Optimizing COCOMO II Parameters using Artificial Bee Colony Method

Rayandra Yala Pratama[1], Riyanarto Sarno[2], Sholiq[3]

[1]Department of Information Technology Management
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
rayandra.pratama16@mhs.mmt.its.ac.id

[2]Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
riyanarto@if.its.ac.id

[2]Department of Information Systems
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
sholiq@is.its.ac.id

***Abstract*—Cost estimation is a crucial and essential process in software industry. The more accurate cost estimated, the more efficient the project became. This cost estimation become a challenge for software industry to bring accurate result. There are many methods to solve this problem. Constructive Cost Model is usual method that is used to estimate software cost. This model was proposed in 1981 by using regression analysis with 63 types of project data. In 2000, COCOMO II was introduced. This new model of COCOMO use cost drivers, scale factors, and project size that measured by line of code. COCOMO II has 4 parameters A, B, C and D. However, using this parameters are not guarantee accurate result. This paper proposed Bee Colony Optimization to calibrate the COCOMO II model parameter to be more accurate for effort estimation. This Bee Colony Optimization is applied on Nasa93 dataset that consisted of 93 projects which each project has 22 cost drivers, project's size, effort, and development time. This proposed method gives MMRE result 50.584% on effort and 14.192% on development time.**

Keywords—Bee Colony Optimization; COCOMO; Bee Algorithm; MRE; Software Cost Estimation

I. Introduction

Software cost and effort estimation is important for industry to get early information. The information is useful for the industry to manage and to control the project. Having early project's effort also can be useful for resource allocation. It is challenging to estimate cost and effort software project accurately. The Mean Magnitude of Relative Error (MMRE) is used for indicating the accuracy of software estimation. The least MMRE value, the more accurate the estimation become.

Various methods have been used for software cost estimation in last decades. The methods are classified into 3 categories, Algorithmic method, Expert judgment, and Analogy based method. The most well-known method to estimate software cost is Constructive Cost Model (COCOMO). Various heuristic methods for optimization are used in optimization problems. Those methods are Particle Swarm Optimization [1] [2] [3] [4] [5], Neural Networks [6], Multi-Objective Particle Swarm Optimization [7], Genetic Algorithm [8], Bee Colony Optimization [9], Firefly Algorithm [10], and many more. In this paper, measurement of COCOMO model using Bee colony optimization use MMRE as a fitness function.

There are three parameters that are used in cost estimation. The A and B are used to calculate Effort that need to develop the software. On the other hand, the C and D are used to calculate time development (TDEV) that needed in building software. The purpose of this research is to optimize the parameters of COCOMO II that will produce minimum error (MRE).

The aims of this research is to determine the values of parameters A, B, C, and D in COCOMO II so as to obtain the estimation of effort that is closest to actual effort (optimal). To measure the most optimal estimation effort is indicated by the smallest MRE and MMRE. The researchers' motivation to improve the estimation level is to get better estimation results than the original COCOMO II estimation. It is intended to provide a reference to the use of constants A, B, C, and D by software project managers when estimating the effort and cost using COCOMO II.

This paper contain some chapters that the discussion will be taking place. Literature review will be discussed in chapter 2. Chapter 3 will discuss about previous research on cost estimation and Bee Colony Optimization. The methodology will be discussed in chapter 4. The evaluation criteria and dataset used are discussed in chapter 5. The experiment and result are discussed in chapter 6 and last chapter is conclusion about this research.

II. LITERATURE REVIEW

A. Software Cost Estimation

In software industry, software cost estimation is required to predict effort and schedule accurately. This prediction can provide initial effort that indicates in person-month and how long does the project take. With this information, resources will be assigned efficiently regarding limited resources that industry has. Yet, estimating software cost is still difficult to perform. This difficulties are came from uncertainty data and complicated process that make software development process taking effect.

To handle this issues, some procedures and techniques are provided. Both algorithm and non-algorithm solutions can solve the estimation problems. Linear Regression is common method that used as algorithm solution by using previous data to perform next prediction. Otherwise, non-algorithm solution usually use rules construction that include analogy method, genetic algorithm, fuzzy, and artificial neural network. Effort and time development are usual measurement for cost estimation. Effort indicates how much time required by a person to do the development, while time development is how many months a project is scheduled. The more efforts and time development, the more cost has to be paid.

B. COCOMO II

Constructive Cost Model (COCOMO) is a model to estimate software cost that developed by Barry Boehm based on regression. COCOMO is well-known model for software estimation. COCOMO delivers effort and schedule estimation as main models. This model was introduced in 1981 and developed from the 63 projects dataset. Each of those has 16 variables and divided into 3 aspects, Effort Multiplier (EM), Scale Factors (SF), and Line of Code/Size (LOC). To calculate effort in person-months (PM), those variables will be calculated using two equations.

COCOMO II is the successor of COCOMO that was published in 2000. The models that used in COCOMO II is different from the ones from COCOMO (COCOMO 81). They are:

- Application Composition Model

  This model is new Object-Points base. This is the right choice for projects that using modern GUI-builder tool.

- Early Design Model

  This model uses small set of Cost Drivers and new equations of cost estimation. Those will produce rough estimation cost and duration of a project.

- Post-Architecture Model

  After overall architecture of the projects have been determined, this model is used. This is the most detail model. Line of Code is used as size to estimates the cost.

COCOMO II has 17 Effort Multipliers and 5 Scale Factors. Line of Code is project's size that usually count as thousand line of code (KLOC). Equation (1) is used to calculate Effort in Person-Month (PM) where Effort Multipliers and Scale Factors are needed.

$$Effort\ (PM) = A * Size^{E} * \prod_{i=1}^{17} EM_i \ \ldots\ldots (1)$$

Where,

$$E = B + 0.01 * \sum_{j=1}^{5} SF_j$$

In Equation (1), A is multiplicative constant that has value 2.94. Size is how many lines of source code need to build the project. E is exponential value that calculated with parameter B and sum of Scale Factors. B has value 0.91.

For Time Estimation, Equation (2) is used to calculate project's development time (TDEV).

$$TDEV_{NS} = C * (PM_{NS})^{F} \ \ldots\ldots (2)$$

Where,

$$F = D + 0.2 * 0.01 * \sum_{j=1}^{5} SF_j \ \ or\ F = D + 0.2 * (E - B)$$

C and D are constant of development time that have value 3.67 and 0.28.

This paper propose COCOMO II parameters optimization using Bee Colony Optimization on effort and schedule calculation with Nasa93 dataset.

C. Bee Colony Optimization

The Bee Colony Optimization is an example of swarm intelligence that simulates the behavior of bees. Complex combinatorial optimization problem is solved by artificial bee agents. Bee colony optimization is a meta-heuristic algorithm that is good to solve combinatorial problem using collective intelligence and swarm behavior of bees. Bee Algorithm can be used to find possible solution that is a responsibility of artificial bees. Two types of alternating pass that consisted in Bee Algorithm are forward and backward pass. Both passes are problem dependent.

Through forward pass process, empty problems are assigned to every bee. Partial/Complete solutions are computed by every bee after exploring search space for number of predefined moves. The evaluation is determined from past experience and individual exploration. Backward pass or second phase is performed after the bees go back to colony or hive.

All bees are participated for making a decision during backward pass. Every bees share all evaluated solutions by performing waggle dance. The waggle dance is basically a shape of digit '8' movement. Every bee produce different solutions. During this only process, bees are communicating with each other. The best solution is a loyal solution that considered as partial/complete solution. Equation (3) is used for selecting loyal solution.

$$loyalty = \frac{\max_{solution} - \text{current}_{solution}}{\max_{solution} - \min_{solution}} \dots \dots (3)$$

For every bee, loyalty was checked within a single move. $\max_{solution}$, $\text{current}_{solution}$ and $\min_{solution}$ are values from a set of solutions.

III. Related Work

There are many researches done to optimize COCOMO II parameters to improve the accuracy of cost estimation.

Langsari and Sarno optimized COCOMO II parameters A and B using Particle Swarm Optimization [1]. The research used Turkish software industry dataset that has 12 number of projects. Each project has project's size that used in effort calculation. This research proposed Manhattan Distance and Mean Magnitude of Relative Error as fitness functions for proposed method. It also compared 3 models input, COCOMO II Model, Tabu Search Effort, and PSO Effort. The result show that PSO is 698.9461% MMRE and 495.3469% MD better than general COCOMO II model and 104.876% MMRE and 47.332% MD better than Tabu Search.

In other research, Langsari and Sarno optimized effort and time parameter of COCOMO II using Fuzzy Multi-Objective PSO [7]. This research was using Nasa93 dataset that has 93 projects. Each project has effort multipliers, scale factors, line of code, effort, and time to develop. To calibrate and optimize the COCOMO II parameters, they use Gaussian Membership Function (GMF) Fuzzy Logic and Multi-Objective Particle Swarm Optimization. The proposed method has better result than previous research with 11.891% different on effort and 8.082% different from TDEV on original COCOMO II method. Also using Neural Network is also done by the same researchers can improve the level of accuracy [11].

Chalotra, et.al [9] tuned COCOMO's parameter using Bee Colony Optimization. This research used IVR dataset that has 48 projects. Each project has line of code, actual effort and time duration (TDEV) in month. The research compare the proposed method with COCOMO, COCOMO II, SEL, Balley Basil and Halstead. This research resulted the proposed method has the best MMRE value among the others. The proposed method obtained 0.11 on MMRE while COCOMO II obtained 0.24.

IV. Methodology

To perform effort estimation has many uncertainties on COCOMO II parameters and cost driver. Performing another method to optimize its values was delivering various results. The best values were not discovered yet. This paper proposed Bee Colony Algorithm to optimize COCOMO II parameter that are effort and time development. This method is searching the best value for COCOMO II parameters to get minimum error.

This paper uses Nasa93 dataset with 93 projects that have cost drivers (effort multipliers and scale factors), project's size (in KLOC), actual estimation (in person-month) and time development (in month). For this method, all of the project's compositions are used for calculation. Meanwhile, the output will be new value of parameter A and parameter B. The equation used for COCOMO II effort estimation is shown in Equation (1), while Equation (2) is used for calculating time development.

Size that used in effort estimation is came from project size that measure in kilo line of code (KLOC). The performance of each project is measured by magnitude of relative error (MRE) that indicate how many different between calculated value with actual effort that obtained from projects. MRE equation is shown in Equation (4).

$$MRE = \frac{\text{actual effort} - \text{predicted effort}}{\text{actual effort}} \dots \dots (4)$$

Meanwhile, the performance perform by individual method or overall projects is measured by Mean Magnitude of Relative Error (MMRE) that indicate average error from all projects. MMRE is calculated using Equation (5).

$$MMRE = \frac{1}{n}\sum_{i=1}^{n} \frac{\text{abs(actual effort} - \text{predicted effort)}}{\text{actual effort}} \dots (5)$$

Where actual effort is obtained from Nasa93 dataset on each project. The predicted effort is an effort that produced by Equation (1). The proposed Bee Colony Algorithm process is:

Input: Cost drivers (effort multipliers and scale factors), project's size (KLOC), actual effort (PM), and time development (month) that taken from Nasa93 dataset.

Output: Optimized value of parameter A and B of COCOMO II

Initialization parameters: B (number of bees), NC (number of constructive moves)

1) Read data.

   Nasa93 dataset still contains data with ordinal scale and need to be converted into number value. The value is available in COCOMO II manuals.

2) Specify number of bees, constructive moves and criteria to stop the process.
3) Initialize parameters of COCOMO II model.
4) Start BCO module.
5) For each bee, repeat step 6 to step 9.
6) Calculate effort using Equation (1) and time development using Equation (2) in forward pass process.

7) For each bee, generate partial solution by editing according to changes required using Equation (4) in forward pass.

8) Start backward pass. During backward pass, evaluate every complete solutions and choose the best bee according to fitness function that calculated using Equation (5). Proposed goal is to minimize MMRE value.

9) For each bee, calculate and check loyalty using Equation (3).

10) Continue the solutions that are loyal to their solutions.

11) Get global best solution from the best bee that have minimum MMRE value.

12) Finish BCO module.

V. Evaluation Criteria and Data Set

To evaluate our research, we use Mean Magnitude of Relative Error (MMRE) as fitness function for the proposed Bee Colony Optimization.

The main purpose of evaluation is to prove that the prediction of our method is accurate according to actual effort from dataset. The effort that calculated, *Predicted Effort*$_i$, should be as close as possible to the effort that obtained from dataset, *Actual Effort*$_i$. The bigger difference that produce between *Predicted Effort*$_i$ and *Actual Effort*$_i$, the less accurate the prediction will be. If the accuration is low, it will affect the project development and project management. MMRE calculated over MRE which considered as individual error of each project.

Nasa93 dataset is used in our research to optimize COCOMO II parameter A and B using Bee Colony Optimization algorithm. This dataset has of 27 attributes that are consisted of project's ID, 22 cost drivers (5 scale factors, 17 effort multipliers), project size in KLOC, effort in month (actual effort), total defects, and time development in month. This cost drivers are measured in ordinal scale that has interval value from Very Low (vl) to Extra High (xh). The actual effort is project effort in a month that consisted of 152 hours and development and management hours are included. All data from Nasa93 dataset will be used in calibration of COCOMO II parameters.

VI. Experiment And Result

The experiment and the result of applying Bee Colony Optimization to the Nasa93 dataset are showed in this section. The purpose of this research is to reduce the uncertainties of COCOMO II model parameters A, B, C, and D using Bee Colony Optimization. This paper also compare the result that obtained from the proposed method and from basic COCOMO II model parameter.

The calculation is performed using excel application and conducted in 3 iterations. After 3rd iteration, the parameters that has best MRE value are obtained. The results are A = 2.609, B = 1.042, C = 2.410, and D = 0.323 instead of the basic COCOMO II parameter values are A = 2.94, B = 0.91, C = 3.67 and D = 0.28.

Reducing MRE and MMRE values are the purposed of this implementation, the smaller MRE and MMRE value the closer the result with the actual effort and development time. On Project ID 11, the MRE value is 55.07% and 26.14% from proposed BCO method where basic COCOMO II model gives 60.03% and 1.89%. This result shows that MRE on effort is reduced by 4.96% but MRE on development time is increased by 24.25%. More detailed comparison is shown in Table I.

The MMRE of each method represent MEAN from differences between actual values with calculated value from proposed method. The MMRE value of COCOMO II model, Bee Colony Optimization as 50.58% and 60.82% for effort estimation and 14.19% and 25.65% for development time. This results show that Bee Colony Optimization has increased MMRE value by 10.24% for effort estimation and 11.46% for development. The result on MMRE show that effort and development time by proposed method is delivering worse solution when compared to basic parameter of COCOMO II model.

TABLE I. Comparison between proposed method and Basic parameter

Project No.	COCOMO II Effort	BCO Effort	COCOMO II TDEV	BCO TDEV
6	17.862	19.115	5.996	23.725
12	56.932	29.809	11.670	25.289
18	2.363	36.619	26.937	42.786
24	48.199	34.279	4.217	23.674
30	9.985	11.839	5.716	26.063
36	36.747	28.136	3.929	25.913
42	62.029	28.359	1.145	7.179
48	48.846	24.433	12.042	26.737
54	52.632	14.385	8.324	16.312
60	25.527	43.201	22.905	32.819
66	39.515	3.997	19.065	28.881
72	7.782	49.894	12.179	26.300
78	523.142	984/98	52.029	61.65
84	78.265	70.659	20.122	2.379
90	76.050	56.357	6.195	2.748
93	26.181	24.269	6.829	28.458
MMRE (%)	**50.584%**	**60.821%**	**14.192%**	**25.655**

VII. Conclusion

Providing accurate result on software estimation is a crucial issue which is important to both developers and clients. It is very important to have accurate estimation in the early development process. The proposed method of Bee Colony Optimization is used to solve complex optimization problems. The proposed method are implemented using Nasa93 dataset that has 93 projects with each project has effort and development time. Evaluation is compared with Nasa93 dataset. The method is assessed with evaluation criteria. The proposed method give result that the MMRE of effort and MMRE of development time are higher than COCOMO II model MMRE.

References

[1] K. Langsari and R. Sarno, "Optimizing COCOMO II Parameters using Particle Swarm Method," *International Conference on Science and Information Technology (ICSITech),* 2017.

[2] J. Parkash and Karambir, "COCOMO II Model Parameter Optimization using PSO and Effort Estimation," *Int. Journal of Information Technology & Mechanical Engineering - IJITME,* vol. 1, no. 4, pp. 1-11, 2014.

[3] G. S. Rao, C. V. P. Krishna and K. R. Rao, "Multi Objective Particle Swarm Optimization for Software Cost Estimation," *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of CSI,* vol. 32, pp. 125-132, 2014.

[4] A. Kumar, A. Sinhal and B. Verma, "A Novel Technique of Optimization for Software Metric Using PSO," *International Journal of Soft Computing And Software Engineering (JSCSE),* vol. 3, pp. 1-8, 2013.

[5] C. V. M. K. Hari and P. V. G. D. P. Reddy, "A Fine Parameter Tuning for COCOMO 81 Software Effort Estimation using Particle Swarm Optimization," *Journal of Software Engineering,* pp. 38-48, 2011.

[6] R. Sarno, J. Sidabutar and Sarwosri, "Improving the Accuracy of COCOMO's Effort Estimation Based on Neural Networks and Fuzzy Logic Model," *International Conference on Information, Communication Technology and System (ICTS),* 2015 doi:10.1109/ICTS.2015.7379898.

[7] K. Langsari and R. Sarno, "Optimizing Effort and Time Parameters of COCOMO II Estimation using Fuzzy Multi-Objective PSO," in *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2017.

[8] R. K. Sachan, A. Nigam, A. Singh, S. Signh, M. Choudhary, A. Tiwari and D. S. Kushwaha, "Optimizing Basic COCOMO Model using Simplified Genetic Algorithm," *Procedia Computer Science,* vol. 89, pp. 492-498, 2016.

[9] S. Chalotra, S. K. Sehra, Y. S. Brar and N. Kaur, "Tuning of COCOMO Model Parameters by using Bee Colony Optimization," *Jounal of Science and Technology,* p. Vol 8 (14), 2015.

[10] N. Ghatasheh, H. Faris, I. Aljarah and R. M. H. Al-Sayyed, "Optimizing Software Effort Estimation Models Using Firefly Algorithm," *Journal of Software Engineering and Applications,* vol. 8, pp. 133-142, 2015.

[11] R. Sarno, J. Sidabutar and Sarwosri, "Comparison of different Neural Network architectures for software cost estimation," in *2015 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 2015 doi: 10.1109/IC3INA.2015.7377748.

This page intentionally left blank