ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Improved Software Development Effort Estimation Based on Fuzzy Logic Functions

Rshma Chawla*, Deepak Ahlawat, Mukesh Kumar
*Assistant Professor at MMICTBM, MMU Mullana, India
MPhil Research Scholar at MMICTBM, MMU Mullana, India
Administrator cum HOD, PDM Group of Institutions, Karsindhu (Safidon) Jind, India

Abstract

The systems and software development industry is characterized by a paradigm of project failure. One of the known contributing causes of these project failures is poor requirements engineering and management, which has been repeatedly and widely discussed and documented. But there are other factors also like poor software project management practices, poor design strategy and inefficient testing principles also contributing to project failures.A fuzzy model is more pertinent when the systems are inadequate for analysis by conventional means or when the available data is uncertain, inaccurate or vague.

In this paper, software development effort estimation using Fuzzy Triangular Membership Function, GBell Membership Function, Gauss2 Membership Function and Trapezoidal Membership Function is implemented using Mamdani Type Fuzzy inference system of Fuzzy Logic Toolbox Software of Matlab R2013a and the results of these membership functions are compared with each other and with COCOMO model. It is found that the Fuzzy Logic Model using Gaussian2 Membership Functionprovided best results.

Keywords: Software Engineering, Software Effort Estimation, Fuzzy Logic, COCOMO Model

Introduction

Software development effort estimation

Software metric and especially software estimation is baseon measuring of software attributes which are typicallyrelated to the product, the process and the resources ofsoftware development [1]. This kind of measuring can beused as parameters in project management models [2] whichProvide assessments to software project managers inmanaging software projects to avoid problems such as costoverrun and behind the schedule. One of the most widelyresearched areas of software measurement is software effortestimation. Software effort estimation models divided intotwo main categories: algorithmic and non-algorithmic.The most popular algorithmic estimation modelsinclude Boehm's COCOMO [3], Putnam's SLIM [4] andAlbrecht's Function Point [5]. These models require asinputs, accurate estimate of certain attributes such as lineof code (LOC), complexity and so on which are difficult toobtain during the early stage of a software developmentproject. The models also have difficulty in modelling theinherent complex relationships between the contributingfactors, are unable to handle categorical data as well as lackof reasoning capabilities [6]. The limitations of algorithmicmodels led to the exploration of the non-algorithmictechniques which are soft computing based.These include artificial neural network, evolutionarycomputation, fuzzy logic models, case-based reasoning,combinational models and so on. Artificial neural networkare used in effort estimation due to its ability to learn fromprevious data [7][8]. It is also able to model complexrelationships between the dependent (effort) and independentvariables (cost drivers) [7][8]. In addition, it has the abilityto generalise from the training data set thus enabling it toproduce acceptable result for previously unseen data. Mostof the work in the application of neural network to effortestimation made use of feed-forward multi-layer Perception,Backpropagation algorithm and sigmoid function [7].Selecting good models for software estimation is very criticalfor software engineering. In the recent years many softwareestimation models have been developed [4, 5, 6, 7 , 8, 9].Gray and MacDonell compared Function Point Analysis,Regression techniques, feed-forward neural network andfuzzy logic in software effort estimation. Their resultsshowed that fuzzy logic model achieved good performance,being outperformed in terms of accuracy only by neuralnetwork model with considerably more input variables. Alsothey developed FULSOME (Fuzzy Logic for SoftwareMetrics) which is a set of tools that helps in creating fuzzymodel. Fei and Lui [10] introduced the f-COCOMO modelwhich applied

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

fuzzy logic to the COCOMO model forsoftware effort estimation. Since there was no comparisonof the results between the f-COCOMO and other effortestimation models in their study, the estimation capabilityof the former is unknown. Roger [11] also proposed a fuzzyCOCOMO model which adopted the fuzzy logic method tomodel the uncertainty of software effort drivers, but theeffectiveness of the proposed model is not mentioned. Idri[7, 8] further defined a fuzzy set for the linguistic values ofeach effort driver with a trapezoid-shaped membershipfunction for the fuzzy COCOMO model. The effortmultipliers in the original COCOMO model were obtained

from the fuzzy sets. This fuzzy COCOMO model was lesssensitive to the software effort drivers as compared to theintermediate COCOMO81. In 2004, Xue and Khoshgoftaar[13] presented a fuzzy identification effort estimationmodeling technique to deal with linguistic effort drivers,and automatically generated the fuzzy membershipFunctions and rules by using the COCOMO81 database. Theproposed fuzzy identification model provided significantlybetter effort estimates than the original three COCOMOmodels, i.e., basic, intermediate, and detailed.

Fuzzy logic approach

Since fuzzy logic foundation by LotfiZadeh in 1965, it hasbeen the subject of important investigations [12]. It is amathematical tool for dealing with uncertainty and also it provides a technique to deal with imprecision andinformation granularity [11]. The fuzzy logic model usesthe fuzzy logic concepts introduced by LotfiZadeh [12].Fuzzy reasoning consists of three main components [11, 12,13, 14]: fuzzification process, inference from fuzzy rulesand defuzzification process. Fuzzification process is wherethe objective term is transformed into a fuzzy concept. Themembership functions are applied to the actual values ofvariables to determine the confidence factor or membershipfunction (MF). Fuzzification allows the input and output tobe expressed in linguistic terms. Inferencing involvesdefuzzification of the conditions of the rules and propagationof the confidence factors of the conditions to the conclusionof the rules. A number of rules will be fired and the inferenceengine assigned the particular outcome with the maximummembership value from all the fired rules.

Parameters analysis

The main parameter for the evaluation of cost estimationmodels is the Magnitude of Relative Error (MRE) [13] whichis defined as follows

$$\text{Magnitude Relative Error (RE)} = \frac{\left|E - \hat{E}\right|}{E}$$

Where E = Estimated Effort, $\hat{E}$ =Actual Effort.

The MRE value is calculated for each observation iwhose effort is predicted. The aggregation of MRE overmultiple observations (N), can be achieved through the MeanMRE (MMRE) as follows:

MRE = 1/N(Magnitude Relative Error)

Fuzzy identification

A fuzzy model [20,8] is used when the systems are notsuitable for analysis by conventional approach or when theavailable data is uncertain, inaccurate or vague [15]. The pointof Fuzzy logic is to map an input space to an output spaceusing a list of if-then statements called rules. All rules areevaluated in parallel, and the order of the rules is unimportant.For writing the rules, the inputs and outputs of the system areto be identified. To obtain [21] a fuzzy model from the dataavailable, the steps to be followed are,• Select a Mamdani type Fuzzy Inference System.

- Define the input variables mode, size and output variableeffort.
- Set the type of the membership functions (TMF orGBellMF orGauss2 or Trapezoidal) for input variables.
- The data is now translated into a set of if–then ruleswritten in Rule editor.
- A certain model structure is created, and parameters ofinput and output variables can be tuned to get the desiredoutput.

Fuzzy approach for prediction of effort

The Intermediate COCOMO model data is used fordeveloping the Fuzzy Inference System (FIS)[10]. The inputsto this system are MODE and SIZE. The output is FuzzyNominal Effort. The framework [16] is shown in “Fig. 1”.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

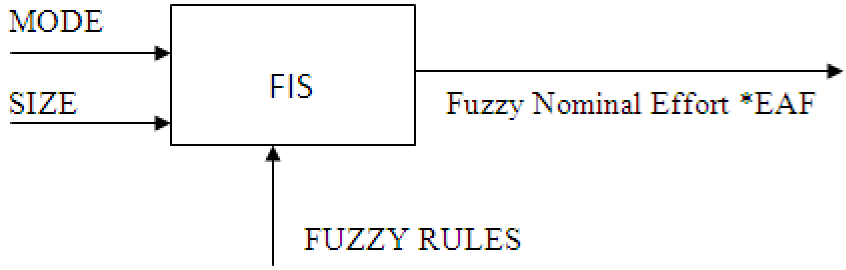


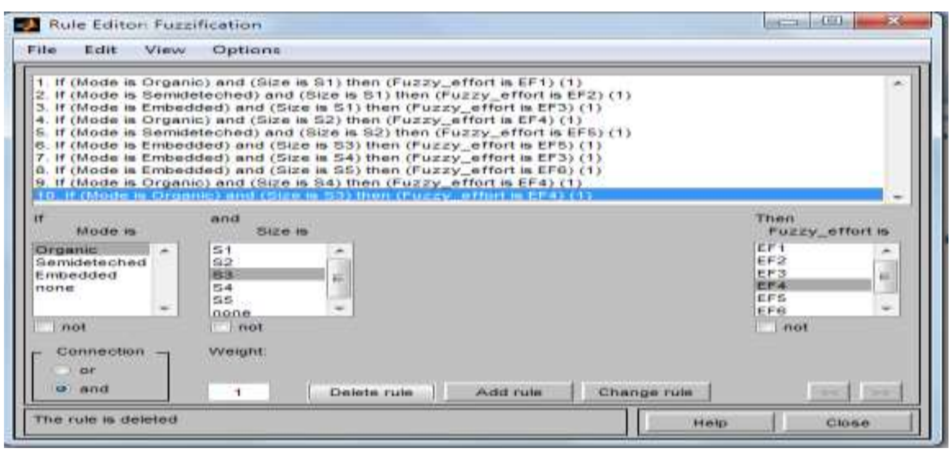
Figure 1: Fuzzy Framework.

Fuzzy approach [17] specifies the SIZE of a project as a range of possible values rather than a specific number. The MODE of development is specified as a fuzzy range .The advantage of using the fuzzy ranges[18] is that we will be able to predict the effort for projects that do not come under a precise mode i.e. comes in between 2 modes. This situation cannot be handled using the COCOMO. The output of this FIS is the Fuzzy Nominal Effort. The Fuzzy Nominal Effort multiplied by the EAF gives the Estimated Effort. The FIS[19] needs appropriate membership functions and rules.

Fuzzy rules

Our rules are based on the fuzzy sets of MODE, SIZE and EFFORT appears in the following form:
If MODE is Organic and SIZE is S1 then EFFORT is EF1
If MODE is Semidetached and SIZE is S1 then EFFORT is EF2
If MODE is Embedded and SIZE is S1 then EFFORT is EF3
If MODE is Organic and SIZE is S2 then EFFORT is EF4
If MODE is Semidetached and SIZE is S2 then EFFORT is EF5
If MODE is Embedded and SIZE is S3 then EFFORT is EF5
If MODE is Embedded and SIZE is S4 then EFFORT is EF3
If MODE is Organic and SIZE is S3 then EFFORT is EF4
If MODE is Embedded and SIZE is S5 then EFFORT is EF6
If MODE is Organic and SIZE is S4 then EFFORT is EF4

This is represented in MATLAB as shown in figure below:



Membership functions

A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The input space is sometimes referred to as the universe of discourse, a fancy name for a simple concept.One of the most commonly used examples of a fuzzy set is the set of tall people. In this case the universe of discourse is all potential heights, say from 3 feet to 9 feet, and the word tall would correspond to a curve that defines the degree

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

to which any person is tall. If the set of tall people is given the well-defined (crisp) boundary of a classical set, we might say all people taller than 6 feet are officially considered tall. But such a distinction is clearly absurd. It may make sense to consider the set of all real numbers greater than 6 because numbers belong on an abstract plane, but when we want to talk about real people, it is unreasonable to call one person short and another one tall when they differ in height by the width of a hair.

Fuzzy Logic Membership Functions used in Fuzzy Logic toolbox.

1. Trimf - Triangular-shaped built-in membership function

Syntax: y = trimf(x,params)

y = trimf(x,[a b c])

Description: The triangular curve is a function of a vector, x, and depends on three scalar parameters *a*, *b*, and *c*, as given by

$$f(x;a,b,c) = \max\left(\min\left(\frac{x-a}{b-a},\frac{c-x}{c-b}\right),o\right)$$

2.Gbellmf-Generalized bell-shaped built-in membership function

Syntax:y = gbellmf(x,params)

Description: The generalized bell function depends on three parameters *a*, *b*, and *c* as given by

$$f(x;a,b,c) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}}$$

Where the parameter *b* is usually positive. The parameter *c* locates the center of the curve. Enter the parameter vector params, the second argument for **gbellmf**, as the vector whose entries are *a*, *b*, and *c*, respectively.

1. Trapmf-Trapezoidal-shaped built-in membership function

Syntax: y = trapm f(x, [a b c d])

Description: ***The trapezoidal curve is a function of a vector, x, and depends on four scalar parameters a, b, c, and d, as given by***

$$f(x;a,b,c,d) = \max\left(\min\left(\frac{x-a}{b-a},1,\frac{d-x}{d-c}\right),o\right)$$

The parameters *a*and*d* locate the "feet" of the trapezoid and the parameters *b* and *c* locate the "shoulders."

4. gauss2mf - Two-sided Gaussian membership function.

Synopsis

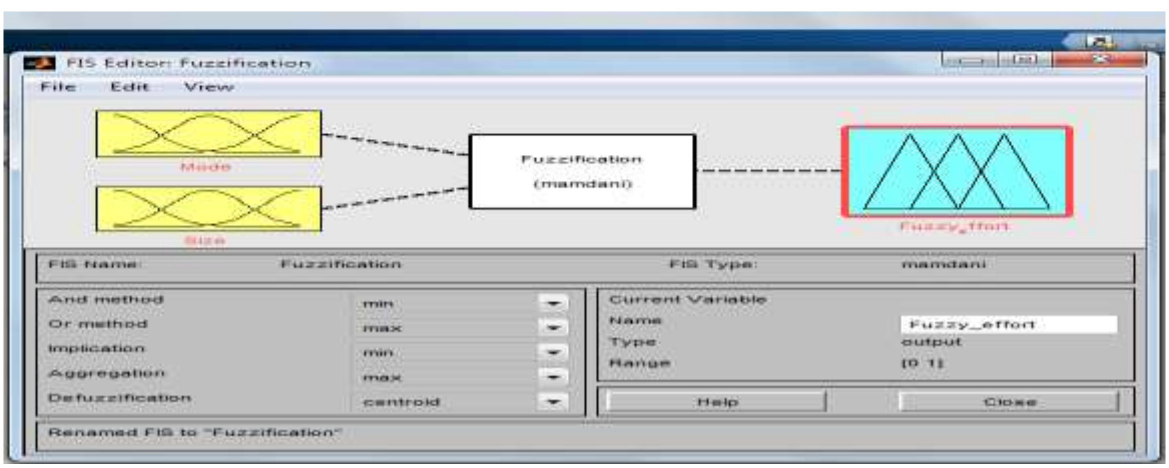
y = gauss2mf(x,params)

y = gauss2mf(x,[sig1 c1 sig2 c2])

3.Experimental Results

Experiments were done by taking original datafrom COCOMO dataset [14]. The software development efforts obtained when usingCOCOMO and other membership functions were observed. After analyzing the results attained bymeans of applying COCOMO, trapezoidal MF forcost drivers, and Gaussian MF for both size and costdrivers together, it is observed that the effortestimation of the proposed model is giving moreprecise results than the other models.

COCOMO used Mamdani FIS method due to its intuitive,widespread acceptance and well suited for human inputnature. Figure 2 show the fuzzification of costattributes using MATLAB.



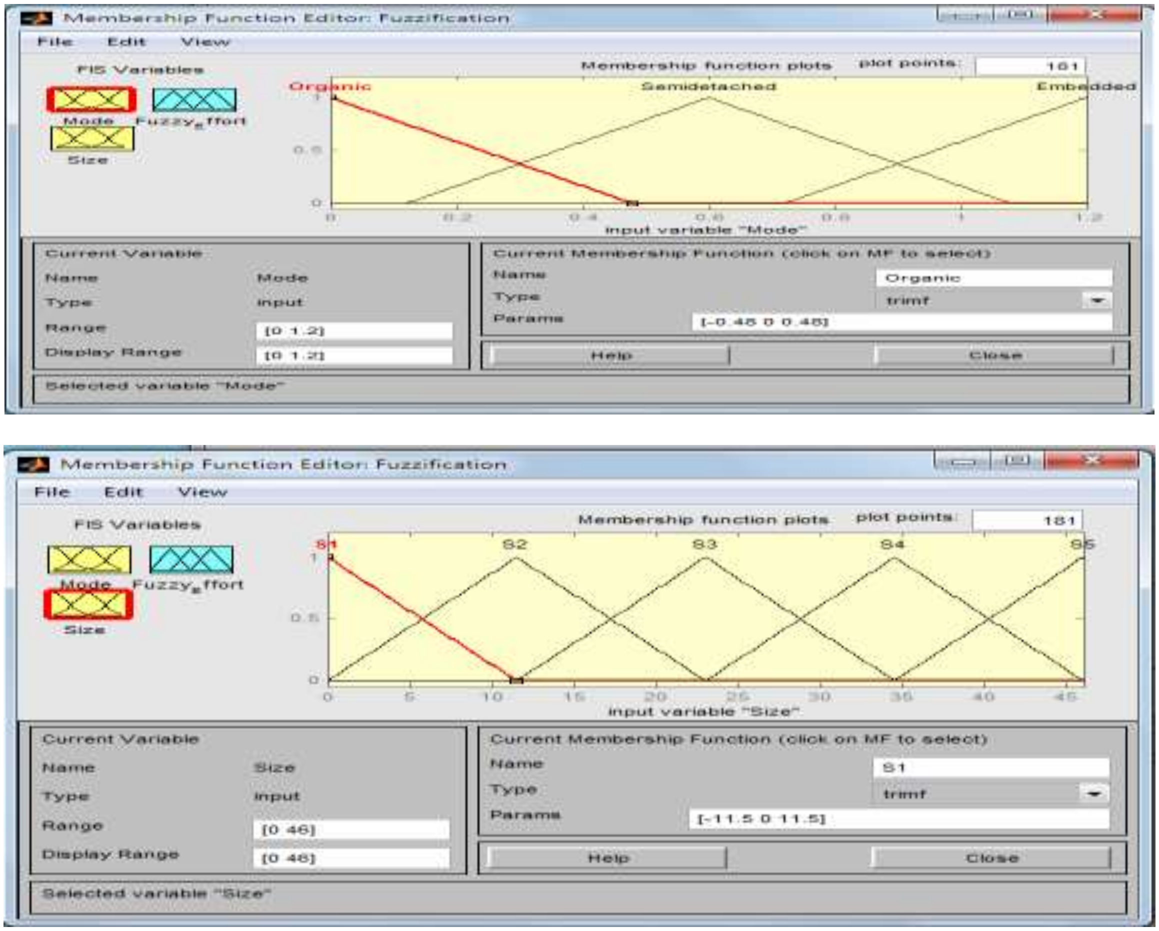


Fig 2: Fuzzification of various cost using FIS tool in the MATLAB software.

Table 1 -Comparison between obtained results from COCOMO 81 and FL-COCOMO in terms of MMRE

MODEL	MMRE
Basic COCOMO 81	0.60197
Intermediate COCOMO 81	0.18889
Detailed COCOMO 81	0.18829
FL-COCOMO (Using Triangular function)	0.2454
FL-COCOMO (Using Trapezoidal function)	0.1953
FL-COCOMO (Using Gaussian2 function)	0.1799
FL-COCOMO (Using GBell function)	0.1832

The effort estimated by means of fuzzifying size and cost drivers together and using Gaussian MF isyielding better estimate which is very nearer to the actual effort. Therefore, using fuzzy sets, size and cost drivers of a software project can be specified by distribution of its possible values, by means of whichwe can evaluate the associated imprecision residingwithin the final results of cost estimation.

Conclusions and future research

This research work is to provide a technique for software cost estimation that performs better than other techniques on the accuracy of effort estimation.
In this research an improved approach to software project effort is projected by the use of fuzzy sets rather than classical intervals in the COCOMO model.This study explores four fuzzy logic membership functions Fuzzy Triangular Membership Function, GBell Membership Function,Gauss2 Membership Function and

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Trapezoidal Membership Function is implemented and compared with COCOMO. The Gaussian2 Membership Function used in this research has shown good results by handling the imprecision in inputs quite well and also their ability to adapt further make them a valid choice to represent fuzzy sets.
Mean Relative Error shows the comparison between Fuzzy Membership Functions. Trapezoidal Membership Function has highest Mean Relative Error this implies it has lowest accuracy. Mean Magnitude Relative Error of Gaussian2 Membership Function shows better software effort estimates as compared to the traditional COCOMO. Lower the MMRE better is the prediction accuracy of the model. The above research work can be analyzed in terms of feasibility and acceptance in the industry.sssIt can be deployed on COCOMO II environment with experts providing required information for developing fuzzy sets and an appropriate rule base.

References

1. N. E. Fenton, S. L. Pfleeger, "Software Metrics, A Rigorous and Practical Approach", 2nd Edition, PWS Publishing Company, Thomson Publishing, Boston, 1997.
2. A. R. Gray, S. G. MacDonell, "Applications of Fuzzy Logic to Software Metric Modelsfor Development EffortEstimation". *Fuzzy Information Processing Society 1997NAFIPS! 97*, Annual Meeting of the North American, 21& 24 September 1997, pp. 394 & 399.
3. B. W. Boehm, Software Engineering Economics,Englewood Cliffs, NJ, Prentice Hall, 1981.
4. L. H. Putnam, "A General Empirical Solution to theMacrosoftware Sizing and Estimating Problem". *IEEETransactions on Software Engineering*, SE-4(4), 1978, pp345-361.
5. Attar Software. 2002. "Fuzzy Logic in KnowledgeBuilder", White Paper. http://www .i n t ellicraf t ers.com/ fuzzy.htm.
6. M. O. Saliu, M. Ahmed and J. AlGhamdi. "OwardsAdaptive Soft Computing based Software EffortPrediction", *Fuzzy Information, 2004. Processing NAFIPS'04.IEEE Annual Meeting of the North American FuzzyInformation Processing Society*, 27-30, June 2004, **1**, pp.16-21.
7. A. Idri, T. M. Khoshgoftaar, A. Abran. "Can NeuralNetworks be Easily Interpreted in Software CostEstimation", *IEEE Trans. Software Engineering*, **2**, 2002,pp. 1162 & 1167.
8. A. Idri,, A. Abran,, T.M. Khoshgoftaar. #EstimatingSoftware Project Effort by Analogy based on LinguisticValues" *in.Proceedings of the Eighth IEEE Symposium on Software Metrics*, 4-7 June 2002, pp. 21 & 30.
9. C. J. Burgess, M. Lefley. "Can Genetic ProgrammingImprove Software Effort Estimation, A ComparativeEvaluation", *Information and Software Technology*, **43**,No.14, 2001, pp. 863-873.
10. Fei Z, Liu X (1992) f-COCOMO: :Fuzzy Constructive CostModel in Software Engineering". *In: IEEE InternationalConference on Fuzzy Systems*, pp 331&337.
11. Roger JS (1993) ANFIS: "Adaptive Network based FuzzyInference Systems". *IEEE Trans Syst Man Cybern*,3(3):665&685.
12. L. A. Zadeh. "Fuzzy Sets".*Information and Control*, **8**,1965, pp. 338-353.
13. Xu Z. and Khoshgoftaar, T. M. Identification of FuzzyModels of Software Cost Estimation, 2004.
14. S. G. MacDonell, A. R. Gray, J. M. Calvert. #FULSOME:Fuzzy Logic for Software Metric Practitioners andResearchers$ in Proceedings of the 6th InternationalConference on Neural Information Processing ICONIP!99,ANZIIS!99, ANNES!99 and ACNN!99, Perth, WesternAustralia, IEEE Computer Society Press, 1999, pp.308-313.
15. MoshoodOmoladeSaliu, Adaptive FuzzSoftware Development Effort Prediction, King Fahd University ofPetroleum & Minerals, April 2003
16. ImanAttarzadeh and Siew Hock Ow, Software Development EffortEstimation Based on a New Fuzzy Logic Model, IJCTE, Vol. 1, No.October2009.
17. Xishi Huang, Danny Ho,Jing Ren, Luiz F. Capretz, A soft computingframework for software effort estimation, Springer link, Vol 10, No 2Jan-2006.
18. Ryder J., "Fuzzy Modeling of Software Effort Prediction", Proc. of

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IEEEInformation Technology Conference, Syracuse, NY, 1998.
19. M. W. Nisar, Yong-Ji Wang, M. Elahi and I.A Khan, "Software Development Effort Estimation Using Fuzzy Logic", Information Technology Journal, 2009 Asian Network for Scientific Information,2009.
20. Zonglian F. and Xihui L., "f-COCOMO: Fuzzy Constructive Cost Modelin Software Engineering", Proc. of IEEE Int. Conf. On Fuzzy Systems,IEEE, 1992, 331-337.
21. M. Braz and S. Vergilio, "Using Fuzzy Theory for Effort Estimation ofObject-Oriented Software", Proceedings of the 16th IEEE InternationalConference on Tools with Artificial Intelligence, ICTAI 2004.