optimization:A Review.IJCA Proceedings on2nd National Conference on Information and Communication Technology NCICT(4):1-5, November 2011.

[19] van den Bergh, Frans, and A.P. Engelbrecht."A new locally convergent particle swarm optimizer." In Proceedings of the IEEE international conference on systems, man, and cybernetics, vol.7, pp.6-9. 2002.

[20] Parsopoulos, Konstantinos E., and Michael N. Vrahatis."Particle swarm optimization method in multiobjective problems." In Proceedings of the 2002 ACM symposium on Applied computing, pp.603-607. ACM, 2002.

[21] Ahmet,Bestoun,and Kamal Zamli."A Greedy Particle Swarm Optimization Strategy for T-way Software Testing."Journal of Artificial Intelligence 5.2 (2012): 85-90.

[22] Laskari, E.C., K.E. Parsopoulos, and M.N. Vrahatis. "Particle swarm optimization for minimax problems." In Evolutionary Computation, 2002. CEC'02.Proceedings of the 2002 Congress on, vol.2, pp.1576- 1581. IEEE, 2002.

[23] Dorigo, Marco, and Luca Maria Gambardella. "Ant colony system: a cooperative learning approach to the traveling salesman problem." Evolutionary Computation, IEEE Transactions on 1, no.1 (1997): 53-66.

[24] Bullnheimer B,Hartl RF, Strauß C" A new rank based version of the ant system-a computational study." Central European Journal for Operations Research and Economics 7(1997): 25–38.

[25] Dorigo M, Di Caro G," The ant colony optimization meta-heuristic", McGraw-Hill Ltd., UK, Maidenhead, UK, England, (1999) pp 11–32.

[26] Hong Liu, Ping Li and Yu Wen, "Parallel Ant colony optimization algorithm," in Proc. WCICA, 2006, pp 3222-3226.

[27] Maniezzo,Vittorio, and Alberto Colorni."The ant system applied to the quadratic assignment problem."Knowledge and Data Engineering,IEEE Transactions on 11.5 (1999): 769-778.

[28] Colorni, Alberto, et al. "Ant system for job-shop scheduling." Belgian Journal of Operations Research, Statistics and Computer Science 34.1 (1994): 39-53.

[29] R.Schoonderwoerd, O.Holland, J.Bruten et L. Rothkrantz, Ant-based load balancing in telecommunication networks, Adaptive Behaviour, volume5, numéro 2, pages 169-207, 1997.

[30] Bullnheimer, Bernd, Richard F. Hartl, and Christine Strauss. "An improved ant System algorithm for thevehicle Routing Problem." Annals of operations research 89 (1999): 319-328.

© 2021 IJSRET