Cost Estimation Based On Artificial Bee Colony

Krishna Mohan.G, Venkata Naresh Mandhala, J. Satish Babu, Havisha.B, Haritha Vardhini.Y, Lakshmi Priya.R

Abstract: The reliability of the software is mainly achieved by modeling and estimating. The existing models of software reliability are nonlinear and it is difficult to estimate the parameters of these models. Several methods of optimizing nonlinear function problems are available, such as the artificial bee colony algorithm (ABC). It is necessary to know how much development it will cost and how long it will take for any new software project to develop. World technologies and new software project management models are developing rapidly, but there is still a need for smart and precise software project management, That causes this issue to fail, frustration within this rising way, we are attempting to use computational intelligence approaches such as artificial intelligence and statistical techniques along with smart software project management. Statistical analysis found that most software projects were unable to reach the end based on the original projected time and cost. In this paper, we focus on algorithmic models and address their difficulties within parameters of modification and tuning to improve the reliability of software cost estimation using artificial bee colony algorithm (ABC).

Index Terms: Artificial Bee Colony, COCOMO Model, Cost Estimation.

---

1. INTRODUCTION

Present days, increasing needs of software production and development makes software the most expensive components of computer. The success rate of software developer organizations, based on estimated costs and time, depends on the management and planning of resource usage. So for estimating cost we use several algorithm techniques like COCOMO and ABC for optimizing the cost. Both underestimated and overestimated efforts are detrimental to projects under construction. proposal or to the allocation of excess assets to the project. Over the past several decades, A variety of cost-effective methods of estimation have been developed and can be grouped into three Underestimation results in a situation where project commitments can not be fulfilled due to lack of time and/or resources. On the other hand, overestimation may lead to the rejection of project main categories. These groups are as follows:

1. Expert judgment: A widely used approach is based on an expert's previous experience on similar projects to collect, assess, interpret, and analyze data on a particular project to provide estimation.
2. Algorithmic models: parametric models that seek to reflect the relationship between the effort and features of the project are also called. The size of the code, usually measured by Kilo Lines of Code (KLOC) or feature level, is the main cost driver of such models. These are versions of COCOMOI, COCOMO II.

3. Machine learning: In recent years, machine learning methods have been used in combination with or as an alternative to the above two techniques. The most well-known tool for estimating technology costs is the Constructive value model (COCOMO).

- *Krishna Mohan.G, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*
- *Venkata Naresh Mandhala, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*
- *J. Satish Babu, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*
- *Havisha.B, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*
- *Haritha Vardhini.Y, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*
- *Lakshmi Priya.R, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Vaddeswaram, A.P., India.*

1.1 COCOMO-MODEL:

It is a LOC-based model, i.e. variety of code lines. it's a applied mathematics budget estimation model for code comes and is usually used as a tool to faithfully predict the various parameters related to creating a project like scale, effort, price, time and performance. it had been projected in 1970 by Barry mystic and is predicated on sixty three comes studied. that build it one in every of the best-documented models. Organic: The organic class includes relatively small projects undertaken by highly experienced teams. Semi-detached: Semidetached class includes projects with an average size of 100 to 300 KLOC that are neither complex nor simple. Embedded: Embedded group comprises more than 300 KLOC works. This class is used when previously specified hardware and operations and needs no changes.

$$PM = \mathrm{l} * (size)\mathrm{m} * 15\ EMi$$

Types of Models: COCOMO is a hierarchy of three types that are becoming increasingly detailed and accurate. Depending on our criteria, any of the three types can be adopted. These are COCOMO model types:

1.1.1 Basic COCOMO Model:

First level, Simple COCOMO can be used to measure Computer Costs easily and slightly roughly. Due to the lack of adequate variable factors, its accuracy is somewhat limited.

E= l(KLOC)^m

1.1.2 Intermediate COCOMO Model:

Intermediate COCOMO takes these price drivers under consideration and elaborate It additionally accounts for the result of individual project phases, i.e. within the case of elaborate, each these price drivers are accounted for and part wise estimates are done henceforward providing a additional correct result. These 2 models are mentioned more below. It takes these price drivers under consideration and elaborate COCOMO additionally accounts for the result of individual project stages, i.e. within the case of elaborate, each these price drivers are accounted for and phase-wise estimates are administered henceforward providing a additional correct result. Such 2 versions are mentioned more below.

E= (l(KLOC)^m)*EAF In Intermediate model the values of l and m are as follows:

1.1.3 Detailed Model:

Comprehensive COCOMO integrates all the functionality of the intermediate version with an analysis of the effect of the cost driver on each phase of the software engineering process. For each cost driver feature, the comprehensive model uses various effort multipliers. The entire program is catageroised into many models in comprehensive cocomo, and then by applying COCOMO to estimate effort in several modules and add the effort.

There will be six phases in COCOMO MODEL.
The effort is measured as a function of the size of the system and according to each step of the software life cycle, a set of cost drivers is given.

1.2 ARTIFICIAL BEE COLONY:
To optimize math functions, Karaboga used the ABC algorithm. The ABC has three bee groups: Employed Bees (EB), Onlooker Bees (OB) and Scouts Bees (SB Bees who linger in the dance area are named OB to make the decision for choosing a food source. Bees going to a designated source of food are called EB. Bees doing a random search are called SB
There are three stages in the ABC algorithm for each search cycle:
1. Give EB to sources of food and then calculate the quantity of their nectar.
2. Select the food source via OB and by sharing info via EB and assessing the quantity of nectar in food sources.
3. Specify the SB and attach it to new food sources.

2 LITERATURE REVIEW
Krishna Mohan, et al. (2016) the performance of SRGM is judged by its ability to suit the software system failure data. How sensible will a mathematical model fit the information and reliableness of the software system is bestowed. Rayandra Yala Pratama, et al. (2017) Optimizing COCOMO II Parameters using Artificial Bee Colony This paper uses Nasa93 dataset with 93 projects that have cost drivers, the scale of the project, real estimate and production of time. All the compositions of the project are used for calculation for this process. The suggested method give result that the MMRE of effort and development time are greater than COCOMO II model MMRE. Vahid Khatibi Bardsiri,(2016) Proposed Improved COCOMO based model to Estimate the effort of Software Projects. Most approaches to cost estimation are focused on an estimate based on an individual month. The bee algorithm proposes multiple times through goal function a and b coefficients. This research used the bee colony algorithm used to estimate the effort of software development to maximize coefficients [3]. Srinivasa, et al., (2011) suggested a replacement model structure supported Alaa F. Sheta by using formal logic for the uncertainty of dominant prediction and tuned the parameters of the cost model by using swarm intelligence-Particle Swarm improvement. The verification of projected model results and comparison with the prevailing models was through with the National Aeronautics and Space Administration software package dataset. Reddy, et al., (2010) proposed software system effort estimation principles supported artificial neural networks. The figures intended to enhance the fulfilling of the network that suits to the COCOMO Model.

3 METHODOLOGY
The estimation of effort is done by COCOMO required (Person-Months – PM) primarily based on your estimation of scale of the software project (measured in KLOC):

Effort = 2.94 * EAF * (KLOC)^E
MARE= (ACTUAL EFFORT-ESTIMATE EFFORT)/(ACTUAL EFFORT)
Where EAF is Cost Drivers of Effort Adjustment Factor.
E Is an exponent of the five Scale Drivers.
The COCOMO II timeline formula estimates how many months the code plan will need to complete. A project's length is dependent on the effort estimated by the formula of effort[9].

Time=3.67*(Effort)^SE

Where Effort is taken from the COCOMO II effort equation.
cost=effort*time
SE Is the schedule equation exponent derived from the five Scale Drivers.

4 RESULT
In this paper we found the best and unique technique for estimating the cost using ABC (Artificial Bee Colony). Output costs are calculated from low to high up to the accurate result for the software. We used ABC for calculating the result.

TABLE 1. *Best Cost for the iterations*

Iteration	Best Cost
1	2.9733
2	2.9733
3	1.1023
4	1.1023
5	0.68359
6	0.36185
7	0.36185
8	0.14082
9	0.082981
10	0.082980

5 CONCLUSION
There are several factors that can cause project failure, some of which are initial estimate errors, lack of on-time estimates, lack of integrated businesses, technical changes during the project, difficulty of the project, changes in market conditions and budget cuts. Such variables can impact other project problems, such as estimates of effort and time, which may be correlated with negative results of project management outcomes in this case. Therefore, we can estimate the cost reliably using artificial bee colony.

6 FUTURE SCOPE
These estimates can be used for assessing software reliability through SPC (Statistical Process Control)[1]. and SPRT (Sequential Probability Ratio Test)[9].

REFERENCES

[1] Satyaprasad, R. Suryakiran, C. Krishnamohan, G (2016). “SPC based software reliability using modified genetic algorithm: Rayleigh model” JARDC 10(4 Special Issue), pp. 500-506.

[2] Nashat Mansour, HratchZeitunlian and Abbas Tarhini,(2004) “Optimization Meta-heuristic for Software Testing” Volume 175, pp 463-474.

[3] BalwantKumar,DrDharmender Kumar,(2004) "A review on Artificial Bee Colony algorithm", International Journal of Engineering and Technology,2(3)(2013) 175-186 [16]“Genetic”,[Online].AvailableURL:http://www.tutorialspoint.com/software_testing/testing_ov erview.htm.

[4] Eugenia.D, Javier.T, Blanco,(2003) “AST Using a Metaheuristic Technique Based on Tabu Search”, 18th IEEE International Conference on ASE, pp 310-313,

[5] McMinn. P, “Search-based software test data Generation(2004): A survey,” Software Testing, Verification and Reliability ; 14(2): pp. 105-156.

[6] Watkins.A and Hufnagel.E,(2006) “Evolutionary test data generation: a comparison of fitness functions,” Software Practice & Experience ; 36: pp. 95– 116

[7] Dervis.Karaboga ,BahriyeBasturk ,(2007)" A powerful and efficient algorithm for numerical function optimization: artificial bee colony(ABC) algorithm", Springer Science Business Media B.V.

[8] A.Srisaila, G. Krishna Mohan, R. Satya Prasad, "Spc and order statistics: Plp model", International Journal of Applied Engineering Research 2015; 10(4), pp. 10985-10994.

[9] Krishna Mohan, G. Indira, B. Satya Prasad, R. (2014) "SPC and order statistics: LPM" International Journal of Applied Engineering Research ; 9(21), pp. 9971-9982

[10] Hana Rashied Ismaeel and Abeer.Salim Jamil(2007) "Software Engineering Cost Estimation Using COCOMO II Model"