Journal of Scientific Research and Development 1 (2): 44-51, 2014
Available online at www.jsrad.org
ISSN 1115-7569

Artificial bee colony based constructive cost model for software cost estimation

Farhad Soleimanian Gharehchopogh [1,*], Isa Maleki [2], Amin Kamalinia [3], Habibeh Mohammad Zadeh [3]

[1]*Department of Computer Engineering, Hacettepe University, Beytepe, Ankara, Turkey*
[2]*Young Researchers and Elite Club, Urmia Branch, Islamic Azad University, Urmia, Iran*
[3]*Department of Computer Engineering, Urmia Branch, Islamic Azad University, Urmia, Iran*

Abstract: With rapid changes in technology, software projects' development and production should be of greater operational speed in order to adjust and prepare the possibility of competition against rival companies. Hence, Software Cost Estimation (SCE) for projects is extremely vital. Nowadays, in most of large and small software companies, managers are facing with numerous challenges in allocating resources. In models such as COCOMO algorithm, due to the linearity, there is no possibility to test and train the factors of the current project with the former project. Therefore, in order to reduce cost and time and increase the quality of the projects, algorithmic models such as COCOMO the SCE must be changed to Artificial Intelligence (AI) models in order to meet performance of project during its lifecycle. In this paper, with using Artificial Bee Colony (ABC) algorithm the dependence between factors of COCOMO model has been measured and the best value to estimate the cost and effort of projects has been provided. The results show that compared with COCOMO, COCOMO-ABC model has less MMRE value and increased the accuracy of PRED (30) to about 1.77 times.

Key words: Software cost estimation; COCOMO; Artificial bee colony

1. Introduction

SCE is a major factor in the success of any project, as well as critical infrastructure and main platform for application of cost and effort in software projects (Buglione and Gencel, 2008; Maleki et al., 2014). Allocation and distribution costs for the project have this benefit which determines which part of the Company acknowledges the need to support the effort. Time and cost are two important parameters in software development and production that usually successful completion of the project largely depends on these two factors. The impact of the cost on project is quantifiable, because costs are calculated simply, but to evaluate the effect of timing on projects is not so easy, because scheduling can influence on how to do, the speed and range of effort and necessarily does not exist a direct relationship between the projects' factors and the total duration of the project (Gharehchopogh et al., 2014; Maleki et al., 2014). Also, due to the floating of the activities of the project, overall project scheduling maybe difficult.

Selecting of manpower is the main anxiety of project managers, because the nature of activities including time limits and periodically requires the people to capture the activities quickly and often do arduous tasks and activities with previous experience. Since software projects' managers define the implementation, monitoring and controlling the projects so, two important components for administrators include time management and cost management (Gharehchopogh et al., 2014). Time management expresses the structure and requirements of the procurement and scheduling control in projects. The SCE will be difficult in the process before planning. Because prior to the completion of the planning phase, the project is uncertain and probably needs to compare and choose between different projects. And because the projects' information is limited, therefore, to evaluate the cost and effort needed to spend time. In the development and application of software projects should be considered the following factors (Trendowicz et al., 2013; Maleki et al., 2014):

- *Time Estimation*: estimating the time of doing the activities is one of project time management processes. Time estimation has been done depending on the scope of activities, various required resources, the amount of resources and the availability of resources .in the estimating the time of the activities, similar projects that contain information about the cost and time considered the attention of the project's team and set the time of activities with respect to them. Finally, the project team with considering the impact of the success of similar projects estimate the length of the base for each activity.
- *Scheduling*: Scheduling is one of the process of time management that take place after the estimating the duration of the project activity. This process which is iterative, determines the time scheduled for the beginning and end of each activity of the project. During this process it may

* Corresponding Author.

be necessary to estimate the duration of activities and also to review the resource estimation.

In order to be able to succeed, software projects must be evaluated qualitatively and quantitatively (Liu and Mintram, 2005; Heidrich, 2013). Typically, qualitative evaluation method requires less effort and is simple to implement. The basis of the method relies on the estimations of similar or earlier projects. Unlike the qualitative method, quantitative method of evaluation requires further effort concerning more accurate outputs. Offering tangible numerical results, this method can be directly used in time and budget planning. It should be noted that the level of the accuracy of this type of evaluation depends on the input data. Therefore, its results seek expert`s opinion and must be applied with caution. Most software experts believe that qualitative evaluation in projects is required, because it helps understand and handle risks remarkably, and effectively. However, evaluating risks quantitatively is a process which is effective in specific projects and particular circumstances. In fact, the type, size and degree of accuracy required in evaluations depend on the type of project and other management factors and should be identified and included in the risk management program. Some of the weaknesses of quantitative evaluation method are as follows (Gharehchopogh et al., 2014; Lagerstrom et al., 2012; Sunkle and Kulkarni, 2012):

- The use of these methods usually requires software. Although they have good function and ability in accurate analysis, but they impose higher cost on project management and their implementation requires skilled manpower. They also need to be coordinated with other management tools of project.
- The analysis of the outputs of this method requires interpretation and judgment of the experienced people Otherwise, misdiagnosis is likely to happen.
- Over trust to outputs and software results without the judgment of software analyst can cause error.

The most prevailing methods for estimating the cost of software projects include Algorithmic models such as COCOMO (Boehm, 1981, 2000) and Function Point (Albrecht and Gaffney, 1983). These models work on the basis of a linear function and a number of input parameters that are really effective in the projects. Algorithmic models use characteristics such as LOC and complexity for estimating the development of software projects. In this paper, a hybrid model is presented for training and testing the parameters of COCOMO model. Being inspired form social life of bees; ABC Algorithm was invented to solve optimization problems (Karaboga, 2005). This Algorithm is the simulation of the searching-for-food behavior of bees for solving optimization problems.

This paper is organized as follows: In Section 2, we will discuss the previous works that have been done; in Section 3, the proposed model will be discussed; in Section 4, we will describe the evaluation and results of the proposed model, and finally, in Section, 5 we will come to the conclusion and the future works.

2. Related works

Many studies have been performed in the SCE; but, it should be noted that as long as all these studies are not detailed and accurate, they will not show their effectiveness in dealing with the attempt and cost of the project. PSO-FCM and PSO-LA hybrid models have been proposed for the SCE (Gharehchopogh et al., 2014). Evaluation was performed on NASA60 Dataset. In the hybrid model of PSO-FCM, the minimum of the inter-cluster distances and the total of intra-cluster distances, along with the number of clusters have been used as the parameters of fitness and for the optimization of Particle Swarm Optimization (PSO) algorithm. The use of Fuzzy C-Means (FCM) makes the particles gather in the best clusters and causes the fitness function to have many local optimum points. In order to improve the performance of the PSO algorithm Learning Automata (LA) is used to regulate the behavior of particles. All particles in the PSO-LA hybrid model search for a place in search space simultaneously. In the hybrid PSO-LA model, the strategy of LA model according to the reward parameter for PSO algorithm enables the particles to achieve several local optimums. The results of their experiments show that the MRE error in the hybrid PSO-FCM model is less in comparison to the PSO-LA hybrid models. The amount of MMRE in the PSO-FCM model is 25.36, 24.56, 24.22, and 23.86 in the PSO-LA model equal 26.32. The accuracy of PRED (25) on COCOMO Model is 40 and on Model PSO-FCM is 61.6, 58.3, 65 and 68.3, respectively. It is also 63.3 in PSO-LA models.

A new model for estimating effort was proposed using PSO algorithm (Gharehchopogh et al., 2014). Effective parameters in effort estimation have been analyzed using PSO algorithm. The evaluation has been conducted on KEREMER dataset with 15 projects. Test results show that the amount of MMRE in the proposed model is 56.57 and in COCOMO model is 245.39.

Ant Colony Optimization (ACO) and GA hybrid model based on the training and testing of software factors have been proposed for SCE (Maleki et al., 2014). The evaluation has been don on NASA60 dataset. Using the ACO the process of data training has been done using ACO and the process of data testing was performed using GA. The results with 10 projects in the hybrid model show that the hybrid model compared with COCOMO model has about 0.9 less MRE. The value of MMRE for 60 projects in the hybrid model is equal to 27.53 and in the COCOMO model is 29.64. The hybrid model is reduced the value of MMRE to about 1.07 times.

Models of K-Means, PSO and Differential Evolution (DE) have been proposed for the SCE (Jin-Cherng et al., 2013). Evaluation was performed on COCOMO dataset with 63 project. The results showed that MMRE vale in the hybrid model of DE & K-Means with 3 clusters is equal to 0.21, with 4 clusters is 0.20 and with 5 clusters is equal to 0.25 and in the COMOCO model is 0.26. also the PRED (25) value in the DE & K-Means hybrid model with 3 clusters is 0.55, with 4 clusters is 0.62 and in K-Means & DE hybrid model with 3 clusters is 0.57, with 4 clusters is 0.62 and with 5 clusters is 0.57. The value of PRED (25) in the COCOMO model is equal to 0.54. Comparisons showed that K-Means & PSO model with 4 clusters has increased the PRED accuracy to about 1.14 and the K-Means & DE model with 5 clusters have increased the PRED accuracy by approximately 1.05 times.

K-Means & PSO hybrid model is proposed for the SCE (Srinivasa et al., 2013). Evaluation was performed on COCOMO81 dataset. Dataset has been divided into 3 clusters with different projects. Training and testing of clusters was conducted on 35 projects. The results show that the MARE value in the COCOMO model in the training and testing processes is 23.13 and 15.63, respectively and in the hybrid model is 19.49 and 10.96, respectively.

PSO and fuzzy model with PSOI, PSOII and PSOIII have been suggested for SCE (BalaKrishna and Krishna, 2012). Evaluation was conducted on the NASA dataset. In the fuzzy model, Triangle membership function has been used. The results show that the VAF value in the models of PSOI, PSOII and PSOIII is 98.92, 98.54 and 98.65 respectively, and the MARE value is 6.16, 6.53 and 6.47, respectively, and also the VARE value is 0.26, are 0.23 and 0.20. PSOII model has less VAF value in comparison with PSOI and PSO III and also the model of PSOI has less MARE value than PSOII and PSOIII.

The techniques of Liner Regression (LR), Artificial Neural Network (ANN), SVR and K Nearest Neighbors (KNN) are used for SCE (Khalifelu and Gharehchopogh, 2012). LR model can be used to determine the dependence of the effective attributes in SCE. LR model finds the relationship between the dependent and the independent factors among the data. ANN tries to reduce the amount of MRE error with training and testing the data. SVR model is used for the optimization of effective factors in SCE. KNN is a technique in data-mining which has been used for classifying data in a collection of data that has already been classified and whose specifications have been identified. Using KNN, the weight of effective traits in SCE is determined. Prediction accuracy on the training data in LR, ANN, SVR and KNN models are 74%, 87%, 95% and 68%, respectively. Also prediction accuracy on the testing data in LR, ANN, SVR and KNN models is 60%, 95%, 80% and 60%, respectively. The amount of MAE in LR, ANN, SVR and KNN models is 39.17%, 12.07%, 36.32% and 1.12%, respectively. The amount of RRSE in LR, ANN, SVR and KNN models is 11.6%, 2.37%, 20.08% and 0.77%, respectively. Their results showed that KNN model has lower amount of MAE than the other models.

ANN-MLP model is one of the conventional methods in the SCE (Gharehchopogh, 2011). In order to demonstrate the performance of ANN, from 60 projects available in NASA software dataset which have been tested and trained using ANN, 11 projects have been compared with COCOMO model and it has been shown that the amount of MRE in COCOMO model is lower than ANN model. 80% of the projects are used for training and 20% is used for testing. The results show in more than 90% of the projects, ANN model has presented much better estimation than the COCOMO model.

Another model proposed by researchers for SCE is Tabu Search (TS) algorithm (Ferrucci et al., 2009). Evaluation is performed on *Dasharnais* dataset. The results of experiments of TS have been compared with Case-Based Reasoning (CBR) (Kadoba and Shepperd, 2001) and Stepwise Regression (SWR) (Kitchenham, 1998) models. The results showed that the amount of MMRE on TS model is 0.45 and 0.48, respectively; it is 0.48 and 0.39 in the SWR and CBR models, by the way. The accuracy of PRED (25) on the TS model is 0.39 and in the CBR and SWR models is 0.55 and 0.22, respectively; and the amount of MDMRE in CBR and SWR models is 0.22 and 0.38, respectively. Comparisons show that MMRE on TS models is less than CBR and SWR models, and the SWR has lower amount in comparison with the CBR. The accuracy of PRED (25) on the CBR is higher than TS and SWR models.

KNN and C4.5 models also have been proposed for the SCE (Song et al., 2008). Evaluation has been done on Desharnais, COCOMO81, ISBSG, and Albrecht datasets. In most of the projects, the accuracy of C4.5 model is more than KNN. Detecting and training of the data in C4.5 model has better prediction. KNN model in the COCOMO81 and Albrecht datasets enjoys more PRED accuracy.

C4.5 model is more effective in detecting projects which do not belong to a certain group of projects. Neuro-Fuzzy technique has been proposed for calibrating the parameters of COCOMO model (Huang et al., 2007). The proposed model emphasizes on training and testing the data. Evaluation was conducted on the dataset of the software projects of COCOMO. Test results show that the accuracy of PRED (20) and PRED (30) in comparison to COCOMO model has increased to about 15% and 11%, respectively.

3. Proposed model for SCE

SCE is used as an important process in cost and effort conditions in order to achieve the objectives of the project. SCE contains all areas of project management, including cost and effort management. Several factors have been proposed to evaluate the performance of these factors. These factors are defined in the scope of quality, time and cost that are the framework of working domain of a project. The most important point is that these factors must be completely and practically defined and combined as an overall index. Finally, the projects would be evaluated on the basis of this overall index. In the NASA93 software project dataset, the most important factors for estimating cost and effort have been defined on the basis of Effort Multipliers (EMs) (Menzies, 2006). EMs factors assess and measures the results and the evaluation of the projects. When LOC has been identified and analyzed in the projects, suitable values must be adopted for EM factors. In the proposed model, the values of these factors which are on the basis of ABC algorithm have been determined for the parameters of COCOMO model. Fig. 1 shows the overall structure of the COCOMO-ABC model for the optimization of the parameters of the COCOMO model. In the hybrid model, EMs factors and also the size of the projects have been considered as the input of the data, and MMRE has been considered as the function's objective.

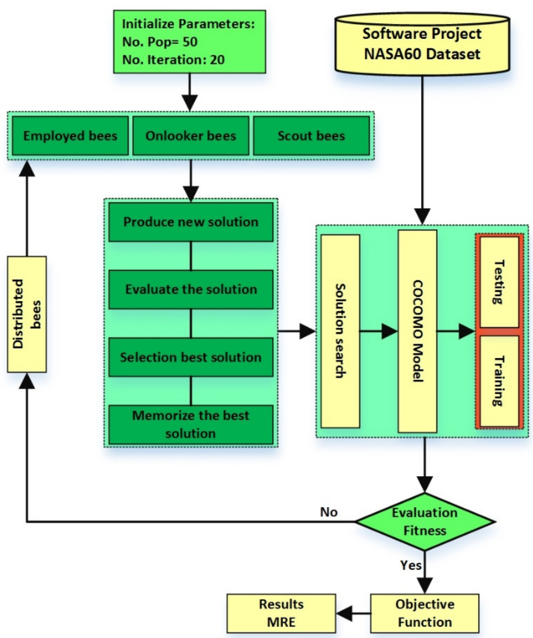

Fig. 1: Flowchart of the proposed model

In the COCOMO model, the way of estimating the cost and effort of the software projects is defined based on the Eq. (1). The parameters (a) and (b) are fixed constants whose amounts depend on the data of the dataset. The Size parameter is the size of project on the basis of thousands of source lines of code. EMs parameters decrease and increase the amount of effort on the basis of person and month (Boehm, 1981, Boehm, 2000).

$$PM = a * (Size)^b * \prod_{i=1}^{15} EM_i \qquad (1)$$

In the hybrid model, a colony is assigned to any of the EMs factors. Then, in each colony, an initial population is generated randomly and uniformly. Individuals of the colony start to work in a parallel way, and searching the space. At the end of each iteration, the best obtained value is stored and computed in each colony. To calculate the best value for EM factors, it is enough to obtain the corresponding position with the optimized amount of each colony in the objective function and then replace the amount obtained value of each colony in the COCOMO model. In each iteration, all the bees update their locations with estimating the new location and move to the optimized location. New position is sum of new values which is used for reducing MMRE. Unless the MMRE is reduced, bees will move to the new location, and the previous values will be thrown away; testing the parameters of the COCOMO model using ABC will continue to the where that the value of the parameters is optimized through the colony's individuals and the objective function is satisfied. Once training process is completed, values for calculating MMRE are tested. In the hybrid model, MMRE has been considered as the fitness function. The objective of the fitness function in the hybrid model is minimizing the amount of MMRE in comparison with COCOMO model. In the hybrid model, until the value of MMRE is reduced in a desirable way, it will be repeated. Fitness function for the hybrid model is defined according to Eq. (3) (MacDonell and Gray, 1997).

$$MRE_i = \frac{|act_i - est_i|}{act_i} \times 100 \qquad (2)$$

$$MMRE = \frac{1}{n}\sum_{i=1}^{n} MRE_i \, , i = 1,2,\ldots,n \qquad (3)$$

Using Eq. 3 we can estimate the total obtained error by comparing the estimation models. PRED is also considered an important factor in the accuracy of the estimation. MMRE and PRED are the most prevailing methods to evaluate the accuracy of prediction. PRED (x) is defined according to Eq. (4) (MacDonell and Gray, 1997).

$$PRED(x) = \frac{1}{n} \times \sum_{i=1}^{n} MRE \le x \qquad (4)$$

PRED (x) which is defined based on MRE is the most effective in estimating of accuracy and provides a good illustration of how the models work. Evaluating the estimation criteria, the model which has lower amount of MRE is better in comparison with a model which has higher amount of MRE; the model which has lower amount of MRE is better than the one which has higher amount of MMRE. Also, the model which has higher amount of PRED is better than the one with lower amount of PRED.

4. Evaluation and results

Experiments have been conducted on NASA93 software projects dataset. Simulation of the models has been conducted in VC#.NET 2013 programming environment. The evaluation of models has been measured using MMRE criterion. In the ABC algorithm, the most important parameter is the number of iterations and the rate of population's distribution. Number of iterations has been considered with an average size of 20 and the population size of 50. Based on the obtained results of Table (1) can be said that in ABC algorithm was effective in hybrid with COCOMO.

Table 1: Mparison of MRE of the models

No. Projects	KSLOC	Actual Effort	MRE COCOMO	MRE ABC	MRE COCOMO-ABC
1	0.9	8.4	72.12	23.12	16.84
2	2.2	8.4	24.12	11.2	3.62
3	3.5	10.8	0.73	3.25	4.82
4	6.2	12	638.09	265.13	326.98
5	5.5	18	1.19	2.19	1.05
6	6	24	56.16	35.31	28.81
7	9.7	25.2	33.25	19.45	11.32
8	7.7	31.2	16.9	8.14	6.85
9	8.2	36	22.72	10.16	7.84
10	11.3	36	22.23	10.65	7.28
11	3	38	2.46	3.26	1.2
12	6.5	42	23.13	13.54	9.74
13	8	42	15.66	5.65	4.32
14	10	48	35.22	21.56	19.84
15	15	48	38.84	19.54	12.36
16	20	48	15.58	7.85	2.95
17	10.4	50	27.39	13.08	5.64
18	13	60	1.69	2.65	6.51
19	14	60	15.59	7.32	4.72
20	19.7	60	23.74	15.18	6.84
21	32.5	60	134.17	78.32	45.71
22	31.5	60	24.65	10.65	5.82
23	12.8	62	18.47	9.64	11.39
24	15.4	70	11.17	2.31	4.07
25	7.5	72	36.85	21.56	12.31
26	20	72	54.19	32.41	13.42
27	34	72	85.09	56.84	62.37
28	16.3	82	19.19	13.26	11.54
29	15	90	30.87	14.63	9.82
30	165	97	1014.53	752.4	650.09
31	11.4	98.8	35.56	11.03	9.32
32	21	107	58.99	28.31	15.48
33	16	114	25.04	10.42	15.94
34	24.6	117.6	19.03	8.92	5.08
35	25.9	117.6	14.23	5.65	1.17
36	29.5	120	2.75	1.3	5.35
37	40	150	45.37	21.46	16.05
38	19.3	155	25.93	19.85	13.77
39	90	162	18.52	6.31	9.81
40	32.6	170	15.28	9.81	7.4
41	35.5	192	17.47	11.23	8.55
42	240	192	199.07	94.55	65.42
43	38	210	13.06	8.42	3.07
44	100	215	92.47	52.13	54.39
45	48.5	239	5.97	6.15	4.1
46	20	240	73.35	33.21	21.43
47	47.5	252	22.92	13.06	15.94
48	70	278	0.02	2.64	1.81
49	66.6	300	3.16	6.42	4.63
50	85	300	31.88	21.03	16.8
51	98	300	77.86	45.13	33.49
52	150	324	76.37	52.24	46.93
53	66.6	352.8	17.66	12.66	8.07
54	100	360	13.52	10.72	5.35
55	100	360	16.17	11.06	9.13
56	50	370	21.98	18.73	15.62
57	79	400	30.04	16.3	10.47
58	60	409	5.31	6.83	3.21
59	190	420	4.03	5.68	6.02
60	24	430	65.94	36.02	28.14
61	151	432	42.47	23.11	35.26
62	90	444	18.81	5.64	6.4
63	339	444	492.88	289.25	315.68
64	70	458	2.99	3.49	5.79
65	16.3	480	41.14	32.56	28.34
66	53	480	27.66	19.65	15.19
67	78	571.4	3.98	10.28	5.23

68	144	576	42.55	32.39	28.49
69	41	599	53.51	40.64	46.81
70	111	600	27.33	19.21	15.06
71	137	636	14.63	9.62	5.61
72	7.25	648	83.99	88.23	46.11
73	100	703	1.7	2.47	3.4
74	350	720	67.99	52.38	39.44
75	101	750	19.64	11.17	8.31
76	162	756	43.76	33.19	25.65
77	150	882	1.11	3.65	2.4
78	284.7	973	39.13	45.72	31.68
79	227	1181	4.71	5.64	6.04
80	352	1200	115.24	95.24	90.13
81	177.9	1248	1.79	3.26	2.06
82	32	1350	10.21	4.79	5.92
83	282.1	1368	16.7	10.86	7.4
84	70	1645.9	32.83	21.73	13.07
85	65	1772.5	34.4	25.94	19.11
86	50	1924.5	55.9	65.92	54.32
87	219	2120	28.79	25.19	16.03
88	302	2400	31.6	29.83	25.01
89	423	2400	52.34	44.61	56.44
90	271	2460	2.68	3.21	1.3
91	165	4178.2	14.89	8.54	6.18
92	980	4560	442.25	375.91	326.45
93	233	8211	34.48	25.6	17.46

In Table 2, the results of models based on the MMRE and PRED (30) criteria are shown. In the COCOMO model, the MMRE value is equal to 58.80 and in the models of COCOMO-ABC and ABC is 38.13 and 33.22. The comparison shows that the value of MMRE in the model of COCOMO-ABC has been decreased by about 1.77 times in comparison with COCOMO.

Table 2: A Comparison of MMRE and PRED (30) on the Models

Models	MMRE	PRED (30)
COCOMO	58.80	56.98
ABC	38.13	74.19
COCOMO-ABC	33.22	81.72

Results of Table 2 shows that the model COCOMO-ABC is largely reduced the MRE criterion in comparison with ABC and COCOMO models. Therefore, the hybrid model was useful to estimate and has less estimation error than the COCOMO and ABC models. In Fig. 2, the diagram of the models based on the MMRE criterion is shown.

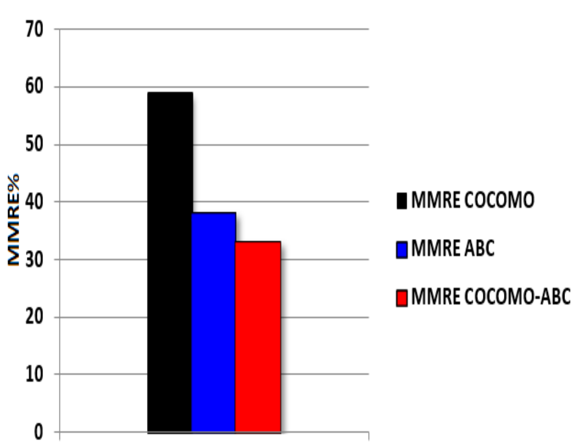

Fig. 2: The Comparison of MMRE Criterion in the Models

In Fig. 3 the graph of the models based on the criteria of PRED (30) is shown. The results of models showed that COCOMO-ABC model has increased the PRED accuracy to about 1.43 times than the COCOMO model and to 1.10 times greater than the ABC model.

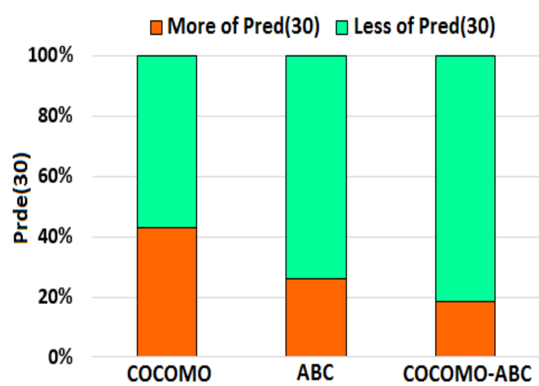

Fig. 3: the comparison of PRED criterion on models

The results of the implementation on NASA93 software projects showed that high accuracy and the ability to accurately estimate the cost in the model of COCOMO-ABC are better than ABC and COCOMO models.

5. Conclusions and future works

Projects' failure can be affected by many factors, some of these factors are errors in the initial estimates, lack of on-time estimate, lack of integrated companies, technical changes during the project, the complexity of the projects, changes in market conditions and budget cuts. These factors can effect on other issues of projects such as estimates of effort and time that in this case they might be associated with negative outcomes for project management. Inaccurate SCE can result in a change in the contract time of software projects and direct and indirect costs or both of them. So, project management team software must be able to effectively estimate the costs in order to reduce the negative impacts of the project to a minimum. In this paper, COCOMO-ABC model was proposed to better estimate and outperform the COCOMO model. Results indicated that the COCOMO-ABC model has reduced the MMRE value to about 1.77 times than the COCOMO model. We hope with providing this paper present applicable models in the future for SCE using AI technique.

References

Albrecht, A.J. Gaffney, J. (1983). Software Function, Source Lines of Code, and Development Effort Prediction: a software science validation, IEEE Transactions on Software Engineering SE, 9(6): 639-648.

BalaKrishna, A. Krishna, T.K.R. (2012). FUZZY AND SWARM INTELLIGENCE FOR SOFTWARE EFFORT ESTIMATION, Advances in Information Technology and Management (AITM), 2(1), 246-250.

Boehm, B.W. (1981). Software Engineering Economics, Prentice-Hall, Englewood Cliffs, New Jersy.

Boehm, B.W. (2000). Software Cost Estimation with COCOMO II, Prentice Hall PTR, Englewood Cliffs, New Jersy.

Buglione, L. Gencel, C. (2008). Impact of Base Functional Component Types on Software Functional Size Based Effort Estimation, Product-Focused Software Process Improvement, Vol. 5089, pp. 75-89.

Ferrucci, F. Gravino, C. Oliveto, R. Sarro, F. (2009). Using Tabu Search to Estimate Software Development Effort, Springer-Verlag Berlin Heidelberg 2009, LNCS 5891, pp. 307-320.

Gharehchopogh F.S., Maleki I., Sadouni S. (2014). Artificial Neural Networks Based Analysis of Software Cost Estimation Models, MAGNT Research Report, 2 (6): 597-505.

Gharehchopogh, F.S. (2011). Neural Networks Application in Software Cost Estimation: A Case Study, 2011 International Symposium on Innovations in Intelligent Systems and Applications (INISTA 2011), pp. 69-73, IEEE, Istanbul, Turkey.

Gharehchopogh F.S., Maleki I., Ghoyunchizad N., Mostafaee E. (2014). A Novel Hybrid Artificial Immune System with Genetic Algorithm for Software Cost Estimation, MAGNT Research Report, 2 (6): 506-517.

Gharehchopogh, F.S. Maleki, I. Khaze, S.R. (2014). A Novel Particle Swarm Optimization Approach for Software Effort Estimation, International Journal of Academic Research, Part A, 6(2), 69-76.

Gharehchopogh, F.S. Ebrahimi, L. Maleki, I. Gourabi, S.J. (2014). A Novel PSO based Approach with Hybrid of Fuzzy C-Means and Learning Automata in Software Cost Estimation, Indian Journal of Science and Technology, 7(6), 795-803.

Gharehchopogh. F.S, Talebi. A, Maleki. I (2014). Analysis of Use Case Points Models for Software Cost Estimation, International Journal of Academic Research, Part A, 6(3), 118-124.

Heidrich, J. (2013). Software Effort Estimation and Risk Management, Product-Focused Software Process Improvement, Springer Berlin Heidelberg, Vol. 7983, pp. 370-371.

Huang, X. Ho, D. Ren, J. and Capretz, L.F. (2007). Improving the COCOMO Model using a Neuro-Fuzzy Approach, Applied Soft Computing, 7 (1): 29-40.

Jin-Cherng Lin, Yueh-Ting Lin, Han-Yuan Tzeng, Yan-Chin Wang (2013)., Using Computing Intelligence Techniques to Estimation Software Effort, International Journal of Software Engineering & Applications (IJSEA), 4(1): 43-53.

Karaboga, D. (2005). An idea based on honeybee swarm for numerical optimization, Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Kadoba, G. Shepperd, M. (2001). Using Simulation to Evaluate Predictions Techniques, In: Proceedings of International Software Metrics Symposium, pp. 349-358, IEEE Press, Los Alamitos.

Khalifelu, Z.A. Gharehchopogh, F.S. (2012). Comparison and Evaluation Data Mining Techniques with Algorithmic Models in Software Cost Estimation, Elsevier, Procedia-Technology Journal, ISSN: 2212-0173, Vol. 1, pp. 65-71.

Kitchenham, B.A. (1998). A Procedure for Analyzing Unbalanced Datasets, IEEE TSE, 24(4): 278-301.

Lagerstrom, R. Würtemberg, L.M. Holm, H. Luczak, O. (2012). Identifying Factors Affecting Software Development Cost and Productivity, Software Quality Journal, Springer Berlin Heidelberg, 20(2): 395-417.

Liu, Q. Mintram, R.C. (2005). Preliminary Data Analysis Methods in Software Estimation, Software Quality Journal, Springer Berlin Heidelberg, 13(1): 91-115.

Maleki, I. Gharehchopogh, F.S. Ayat, Z., Ebrahimi, L. (2014). A Novel Hybrid Model of Scatter Search and Genetic Algorithms for Software Cost Estimation, MAGNT Research Report, 2(6): 359-371.

Maleki, I. Ebrahimi, L. Gharehchopogh, F.S. (2014). A Hybrid Approach of Firefly and Genetic

Algorithms in Software Cost Estimation, MAGNT Research Report, 2(6): 372-388.

Maleki. I, Ebrahimi. L, Jodati. S, Ramesh. I (2014). Analysis of Software Cost Estimation Using Fuzzy Logic, International Journal in Foundations of Computer Science & Technology (IJFCST), 4(3) 27-41.

Maleki, I. Ghaffari, A. Masdri, M. (2014). A New Approach for Software Cost Estimation with Hybrid Genetic Algorithm and Ant Colony Optimization, International Journal of Innovation and Applied Studies, 5(1), 72-81.

Menzies, T. (2006). The PROMISE Repository of Software Engineering Databases. School of Information Technology and Engineering, University of Ottawa, Canada (2006), http://promise.site.uottawa.ca/SERepository [Last Access: Oct 2014]

MacDonell, S.G. Gray, A.R. (1997). A Comparison of Modeling Techniques for Software Development Effort Prediction, in Proceedings of International Conference on Neural Information Processing and Intelligent Information Systems, pp. 869-872.

Srinivasa Rao T, Hari CH.V.M.K, Prasad Reddy P.V.G.D, (2013). Predictive and Stochastic Approach for Software Effort Estimation, International Journal of Software Engineering, 6(1): 97-115.

Song, Q. Shepperd, M. Chen, X. Liu, J. (2008). Can K-NN imputation improve the performance of C4.5 with small software project data sets? A comparative evaluation, The Journal of Systems and Software, 81, 2361-2370.

Sunkle, S. Kulkarni, V. (2012). Cost Estimation for Model-Driven Engineering, Model Driven Engineering Languages and Systems, Springer Berlin Heidelberg, Vol. 7590, pp. 659-675.

Trendowicz, A. (2013). Why Software Effort Estimation, The Fraunhofer IESE Series on Software and Systems Engineering, Springer Berlin Heidelberg, pp. 3-7.