xIlkogretim Online - Elementary Education Online,2021; Vol 20 (Issue 5): pp.1877-1887
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.01.200

Optimized Cost Estimation Model in early stage software- A review

Shweta KR, Ph.D Research Scholar, Dept. of Computer Science, Chikkanna Govt. Arts College, Tirupur, Tamilnadu, India.
Duraisamy S, Assistant Professor, Dept. of Computer Science, Chikkanna Govt. Arts College, Tirupur, Tamilnadu, India.
Latha Maheswari T., Assistant Professor, Dept. of Computer Science, Sri Krishna College of Engineering and Technology, Coimbatore, Tamilnadu, India.

Abstract: Software Project Management is an esteemed activity by team effort for giving assurance to product delivery. Three primary required factors are cost, quality and time. Estimation of those three factors in earlier stage is a difficult process due to under and over estimate constraints. Both the under and over estimate constraints in project life cycle leads loss of project. Hence we required a best suitable model for selecting the optimal criteria's for cost estimation as well as effort estimation. Cost by project manager and effort by manpower needs an indicator to fine tuning the method for improving the accuracy of an efficient industry. For this purpose more than the last 30 years many researchers proposed many algorithmic and non-algorithmic estimation methods using optimization. This paper focuses on the brief summary of the various optimized cost estimation models in the latest trends. Based on this area the target of cost and effort estimation in the earlier stage is optimized in future.

Keywords: Cost, Effort, Time, Quality, Estimation, Optimization

I. INTRODUCTION

In a project, quite possibly the most significant is to do an arranging and spending plan for the undertaking cost programming. Cost assessment for programming project is a troublesome issue in administration programming, programming project costs remember the product for the way toward being framed the amount of the relative multitude of costs, and it incorporates the expense of from improvement to a test, programming the executives and quality administration costs, and incorporates the expense, inadvertent revise, and cost assessment is going to be programming advancement or is creating programming project required responsibility and work progress conjecture, bringing about adequate, gives out a gathering of guess blunder scope of preparation.

Comprehend the product cost assessment of the idea and strategy for development, which can assist us with improving programming cost assessment model, in order to build up the down to earth programming cost assessment apparatuses Software improvement cost assessment fundamentally alludes to the product advancement measure costs responsibility and the comparing cost. Not the same as the customary modern items, programming costs do exclude the crude materials and energy utilization, principally is the human work utilization. Moreover, the product likewise without a reasonable assembling measure, its improvement costs are in one-time advancement interaction of the all out cost to compute. Hence, the product improvement quotes, ought to from programming plans, necessity examination, plan and coding, unit test, mix test to the affirmation test, spent the whole advancement measure dependent on the cost.

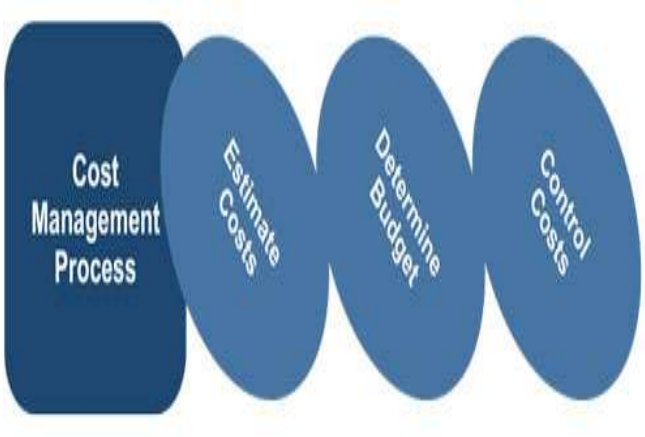

Figure 1: Management of Cost Estimation

As per the reports, in 1994, 8400 activities in Information Technology (IT) industry at Standish Group in America of the aftereffects of the examination show that: the project all out normal spending plan past amount is 90%, progress past the measure of complete 120%, project for both the spending plan, and 33 percent, in a major organization postpone progress of the undertaking, 9% as indicated by spending plan, the advancement finish.

II. LITERATURE REVIEW

The main objective of this paper is to give a brief detail about software cost estimation models especially using optimization methods. Collection of well defined activity combined together is called as project. All the methods have its own uniqueness. So no one can predict the accurate method for specified project activity. Thus incorporate the study of all the existing methods gives adequate details to producing new models for achieving the goal of a project.

Cost Estimation:

Programming frameworks and applications are the most costly piece of the PC frameworks because of the human exertion that is utilized in delivering programming frameworks. This reason (broad programming improvement) motivates numerous analysts to zero in on this part of examination. Programming cost assessment or what is called by different analysts "exertion expectation" is the way toward assessing programming cost precisely. In the course of recent years, numerous investigations have been led in the product cost assessment field where two primary sorts of cost assessment techniques have been examined including the algorithmic and non-algorithmic strategies.

Algorithmic models:

Algorithm is a step by step procedure. All the sequence of steps is used to accomplish a particular task using some mathematical equations. Attributes are configured using some inputs Line of Source Code (LoSC), functions (F) to perform specified operations, overall planning and design with Risk factor Analysis (RA) and so on.

Empirical model (COCOMO)

New project is developed by using previous project ideas called as empirical models. Such type of Constructive Cost Model is called COCOMO by Boehm for achieving software quality through effort and schedule attributes.

At present as of now improvement COCOMO cost assessment model for COCOMO2. The dispatch of COCOMO2 primary reason for existing is to construct programming advancement cost information base and instruments

to help ceaseless turn of events; assessment model Provide quantitative examination structure, apparatus sets and abilities, to survey the product advancement strategies to improve the product improvement exertion and the impact of working timetable [3].

Analytical model (SEM)

Based on some common assumptions new predictions are developed to create a successful project. Putnam developed software effort estimation empirical model (SEM) for effort estimation.

Dimensional model (FP)

Allan developed function point analysis for measuring the size of function points to calculating infinite numbers.

Size model (SM)

Leung and Fan suggested some global size metrics to calculating infinite numbers.

Non-Algorithmic models:

These methods used only simple ideas and thoughts from various methods without any formulas. So user readability and understandability is always available in this method.

Top down models:

Top-down assessing strategy is additionally called Macro Model. Utilizing top-down assessing technique, a general expense assessment for the task is gotten from the worldwide properties of the product venture, and afterward the undertaking is divided into different low-level instrument or parts. The main technique utilizing this methodology is Putnam model. This strategy is more material to early cost assessment when just worldwide properties are known. In the beginning stages of the product advancement, it is valuable in light of the fact that there is no depth gritty data accessible.

Bottom up models:

Utilizing base up assessing technique, the expense of every product segments is assessed and afterward joins the outcomes to show up at an expected expense of generally speaking undertaking. It targets developing the gauge of a framework from the information amassed about the little programming segments and the communications. The main strategy utilizing this methodology is COCOMO's definite model.

Optimization models:

Ramaraj and Duraisamy, et al., (2007) proposed a quality model to survey the plan period of any article arranged framework dependent on crafted by Chidamber, Kemrer and Basili and recommend two new measurements. Their examination centers around dissecting a bunch of measurements, which has direct impact on the nature of the product and making a measurements instrument dependent on java that can be utilized to approve the item situated tasks against these measurements. Their investigation is done on a bunch of true activities planned utilizing Unified Modeling Language, which are utilized as experiments. These measurements and models are proposed to add greater quality data in refining any item situated framework during the beginning phases of plan itself [1].

Prabu P, Duraisamy S et al., (2015) built up a model for improving the viability of programming advancement measure utilizing Pair Programming by directing tests among representatives of valtech Software Company with extensively having different degrees of programming abilities. Their examination demonstrates that, subsequent to encountering pair programming, most representatives called attention to a more grounded inclination to work with another worker, accepted that matching made them more coordinated, and accepted that blending saved time on reasonable meetings. Their test results found that regardless of certain downsides, pair programming can be amazingly useful [2].

Xiaotie Qin et al., (2011) initially presented the product cost assessment, and afterward present three most valuable techniques for programming cost assessment, and on the grounds that COCOMO2 is utilized broadly, the article momentarily examines the COCOMO2 module finally. They discussed the product cost the executives is quite possibly the main elements in the advancement of programming industry. The expense the executives is a fundamental substance of programming project the board, as of now; the expense the board is a more vulnerable field of programming project the executives than different fields, while cost assessment is a vital field of compelling expense the board [3].

Iqbal e t al., (2020) used Function Points and applied some new models to get an unrivaled assessment of programming properties. The use of thoughts and qualities from the fluffy set speculation to loosen up work directs examination toward fluffy capacity focuses investigation. Fluffy theory attempts to develop formal quantitative plan prepared for emulating imprecision of the human data. With the capacity focuses made by Fuzzy FPA, a gauge an incentive for instance, costs/cost and programming improvement can be all the more effectively decided [4].

Seyed Hossein Hashemi et al., (2020) proposed new technique for assessing the exertion utilizing the cuckoo inquiry enhancement calculation and dependent on the past examination and with the point of deciding the ideal loads of upgrades of article arranged projects cost. The proposed strategy has been recreated and the outcomes show the nature of the proposed technique contrasted with other transformative calculations just as past examinations. The proposed technique has a lower size mistake rate than different strategies, so the blunder of proposed strategy utilizing cuckoo inquiry improvement calculation to decide ideal weight dependent on MMRE standard is not exactly other transformative calculations and crude examination [5].

Elnaz Eskandarian Miandoab et al., (2016) proposed, a mixture calculation that consolidates COA-Cuckoo improvement and K-Nearest Neighbors (KNN) calculations is utilized. The purported piece calculation runs on six distinctive informational indexes and is assessed dependent on eight assessment standards. The outcomes show an improved precision of assessed cost [6].

Anupama Kaushik et al., (2020) used Long momentary memory (LSTM) and intermittent neural organizations (RNN) to propose the early model for computing the exertion expected to create programming. The consequence of the model is assessed on COCOMO'81, NASA63 and MAXWELL datasets. The exploratory outcomes showed that LSTM-RNN with direct initiation work (LAF) alters the accuracy of cost assessment in correlation with different models used in the examination [7].

Nazeeh Ghatasheh et al., (2015) Firefly Algorithm is proposed as a metaheuristic streamlining strategy for improving the boundaries of three COCOMO-based models. These models incorporate the fundamental COCOMO model and other two models proposed in the writing as expansions of the essential COCOMO model. The created assessment models are assessed utilizing distinctive assessment measurements. Trial results show high exactness and huge blunder minimization of Firefly Algorithm over other metaheuristic advancement calculations including Genetic Algorithms and Particle Swarm Optimization [8].

Ping-Hao Zhang et al., (2020) the cycle of decided the ideal gathering factors in the choice tree calculation is chosen to understand the variable choice of the task. By examination with other expectation consequences of all-factor demonstrating, it is shown that this technique can be utilized for other information investigation, information mining and different applications. Simultaneously, with the expanding intricacy of the examination object and the expanding number of case tests, the meaning of the variable choice strategy and the improvement of displaying productivity will turn out to be increasingly huge. Accordingly, as another variable determination strategy, it will be a significant information technique saves in the treatment of explicit designing issues. Specifically, the low-dimensional examination and quick demonstrating of complex issues will assume a significant part [9].

Zahra Asheghi Dizaji et al (2014) have endeavored to appraise the expense of programming projects by utilizing Particle Swarm Optimization (PSO) calculation and Tent planning as Chaos Optimization calculation. We have utilized the NASA datasets as preparing and testing sets, and MARE to assess the exhibition of the proposed technique. Furthermore, the outcomes have been contrasted with the moderate COCOMO model proposed by Boehm (1981), which shows an Absolute Relative Error decrease in cost assessment up to 0.0797% [10].

Hassan Najadat et al., (2012) examined, a mixture cost assessment model is proposed to deliver a reasonable forecast model that thinks about programming project, item, measure, and ecological components. An expense assessment dataset is worked from countless open source projects. Those activities are partitioned into three spaces: correspondence, account, and game undertakings. Results showed that games applications have higher estimations of the SLOCmath, coupling, cyclomatic intricacy, and MCDC measurements. Data acquire is utilized to assess the capacity of item arranged measurements to foresee programming intricacy [11].

Chamkaur Singh et al., (2019) dissected the different exercises as per the current programming advancement climate and understanding the key factors that influence the support cost of the product. Alongside sorts of programming support assessment, different programming upkeep exercises are grouped into three classes. The other area contains the survey on work done by different scientists on the upkeep cost of programming and different components that influences it. These components are ordered into specialized and non-specialized elements [12].

Nishu Dewan et al., (2014) pronounced subterranean insect state enhancement was utilized to improve the boundaries of COCOMO model which were utilized to gauge the exertion. Different ways were created and ACO will pick the best way by registering likelihood of every way produced by insect province framework. The exhibition measure utilized for correlation is Mean Magnitude of Relative Error. The MMRE estimations of all the three created way is contrasted with the COCOMO esteem. The outcomes acquired showed that the proposed model (Path C) performed better compared to the COCOMO model as %MMRE an incentive for the proposed model were least when contrasted with different models. Subsequently gives better outcomes [13].

Hamayoon Ghafory et al., (2020) reviewed software cost assessment is the way toward anticipating the measure of exertion needed to construct a product framework. Models gives at least one numerical calculations that figure cost as an element of various factors, size is essential expense factor in many models and can be estimating utilizing lines of code (LOC). The models ought to be utilized to assess the expense of programming, SLIM is a helpful model to gauge the expense and it is a decent model for huge projects, likewise this model required a few boundaries to produce an assessment. [14].

Farhad Soleimanian et al., (2014) they contemplated SCE by utilizing a cross breed of Genetic Algorithm (GA) and Artificial Bee Colony (ABC) which are Meta-Heuristic Algorithms. Test outcomes show that proposed model, GA and ABC calculations have less MRE mistakes esteems than the COCOMO model. Likewise, the half breed model has better intermingling contrasting and the GA and ABC calculations [15].

Sabbagh Jafari et al., (2016) proposed COCOMO exertion assessment technique is enhanced utilizing Meta-heuristic amicability search Algorithm. Nasa dataset was utilized to test the outcomes. The reason for advancement strategies in programming endeavors assessment is to diminish the Mean Magnitude of Relative Error which for this situation prompted practically 21% improvement [16].

From the most recent couple of many years, numerous specialists work on advancement of the COCOMO model by utilizing normally motivated calculations. Such improvement calculations help the product business in anticipating exact and authentic estimations of cost and exertion utilized for programming project advancement and support. **Muhammad Sufyan Khan et al., (2016)** utilized another meta-heuristic calculation roused by the strawberry plant for streamlining of COCOMO exertion assessment technique. NASA 93 informational index is utilized in the proposed approach. The Magnitude of Relative Error (MRE) and Mean Magnitude of Relative Error (MMRE) is assessed. Test aftereffects of the proposed technique with the COCOMO model shows a decrease in MMRE to 23.8% [17].

Several software engineering steps depend on estimations, e.g., estimating the complexity and the effort for implementing a particular task or estimating the time and costs of implementing a system. Consequently, there are plenty of methods and models suggested for these estimation tasks. **Chris Kopetschny et al., (2019)** introduced known methods and models for software estimation and continue with problem description and analysis to end the introduction presenting their research aims [18].

Carlos Eduardo Carbonera et al., (2015) presented a comprehensive overview of these approaches, and pinpoints research gaps, challenges, and trends. Method: A systematic mapping of the literature was designed and performed based on well-established practical guidelines. In total, 120 primary studies were selected, analyzed and categorized, after applying a careful filtering process from a sample of 3,746 candidate studies to answer six research questions. Their study benefits practitioners and researchers by providing a body of knowledge about the current literature, serving as a starting point for upcoming studies [19].

Sang Hun Lee et al., (2020) proposed a framework for optimizing software development and V&V qualities by incorporating and estimating various risk-cost factors related to SDLC processes. The model determines the optimal software development or V&V quality that gives the minimum value of an objective function which includes three risk-cost factors: 1) software quality control cost; 2) software defect fixing cost; and 3) the cost from NPP accident consequence considering software failure probability. We present a case study using OPR-1000 plant model. Their study results showed important SDLC phases where the risk-cost factors can be minimized by achieving high software quality. The proposed method is expected to be useful to software developers and decision makers for assisting NPP software project and quantifying risk from software life cycle on NPP safety [20].

Neelamdhab Padhy et al., (2017) created programming measurements calculations and their essential developments for assessing the measurements from the UML/class outlines. It is attainable to determine an effective and powerful reusability forecast model for web-administration items utilizing object-arranged measurements. Here, it was additionally discovered that OO - CK measurements, especially intricacy, attachment and coupling-related measurements can be useful in foreseeing reusability in web-administration programming items. Thinking about the previously mentioned key commitments, it very well may be expressed that the proposed exploration could be of fundamental importance in cutting edge programming calculation frameworks, essentially for programming part reusability, unwavering quality, survivability, maturing forecast and security, and for programming greatness affirmation purposes [21]

Miikka Kuutilaa et al., (2020) introduced their work means to give an outline of studies identified with time pressure in programming; explicitly, existing definitions, potential causes, and measurements applicable to time pressure were gathered, and a planning of the investigations to programming cycles and approaches was performed. Also, they orchestrate consequences of existing quantitative examinations on the impacts of time tension on programming improvement, and offer useful takeaways for specialists and scientists, in light of exact proof [22].

Rohit Kumar Sachan et al., (2016) proposed an improved on hereditary calculation based model. An improved on hereditary calculation is utilized for upgrading the boundaries of the fundamental COCOMO model. The proposed approach is applied on NASA programming project dataset. Trial results show better practical assessment over the essential COCOMO [23].

Tribhuvan Singh et al., (2018) created IEAM-RP for cost assessment model to tuning the ascribes of sheta model. The principle objective of proposed model is to limit the distinction between estimated exertion and assessed exertion. Combination speed and variety are primary ascribes of elite during the enhancement cycle [24].

Mohsen Hasanluo et al., (2016) introduced another strategy to tackle the issue of SCE by utilizing cross breed molecule swarm improvement (PSO) calculation and K-closest neighbor (KNN) calculation. The technique was assessed on 6 various datasets with 8 distinctive assessment models. Gotten results show the more exact presentation of the proposed technique [25].

Ismail M. Keshta (2017) saw that it is ideal to utilize various diverse assessing methods or cost models, and afterward think about the outcomes prior to deciding the purposes behind any of the huge varieties. None of the strategies are essentially preferred or more terrible over the others. We found, indeed, that their qualities and shortcomings frequently supplement one another. Hence, the principle end is that there is nobody single method that is best for each circumstance, and the aftereffects of various methodologies should be deliberately considered to find what is the well on the way to deliver gauges that are practical [26].

Pahariya et al., (2009) introduced computational insight strategies for programming cost assessment. They proposed another repetitive engineering for Genetic Programming (GP) all the while. Three straight outfits dependent on (I) number-crunching mean (ii) mathematical mean and (iii) symphonious mean are executed. They likewise performed GP based component determination [27].

SevgiYigit-Sert et al., (2018) addressed an assessment model for the exertion needed for the advancement of programming projects utilizing a variation of fake honey bee province (ABC) calculation. The proposed model is performed over a dataset comprising of NASA programming projects and has preferable execution over the past investigations [28].

Zahra Ashegi Dizaji et al., (2014) they planned to improved the exactness of the expense assessment by utilizing Bee Colony Optimization calculation. It ought to be referenced that the proposed technique is contrasted and the middle COCOMO. The outcomes demonstrate that the proposed strategy have decreased the mean outright relative blunder to 0.1619 [29].

Shailendra Pratap Singh et al., (2018) new change methodologies are proposed to improve the assembly pace of differential development (DE) calculation. This calculation improves the precision of the semidetached model. This model streamlines the boundaries utilizing the upgrade based differential development calculation (EABMO). The proposed approach does a superior exhibition of the majority of the benchmark work (f1–f24). Further, this methodology applied the genuine use of the product business for decreasing the expense assessment and blunder estimation. The proposed approach limits the blunder contrasting execution of semidetached model (project) like size of relative mistake, mean extent relative blunder and mean squared blunder. The outcome checks that our proposed EABMO calculation performs better compared to the semidetached model based DE, GA and PSO calculation [30].

YansiKeim et al., (2014) sums up programming cost assessment models: COCOMO II, COCOMO, PUTNAM, STEER and ESTIMACS dependent on the boundaries carry out capacity, extensibility, adaptability and recognizability and methods used to gauge programming costs [31].

Sangeetha et al., (2012) gave an examination of existing programming cost assessment strategies remembering the new advances for the field. I have featured the expense assessment models that have been proposed and utilized effectively. Models might be grouped into 2 significant classes: algorithmic and non-algorithmic. Each has its own qualities and shortcomings. They think about the most well known algorithmic models used to appraise programming costs [SLIM, COCOMO, Function Points, SEER-SEM, etc). A vital factor in choosing an expense assessment model is the exactness of its appraisals [32].

Mohammed Aljohani (2017) recommended method like straight relapse; both algorithmic and non-algorithmic are applied. Model, composite and relapse strategies are utilized to determine COCOMO, COCOMO II, SLIM and direct various separately. Also, mastery based and straight based principles are applied in non-calculation techniques. In any case, the strategy needs some progression to decrease the mistakes that are capable during the product advancement measure. Consequently, this paper corresponding to programming assessment strategies has proposed a model that can be useful to the data innovation firms, analysts and different firms that utilization data innovation in the cycles, for example, planning and dynamic cycles [33].

Abhishek Singh Verma et al., (2020) proposed a self versatile butterfly strategy to decrease the expense of relapse test. Bat search calculation is utilized to think about the proposed strategy utilizing shortcoming discovery. Diverse test outcomes show the productivity of proposed strategy Test case optimization in software testing and regression testing is solved by using some nature inspired algorithms [34].

Nature is a guide to give some guidelines to all the learners and researchers to develop optimal methods to solving complex problems. **Krishnaveni et al., (2019)** suggested many metaheuristic methods play an important role in esteemed research areas like bio inspired computing, cloud computing, software project management, image processing, data mining and so on [35].

III. PROBLEMS AND DIRECTIONS

The main challenge in software engineering is to develop a good project that must be satisfied the cost, effort, quality and time constraints in both the developer side and customer side. In software life cycle model requirements gathering, Analysis, Planning, Designing, Coding, Testing, Maintenance and Deployment Stages are playing an individual role to satisfying the customer needs. An experiment is never a failure solely because it fails to achieve predicted results. Forecasting the cost estimation and also effort estimation is needed and in an accurate manner. Optimization algorithms are used to achieving the best cost estimation through new metaheuristic methods in an earlier stage.

Figure 2: Challenges of Cost Estimation

Quantifying cost impacts

Estimators need to determine if cost is sensitive to that parameter and to determine how it responds. Estimators must also analyze cost impacts for new products or approaches that are unknown or intangible.

Resource constraints

Time constrains many aspects of estimating, including data collection and validation, data quality, and consistency.

Quality of available data

Resource constraints challenge the quality and quantity of data that estimators can obtain. If there is not sufficient time available, estimators may use secondary data sources, manipulated from the original source. Secondary data, especially that lack thorough documentation, have limited usefulness.

Large number of organizations involved

Collecting data involves many different data sources and organizations of which estimators must be aware. Estimators need to communicate and coordinate with all of these sources.

Consistency

Once analysts collect all these data from the different sources through many time periods, they must make it consistent and comparable.

To improve the performance of the existing methods we propose the new method based on optimization. Sandish group of 8380 projects in 1994 reported 53% of projects cost 189% more than initial estimates. In the same year 55% of projects over budget, 24 companies that launched huge distributed systems.

Table 1: Pros and cons of algorithmic and non algorithmic cost estimation model

Model	Type	Strength	weakness
COCOMO	Algorithmic	Neat outcomes.	Huge amount of data is required.
LOC	Algorithmic	Size of the project estimation is done.	It is language dependent.
Linear	Algorithmic	Linear regression technique is used for flexibility.	Error is needed for calculation.
Seer-Sem	Algorithmic	Large projects are handled easily	Input parameter size leads complexity.
Putnam	Algorithmic	Probability method for huge project	It is not suitable for small projects.
Expert Judgment	Non-Algorithmic	Special projects.	Experienced experts are needed.
Estimation by analogy	Non-Algorithmic	Actual experience is required	Similar previous projects are rare to analysis.
Parkinson Method	Non-Algorithmic	Experience in comparison and analysis.	Not good practice.
Top down	Non-Algorithmic	System level component analysis needs minimum data.	Not give an adequate detail.
Bottom up	Non-Algorithmic		Time consuming.
Price to win	Non-Algorithmic	Contract basis.	Long term run.

IV. CONCLUSIONS

Cost and effort estimation are two basic building components of software engineering. Prediction is the primary field in the earlier stage of cost analysis. If prediction is wrong in the earlier stage then entire project face the troubles in later stage. To select the best metrics for both cost and effort estimation needs optimization algorithms. Minimizing the cost and maximizing the effort leads precaution of software success as well as customer satisfaction. In future we propose the new optimal algorithm for making effectual cost estimation and avoid any failure in the project.

REFERENCES

[1] E. Ramaraj and S. Duraisamy, "Design Optimization Metrics for Uml Based Object-Oriented Systems", International Journal of Software Engineering and Knowledge Engineering, Vol.17, No.03, pp.423-448 (2007)

[2] Prabu P, Duraisamy S, "Impact of pair programming for effective software development process", International Journal of Applied Engineering Research (2015)

[3] Xiaotie Qin, Miao Fang, "Summarization of Software Cost Estimation",Advanced in Control Engineeringand Information Science, Procedia Engineering 15, 3027 – 3031(2011)

[4] S. Z. Iqbal, Kashif Saghar, "Improving Software Cost Estimation with Function Points Analysis Using Fuzzy Logic Method", LC International Journal of STEM (Science, Technology, Engineering, and Math), Volume-01, Issue Number-01(2020)
[5] Seyed Hossein Hashemi, Seyedeh Nafiseh Hashemi, "Determining the Parameters of Object-Oriented Software Cost Estimation Using Cuckoo Search Optimization Algorithm", Journal of Xi'an University of Architecture & Technology, Volume XII, Issue IX, pp.420-442 (2020)
[6] Elnaz Eskandarian Miandoab, Farhad Soleimanian Gharehchopogh, "A Novel Hybrid Algorithm for Software Cost Estimation Based on Cuckoo Optimization and K-Nearest Neighbors Algorithms", Engineering, Technology & Applied Science Research Vol. 6, No. 3, 1018-1022 (2016)
[7] Anupama Kaushik, Nisha Choudhary, and Priyanka, "Software Cost Estimation UsingLSTM-RNN"Proceedings of International Conference on Artificial Intelligenceand Applications, Advances in Intelligent Systems and Computing 1164, Springer (2020)
[8] NazeehGhatasheh, HossamFaris, Ibrahim Aljarah, Rizik M. H. Al-Sayyed, "Optimizing Software Effort Estimation Models Using Firefly Algorithm", Journal of Software Engineering and Applications, 8, 133-142 (2015)
[9] Ping-Hao Zhang, And Kang Fu,"Research on the Application of Decision Tree Algorithm in the Selection of Variables in Cost Estimation Model", 2020 5th International Conference on Social Science and Management (ICSSM 2020)
[10] Zahra Asheghi Dizaji, Kamal Khalilpour, "Particle Swarm Optimization and Chaos Theory BasedApproach for Software Cost Estimation", International Journal of Academic Research, Vol. 6. No. 3. May, (2014)
[11] Hassan Najadat, IzzatAlsmadi,and YazanShboul, "Predicting Software Projects Cost Estimation Based onMining Historical Data", International Scholarly Research Network ISRN Software Engineering, Article ID 823437, 8 pages (2012)
[12] Chamkaur Singh, Neeraj Sharma, Narender Kumar, "Analysis of Software Maintenance Cost Affecting Factors and Estimation Models", International Journal of Scientific & Technology Research Volume 8, Issue 09, September (2019)
[13] NishuDewan, SumeetKaurSehra, "Ant Colony Optimization Based Software Effort Estimation", IJCST Vol. 5, Issue 3, July - Sept (2014)
[14] HamayoonGhafory and Faqeed Ahmad Sahnosh, "The review of software cost estimation model: SLIM", International Journal of Advanced Academic Studies 2(4) pp.511-515 (2020)
[15] FarhadSoleimanian, Isa Maleki, Akbar Talebi, "Using Hybrid Model of Artificial Bee Colony andGenetic Algorithms in Software Cost Estimation", IEEE (2014)
[16] S. M. SabbaghJafari,F. Ziaaddini, "Optimization of Software Cost Estimation usingHarmony Search Algorithm", 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC2016), Higher Education Complex of Bam, Iran, (2016)
[17] Muhammad Sufyan Khan, CH Anwar ul Hassan, Munam Ali Shah, AzraShamim, "Software Cost and Effort Estimation using a New Optimization Algorithm Inspired by Strawberry Plant", IEEE (2016)
[18] Chris Kopetschny, Morgan Ericsson,Welf Loweand Anna Wingkvist, "Optimization of Software Estimation Models", In Proceedings of the 14th International Conference on Software Technologies (ICSOFT 2019), pages 141-150 (2019)
[19] Carlos Eduardo Carbonera, KleinnerFarias, Vinicius Bischoff, "Software development effort estimation: A systematic mapping study", IET Research Journals, pp.1-14 (2015)
[20] Sang Hun Lee, Seung Jun Lee, SeoRyong Koo, AthiVaruttamaseni, MengYue, Ming Li, Jaehyun Cho,Hyun Gook Kang, "Optimization of software development life cycle quality for NPP safety software based on a risk-cost model", Annals of Nuclear Energy 135 106961(2020)
[21] NeelamdhabPadhy, R.P. Singh, Suresh Chandra Satapathy, "Software reusability metrics estimation: Algorithms, models and optimization techniques", Computers and Electrical Engineering, 1–16(2017)
[22] MiikkaKuutilaa, Mika Mantyla, Umar Farooq, MaelickClaes, "Time Pressure in Software Engineering: A Systematic Review", Information and Software Technology (2020)
[23] Rohit Kumar Sachan, Ayush Nigam, Avinash Singh, Sharad Singh, Manjeet Choudhary, AvinashTiwari and Dharmender Singh Kushwaha, "Optimizing Basic COCOMO Model Using Simplified Genetic Algorithm", Procedia Computer Science 89 492 – 498, (2016)

[24] Tribhuvan Singh, Ranvijay Singh, Krishn Kumar Mishra, "Software Cost Estimation Using Environmental Adaptation Method", 8th International Conference on Advances in Computing and Communication (ICACC-2018), Procedia Computer Science 143 325–332(2018)
[25]Mohsen Hasanluo, Farhad Soleimanian Gharehchopogh, "Software Cost Estimation by a New Hybrid Model of Particle Swarm Optimization and K-Nearest Neighbor Algorithms", Journal of Electrical and Computer Engineering Innovations (JECEI), Vol. 4, No. 1, (2016)
[26] Ismail M. Keshta, "Software Cost Estimation Approaches: A Survey", Journal of Software Engineering and Applications, 10, 824-842 (2017)
[27] J.S.Pahariya, V. Ravi, M. Carr, "Software Cost Estimation using Computational Intelligence Techniques", IEEE (2009)
[28] SevgiYigit-Sert, Pinar Kullu, "Software Cost Estimation using Enhanced Artificial Bee Colony Algorithm",(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 9, No. 4, (2018)
[29] Zahra AshegiDizaji, Reza Ahmadi, HojjatGholizadeh, FarhadSoleimanian, "A Bee Colony Optimization Algorithm Approach for Software Cost Estimation", International Journal of Computer Applications (0975 – 8887) Volume 104 – No 12, (2014)
[30] ShailendraPratap Singh, "Cost estimation model using enhance-based differential evolution algorithm", Iran Journal of Computer Science, Springer (2019)
[31] YansiKeim, Manish Bhardwaj, ShashankSaroop, AdityaTandon, "Software Cost Estimation Models and Techniques: A Survey", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 2, (2014)
[32] Y.Sangeetha, P.MadhaviLatha, R.Satya Prasad, "Software Cost Models", International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 7, (2012)
[33] Mohammed Aljohani and RizwanQureshi, "Comparative Study of Software Estimation Techniques",International Journal of Software Engineering & Applications (IJSEA), Vol.8, No.6, (2017)
[34] Abhishek Singh Verma, AnkurChoudhary, ShaileshTiwari, "Test case optimization using Butterfly Optimization Algorithm", IEEE (2020)
[35] A.Krishnaveni, R.Shankar, S.Duraisamy, "A Survey on Nature Inspired Computing (NIC): Algorithms and Challenges", Global journal of computer science and technology: D Neural & Artificial Intelligence volume 19 issue 3 version 1.0 Year (2019)