European Journal of Applied Sciences 8 (2): 126-139, 2016
ISSN 2079-2077
© IDOSI Publications, 2016
DOI: 10.5829/idosi.ejas.2016.8.2.22892

Software Effort Estimation of Gsd Projects Using Calibrated Parametric Estimation Models

[1]*S. Ramacharan and* [2]*K. Venu Gopala Rao*

[1]Department of IT, Vasavi Engineering College, Hyderabad, India
[2]Department of CSE, Osmania University, Hyderabad, Telangana, India

Abstract: Software Engineering is the domain of computer science developed for designing, coding and testing of various software projects of computer and other electronic devices. Global Software Development (GSD) is the environment for developing software projects at geographically isolated areas beyond cultural peripheries in a harmonized vogue comprising synchronous and asynchronous communication. The conventional methods are utilized to estimate effort for co-located projects which are not efficient for GSD projects. The parametric effort estimation models COCOMO II and SLIM can estimate effort for both co-located and GSD projects but it has some impact factors on GSD projects based on their cultural difference and distance. Here we have introduced a model for effort estimation named as Scheduling-based model. In this research calibration on three models for estimating effort in GSD environment is proposed to achieve better performance. The performance can be evaluated based on the accuracy of effort measured from the deviation of actual effort from completed projects and estimated effort before and after calibration. The calibrated estimation shows better accuracy in GSD projects than the estimation before calibration.

Key words: GSD · COCOMO II · SLIM · Calibration · Effort Estimation · Scheduling-based models

INTRODUCTION

Global Software Development (GSD) is constantly employing the software industry because of the significant reduction in cost and development time, access to specialized skilled personal across the globe. Building vigorous communication and coordination mechanisms among the distributed teams for utilizing their full potential is the vital challenge in distributed development systems [1]. The resource requirements of software development in GSD environment have two hypotheses, namely coherence and collocation. Coherence represents that splitting of work is depends on product functionality and Collocation involves pursuing of the similar set of functionality or component [2]. Organization which have responsibilities in communication aids to enhance shared understanding of requirements among GSD team members [3].

Efforts estimation for Global Software Development is the crucial activity of evaluating the most pragmatic use of effort needed to maintain software. The cost and effort for developing a software product can accurately determine utilizing the model for the success of whole development project [4, 5]. Schedule, productivity, cost, resources and quality are the factors of estimating software product development projects. Software estimation models are categories into three models they are Expert estimation, Formal estimation and Combination based estimation or Hybrid estimation model [6]. The baseline effort can be obtained using expert estimation, Formal estimation model is based on mechanical processes, whereas Hybrid estimation model is based on judgmental or mechanical grouping of estimates from different sources [7].

COCOMO II is an intention cost model for planning and executing software projects, which includes two inherent information models. The first is architecture for describing a software project and second is a knowledge base that can be used to estimate the similar data sets [8]. SLIM is an algorithmic model based on, accurate estimate of size of software regarding line of code (LOC) and statistical analysis of prior data [9]. The International Software Benchmarking Standards Group (ISBSG) warehouse (release 8) consists of 2027 projects collected

Corresponding Author: S. Ramacharan, Department of IT, Vasavi Engineering College, Hyderabad, India.

from twenty countries around the world among that Australia (21%), Japan (20%) and the United States (18%) are the vital contributors. Missing data in ISBSG dataset can be handled by the Mean or Mode Single Imputation (MMSI) method [10].

Impact on the role of the respondents, the data collection approach and analysis methods are the major cause of estimation error. The estimation errors are measured by mean magnitude of relative Error and Mean relative error. Deviation is the difference between emerged effort and actual value which is given by MMRE and the bias of estimation is given by MRE [11].

Related Work: Plenty of researches have been implemented for project management and effort estimation in Global Software Development by utilizing different methods. Some of the recent related works are go through in the following section.

S. Arun Kumar and T. Arun Kumar [12] had proposed requirements management framework intended to facilitate the organizations that will result in forming a benchmarked approach from effectively manipulating requirements engineering controversies over various levels. Requirements in GSD were effectively managed by implementing the requirements management framework with an integrated approach and a validated model. The system utilized four steps, at first a Framework was formulated and in second step a combined organization structure of both traditional approaches and agile approaches was designed. In Third steps Ontology based Knowledge Management Systems were applied for both the approaches and finally propose requirements management metrics during the development of information systems to measure and manage software process.

Poonam Pandey [13] had performed an analysis of most of the algorithmic techniques which has been developed for Software Cost Estimation. It is very vital to estimate the involved Software Cost and Effort accurately for managing any software project effectively. All techniques are classified into two categories, the parametric models are derived from the historical projects data and the Non-Parametric models were based on a set of artificial intelligence techniques.

Saleem Basha and Dhava chelvan P [14] performed an analysis of empirical effort estimation models such as COCOMO, SEER-SEM, SLIM, REVIC, SASET and COSTMODL. Effort, cost and schedule for software projects were estimated by COCOMO model. SEER-SEM model was designed especially to estimate, plan and monitor the effort and resources recommended for any type of software development and/or maintenance project. SLIM model utilizes Rayleigh curve for effort prediction which represents manpower measured in person per time as a function of time and it is applicable for large projects exceeding 70, 000 lines of code. REVIC model is a direct descendent of COCOMO. REVIC includes additional cost multipliers such as requirements volatility, security, management reserve and an Ada mode. SASET provides derived software sizing values by using a hierarchically structured knowledge database of normalized parameters. COSTMODL is a COCOMO based estimation model which includes the basic COCOMO equations and modes along with some modifications.

Kavita Choudhary [15] had presented a parametric estimation of software systems. Effort Estimation required for the development of software projects utilized genetic algorithm approaches and values of actual effort, effort using genetic algorithm, magnitude of relative error (MRE) and mean of magnitude of relative error (MMRE) are computed. This work examined the correlation between different dimensions of software projects such as models, project size and effort.

Yousef Alkouni Alghdr *et al*. [16] estimated software Process maturity using COCOMO II's effort estimation based on CMMI. The new process maturity values was identified with the ideal scale factor method and with the aid of our datasets which better reflect the impact of CMMI based process maturity on software development effort.

Manpreet Kaur [17] had designed a simulator to perceive which model is better in terms of cost, effort, persons per month and source lines of codes by comparing the cost models such as COCOMO81 and COCOMO2.0. Inputs are taken as Source lines of code which is generated using random numbers. Cost drivers values and inputs are the major source for cost and effort estimation. Comparison of cost models was relies on the cost driver's values.

Vahid Khatibi and Dayang N.A. Jawawi [18] had researched software cost estimation methods to realize inaccurate estimations are the major cause for software project failures. COCOMO II utilizes function point (FP) or source line of code (SLOC) as the size of the project and constitute of 17 Effort Multipliers and 5 scale factors which uses rating level as its weight by assigning quantitative values for each levels. In SLIM model effort estimation is based on manpower distribution and the evaluation of many software projects. Calibration of parameters increases the accuracy of such methods.

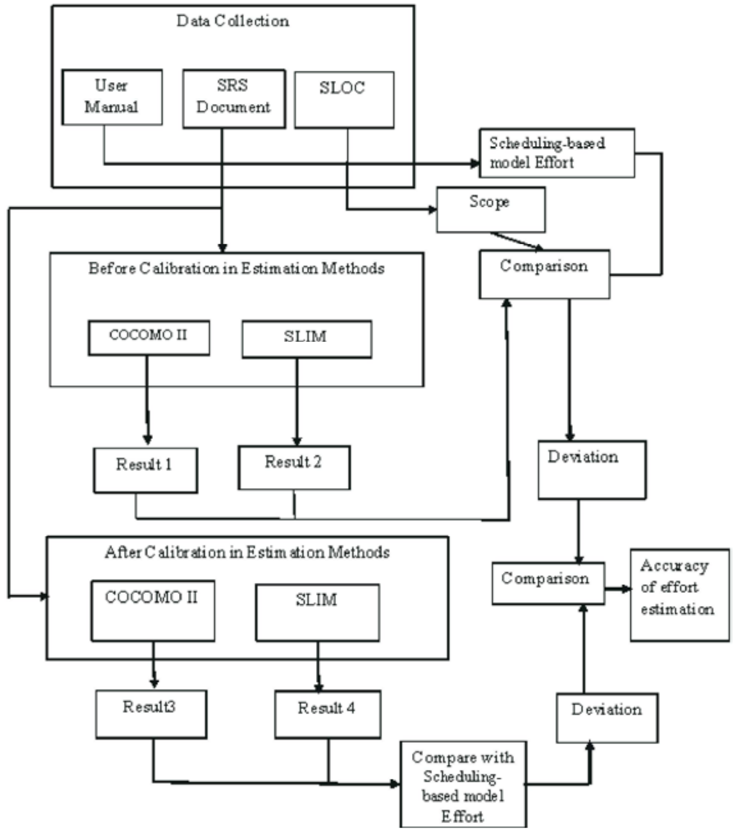

Fig. 1: Process of Proposed System

The objective of the above mentioned works is improving the accuracy of effort estimation. An accurate estimation of the effort remains complex and hardly realizable. In this system, Calibration and validation of parameters in COCOMO II and SLIM is proposed for achieving accurate effort estimation with low cost.

Accurate Software Effort Estimation in GSD by Calibrating the Parameters in Estimation Methods: This section illustrates the system design and Estimation Models used in the proposed system. In this system Effort estimation of GSD projects by calibrating values in parametric models are proposed. The overall process of the proposed work is illustrated in Fig. 1. At first Actual effort is calculated from organization history then the second phase utilizes the parametric estimation models such as COCOMO II, SLIM and Scheduling-based model to estimate the efforts for Software development projects and find the deviation. In third Phase measure the impact on GSD projects. Finally, calibrate the parametric values in estimation models to accurately estimate the effort of GSD projects.

Management of Project Development in GSD Environment: Differences in time zone, geographic location and communication are major challenges of project management in GSD environment which oscillate project's success. Project managers are updating themselves based on new patterns of GSD. Standardized approaches and problem solutions are lack to provide by literature and industry. GSD suggested controlling problems of communication manner. Overall result of the project was affected by wrong approach in project process distribution which is associated with task distribution. Effort estimation is the major activity of project management which is closely related to risk management. GSD factors such as multi sourcing, geographical distribution, temporal diversity, socio-cultural diversity, linguistic diversity, contextual diversity and political & legislative diversity are the vital cause of GSD threats that directly affects the GSD project management activities.

Collecting Data from Executed GSD Projects: Collect data about executed GSD projects from different organizations and it consist of user manuals, SRS documents and size of project in SLOC. The vital process of gathering data is by Exchanging emails and online interviews. Parameters of estimation methods were considered during the interviews to excite data according to these parameters. Essential data such as delay factors and recommendations from project managers to improve project estimation accuracy are also considered in data collection process.

Parametric Models for Effort Estimation: Effort estimation takes an important part in software engineering domain to estimate effort and duration need for completing a project. One of the major activities in software development is accurate the estimation of software projects that can solve the hurdles in projects. Effort estimation is essential for software development projects to set cost and time limit. Estimation of new projects utilizes history of previous projects to ensure optimistic results [19-25].

COCOMO II: COCOMO II is derived from COCOMO 81 developed by Dr. Barry Boehm in 1981. The word COCOMO is an acronym derived from Constructive Cost Model. Early stages the cost and schedule of software projects are most commonly estimated by COCOMO model. COCOMO II estimates the cost of project development and number of hours a person spends on working for software development project in a calendar month (PM).COCOMO II utilizes Source Lines Of Codes (SLOC) or Functional Points (FP) as an input for estimate the software projects' size. It uses algorithmic methods as well as data from previous projects or expert's knowledge to conclude the actual size of the project.

COCOMO II is classified into three sub models

- Application composition model
- Post architecture model
- Early design model

COCOMO II has 17 cost drivers as Effort Multipliers (EM) for post architectural model and 5 scale factors which influence the estimation of software projects.

Application composition model is used for estimating effort and schedule of the project which uses Integrated Computer Aided Software Engineering tools. The projects such as embedded system, software infrastructure and large applications utilize The Early Design and Post Architecture models for effort and schedule estimation. The size of the project and scale factors is perceived by Early Design and Post Architecture models.

Table 1: Effort Multipliers in COCOMO II

	EFFORT MULTIPLIRS
PRODUCT FACTORS	• Required Reusability (RELY) • Database Size (DATA) • Product Complexity (CPLX) • Developed for Reusability (RUSE) • Documentation match to life-cycle-model (DOCU)
PLATFORM FACTORS	• Execution Time Constraint (TIME) • Main Storage Constraint (STOR) • Platform Volatility (PVOL)
PERSONNEL FACTORS	• Analyst Capability (ACAP) • Programming Capability (PCAP) • Application Experience (APEX) • Platform Experience (PLEX) • Personal Continuity (PCON)
PROJECT FACTORS	• Use of Software Tools (TOOL) • Multisite Development (SITE) • Scheduling Factor (SCED)

Software Size Estimation in COCOMO II: COCOMO II calculates effort and schedule estimation based on size of the software for reliable estimation. Size of software includes new and reused code and modified code so it is difficult to estimate size of the project. Overall size of the project is used in COCOMO II for new and reused code with updates. Amount of changes in design, code and testing are taken into consideration for these adjustments. Source Lines Of Code (SLOC) and Function Points (FP) are the two major types of size used for effort and schedule estimation in COCOMO II.

Effort Multipliers in COCOMO II: As compared with COCOMO 81 more cost drivers are added with COCOMO II, in which the cost drivers are act as effort multipliers. In Post-Architecture Model (the sub model of COCOMO II) the cost drivers are classified into four categories that are shown in Table 1.

Scale Factors in COCOMO II: The scale factors are associated with the features of organization and teamthat permits the effort estimation team to build better approximation of influencing factors. Rating levels of each scale factor varies from a range of very low to extra high weight/value. For different organizations and projects, the weight of scaling factors could also be different. Table 2 shows the scaling factors of COCOMO II.

SLIM: An algorithmic technique SLIM, which is developed for measuring the whole size of a project based on its estimated SLOC and it is used to calculate effort

Table 2: Scale Factors in COCOMO II

Precedentedness	PREC
Development Flexibility	FLEX
Architecture/ Risk Resolution	RESL
Team Cohesion	TEAM
Process Maturity	PMAT

and schedule for projects. Rayleigh curve model is used to customize SLIM for effort estimation. Quantitative Software Management (QSM) developed SLIM tool, which is a metrics-based estimation tool using validated data of over 2600 projects. The effort and time required to build medium and large software projects of the management can be estimated by using SLIM tool that can be customized based on a specific organization.

Management Indicators in SLIM: Productivity Index (PI) and Manpower Buildup Index (MBI) are the two crucial management indicators in which PI could be taken as process productivity and MBI is a measure of staff buildup rate. PI values were noticed from 0.1 to 34 in which a high PI value means that project's productivity is high and it is low complex and PI having values below average implies 10% more development time and 30 % more cost. To overcome future contingencies the PI values were given as 0.1 to 40.0 in SLIM tool by QSM. MBI values are noticed in the range of -3 to 10 and the factors in which MBI influence are schedule pressure, task concurrency and resource constraints.

Actual effort Estimation

Project A

Table 3: Actual Effort for Project A

Phase	Number of Persons	Weeks	Days
Define and Discovery	10	4	200
Design	11	4	220
Build	15	8	600
SIT	11	3	165
UAT	10	4	200
Deployment + Training	7.5	2	75
Total	64.5	25	1460

Project B

Table 4: Actual Effort for Project B

Phase	Number of Persons	Person Hours	Person Days
UBR	1	5	9.375
Analyze	2	2.23	10.035
Prepare Master Test Plan	15	0.5	17.578125
Execute Test	1	18.5	18.5
Update Test Plan	22	0.5	7.5
Update Test Scenarios	22	0.2	4.4
Total	63	26.93	71.788125

Effort Estimation with COCOMO II: COCOMO II model estimates the effort (in Person-Months) required for completing a project based on the size of software projects, which is supposed to be effected SLOC, excluding blank spaces and comments. This size is calculated into thousands source lines of code (KSLOC) of software project.

Formula for estimating effort in COCOMO II is

Effort in $PM = C*\left(Sizeof\ project\right)^{ESF*}\prod_{\times=1}^{17} A_{\times}$ (1)

where C= 2.94 (for COCOMO II)

A is the effort multiplier in which all cost driver values related to the project are multiplied with each other. ESF is the exponent derived from scale factors that can be calculated by the following formula

$$ESF = D + 0.01 * \sum_{y=1}^{5} B_y \quad (2)$$

where D= 0.91 (for COCOMO II)

B_y is the scale factor introduced in COCOMO II that mostly contains organizational and project team characteristics which is calculated by adding all five-scale factor values.

Effort Estimation for Project A with COCOMO II: Table 5 and 6 represents values of effort multipliers and scale factors used for Project A, which were accumulated through interviews and characteristics of organization.

Effort Estimation for Project B with COCOMO II: Table 7 and 8 constitute of effort multipliers and scale factors used for Project B, which were collected through interviews and characteristics of organization.

Effort Estimation with SLIM: SLIM model can estimate the effort and time required for consummate software projects based on Source Lines Of Codes and the Productivity parameter depends on the range of productivity index. Formula for estimating effort in man years by using SLIM model is:

$$Effort\left(MY\right) = \left(\frac{sizeof\ SLOC}{PRP * duration\left(Y\right)^{4/3}}\right)^3 * SSF \quad (3)$$

where, Duration(Y) is the time taken for completing project in years, SSF is a special skill factor and PRP is the productivity parameter that can be calculated by using the following formula.

Table 5: Scale Factors for Project A

B_y (Scale Factors)	Range
Precedentedness (PREC)	VH: 1.24
Development Flexibility (FLEX)	H: 2.03
Architecture/ Risk Resolution (RESL)	H: 2.83
Team Cohesion (TEAM)	N: 3.29
Process Maturity (PMAT)	H: 3.14

Table 6: Effort Multipliers for Project A

Cost Drivers	Range
PRODUCT FACTORS	
• RELY	N: 1.00
• DATA	N/A
• CPLX	L: 0.87
• RUSE	N: 1.00
• DOCU	N:1.00
PLATFORM FACTORS	
• TIME	N/A
• STOR	N/A
• PVOL	L: 0.87
PERSONNEL FACTORS	
• ACAP	H: 0.85
• PCAP	H: 0.88
• APEX	VH: 0.81
• PLEX	VH: 0.85
• LTEX	VH: 0.85
• PCON	N/A
PROJECT FACTORS	
• TOOL	VH: 0.78
• SITE	VL: 1.22
• SCED	L: 1.14

Table 7: Scale Factors for Project B

B_y (Scale Factors)	Range
Precedentedness (PREC)	VH: 1.24
Development Flexibility (FLEX)	VH: 1.01
Architecture/ Risk Resolution (RESL)	VH: 1.41
Team Cohesion (TEAM)	H: 2.19
Process Maturity (PMAT)	H: 3.14

Table 8: Effort Multipliers for Project B

Cost Drivers	Range
PRODUCT FACTORS	
• RELY	VL: 0.82
• DATA	N/A
• CPLX	L: 0.95
• RUSE	L: 0.91
• DOCU	L:0.87
PLATFORM FACTORS	
• TIME	N/A
• STOR	N/A
• PVOL	L: 0.87
PERSONNEL FACTORS	
• ACAP	H: 0.85
• PCAP	H: 0.88
• APEX	VH: 0.81
• PLEX	VH: 0.85
• LTEX	VH: 0.84
• PCON	N/A
PROJECT FACTORS	
• TOOL	VH: 0.78
• SITE	VL: 1.22
• SCED	VL: 1.43

Table 9: Values for SLIM

Variables	Project A	Project B
SSF (special skill factor)	27.13	8.5
PRP (Productivity Parameter)	4.4501	5.10
Duration(Y)	65	78
Size of SLOC	2500	2500

$$PRP = \frac{sizeof\ SLOC}{\left(\frac{Effort(MY)}{SSF}\right)^{1/3} * duration(Y)^{4/3}} \tag{4}$$

Effort Estimation for Project A with SLIM: From the variables value given in Table 9 Productivity Parameter can be calculated which is equal to the Productivity index of business applications.

PRP= 4.4501

By using PRP and other variable values in SLIM equation, effort in person month can calculated.

Effort (MY) = 264

Effort Estimation for Project B with SLIM: From the variables value given in Table 9 Productivity Parameter can be calculated as given below;

PRP= 5.10

By using PRP and other variable values in SLIM equation, effort in person month canbe calculated as given below;

Effort (MY) = 67

Effort Estimation after Calibration in Parametric Models: COCOMO II and SLIM are two algorithmic and well-organized methods with various parameters that can be used to calibrate this methods according to the needs of project and organization and can use or exclude some parameters if not associated with specific project or organization.

In GSD projects cultural differences are the major impact so that new cost drivers are introduced on COCOMO II model and set their values between 1 and 2 for different scales which can be derived by studying the culture difference between onshore and offshore sites (Same culture would lead to a low scale value and very different culture would be very high or extra high scale value). Likewise, GSD factors are added in SLIM model for calculating PRP value and productivity index for GSD organization that will affect the figure of PRP, which automatically affects the result. Calibration is needed when estimating effort for GSD projects by using parametric models in which SLIM model uses very few parameters for estimation as compare to COCOMO II and makes the process easy.

Calibration in COCOMO II: For GSD environment, some GSD factors are additionally added with effort estimation models for accurate estimation. In COCOMO II eleven effort multipliers are added as GSD factorsand calibrate the scale factor values which can accurately estimate the effort of GSD projects. The range of each factor is varied from very low to very high value based on the characterization of each GSD factor. Effort Multipliers for GSD Projects are shown in the following Table 10.

Formula for estimating effort after calibration in COCOMO II is

$$Effort\,in\,PM = C*\left(Size of\ project\right)^{ESF}*\prod_{Y=1}^{11} AO_y \qquad (5)$$

where, AO_y is Outsourcing effort multiplier.

Effort of Project a after Calibration with COCOMO II: Table 11 and 12 shows the scale factors and effort multipliers of project A after calibration in COCOMO II model.

Effort of Project B after Calibration with COCOMO II: Table 13 and 14 shows the scale factors and effort multipliers of project A after calibration in COCOMO II model.

Calibration in SLIM: SLIM is an algorithmic model of estimating effort of various projects with parameters such as Productivity Index and Manpower Buildup Index. In GSD projects some parameters of SLIM can be calibrated to accurately estimate the effort based on GSD Factors.

Effort of Project a after Calibration with SLIM: Effort of Project A can be estimated after the calibration of parameters in SLIM given in Table 15 Productivity Parameter can be calculated as follows;

PRP = 3.2017

By using PRP value of Project A and other variable values in SLIM equation after calibration, effort in person month can calculated.

Table 10: Effort Multipliers for GSD Factors

GSD FACTORS FOR COCOMO II	
OUTSOURCING FACTORS	• Cultural Distance (CULT) • Barrier of Language (BALA) • Different Time Zones (TMZN)
BUYERS OUTSOURCING MATURITY	• Buyer's Outsourcing Experience (BOXP) • Buyer's Project Managers (BUMP) • Contract Design (CODS)
PROVIDERS OUTSOURCING MATURITY	• Provider's Outsourcing Experience (POXP) • Provider's Project Managers (PUMP)
COORDINATION FACTORS	• Outsourcer's Fit (OFIT) • Project Management (PMGM) • Team Spirit (TESP)

Table 11: Scale Factors for Project A after calibration

B_y (Scale Factors)	Range
Precedentedness (PREC)	VH: 0.90
Development Flexibility (FLEX)	VH: 0.50
Architecture/ Risk Resolution (RESL)	VH: 0.61
Team Cohesion (TEAM)	H: 1.22
Process Maturity (PMAT)	H: 2.14

Table 12: Effort Multipliers for Project A after calibration

Cost Drivers	Range
OUTSOURCING FACTORS	
• CULT	VL: 0.72
• BALA	N/A
• TMZN	L: 0.85
BUYER'S OUTSOURCING MATURITY	
• BOXP	VL: 0.82
• BUMP	N/A
• CODS	L: 0.75
PROVIDER'S OUTSOURCING MATURITY	
• POXP	VL: 0.62
• PUMP	N/A
COORDINATION FACTORS	
• OFIT	VL: 0.82
• PMGM	N/A
• TESP	L: 0.65

Table 13: Scale Factors for Project B after calibration

B_y (Scale Factors)	Range
Precedentedness (PREC)	VH: 0.80
Development Flexibility (FLEX)	VH: 0.50
Architecture/ Risk Resolution (RESL)	VH: 0.71
Team Cohesion (TEAM)	H: 1.22
Process Maturity (PMAT)	H: 2.14

Table 14: Effort Multipliers for Project B after calibration

Cost Drivers	Range
OUTSOURCING FACTORS	
• CULT	VL: 0.42
• BALA	N/A
• TMZN	L: 0.65
BUYER'S OUTSOURCING MATURITY	
• BOXP	VL: 0.62
• BUMP	N/A
• CODS	L: 0.85
PROVIDER'S OUTSOURCING MATURITY	
• POXP	VL: 0.72
• PUMP	N/A
COORDINATION FACTORS	
• OFIT	VL: 0.72
• PMGM	N/A
• TESP	L: 0.95

Table 15: Values for SLIM after calibration

Variables	Project A: after calibration	Project B: after calibration
SSF (special skill factor)	30	10
PRP (Productivity Parameter)	3.2017	4.9903
Duration(Y)	65	78
Size of SLOC	2500	2500

Effort (MY) = 238

Effort of Project B after Calibration with SLIM: Productivity Parameter can be calculated by using the variables of SLIM model after calibration in parameters given in Table 15 and estimated as follows.

PRP = 4.9903

By using PRP and other variable values in SLIM equation after calibration, effort in person month can calculated.

Effort (MY) = 75

Effort Estimation using ISBSG Checker: This ISBSG Checker is a software checker tool that uses Development & Enhancement Industry Information and regression evaluation to easily generate estimates of the effort, required to undertake and total a software package development challenge. It can be used to create initial rough estimates from the early levels of software package development initiatives, to validate existing challenge estimates and to assess the reasonableness along with likely risk associated with a quoted estimate.

The ISBSG collects data on estimates for:

- Project duration
- Design
- Deployment
- Project size

Only a few the projects inside data set have data for all four types of estimates. From the actual evaluations many experts have concluded how the Enhancement tasks are estimated more accurately than completely new developments. Smaller projects will be estimated accurately as well as overestimated. Estimates seem to be less accurate for tasks involving completely new technologies; completely new languages; as well as with big user basics.

While using the ISBSG Checker tool, estimates with regard to project do the job effort and also elapsed time might be generated by entering the actual project dimensions - portrayed in IFPUG perform points. By default, estimates consider every one of the validated facts points within the current release with the ISBSG Data. Nevertheless, a combination of development platform, primary development language sort and improvement type might be selected to get more particular estimates. Here effort is estimated using SCOPE tool.

Effort Estimation Using Scheduling-Based Model: The Scheduling-based model uses data from the metrics database to determine the labor rates for the various development activities. Scheduling is done manually based on the execution time and the number of labors. Here labor rate is defined as PH/SLOC (PH is person hours). Using this definition, the formula for estimating the cost of a project is as follows:

Scheduling-based model can estimate the effort and time required for consummate software projects based on Source Lines of Codes, Scheduling parameter and the Productivity parameter depends on the range of productivity index. Formula for estimating effort in years by using SLIM model is;

$$Effort = \left(\frac{Size of\ SLOC * PRP}{SIG * Duration(Y)\left(\frac{4}{3}\right)} \right)^3$$

where,

Duration(Y) is the time taken for completing project in years

SLOC is the Source Lines of Code

SIG is the scheduling parameter

PRP is the Productivity Parameter

Table 16: Scale Factors for Project A

B_y(Scale Factors)	Range
Precedentedness (PREC)	VH: 1.30
Development Flexibility (FLEX)	VH: 1.15
Architecture/ Risk Resolution (RESL)	VH: 1.90
Team Cohesion (TEAM)	H: 2.34
Process Maturity (PMAT)	H: 3.25

Table 17: Scale Factors for Project B

B_y(Scale Factors)	Range
Precedentedness (PREC)	VH: 1.20
Development Flexibility (FLEX)	VH: 1.05
Architecture/ Risk Resolution (RESL)	VH: 1.80
Team Cohesion (TEAM)	H: 2.14
Process Maturity (PMAT)	H: 3.05

Table 18: Values for scheduling-based model

Variables	Sample Value
SIG (Scheduling Parameter)	5
PRP(Productivity Parameter)	59
Duration(Hour)	8
Size of SLOC	2000

Table 16 and 17 shows the scale factors for Project A and Project B. Table 18 shows the values for scheduling-based model.

RESULTS

The proposed effort estimation method is implemented in MATLAB R2013a and it is estimated using calibrated COCOMO II, SLIM and scheduling-based models. The existing effort estimation methods lack to provide accuracy for GSD projects. Proposed work make calibration on both models based on GSD requirements to get better accuracy for GSD projects. Deviation of effort is the change in actual effort and estimated effort that gives the accuracy and impact of estimation methods on GSD projects. The performance of the proposed method is analysed by using the statistical measures and comparison is made between calibrated estimation methods and estimation methods without calibration, which shows our proposed method gives optimal estimation on GSD projects.

Performance Evaluation: The performance of the proposed system is related to the accuracy of the estimation evaluated by measuring deviation from actual and estimated effort. Deviation is the difference between estimated effort and actual effort, the estimation is more accurate when there is no deviation among them and if the estimated effort is closer to the actual effort then it will also consider as accurate.

The deviation can be calculated using the following equation.

$$Deviation = actual\ effort - Estimated\ Effort \quad (6)$$

Project-A

Table 19: Deviation of values between actual effort and estimated effort of Project-A

Effort Estimation	Project A	Deviation
Estimated Effort with SLIM after Calibration	238	0.83
Estimated Effort with COCOMO II after Calibration	1461	0
Estimated Effort using SCOPE tool	1461	0
Estimated Effort using scheduling- based model	1461	0

Project-B

Table 20: Deviation of values between actual effort and estimated effort of Project-B

Effort Estimation	Project B	Deviation
Estimated Effort with SLIM after Calibration	75	1.04
Estimated Effort with COCOMO II after Calibration	74	0.07
Estimated Effort using SCOPE tool	71.70	0
Estimated Effort using scheduling- based model	71.75	0

Figure 2 and 3 shows the graph for representing deviation of both project A and B before and after calibration in parametric models COCOMO II and SLIM. The graphs displays that the effort estimation of both project A and B after calibration gives better performance than the estimation before calibration in GSD projects.

Comparative Analysis: The performance of the proposed system is evaluated by comparing the effort estimated using proposed method and effort estimated without calibration with actual effort of finished projects.

Fig. 4 and Fig. 5 are the graphical representation of comparison of effort before and after calibration of COCOMO II for Project A and B, which shows that the effort estimated using proposed method is nearest to actual effort. Fig. 6 and Fig. 7 shows the Comparison of effort before and after calibration of SLIM for Project A and B, that represents the effort estimated by calibrated SLIM model is closest to the actual effort showing less deviation. Fig. 9 and Fig. 10 show the comparison of effort and deviation values calculated using three methods. Hence the comparison of efforts shows that the proposed method provides better accuracy of effort estimation of GSD projects.

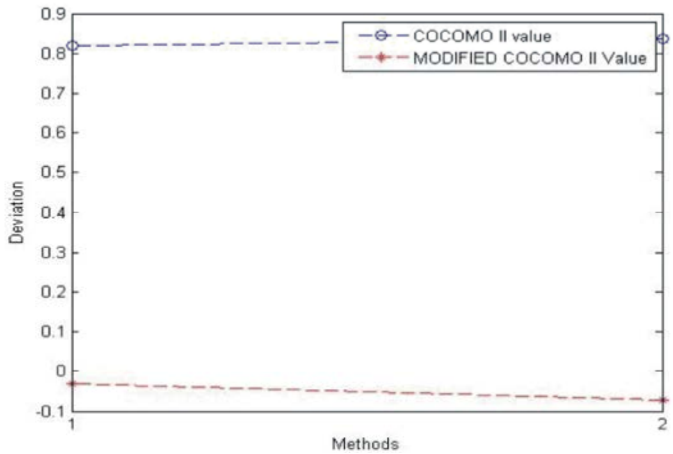

Fig. 2: Deviation of COCOMO II before and after calibration for Project A and B

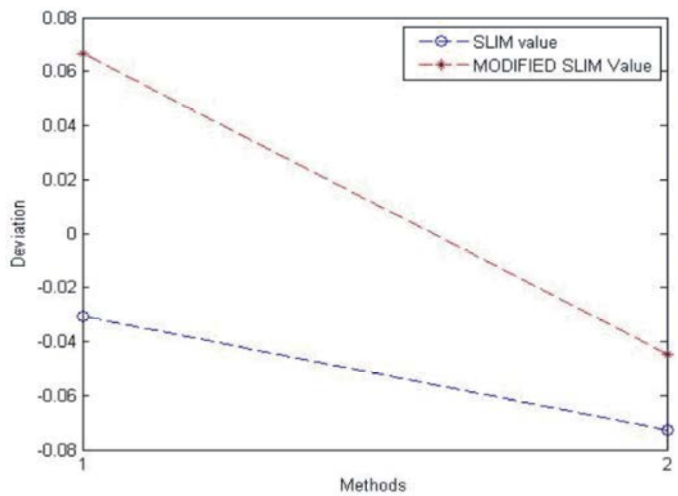

Fig. 3: Deviation of SLIM before and after calibration for Project A and B

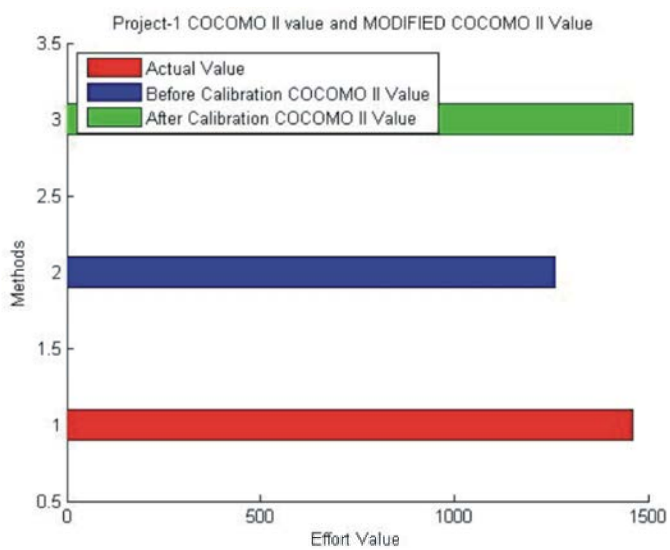

Fig. 4: Comparison of effort before and after calibration of COCOMO II for Project A

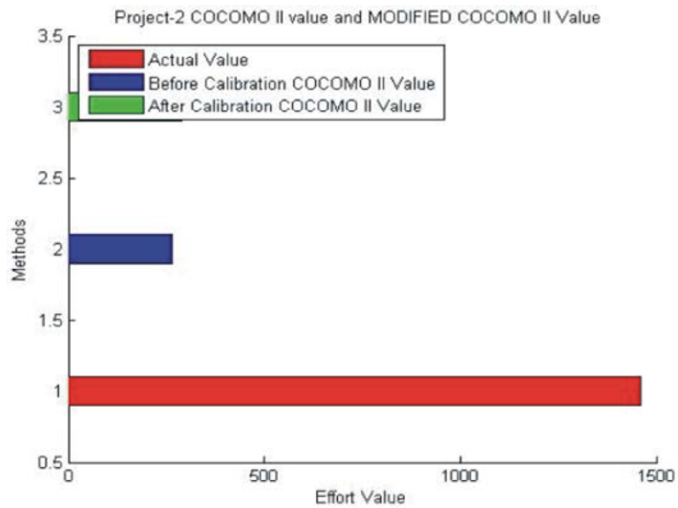

Fig. 5: Comparison of effort before and after calibration of COCOMO II for Project B

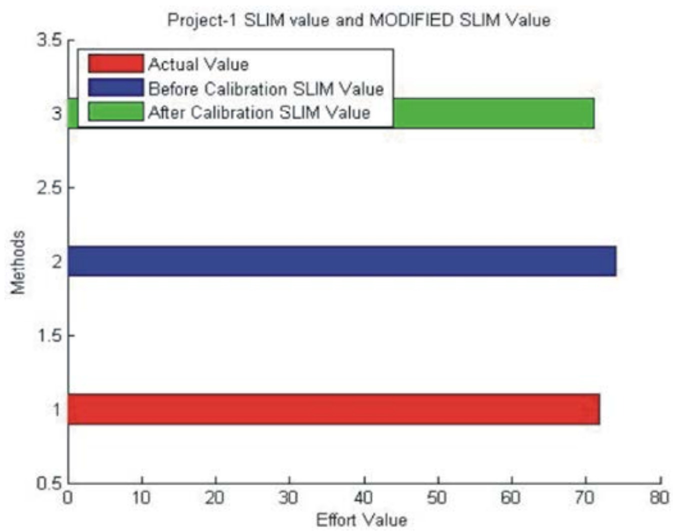

Fig. 6: Comparison of effort before and after calibration of SLIM for Project A

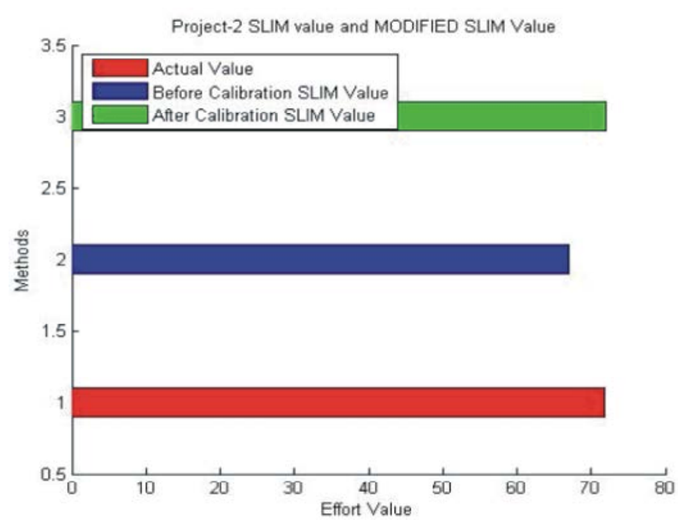

Fig. 7: Comparison of effort before and after calibration of SLIM for Project B

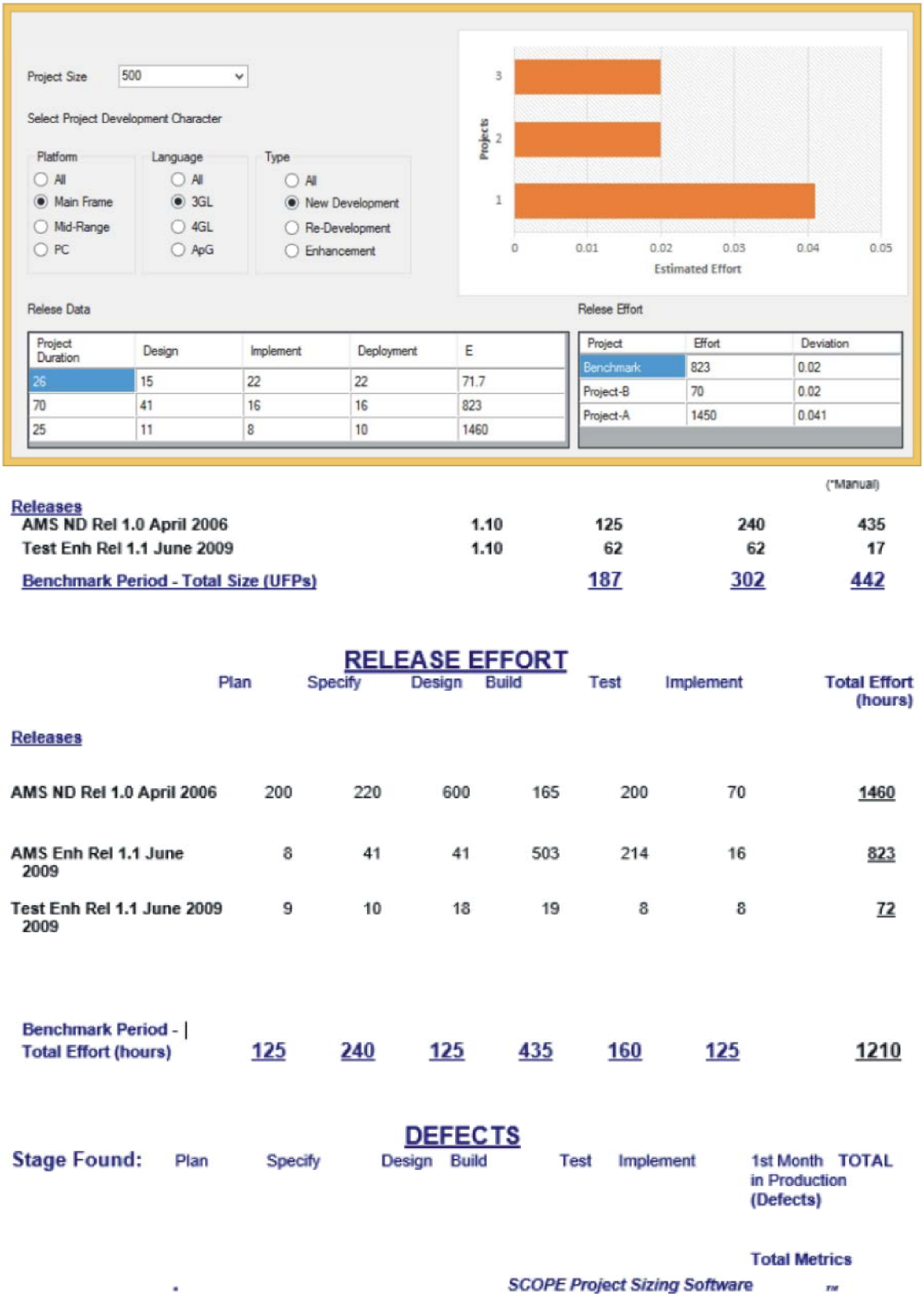

Fig. 8: Scope of the project

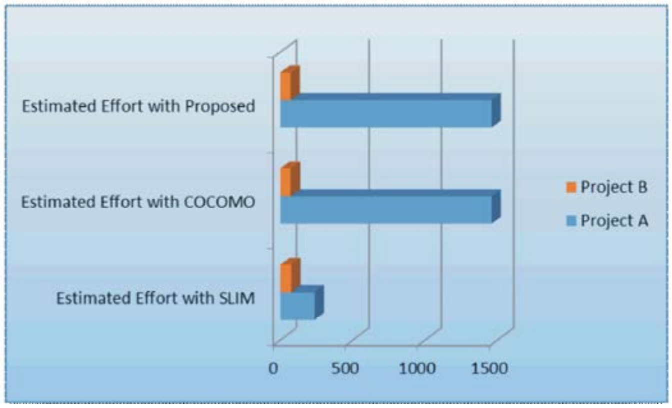

Fig. 9: Comparison of effort values for project A and Project B with proposed method

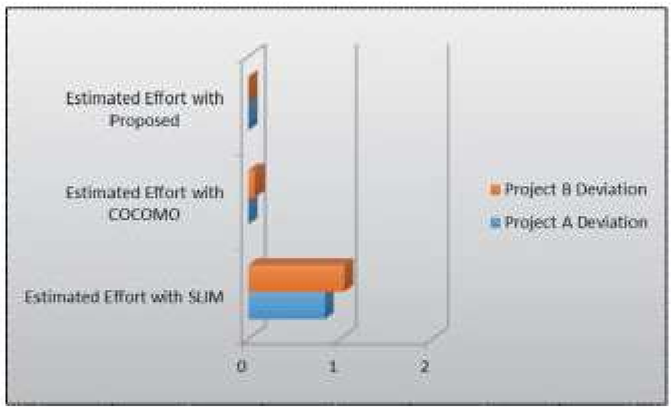

Fig. 10: Comparison of deviation values for project A and project B with proposed method

CONCLUSION

The Proposed Effort Estimation method utilizes calibration in COCOMO II, SLIM and scheduling-based models to attain accuracy while estimating effort in GSD projects. The proposed approach was implemented and its performance was evaluated using MATLAB R2013a. Calibrated values influenced in COCOMO II and SLIM methods provide same accuracy for GSD and Co-located projects. SLIM, COCOMO II and Scheduling-based model furnishes accurate estimation for GSD projects. The result of this scheduling-based model provides better accuracy for GSD projects when compared with accuracy provided by COCOMO II and SLIM models after calibration.

REFERENCES

1. Shah Y.H., M. Raza and S. Ul Haq, 2012. Communication Issues in GSD, International Journal of Advanced Science and Technology (IJAST), pp: 40.
2. Osho L.O., S. Misra and O. Osho, 2013. A Metric in Global Software Development Environment, The Pacific Journal of Science and Technology (TPJST), pp: 14.
3. Humayun, M. and C. Gang, 2013. An Empirical Study on Improving Shared Understanding of Requirements in GSD, International Journal of Software Engineering and Its Applications (IJSEA), pp: 7(1).
4. Tailor, O., J. Saini and P. Rijwani, 2014. Comparative Analysis of Software Cost and Effort Estimation methods: A Review, International Journal of Computer Science and Mobile Computing (IJCSMC), 3(4): 1364-1374.
5. Srivastava, B. and M. Wadhwa, 2013. Relative Analysis of Software Cost and Effort Estimation Techniques, International Journal of Computer Science and Engineering (IJCSE), 2(3): 53-68.
6. Vasantrao, K.V., 2012. Understanding of Software effort Estimation at the early Software Development of the life cycle, A literature View, International Journal of Engineering Research and Applications (IJERA), 2(1): 848-852.

7. Sinhal, A. and B. Verma, 2013. Software Development Effort Estimation: A Review, International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), 3(6).
8. Sharma, T.N., 2011. Analysis of Software Cost Estimation using COCOMO II, International Journal of Scientific and Engineering Research (IJSER), 2(6).
9. Kad, S. and V. Chopra, 2012. Software Development Effort Estimation Using Soft Computing, International Journal of Machine Learning and Computing (IJMLC), 2(5).
10. Elyassami, S. and A. Idri, 2012. Investigating Effort Prediction of Software Projects On the ISBSG Dataset, International Journal of Artificial Intelligence and Applications (IJAIA), 3(2).
11. Jorgensen, M. and K. Molokken-Ostvold, 2004. Reasons for Effort Estimation Error: Impact of Respondent Role, Information Collection Approach and Data Analysis Method, IEEE Transactions on Software Engineering, 30(12).
12. Kumar, S.A. and T.A. Kumar, 2011. Study the Impact of Requirements Management Characteristics in Global Software Development Projects: An Ontology Based Approach, International Journal of Software Engineering and Applications (IJSEA), 2(4).
13. Pandey, P., 2013. Analysis of Techniques for Software Cost Estimation, International Journal of Software Engineering Research and Practices (IJSERP), 3(1).
14. Basha, S. and P. Dhavachelvan, 2010. Analysis of Empirical Software Effort Estimation Modals, International Journal of Computer Science and Information Securitty, 7(3).
15. Choudhary, K., 2011. Parametric Estimation of Software Systems, International Journal of Soft Computing and Engineering (IJSCE), 1(2).
16. Alghdr, Y.A., M.A. Alam and M. Ahmed, 2012. On the Estimation of the Software Process Maturity Using COCOMO II's Effort Estimation based on CMMI, International Journal of Modern Engineering Research (IJMER), 2(4): 2377-2383.
17. Kaur, M., 2011. To Design a Simulator for Performance Comparison of Cost Models Journal of Global Research in Computer Science (JGRCS), 2(7).
18. Khatibi, V. and D.N.A. Jawawi, 2010-2011. Software Cost Estimation Methods: A Review, Journal of Emerging Trends in Computing and Information Sciences (JETCIS), 2(1).
19. Du, W.L., D. Ho and L.F. Capretz, 2010. Improving Software Effort Estimation Using Neuro-Fuzzy Model with SEER-SEM, Global Journal of Computer Science and Technology (GJCST), 10(12).
20. Sawalhi, N.I.E., 2012. Modeling the Parametric Construction Project Cost Estimate using Fuzzy Logic, International Journal of Emerging Technology and Advanced Engineering (IJETAE), 2(4).
21. Mewada, K.M., A. Sinhal and B. Verma, 2013. Adaptive Neuro-Fuzzy Inference System (ANFIS) Based Software Evaluation, International Journal of Computer Science Issues (IJCSI), 10(5): 1.
22. Kashyap, D., A. Tripathi and A.K. Misra, 2012. Software Development Effort and Cost Estimation: Neuro-Fuzzy Model, IOSR Journal of Computer Engineering (IOSRJCE), 2(4): 12-14.
23. Humayun, M. and C. Gang, 2012. Estimating Effort in Global Software Development Projects Using Machine Learning Techniques, International Journal of Information and Education Technology, 2(3).
24. Joseph, K.S. and T. Ravichandran, 2012. Genetic approach for Software Effort Estimation: A Variant for COCOMO, International Journal of Computer Applications and Information Technology (IJCAIT), 1(3).
25. Aljahdali, S. and A. Sheta, 2013. Evolving Software Effort Estimation Models Using Multigene Symbolic Regression Genetic Programming, International Journal of Advanced Research in Artificial Intelligence (IJARAI), 2(12).