Quest Journals
Journal of Software Engineering and Simulation
Volume 4 ~ Issue 8 (2018) pp: 32-38
ISSN(Online) :2321-3795 ISSN (Print):2321-3809
www.questjournals.org

Research Paper

Investigating scope of Hybrid Particle Swarm Optimization-Multi Verse Optimizer in Software quality

Ekta Nehra
Ph.D. Scholar, Dept. of CSE,OPJS University, Churu, Rajasthan, india

ABSTRACT: *Research work is considering dataset of C&K that is to be filtered using optimal solution for effort estimation to provide efficient solution in field of software quality. Present research is optimizing using a novel hybrid meta-heuristic optimization technique called HPSO-MVO (Hybrid Particle Swarm Optimization-Multi Verse Optimizer) (ORPD). An unconstrained benchmark test function is used in conjunction with a hybrid PSO-MVO methodology. Effort estimation components and optimization mechanisms employed in the proposed study include PSO and MVO, respectively. A comparison was made between PSO and MVO in the centre to see which was more successful in terms of simulating optimization in various scenarios.*

KEYWORDS: *Optimization, Software quality, PSO, MVO, Hybrid meta-heuristic optimization*

I. INTRODUCTION

In this context, a new hybrid meta-heuristic optimization algorithm Particle Swarm Optimization-Multi Verse Optimizer (HPSO-MVO) method is used to get optimized solution. This method uses a hybrid PSO-MVO technique on some unconstrained benchmark test functions.

1.1 OPTIMIZATION

In order to properly manage an organization's resources and increase shareholder value, optimization methods are a powerful set of tools. By focusing on a set of prioritized criteria or constraints, the purpose of optimization is to come up with the "best" design feasible. There are several benefits to expanding output, including as increased strength and reliability, longer lifespans, more efficiency, and greater ease of use. This method of decision-making is referred to as "optimization."

1.2 PSO

PSO evolves into a technique of evaluation. It exists in the form of a technique that is easy to apply and put into practice on a regular basis. It has already been determined that such evaluation approaches uncover the best potential solution in a timely way. This strategy may be defined as a way that can optimize any issue in the realm of information technology. It has been noticed that in a PSO-based model, attempts are made one at a time to improve the performance of the candidate solution. It addresses any population-related problem with possible solutions. The dubbed particles travel about in search-space. This methodology works by applying an arithmetical rule on the particle's location and velocity. Its well-known domestic location has a significant influence on its mobility. This site has been modified with improved locations. Other particles may readily identify these sites. The swarm is predicted to migrate toward the best options as a result of this. PSO is a good heuristic because it makes minimal, if any, assumptions about the issue that has to be solved. Meta heuristics like PSO, on the other hand, do not ensure that an optimum solution will ever be identified. In the current circumstance, met heuristics are the most significant and beneficial since they have shown success in a variety of optimization issues when implemented. It's a self-contained system. It defined the degree to which these complex systems were active. In order to deal with optimization issues, a cooperative and intelligent structure employs an exceedingly streamlined model of social behavior.

1.3 MVO

MVO is a new type of invention. It is an effective maximization method which gets encouragement from environment. Mirjalili et al invented this. For putting this in to operation, two customized factors were kept in mind by them. This method is invented by using three ideology of cosmology. In addition to this form, it also becomes famous in new form of meta-heuristic optimization method. It efficiently figures out those problems which are related to OPF. It is a method which gets continuous motivation from living body & social science stand point. In working of this method different ideology of cosmology are bring in to use. In addition to idea of white & black hole, concept of wormhole is also used in this method. One of most important strong point of this method is that it will find out fast rate of intersection. For this purpose it use roulette wheel selection. In addition to this, this algorithm is able to deal with regular & discrete optimization issues.

1.4 HYBRID PSO-MVO

Research work is making use of hybrid PSO- MVO. The derivation process of deriving equation for Hybrid PSO-MVO has been discussed below:

Phase 1: In phase one equation of PSO is considered.

Particle Swarm Optimization

Mechanism has been inspired by social expression of birds or fishes. The PSO consists of P_{best} , G_{best}. Position and velocity are updated over course of iteration from these mathematical equations:

$$v_{ij}^{t+1} = wv_{ij}^{t} + C_1R_1(Pbest^t - X^t) + C_2R_2(Gbest^t - X^t) \ldots\ldots(1)$$

$$X^{t+1} = x^t + v^{2t+1}(i = 1{,}2..NP) And\,(J = 1{,}2..NG) \ldots\ldots(2)$$

Where

$$W = w^{\max} - \frac{(w^{\max} - w^{\min}) * iteration}{\max iteration} \ldots\ldots(3)$$

w^{max} = 0.4

w^{min} = 0.9. v_{ij}^{t} , v_{ij}^{t+1} has been considered velocity of “j” member of “i” particle in iteration number (t) as well as (t + 1) . (Usually $C_1 = C_2 = 2$), r_1 and r_2 Random number (0, 1).

Phase 2: Multi-verse optimizer equation

The major inspiration for the MVO algorithm comes from three concepts: black hole, white hole, and wormhole. Exploitation, exploration, and local search are all modelled mathematically to assess their relative merits. The white hole is thought to be the primary cause of the cosmos' birth. The enormous gravitational pull of black holes attracts everything it touches. As time/space transit conduits, wormholes allow items to move quickly across the cosmos. Steps that MVO worlds take:

1. There is a larger chance of a white hole forming if the inflation rate is higher.
2. The likelihood of a black hole's occurrence decreases with increasing inflation.
3. White holes in universes with higher inflation rates are used to transport substances.
4. More chemicals are accepted by black holes in universes with lower inflation rates.

Deriving hybrid PSO-MVO equation

PSO and MVO are combined in the Hybrid PSO-MVO set. Combining PSO and MVO strengths to get a targeted optimal solution is known as hybrid PSO-MVO. As a result, the PSO Pbest value has been replaced with the MVO Universe value.

$$v_{ij}^{t+1} = wv_{ij}^{t} + C_1R_1(Universes^t - X^t) + C_2R_2(Gbest^t - X^t) \ldots\ldots (11)$$

II. LITERATURE REVIEW

S No.	Author / Year	Title	Methodology	Limitation
1	Van /2003	Data clustering using particle swarm optimization	PSO	Lack of performance and Hybrid Particle Swarm Optimization-Multi Verse Optimizer
2	Wang/ 2010	Effective Feature Selection with Particle Swarm Optimization based One-dimension Searching	PSO	Research work does not perform on 2D and 3D.
3	Xiangying /2010	Parameters Optimization in SVM Based-on Ant Colony Optimization Algorithm	Optimization	Parameters of MVO and PSO doesnot explain.
4	Sun / 2019	A Survey of Optimization Methods From a Machine Learning Perspective	Optimization	In this research, this method is not applicable for PSO.
5	Zhang / 2015	Optimizing parameters of support vector machines using team-search-based particle swarm optimization	PSO	Lack of quality of software.
6	Mohamed / 2019	Software Component Quality Model	Software quality	Research does not contact optimization.
7	Kamaldeep Kaur / 2010	Soft computing approaches for prediction of software maintenance effort	Software quality	Working of software is low due to lack of HPSO.
8	Reenu / 2015	OPEN SOURCE SOFTWARE MAINTENANCE COST EVALUATION USING	Software quality	Research does not work on optimization for software.

		MAINTAINABILITY INDEX AND CODE METRICS		
9	Punia / 2014	Software maintainability prediction using soft computing techniques	Software quality	Techniques donot work on quality of software.
10	Basudhar / 2010	An improved adaptive sampling scheme for the construction of explicit boundaries	Optimization	Building of software is not odes in good quality.

III. PROBLEM FORMULATION

During software development the effort estimation is considered difficult task. However there have been many researchers who have presented the research in this field but they failed to provide optimized solution to detect the effort estimation. Moreover there is need to introduce optimization mechanism that should provide accurate result in less time. Thus there is need to introduce hybrid mechanism to achieve this objective. The proposed model is supposed to provide optimized solution by integration of MVO and PSO.

To put it another way, estimating the amount of work involved in software development is a challenging undertaking. Many researchers, on the other hand, have published their work in this area, but none have been able to come up with an optimum method for estimating the effort. Furthermore, an optimization process must be implemented that delivers correct results in a shorter period of time. This necessitates the use of a hybrid mechanism. According to the theory, the suggested model will provide an optimal solution by combining MVO with PSO.

IV. PROPOSED APPROACH

Proposed work includes elements like pre and post effort estimation components and optimization mechanism used is PSO and MVO respectively. A comparison has been done in the in the middle of PSO and MVO for the purpose of determining their effectiveness.

4.1 Initial Phase of Effort Estimation

The method which is introduced here for the purpose of determining work before coding stage effort estimation derives on the basis of Nageswaran design [5]. On the basis of that, new progresses are proposed here i.e.

Not only the values obtained related to UAW, UUCW, TEF but the determined examination work are also made optimal with the help of PSO and MVO. Act of rendering optimal provide assistance in the determination of weights and maximum values in support of levels related to activation. Act of rendering optimal is carried out on the basis of various efforts. After that, optimal findings is delivered in the company of information in support of those plans for which determination is required. These information are transmitted out of model written details. The optimization mechanism provides with effort in terms of person-months.

4.2 Optimization of Initial effort estimation

During optimization of pre coding effort estimation the data set that has been used for training in previous neural network research is passed to both PSO and MVO optimizers respectively.

Optimization process: In order to get the optimized pre coding phase effort the optimization function is created. Upper bond and lower bond is set on the bases of dataset. Maximum iterations is also set in order to repeated in various iterations. The global best and local best are extracted from this simulation. This global best solution supports the pre coding phase effort estimation.

Comparison operation

The accuracy and performance of both optimizers are checked during execution of optimizers. The simulation would conclude which optimizer is best in order to get more accurate result in less time.

4.3 Final phase effort estimation

It emerges in the form of phase where the written details related to coding are used by the design manager for the purpose of determining examination work. The method which is introduced here considered that the examination work is entirely depends upon the inputs and outputs number strength, code complexity and its criticalness.

A value is given by the all other important parameters.

Variables component: Examination cases figure is directly proportional to input figure. It means it increment and decrement is entirely depends upon input figure. In support of various inputs number of arrangements is already provided. It becomes possible to observe out of Table four. The method which is introduced here considered that character data type doesn't need more than single test data, whereas integer information should need additional examination cases and array variable would need further examination cases in support of examination [1]. It is the way in which allocated weights increase in a proportion manner. Values related to the appearance of all varying figure is demanded by var[i] according to the sequence which is given within the Table four. Var_comp[i] exists in the form of appointed load that are demanded out of Table four. In this way, va-riable var_val is obtained by addition of product of number of their assigned weights & occurrences of variables.

Complexity component:

Exact test case figure needed for the purpose of research is measured on the basis of code abstruseness. Therefore, a value in support of used code abstruseness is provided in the table five. Appointed load is directly proportional to code abstruseness. It means its increment and decrement is entirely depends upon code abstruseness.

V. RESULTS AND DISCUSSION

The simulation has made comparison in case of PSO based and hybrid for pre coding effort estimation

MVO optimization technique has been used in order to improve accuracy &performance during finding optimal solution from precoding and post coding effort estimation.

The results obtained by PSO considering 5000 iteration for initial effort estimation is as follow

Optimal solution found in case of PSO is 0
Best objective value 0.3532
Elapsed time is 22.44 seconds.

The results obtained by hybrid considering 5000 iteration for initial effort estimation is as follow

The best solution obtained by hybrid is: 0.29295
The best optimal value objective function found by hybrid is: 0.349
Elapsed time is 11.63 seconds.

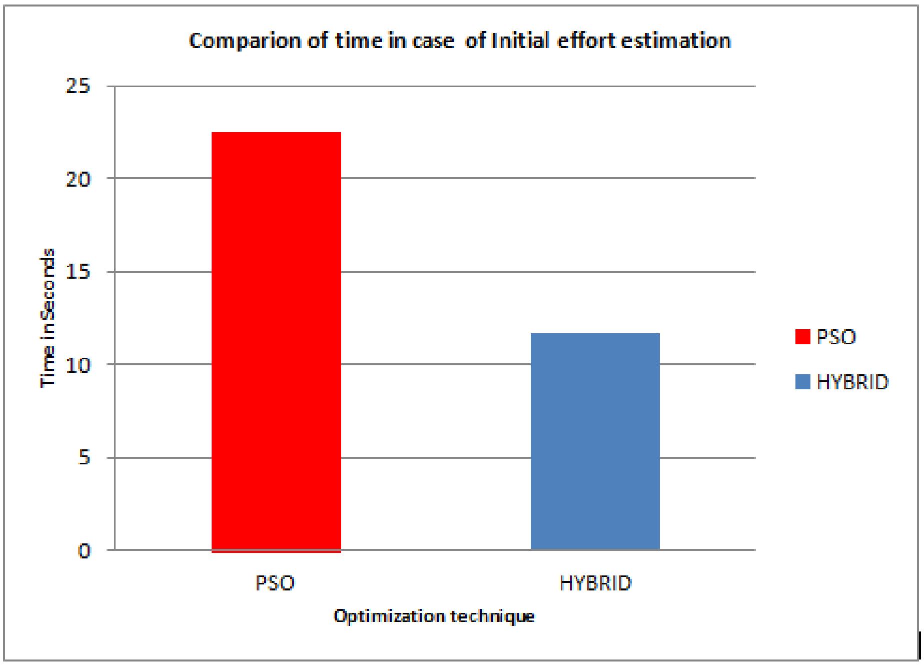

Fig 1 Comparison of time in case of initial effort estimation

COMPARISON IN CASE OF PSO BASED AND MVO BASED POST CODING EFFORT ESTIMATION

The results obtained by PSO considering 5000 iteration for post coding effort estimation is as follow

Optimal solution found is 0.0102
Best objective value 0.3012
Elapsed time is 11.116989 seconds.

The results obtained by hybrid considering 5000 iteration for post coding effort estimation is as follow

The best solution for **post coding effort estimation** obtained by hybrid is : 0.01025

The best optimal value for **post coding (final) effort estimation** of the objective funciton found by MVO is : 0.30118

Elapsed time is 5.142825 seconds.

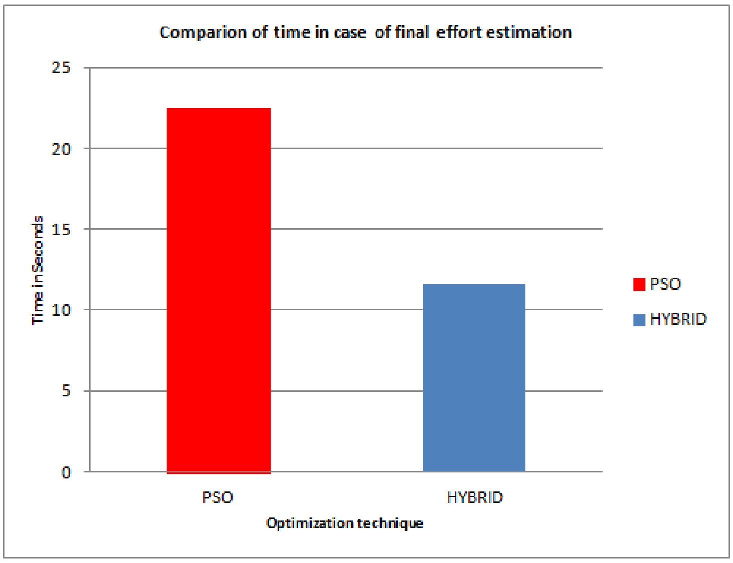

Fig 2 Comparison of time in case of post coding (final) effort estimation

Comparison in case of PSO based and Hybrid based Initial effort estimation considering various iterations

Comparison of time in case of PSO based and hybrid based initial effort estimation has been made in following chart. It has been observed that the time consumption in case of hybrid based pre coding effort estimation is less as compare to PSO based initial effort estimation. Results are presenting if the iterations are growing then the difference in time consumption is also increasing.

Table 4.1 Comparison of time for Hybrid and PSO in case of pre coding effort estimation considering different iterations

Iteration	Pso Based initial Effort Estimation	Hybrid Based initial Effort Estimation
2000	11.42	9.86
3000	16.89	11.30
4000	21.60	14.94
5000	27.63	14.037
6000	36.38	15.58
7000	38.61	17.20
8000	43.97	18.58
9000	52.74	21.36
10000	78.42	21.37

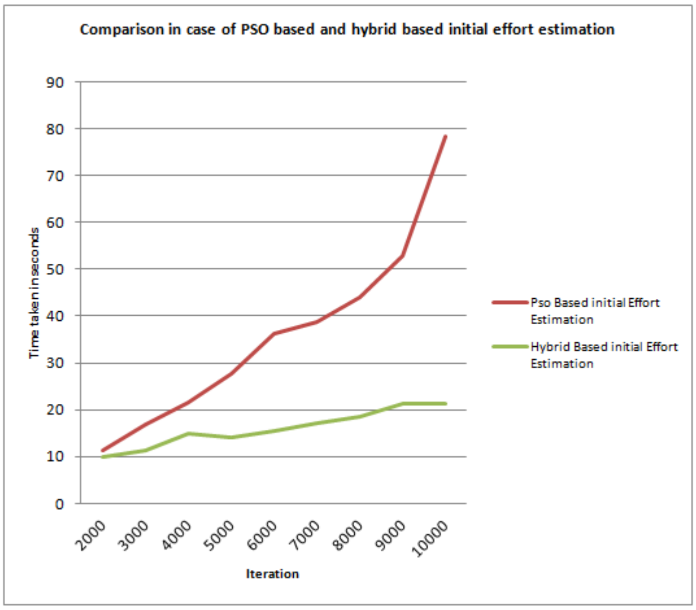

Fig 3 Comparison in case of PSO based and hybrid based initial effort estimation

Comparison in case of PSO based and Hybrid based final effort estimation considering various iterations

Comparison of time for PSO based and hybrid based final effort estimation has been made in following chart. It has been observed that the time consumption in case of hybrid based final effort estimation is less as compare to PSO based final effort estimation. Results are presenting if the iterations are growing then the difference in time consumption is also increasing.

Table 4.2 Comparison of time for Hybrid and PSO in case of post coding effort estimation considering different iterations

Iteration	PSO Based final Effort Estimation	Hybrid Based final Effort Estimation
2000	5.81	4.56
3000	8.33	5.12
4000	10.80	5.75
5000	13.32	6.15
6000	15.93	6.95
7000	18.56	7.42
8000	21.19	8.06
9000	23.84	8.78
10000	26.67	9.22

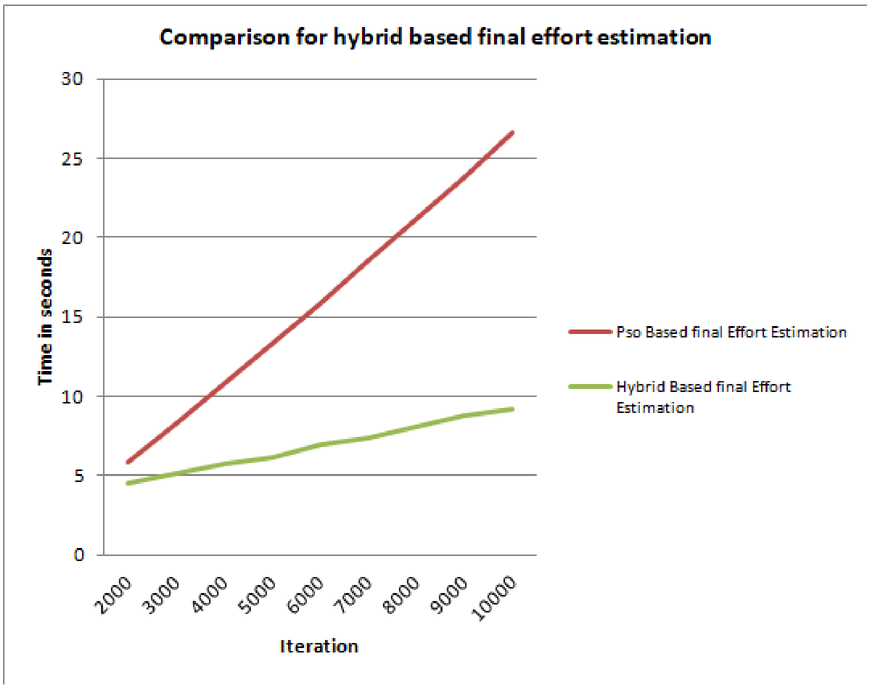

Fig 4 Comparison for hybrid based final effort estimation

VI. CONCLUSION

Effort estimation components and optimization mechanisms employed in the proposed study include PSO and MVO, respectively. A comparison was made between PSO and PSO-MVO hybrid in the centre to see which was more successful in terms of simulating optimization in various scenarios. Simulation concludes that hybrid approach is consuming less time as compare to PSO based effort estimation in post coding as well as pre coding during effort estimation in field of software quality.

Future of research

Such research is significant for the research work where soft computing approaches are used. During prediction and estimation optimization plays significant role. Moreover data filtering before machine learning could reduce the time consumption and increase the accuracy. Present approach could be used in other AI based system.

REFERENCE

[1]. D. W. Van Der Merwe and A. P. Engelbrecht, "Data clustering using particle swarm optimization," 2003 Congr. Evol. Comput. CEC 2003 - Proc., vol. 1, no. June, pp. 215–220, 2003, doi: 10.1109/CEC.2003.1299577.

[2]. Jun Wang, Yan Zhao,Ping Liu "Effective Feature Selection with Particle Swarm Optimization based One-dimension Searching". Systems and Control in Aeronautics and Astronautics (ISSCAA), Pages 702-705. 2010

[3]. Xiangying Liu “Parameters Optimization in SVM Based-on Ant Colony Optimization Algorithm” Advanced Materials Research Vols. 121-122 (2010) pp 470-475

[4]. S. Sun, Z. Cao, H. Zhu, and J. Zhao, “A Survey of Optimization Methods From a Machine Learning Perspective,” IEEE Trans. Cybern., pp. 1–14, 2019, doi: 10.1109/tcyb.2019.2950779

[5]. Zhang, L., & Wang, J. (2015). Optimizing parameters of support vector machines using team-search-based particle swarm optimization. Engineering Computations, 32(5), 1194–1213. https://doi.org/10.1108/EC-12-2013-0310

[6]. Mohamed Abdullahi Ali, Ng Keng Yap, “Software Component Quality Model”, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958,Volume-9, Issue-1, October 2019

[7]. Kaur, Arvinder, Kamaldeep Kaur, and Ruchika Malhotra. "Soft computing approaches for prediction of software maintenance effort." International Journal of Computer Applications 1, no. 16 (2010): 69-75.

[8]. Rani, Reenu, Dipen Saini, and Amandeep Kaur Cheema. "OPEN SOURCE SOFTWARE MAINTENANCE COST EVALUATION USING MAINTAINABILITY INDEX AND CODE METRICS."

[9]. Punia, Mamta, and Amandeep Kaur. "Software maintainability prediction using soft computing techniques." IJISET 1, no. 9 (2014): 431-442.

[10]. Basudhar, A. and Missoum, S. (2010), “An improved adaptive sampling scheme for the construction of explicit boundaries”, Structural and Multidisciplinary Optimization, Vol. 42 No. 4, pp. 1-13.

[11]. Singh, Charu, Amrendra Pratap, and Abhishek Singhal. "An estimation of software reusability using fuzzy logic technique." In 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014), pp. 250-256. IEEE, 2014.

[12]. Kumar, Lov, and Santanu Ku Rath. "Software maintainability prediction using hybrid neural network and fuzzy logic approach with parallel computing concept." International Journal of System Assurance Engineering and Management 8, no. 2 (2017): 1487-1502.

[13]. Kundu, Shivani, and Kirti Tyagi. "Effort estimation of software maintainability using soft computing techniques: a critical literature survey." World Applied Sciences Journal 34, no. 6 (2016): 733-742.

[14]. Dubey, Sanjay Kumar, and Ajay Rana. "A fuzzy approach for evaluation of maintainability of object oriented software system." International Journal of Computer Applications 49, no. 21 (2012).

[15]. Sharawat, Mr Sandeep. "Software maintainability prediction using neural networks." environment 3, no. 5 (2012): 750-755.

[16]. Zavvar, Mohammad, and Farhad Ramezani. "Measuring of software maintainability using adaptive fuzzy neural network." International Journal of Modern Education & Computer Science 7, no. 10 (2015): 27-32.

[17]. Mishra, Swati, and Arun Sharma. "Maintainability prediction of object oriented software by using adaptive network based fuzzy system technique." International Journal of Computer Applications 119, no. 9 (2015).

[18]. Yuan, Xiaohong, Taghi M. Khoshgoftaar, Edward B. Allen, and K. Ganesan. "An application of fuzzy clustering to software quality prediction." In Proceedings 3rd IEEE Symposium on Application-Specific Systems and Software Engineering Technology, pp. 85-90. IEEE, 2000.

[19]. Barbosa, Nelson, and Kechi Hirama. "Assessment of software maintainability evolution using C&K metrics." IEEE Latin America Transactions 11, no. 5 (2013): 1232-1237.

[20]. Zhou, Yuming, and Hareton Leung. "Predicting object-oriented software maintainability using multivariate adaptive regression splines." Journal of systems and software 80, no. 8 (2007): 1349-1361.

[21]. Tahir, Amjed, and Rodina Ahmad. "An AOP-based approach for collecting software maintainability dynamic metrics." In 2010 Second International Conference on Computer Research and Development, pp. 168-172. IEEE, 2010.

[22]. Ludwig, Jeremy, Steven Xu, and Frederick Webber. "Static software metrics for reliability and maintainability." In 2018 IEEE/ACM International Conference on Technical Debt (TechDebt), pp. 53-54. IEEE, 2018.

[23]. Ostberg, Jan-Peter, and Stefan Wagner. "On automatically collectable metrics for software maintainability evaluation." In 2014 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement, pp. 32-37. IEEE, 2014.

[24]. Basudhar, A. and Missoum, S. (2010), “An improved adaptive sampling scheme for the construction of explicit boundaries”, Structural and Multidisciplinary Optimization, Vol. 42 No. 4, pp. 1-13.

[25]. Masmali, Omar, and Omar Badreddin. "Comprehensive Model-Driven Complexity Metrics for Software Systems." In 2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 674-675. IEEE, 2020.

[26]. Shepperd, M. J. "System architecture metrics for controlling software maintainability." In IEE Colloquium on Software Metrics, pp. 4-1. IET, 1990.

[27]. Genero, Marcela, Mario Piattini, Esperanza Manso, and Giovanni Cantone. "Building UML class diagram maintainability prediction models based on early metrics." In Proceedings. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry (IEEE Cat. No. 03EX717), pp. 263-275. IEEE, 2004.

[28]. Garcia, Felix, Francisco Ruiz, and Corrado Aaron Visaggio. "A proposal and empirical validation of metrics to evaluate the maintainability of software process models." In 2006 IEEE Instrumentation and Measurement Technology Conference Proceedings, pp. 1093-1097. IEEE, 2006.

[29]. Abílio, Ramon, Pedro Teles, Heitor Costa, and Eduardo Figueiredo. "A systematic review of contemporary metrics for software maintainability." In 2012 Sixth Brazilian Symposium on Software Components, Architectures and Reuse, pp. 130-139. IEEE, 2012.

[30]. Jin, Cong, and Jin-An Liu. "Applications of support vector mathine and unsupervised learning for predicting maintainability using object-oriented metrics." In 2010 Second International Conference on Multimedia and Information Technology, vol. 1, pp. 24-27. IEEE, 2010.

[31]. Oman, Paul, and Jack Hagemeister. "Metrics for assessing a software system's maintainability." In Proceedings Conference on Software Maintenance 1992, pp. 337-338. IEEE Computer Society, 1992.