METHODICAL SURVEY: SOFTWARE COST ESTIMATION

Santosh Panda*

Sarada Baboo*

ABSTRACT

A key step for instrumenting the method of choice of selecting cost estimation method from the literature and in practice inside the organization from a multiplicity of estimation techniques is a critical and crucial task. As cost of development reflects the quality of software of any organization a proficient way of selecting an estimation technique is highly required. Uncertainty and sensitivity are the main criteria of this review. The current attempt for reviewing is to identify the vastly used cost estimation method satisfying the review questions to find out the uncertainty and most sensible factors that influence and effect most during estimation to reduce the variability between estimated and actual cost of software development because the expected difference in cost has a major effect on the global feasibility of a project. The main objective of this review is to extend the COCOMO model for future economic influence by using simple, open, effective and transparent calibration approach which helps for good decision making for software practitioners for future software development models that is wanting for last 4 decades.

Keywords-Estimation methods, COCOMO, Metrics, FAST, FSS, Reduced parametric model, Machine Learning Technique, Sensitivity analysis, e-FAST and Sobol.

* P.G. Department of Computer Science and Applications, Sambalpur University, Odisha, INDIA.

1. Introduction

1.1 Overview

Software is a product and developed using engineering principles. Cost, quality and durability are the main factors for influencing the demand of a software product. The software product is developed within specific duration or time satisfying the quality with minimum development cost. Software cost depends on the effort and schedule of project. The effort is measured and directly depends on the size as one of the metrics of software. Other product metrics being the complexity, design features and performance. The project cost is a measure of the problem complexity in terms of effort and time required to develop the product. The size metric can be in the form of kilo lines of code (KLOC), function point (FP), use cases or story point. Cost is estimated early in the software development. Cost estimation is one part of project planning which is the responsibility of project manager. The manager's decision making is very critical and crucial part for estimating the accurate cost of the software which reflects the quality because the cost is not only depending on the project size but also on various cost drivers or effort adjustment factors basing on different cost estimation techniques. Software development is an evolutional process having a lot of process models and cost estimating techniques.

1.2 Defining or identifying issues in the area of Cost estimation

The area of concern of this literature review on the general topic is Software Cost Estimation as there is a lot of views and models developed and under research for nearly last 4 decades. Cost estimation is a part of Project management where manager takes a crucial role to estimate the cost and effort prior to the development. A better cost estimation provides better quality. Quality in the sense delivering of the product on demand time of the customer with no defects, risks and low cost of development. However there is no particular and exact estimation technique since the last 40years of software development to till date because of the uncertainty and variability of different scaling or adjustment factor or cost drivers require that are to be incorporated for the estimation of cost of software.

The software or hardware architecture is also evolving; in 1980s (during Basic COCOMO model) the development was on IBM Mainframe using Ada or COBOL. Now it is

evolving from enterprise applications to cloud computing having a lot of services and hardware platforms or resources.

Software development consists of a number of development processes or phases which are dynamic and evolutionary in nature (from waterfall model to agile process model). Another issue is the prediction using past Dataset and the estimation process is done by the managerial decisions or the expert judgment using their past experiences which is black box and vary from organization to organization. Hence cost estimation is sometimes an art than science.

1.3 Applications, Requirements, Key Issues and Challenges

Cost estimation technique is vastly used since last 4 decades in various fields of science and engineering. Following are some of the principle areas of applications:

a. Civil Engineering
b. Department of Defense(DoD) and Military
c. National Aeronautic and Space Agencies (NASA)
d. Software and Hardware Corporates.
    i. End User Programming.
    ii. Infrastructure Sector.
    iii. Intermediate Sector.
    iv. Application Generators.
    v. Application composite Sector.

Cost estimation is a high requirement in any small scale and large scale projects because it estimates the budgetary information and a minute variation in estimation will cost a billion dollar difference for any organization. Cost depends on the effort, duration or schedule, staffing etc. A good estimation leads to a good quality project with proper schedule and staffing and deliver the products as per the customers requirement on time with no defects.

Cost estimation is an essential and critical part of software development and need to be estimated accurately for better quality and to avoid the risk of failure or defects. An efficient and

accurate estimation technique and the most dependent scaling factors (cost drivers or effort multipliers) shall be required and tuned to meet the lacunae in estimation. A new calibrated model must be required which identify the highly sensitive parameters that only be calibrated to get an accurate estimation model.

The key issues and challenging factor for cost estimation is the accuracy of estimation. There are a dozens of estimation methods being developed and is under development as the advance of technology. It is difficult to apply the cost model because of high effort in data collection, calibration model and also to learn software cost models.

Now it is a challenging issue for the Organizations to find more and more accuracy with the flexibility of identifying the key affective and effective input parameters and the metrics on which the cost estimation technique depends.

1.4 Factors effecting the cost estimation

There are various functional and non-functional parameters or attributes which effect the cost estimation that has to be identified earlier than the software development. The key input factors influence the output, effort and the cost of the estimation.

1.5 Traditional and Modern Models or Methodologies of the Topic

Following are the 6 major cost estimation models and different techniques [15].

1.5.1 Model Based –Putnam's Software life-cycle model (SLIM), COCOMO, Checkpoint, System evaluation and estimation of resources (SEER-SEM), PRICE-S, Estimacs, and Softcost.

1.5.2 Expertise Based -Delphi, Rule-Based and WBS

1.5.3 Learning Oriented – ANN, Fuzzy logic, Case-based Reasoning (CBR) and Genetic Algorithm (GA)

1.5.4 Regression Based -OLS, Robust

1.5.5 Dynamics Based - Abdel Hamid-Madnick

1.5.6 Composite Bayesian -COCOMO-II, Reliability Growth Models (RGM)

COCOMO81 (Boehm, 1981) is the most traditional model based parametric model and COCOMO II (Boehm et al, 2000) and its different inherent models like (Bayesian COCOMO-II, COCOTS-2000, COSYSMO-2002, CORADMO-SE-2012) is the existing and vastly used models in different Software Organizations to till date.

Learning oriented models are most resent techniques which provide more accuracy and still under research, but not implemented in any organizations because of its non-transparency and complexity.

1.6. The Constructive Cost Model COCOMO-II

1.6.1. The revised COCOMO-II model for effort calculation derived from USC Center for System and Software Engineering Research Group, COCOMO-II.1997 Reference manual is given below:

$$\text{Effort} = \prod_{i=1}^{17}(EMi)*\text{A}*\left[\left(1+\frac{BREAK}{100}\right)*Size\right]^{\text{B}}+\left[\frac{ASLOC*\frac{AT}{100}}{ATPROD}\right]\text{PM}$$

$$\text{Size} = \text{KNSLOC}+\left[KASLOC\left(\frac{100-AT}{100}\right)\frac{(AA+SU+0.4*DM+0.3*CM+0.3*IM)}{100}\right]\text{KLOC}$$

$$\text{B} = \left(1.01+0.01\sum_{j=1}^{5}\text{SFj}\right)$$

Symbol	Description	Symbol	Description
A	Constant set to2.5	EM	17 Effort multipliers
AA	Assessment and assimilation	IM	Per. Integrating and test modified
AT	Per. Of components Automatically translated	KASLOC	Size of adapted components
ATPROD	Auto. Tran. productivity	KNSLOC	Size of components in new SLOC
BREAK	Per. of code thrown away due to requirement volatility	PM	Person month of effort estimation
CM	Per. Code modified	SF	5 Scale factors
DM	Per. Design modified	SU	Software understanding

1.6.2. 17 Multiplying and 5 Scale Factors of COCOMO-II:

17 Effort Multipliers (EM)

Product attributes

RELY (requirement s/w reliability)

DATA (database size)

DOCU (documentation LC)

CPLX (product complexity)

RUSE (development for reusability)

Platform attributes

TIME (execution time constraint)

STOR (storage constraints)

PVOL (platform volatility)

Personal attributes

ACAP (analysis capability)

AEXP (application experience)

5 Scale Factors (SF)

PREC (precedence)

FLEX (flexibility)

RESL (risk resolution)

TEAM (team cohesion)

PMAT (process maturity)

PCAP (programmer capability)

PEXP (platform experience)

LEXP (language experience)

PCON (personal continuity)

Project attributes

TOOL (use of software tools)

SITE (multisite development)

SCED (required development schedule)

2. Review Process

2.1 Review Objective

The main purpose or objective of this review is to collect and review papers on software cost estimation with respect to estimation topics, estimation approach, research approach, data set to calibrate the existing model, that is vastly used in the software organization which should be transparent and easy to implement for the software practitioners and finally to support further estimation research.

2.2 Considerable source of research journals and books for Review

The considerable source of research journals selected for reviewing about the topic software cost estimation are **IEEE**, **ACM**, **Elsevier**, **IEEE Transaction**, **IJIEE**, **IJCEA**, **USC Research Lab**, **IEEE Xplore**.

Considerable books refereed are:

1. **Software Engineering Economics** by B.W. Boehm, Prentice Hall.
2. **Software Metrics-** A Guide to Planning, Analysis, and Application by C. Ravindranath Pandian, Auerbach Publication.
3. **Software Metrics-** A Rigorous and Practical Approach by Fenton and Pfleeger.

4. **Practical software estimation**-Function point method by M.A.Parthasarathy and N.R.Narayana Murthy, Infosys Press and Addison Wesley.

5. **Fundamentals of Software Engineering** by Rajib Mall.

6. **Software Engineering-** A practitioner's Approach by R.S.Pressman

7. **Sensitivity Analysis in Practice-** A Guide to model Scientific Models by A.Saltelli, Wiley Pub.

2.3 Collection of Review Papers

We have gathered the review papers basing on the criteria i.e. estimation method, calibration of models, size measures, uncertainty, sensitivity, measures of estimation performance, data set used and types of estimation approach.

2.4 Review Questions

While reviewing the journal papers, articles and books on software cost estimation which are related to identify the key software size metrics, identification of the complexity factors that influence and correlates more on the cost model to addresses the following review questions (RQ).

RQ1. What is the widely used, most explored, most repeatedly and most utilizable estimation method?

RQ2.Which software estimation technique has a long term interest?

RQ3. Has parametric methods suppressed by any other methods?

RQ4. Which parametric calibration model and input factors are more applicable for an effective estimation model?

Table 1: Review Questions

Review Questions	Main Motivation
RQ1	To identify the inclinations, possible limitation or opportunities for the focused estimation method.
RQ2	To identify the weakness or openness of software cost estimation
RQ3	To identify the most prominent estimation methods i.e. either parametric or non-parametric.
RQ4	To focus on hypothetically significant journal papers for identifying the calibration model and input parameters that effect more on effort and cost.

2.5 Overall trends that has been published about the topics

Cost is mostly estimated by effort which is directly proportional to size (DLOC/KLOC), function point (FP), object point as the main metric or dominant factor for prediction and some other factors of concern that can be calibrated. Most the software estimation techniques are basing on Model-Based (COCOMO, SLIM, Putnam's model), Expertise-Based (Delphi, WBS), Regression-Based (OLS, Robust), Composite-Based (Bayesian), Learning-Oriented (ANN, Fuzzy Logic, Case Based Reasoning, Machine Learning and GA).

The most resent estimation trend is basing on Learning-Oriented, but still the software organizations familiarize to use the COCOMO model basing on the trained past dataset and experience for the tuning of different factors or cost drivers. Moreover the COCOMO model is evolved from basic to intermediate through CORADMO and many more under USC-CSE research group along with the evolutionary development of software system and is given in the fig. 1.

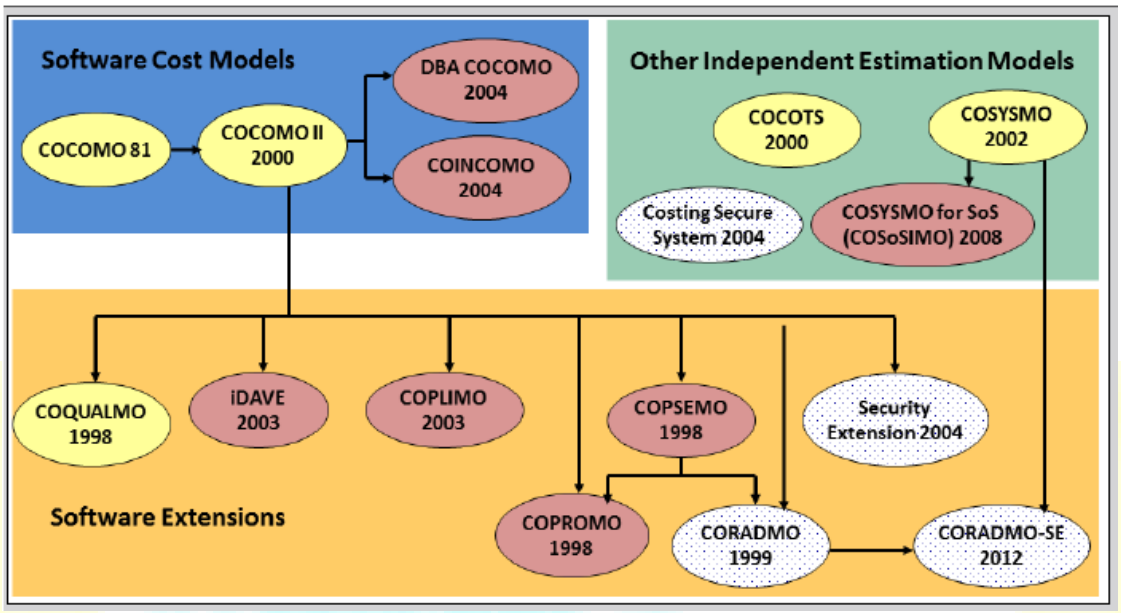

Figure 1. Evolution and extension of COCOMO Model

(Source: USC-center for systems and software engineering, 28th international forum)

2.6 Annexation and Exclusion Principles of Review Papers

The main criteria for including a journal paper in our review are that the paper describes research on software development effort or cost estimation. The paper related to estimation of software size, assessment of software complexity factors and the scale factors that correlate or sensitive more, which is the main purpose of the review to improve the software effort or cost estimation.

The papers that are out of the scope of the requirement of review are excluded.

2.7 Collection of Data set

Data set are collected from PROMISE repository software engineering Databases and are normalized for our application.

2.8 Analysis of the Review Papers

Most of the author's research area of interest is to estimate an accurate estimation technique that can reduce the variability between estimated and actual cost.

3. Literature Review

3.1 **Reviews on the concerned topic, area of interest, methodology, conclusion and limitations.**

Following are some of the reviews on the concerned topic.

R1. Wanger. S et al., 2007 proposed an approach to global sensitivity analysis: **FAST** on **COCOMO**. This paper describes an approach based on global sensitivity analysis to answer which parameter or factor need to be estimated best and the other removed from the model and shows its applicability in a case study on the COCOMO Model with NASA dataset. Earlier the work was done on COCOMO (Boehm, 1981; P.Musilek et al, 2002; Z.Chen et al, 2005) based on local sensitivity analysis which is not able to define the characteristics of GSA. The studies discover that the factor size is most important input parameter but the scale factors are not recognized as most important factors using Fourier Amplitude Sensitivity Analysis (FAST) [4][5].

Limitations: Cannot identify and extracts unusual scale factors to simplify the COCOMO Model. Computationally complex for large set of inputs.

R2. Z. Chen, B.Boehm, 2005 proposed **Feature subset selection (FSS)** based WRAPPER model which can improve software cost estimation accuracy by iterating the subset of input cost drivers. This paper describes that FSS significantly improves the COCOMO model [7].

Limitations: More stratification of dataset requires understanding the effects of FSS confirming the reduced parametric model. Repeated experiments can be done using Genetic Algorithms (GA).

R3. Zhihao Chen, 2006 proposed REDUCED-PARAMETER MODELING FOR COST ESTIMATION MODELS in his dissertation work using reduced parameter model with machine learning technique. In this paper FSS method was used for evaluating the attribute sets using

learning patterns and statistical re-sampling such as cross validation for a set of attributes for accurate estimation using machine learning and Bayesian calibration approach. It is a composite model by aggregating the model based, expert judgment and learning oriented approach. In (S.Chulani, 1999) proposed an approach on Bayesian analysis of empirical software cost estimation technique validating the accuracy of cost estimation [15].

Limitations: Integration of different linear and nonlinear methods made it more complex and the learning Bayesian approach is non-transparent.

R4. Mohd.Sadiq et al, 2013 proposed a linear regression model to predict the software project effort. The estimation error was measured using MRE AND MMRE. Vu Nguyen, B. Boehm proposed a constrained regression technique on COCOMO calibration that used an objective function and a set of constraints. Various approaches like OLS, stepwise, Lasso and Ridge regression are used and validated using cross validation procedure to compute MMRE, PRED and S.D. It is the easier and simpler approach comparative to complex Bayesian approach [8][11].

Limitations: Enterprise can establish their own linear model using their records for accurate cost estimation.

R5. B.A. Kitchenham et al, 1985 proposed a comparison on COCOMO and Putnam model**.** This paper described the Empirical model comparing **Boehm's COCOMO** and **Putnam's Rayleigh** model. It was used to develop a cost model tailored to a particular environment and to improve the precision of the models as they are used during the development cycle by including additional information such as the known effort for the early development phases [9].

Limitations: Only applicable to a particular environment to establish the nature of the effort/size and effort/schedule relationship.

R6. Magne Jorgensen et al, 2007proposed a systematic review of software development cost estimation studies to provide a basis for the improvement of software estimation research through a systematic review of previous papers [2][3].

R7. Capers Jones, 2012 proposed Software Quality Metrics- Three Harmful and Two Helpful Metrics to find out **Helpful and Harmful metrics**. It is concluded that the combination of function point with defect removal efficiency metrics (DRE) can show the true cost of quality and the most cost-effective way to build the software [17].

3.2 Comparing Literature and Organization of Reviews

3.2.1 Comparison Study Matrix

Table 2.Evaluation of the Literature using Comparative Study Matrix.

Sl. No.	Source[Reference]	Year	Author	Methodology	Conclusion	Future Scope
1	Elsevier [9]	1985	B.A. Kichenham et al	Empirical model	Cost models tailored to a particular environment and to improve the precision of the models	To produce a tailored cost model would be to establish the nature of the effort/size and effort/schedule relationship
2	ACM [7]	2005	Z. Chen, B.Boehm	WRAPPER on COCOMO model	Significantly improves the COCOMO model	More stratification can be done to better understand
3	ACM [15]	2006	Zhihao Chen et al	Reduced parametric modeling using Bayesian calibration	Removing cost driver variables and parameters	Extended reduced parameter models are used to get more accuracy

4	IEEE [4][5]	2007	Wanger. S et al	Global sensitivity analysis: FAST on COCOMO	Factor size to be most important but the scale factors are not always identified as important using global sensitivity analysis	To find out more appropriate scale factors that influences the cost for optimization.
5	IEEE Transactions on Software Engineering[2]	2007	Magne Jorgensen et al	Review	Review of software development cost estimation studies	Improvement of software estimation research through a systematic review of previous papers
6	http://www.ifpug.org/Documents/Jones-SoftwareDefectOriginsAndRemovalMethodsDraft5.pdf	2012	Capers Jones	Quality Metrics	Software industry has lacked solid economic understanding of basic topics such as cost of quality and defect removal cost	Combination of function point with defect removal efficiency metrics (DRE) can show the true cost of quality and the most cost-effective way to build the software
7	IJIEE [11]	2013	Mohd. Sadiq et al	Linear regression model	Accurate cost estimation using Linear model	Enterprise can establish their own linear model using their records for accurate cost estimation

3.2.2 Comparison basing on advantages, disadvantages of the Reviewed Methodology.

Table 3. Advantages and disadvantages of review models

Sl. No.	Methodology	Advantages	Disadvantages
1	Empirical model	Improve the precision of the models	Cost models tailored to a particular environment
2	FSS (WRAPPER) on COCOMO model	Significantly improves the COCOMO model	More stratification can be done to better understand
3	Reduced parametric modeling using Bayesian calibration	Removing cost driver variables and parameters	Removing cost drivers may lead to inaccuracy cost
4	Global Sensitivity Analysis: FAST on COCOMO	Both factor size and important scale factors can be identified.	To find out more appropriate scale factors that influences the cost for optimization.
5	Quality Metrics	Identify both quality of software and defect removal cost	Combination of function point with defect removal efficiency metrics (DRE) can show the true cost of quality
6	Linear regression model	Accurate cost estimation using Linear model	Enterprise can establish their own linear model using their records for accurate cost estimation

Table 4: Advantages and Limitations of selected estimation methods [15]

Methods	Advantages	Limitations
Parametric	1 The influence of input factors or Parameters on cost can easily trace out. 2 Repeatable.	1 Could not identify most important input Parameters. 2 Simplistic
Neural Network	1 Accurate estimates possible because of the detail. 2 Can be updated and retrained.	1 Logic not visible 2 Complex 3 Require a large case base to be effective
Expert Judgment	1 Quick to produce. 2 Flexible.	1 Susceptible to bias 2 Unstructured 3 Different experts use different

		Mechanisms
Case-Based Reasoning	1 Can propose solutions rapidly. 2 Plays the role of collective.	1 Need a reliable and large case base. 2 Doesn't handle innovative solutions.

4. Review Summary by Incorporating the Review Questions

Conferring to the review question (RQ1 and RQ2) it is identified that COCOMO Model is vastly used parametric model for cost estimation which is being evolutionary developed for long time [1][8] [13][14][16]. It is also identified from the distribution of estimation methods and type of research approach that, about 49% is regression based estimation method which include the most dominant and common estimation method i.e. COCOMO Model. Nearly 20% of the research topic is about size measurement where 22% popularity is going to Function Point (R7)[17], 7% is for calibration of models and about 46% is for development of estimation methods from the type of research approaches [2].From Table 4 it is distinct that the parametric model advances its popularity because of making clear the influence of parameters on the cost however the limitation, which parameters are to be included, can answer our review (R1, R2, R3 and R4).

While addressing RQ3 the result of previous survey showed that about 59%-79% projects were overestimated effort. The parametric model is somewhat suppressed because most of the projects used expert judgment to make estimation and about 55% is history based evaluation and about 6% is experimental in the type of research approaches [2],[3].

For addressing RQ4, the reviews R1, R2, R3 and R4 it is identified that there must be a global, simpler and transparent calibration model is needed to identify the uncertainty and sensitivity factors that need to tune for accurate estimation of cost.

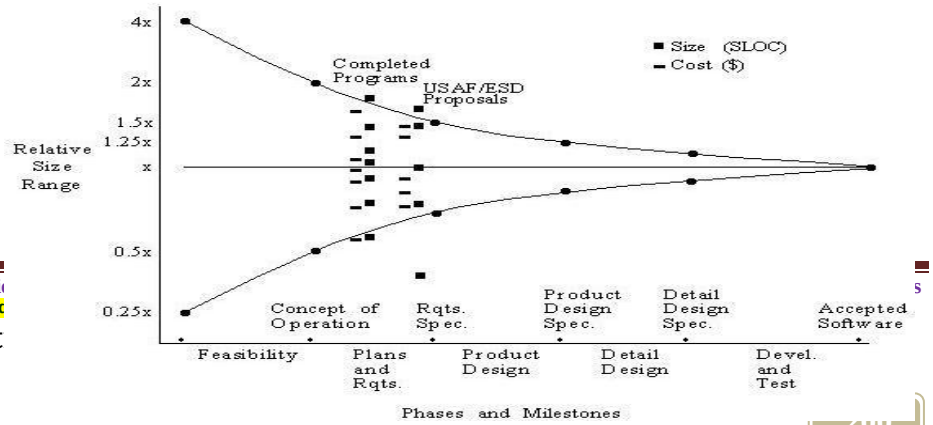

Figure 2. Measures of Uncertainty and estimation of size and cost at different phases of software development

(Source: Software Engineering Economics, B. Boehm)

From Figure 2, It is clear that uncertainty is more at feasibility and requirement phase. The most sensible functional and nonfunctional attributes or parameters are to be found out early in the requirement phase to reduce the propagation of uncertainty, errors and defects such that it will increase the quality of software and the accurate cost of estimation.

Global Sensitivity Analysis (GSA) is a variance based predictor model. It is the more appropriate method to identify the most sensible factors that influence more to minimize the cost [4] [5].

While observing RQ1, RQ2, RQ3 and RQ4 the result we found that COCOMO Model is the vastly applicable parametric regression based model, FP is the most accurate size metrics and GSA can be used to calibrate the needed input parameters easily and effectively..

5. Key Findings, Interpretations and Summary

In the survey of different literatures of the given context while satisfying our review questions it is summarized that function point metric is more appropriate size metric [Capers Jones] and COCOMO model is still a widely used model to till date. Learning-oriented approach like ANN, fuzzy logic, CBR can provide more accuracy but non-transparent and complex, the project managers are more acquainted with the COCOMO model for their expertise and tuning the scale factors with their own dataset from their previous projects.

As per the comparative study from Table 2 and 4 and review summary it is found that Global Sensitivity Analysis model comprises elegant data collection in demand to catch the input parameter factors and to find out which influenced input factors that to be taken for accurate estimation and which will be detached before calibration by calculating the first order and high order indices.

As Extended Fourier Amplitude Sensitivity Test (eFAST) and Sobol are the resent variance based sensitivity analysis technique which has exploration of uncertain parameters , random sampling for efficient parameter sampling and efficient in terms of complexity that can be implemented for calibration on COCOMO-II to find out most sensible input parameter out of 17 multiplying factors and 5 scale factors.

6. Conclusion

From the summary it is concluded that the requirement analysis phase is more uncertain. The main objective of the current review after literature survey is to develop an open or transparent predictive statistical model for identifying the uncertainty and sensitivity of factors that influence more on the estimation of cost of software prior to development that has to be tuned and calibrated for COCOMO-II with collected standardized data set from the repository using the methodology Extended FAST (eFAST) and Sobol as the most recent and widely used methodology for sensitivity analysis which will help to estimate the cost more effectively and accurately.

References

[1] Boehm B, (1981). Software Engineering Economics, Prentice Hall.

[2] Magne Jorgensen and Martin Shepperd, (January, 2007), A Systematic Review of Software Development Cost Estimation Studies, IEEE Trans. on Software Engineering Vol. 33. No. 1.

[3] Yang Da et al., (Oct 9-10, 2008), A Survey on Software Cost Estimation in the Chinese Software Industry, ESEM' 08.

[4] Wagner S., (Sept, 2007), "An Approach to Global Sensitivity Analysis: FAST on COCOMO", IEEE, First International Symposium on Empirical Software Engineering and Measurement, 20-21.

[5] Wagner S, (May 2007), Global Sensitivity Analysis of Predictor Models in Software Engineering, IEEE, International Workshop on Predictor Models in Software Engineering, PROMISE'07 20-26.

[6] Caine Allan and Pidduck Banks Anne, (May 2004), "f2 COCOMO: Estimating Software Project Effort and Cost", *Proceedings of the 26th International Conference on Software Engineering (ICSE)*, The Sixth International Workshop on Economics-Driven Software Engineering Research (EDSER-6), pp. 62-65.

[7] Chen Zhihao et al., (July, 2005), Feature Subset Selection Can Improve Software Cost Estimation Accuracy, ACM SIGSOFT Software Engineering, Proceeding workshop on PROMISE'05, Vol. 30, Issue 4.

[8] Nguyen Vu, Boehm B. et al, "A Constrained Regression Technique for COCOMO Calibration", Proceedings of second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement-ESEM'08.

[9] Kitchenham B.A. and Taylor N.R., Nov. 1985, Software Project Development Cost Estimation, ACM, Journal of System and Software, Vol. 5, Issue 4.

[10] Chulani Sunita, Boehm et al, Calibrating Software Cost Models Using Bayesian Analysis, USC-CSE 98.
http://www.sunset.usc.edu/Research_Group/Sunita/down/tse.pdf for more details.

[11] Sadiq Mohd., May 2013, Prediction of Software Project Effort Using Linear Regression Model, International Journal of Information and Electronics Engineering, Vol. 3, No. 3.

[12] Rashid Ekbal, Patnaik Srikanta et al., Nov. 2012, Software Quality Estimation using Machine Learning: Case-based Reasoning Technique, International Journal of Computer Applications, Vol. 58, No. 14.

[13] Akansha et al., June 2014, A Study on effort estimation models for the software project management, International Journal of Computer Engineering and Applications, Vol. VI, Issue III.

[14] Boehm B. et al., (April 2005), COCOMO Suite Methodology and Evolution, USC-CSE-2005-509, CrossTalk, The Journals of Defense Software Engineering, pp. 20-25.

[15] Chen Zhihao, Reduced-parameter modeling for cost estimation models. http://csse.usc.edu/csse/TECHRPTS/PhD_Dissertations/files/ChenZhihao_Dissertation.pdf for more details.

[16] Boehm B. and Valerdi R., 2008, Achievements and Challenges in COCOMO Based Software Resource Estimation, IEEE.USC-CSSE-2008-831. http://csse.usc.edu/csse/TECHRPTS/2008/usc-csse-2008-831/usc-csse-2008-831.pdf for more details.

[17] Johns Capers, 2012. "Software Quality Matrics: Three Harmful Matrics and Two Helpful Matrics". http://www.ifpug.org/Documents/Jones-SoftwareDefectOriginsAndRemovalMethodsDraft5.pdf for more details.