An Improved Fuzzy Approach for COCOMO's Effort Estimation using Gaussian Membership Function

Ch. Satyananda Reddy, KVSVN Raju
DENSE Research Group
Department of Computer Science and Systems Engineering
College of Engineering, Andhra University, Visakhapatnam, INDIA
satyanandau@yahoo.com, kvsvn.raju@gmail.com

***Abstract* --- In software industry Constructive Cost Model (COCOMO) is considered to be the most widely used model for effort estimation. Cost drivers have significant influence on the COCOMO and this research investigates the role of cost drivers in improving the precision of effort estimation. It is important to stress that uncertainty at the input level of the COCOMO yields uncertainty at the output, which leads to gross estimation error in the effort estimation. Fuzzy logic has been applied to the COCOMO using the symmetrical triangles and trapezoidal membership functions to represent the cost drivers. Using Trapezoidal Membership Function (TMF), a few attributes are assigned the maximum degree of compatibility when they should be assigned lower degrees. To overcome the above limitation, in this paper, it is proposed to use Gaussian Membership Function (GMF) for the cost drivers by studying the behavior of COCOMO cost drivers. The present work is based on COCOMO dataset and the experimental part of the study illustrates the approach and compares it with the standard version of the COCOMO. It has been found that Gaussian function is performing better than the trapezoidal function, as it demonstrates a smoother transition in its intervals, and the achieved results were closer to the actual effort.**

***Index Terms* --- COCOMO, Fuzzy based effort estimation, Gaussian membership function, Software cost estimation, Software effort estimation and Project management.**

I. Introduction

Software cost estimation is one of the most critical tasks in managing software projects. There is inevitable gap between the estimated costs and the actual costs derived from software projects and hence accurate cost estimates are highly desired during the early stages of development. The precision of the effort estimate is very important for software industry because both overestimates and underestimates of the software effort are harmful to software companies. If a manager's estimate is too low, then the software development team will be under considerable pressure to finish the product quickly. On the other hand, if a manager's estimate is too high, then too many resources will be committed to the project. In point of fact, estimating software development effort remains a complex problem and it is very important to investigate novel methods for improving the accuracy of such estimates.

The most popular techniques used for software cost estimation is algorithmic models such as COCOMO [3, 4, 5], IBM-FSD [9], PUTNAM-SLIM [8], SPQR [6] and function points analysis [2, 7]. Fuzzy logic-based cost estimation models are more appropriate when vague and imprecise information is to be accounted for. In this paper, it is proposed to extend the Constructive Cost Model (COCOMO) [3] by incorporating the concept of fuzziness into the measurements of cost drivers; fuzzy set theory is used rather than classical intervals to represent the linguistic values. The advantages of this over quantization are that they are more natural and they mimic the way in which humans interpret linguistic values.

Though, many membership functions were used in the literature [10] to represent the cost drivers, many of them are not appropriate to clear the vagueness in the cost drivers. The triangular, trapezoidal membership functions are being used in COCOMO to replace the conventional quantization by using fuzzy interval values [19]. So, the transition from one interval to another is abrupt rather than gradual. Therefore after studying the behavior of the cost drivers [4], to get emphasize, a way of propagation of uncertainty and to attain smoother transition in the membership function, this paper attempts to achieve a fuzzy based effort by using Gaussian Membership Function, GMF. Hence, in this paper it has been proposed and validated empirically, the use of GMF to represent the cost divers in the COCOMO. It has been found that Gaussian function is performing better than trapezoidal function, as it demonstrates a smoother transition in its intervals, and the achieved results were closer to the actual effort.

Manuscript Received July 11, 2008;
Manuscript Revised December 12, 2008;
Manuscript Accepted February 10, 2009.

The rest of this paper is organized as follows: Section 2 provides the general idea of the methods employed in this paper. Section 3 briefly describes the related work done for estimating the effort through different fuzzy logic approaches. Section 4 presents the proposed fuzzy effort estimation model using Gaussian function and reveals the methodology used in this research. Section 5 presents experimental design and the methodology used in the application of Gaussian function to COCOMO using fuzzy logic tool box. Section 6 summarizes the experimental results. The final section concludes that the accuracy of effort estimation can be improved through the proposed model and the estimated effort can be very close to the actual effort.

II. LITURATURE REVIEW

A. Software Effort Estimation

It is unrealistic to expect very accurate effort estimates of software development effort because of the inherent uncertainty in software development projects, and the complex and dynamic interaction of factors that impact software development effort use. Still, it is likely that estimates can be improved because software development effort estimates are systematically overoptimistic and very inconsistent. Even small improvements will be valuable because of the large scale of software development. It is required to forecast the estimation to the project managers in planning and conducting software development activities because the software price determination, resource allocation, schedule arrangement and process monitoring are dependent upon it. In recent decade, many software effort estimation techniques have been proposed to evaluate their estimation performances. Some of these widely used techniques include the estimation by expert [12], analogy-based estimation [23], algorithmic method [24], rule induction [11], artificial neural network [25] and fuzzy logic [26].

A number of algorithmic models have been proposed as the basis for estimating the effort, schedule and costs of a software project. These are conceptually similar but use different parameter values. Algorithmic cost modeling uses a mathematical formula to predict project costs based on estimates of the project size, the number of software engineers, and other process and product factors. An algorithmic cost model can be built by analyzing the costs and attributes of completed projects and finding the closest fit formula to actual experience. The accuracy of the estimates produced by an algorithmic model depends on the system information that is available.

B. COCOMO Model

The COCOMO model is an empirical model that was derived by collecting data from a large number of software projects. These data were analyzed to discover formulae that were the best fit to the observations. These formulae link the size of the system and product, project and team factors to the effort to develop the system. In COCOMO, effort is expressed as Person Months (PM). It determines the effort required for a project based on software project's size in Kilo Source Line Of Code (KSLOC) as well as other cost factors known as scale factors and effort multipliers by as shown in "(1),"

$$Effort = A \times [Size]^{1.01+\sum_{i=1}^{5} SF_i} \times \prod_{i=1}^{17} EM_i \qquad (1)$$

where A is a multiplicative constant, and the set of Scale Factors (SF) and Effort Multipliers (EM) are defined the model [5]. It contains 17 effort multipliers and 5 scale factors. The standard numeric values of the cost drivers are given in Appendix. This formula proposed by the developers of the COCOMO model reflects their experience and data, but it is an extremely complex model to understand and use. There are many attributes and considerable scope for uncertainty in estimating their values. In principle, each user of the model should calibrate the model and the attribute values according to its own historical project data, as this will reflect local circumstances that affect the model.

When using algorithmic effort estimation models, the cost drivers have to be measured first in order to derive the effort estimate. The cost drivers of a software project being developed are characteristically vague and uncertain at the early stages of its life cycle; hence, it is difficult to generate an accurate effort estimate [28]. The biases usually come out when the measurements of the software cost drivers are based on human judgment. These approaches do not consider the vague and uncertain features that are inhabited in the effort drivers. The vagueness of the cost drivers significantly affects the accuracy of the effort estimates derived from software effort estimation models [27]. Since the vagueness and uncertainty of software effort drivers cannot be avoided, a fuzzy model has the advantage of easily verifying the cost drivers by adopting fuzzy sets. Several researchers have reported the progress made regarding the successful application of fuzzy logic technique in constructing software effort estimation models to enhance the model capabilities.

C. Fuzzy logic

Fuzzy logic is a methodology, to solve problems which are too complex to be understood quantitatively, based on fuzzy set theory, which introduced by Prof. Zadeh in 1965 [20]. Use of fuzzy sets in logical expression is known as Fuzzy Logic (FL) that has been the subject of important investigations. At the beginning of the nineties, fuzzy logic was firmly grounded in terms of its theoretical foundations and used in various fields.

The central assertion underlying this approach is that entities in the real world simply do not fit into neat categories. For example, a project is not small, medium, or large. It could in fact be something in between; perhaps mostly a large project but also something like a medium project. This can be represented as a degree of belonging to a particular linguistic category. Fuzzy sets can be effectively used to represent linguistic values such as low, young, and complex. A fuzzy set can be defined mathematically by assigning to each possible individual in the universe of discourse a value representing its grade of membership in the fuzzy set to a greater or lesser degree as indicated by a larger or smaller membership grade. The fuzzy set is represented as where x is an element in X and $\mu_A(x)$ is a membership function of set A which defines the membership of fuzzy set A in the universe of discourse, X

D. Fuzzy Membership Functions

A fuzzy set is characterized by a membership function, which associates with each point in the fuzzy set a real number in the interval [0, 1], called degree or grade of membership. The membership function may be triangular, trapezoidal, Gaussian etc. A triangular membership is described by a triplet (a, m, b), where 'm' is the modal value, 'a' and 'b' are the right and left boundary respectively. The trapezoidal membership function (see "Fig. 1,") is defined as follows.

$$\mu_Z(x_k, \gamma_k) = \begin{cases} 1 & x_k \in [u_k, U_k] \\ 1-\max(0, \min(1, \gamma_k(u_k - x_k))) & x_k \prec u_k \\ 1-\max(0, \min(1, \gamma_k(x_k - U_k))) & x_k \succ U_k \end{cases}$$

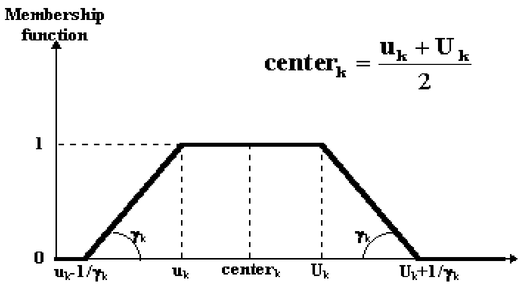

Figure 1. Trapezoidal Membership Function for $\mu_Z(x_k, \gamma_k)$

Another fuzzy membership function that is often used to represent vague, linguistic terms is the Gaussian which is called Gaussian membership function (See "Fig. 2,") is defined as follows.

$$\mu_Z(x_k, \gamma_k) = \exp\left(-\frac{1}{2} \cdot \frac{(x_k - \text{center}_k)^2}{\gamma_k^2}\right)$$

$$\text{center}_k = \frac{u_k + U_k}{2}$$

with $\gamma_k \neq 0$ for any $k \in \{1, 2, ..., n\}$

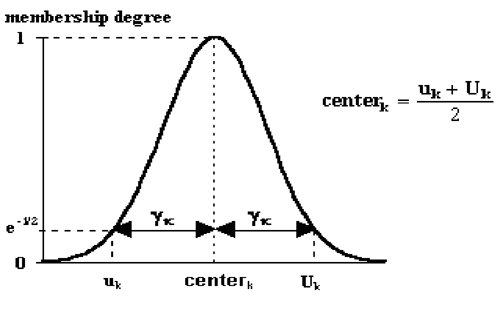

Figure 2. Gaussian Membership Function for $\mu_Z(x_k, \gamma_k)$

III. Related Work

Papers were reviewed regarding aspects related to research on software development effort estimation based on a fuzzy logic model. Studies showed that fuzzy logic model has a place in software effort estimation. Attempts have been made to fuzzily some of the existing models in order to handle uncertainties and imprecision problems. Using real project data, Gray and MacDonell [13] compared Function Point Analysis, Regression techniques, Feed forward neural network and Fuzzy logic in software effort estimation. Their results showed that fuzzy logic model achieved good performance, being outperformed in terms of accuracy only by neural network model with considerably more input variables. In their fuzzy logic model, triangular membership functions were defined for the small, medium, large intervals of size.

Fuzzy logic had also been applied to algorithmic models to cater for the need of fuzziness in the input. The first realization of the fuzziness of several aspects of COCOMO was that of Fei and Liu [17]. It is fact that an accurate estimate of delivered source instruction (KDSI) cannot be made before starting a project, and it is unreasonable to assign a determinate number for it. Ryder [21] researched on the application of fuzzy logic to COCOMO and Function Points models. Musflek et al. [10] worked on fuzzifying basic COCOMO model without considering the adjustment factor. On the other hand, Idri et al., [1] proposed fuzzy intermediate COCOMO with the fuzzification of cost drivers. The effort multiplier for each cost driver is obtained from fuzzy set, enabling its gradual transition from one interval to a contiguous interval. Validation results showed that the fuzzy intermediate COCOMO can tolerate imprecision in its input (cost drivers) and generate more gradual outputs.

Ahmed and Saliu [15] geared up further by fuzzifying the two different portions of the COCOMO model i.e. nominal effort estimation and the adjustment factor. They proposed a fuzzy logic framework for effort prediction by integrating the fuzzified nominal effort and the fuzzified effort multipliers of the intermediate COCOMO model. So far, the mainstream of the work is concentrated on fuzzifying cost drivers with the representation of triangular and trapezoidal membership functions. Hence, in this work, it is proposed to use fuzzy set interval values using GMF for the cost drivers of the project in the effort estimation of Constructive Cost Model.

IV. Research Methodology

A. Problem-Formulation

It is important to stress that uncertainty at the input level of the COCOMO model yields uncertainty at the output [12]. This becomes obvious and, more importantly, bears a substantial significance in any practical endeavor. Fuzzy logic-based cost estimation models are more appropriate when vague and imprecise information is to be accounted for. Cost drivers are often expressed through an unclear category which needs subjective assessment. The effort multipliers and scale factors of the COCOMO were described in natural language as very low, low, nominal, high, very high and extra high and these were represented by fixed numerical values [5]. More conventionally, the problem of software cost estimation using COCOMO relies on a single (numeric) value of cost driver of a given software project to predict the effort. But it is not an appropriate way to fix numerical number to each of these scales.

It is of principal importance to recognize this situation and come up with a technology using which we can evaluate the associated imprecision residing within the final results of cost estimation. The technology endorsed here deals with fuzzy sets. Using fuzzy sets, cost drivers of a software project can be specified by distribution of its possible values. Commonly, this form of distribution is represented in the form of a fuzzy set. By changing the cost drivers using fuzzy set, we can model the effort that impacts the estimation accuracy. Instead of using fixed numbers to characterize the cost drivers, interval values were used and these were represented using various membership functions triangular, trapezoidal etc.

However still there was some linearity by using these functions. Overlapped symmetrical triangles or trapezoids reduce fuzzy systems to precise linear systems [22]. Furthermore there is a possibility when using a trapezoidal function that some attributes are assigned the maximum degree of compatibility when they should be assigned lower degrees. In order to avoid this linearity it is proposed to use more continuous Gaussian function to represent the cost drivers.

B. Proposed Research Method

By studying the behavior of COCOMO cost drivers, in this investigation it is proposed to characterize the use GMF for cost drivers to represent the linguistic values. GMF gives more continuous transition from one interval to another. Gaussian Bell curve sets give richer fuzzy system with simple learning laws that tune the bell curve variance. The Gaussian Function is represented by "(2),"

$$\mu_{A_i}\langle x\rangle = Gaussian\langle x, c_i, \sigma_i\rangle = e^{-(x-c_i)^2/2\sigma_i^2} \quad (2)$$

Where c_i is the center of the i^{th} fuzzy set and σ_i is the width of the i^{th} fuzzy set.

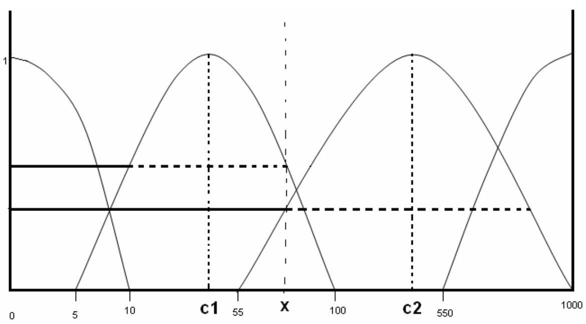

Figure 3. Representation of DATA Cost driver using Gaussian Membership Function

For example, in the case of DATA cost driver, we define a fuzzy set for each linguistic value with a Gaussian shaped membership function μ is shown in "Fig 3,". We have defined the fuzzy sets corresponding to the various associated linguistic values for each cost driver.

In this research, a new fuzzy effort estimation model is proposed by using Gaussian function to deal with linguistic data, and to generate fuzzy membership functions and rules for cost drivers obtained from "(3)". In the next step, we evaluate the COCOMO model using the "(1)," and cost drivers obtained from fuzzy sets (F_EM_{ij}) rather than from the classical EM_{ij}. F_EM_{ij} is calculated from "(4)," the classical EM_{ij} and the membership functions μ defined for the various fuzzy sets associated with the cost drivers.

$$F\_EM_{ij} = F\left[\mu_{A1}^{V_i}(p) \ldots \mu_{A_i}^{V_i}(p), EM_{i1} \ldots EM_{ij}\right] \quad (3)$$

For ease, F is taken as a linear function, where the μ^{Vi}_{Aj} is the membership function of the fuzzy set A_j associated with the cost driver V_i is shown in "(3)".

$$F\_EM_{ij} = \sum_{j=1}^{k_i} \mu_{A_i}^{V_i}(p) \times EM_{ij} \quad (4)$$

V. DESIGN METHODOLOGY

The proposed cost estimation model was implemented using fuzzy logic tool box of MATLAB software. The fuzzy inference system (FIS) is used in order to implement the various processing steps. Options were provided for creating and editing FIS with fuzzy logic tool box software using graphical tools or command line functions. This GUI tool allows us to edit the higher level features such as number of input and output variables of the FIS. Using FIS editor, membership functions can be added for each cost driver using 'addmf' command. Each cost driver in fuzzy COCOMO can be defined with membership function. The membership function editor 'mfedit' that allows us to inspect and modify all the membership functions. For each membership function we can change the name, type and parameters. All the cost drivers are defined and customized to the GMF using the command 'gaussmf' (x, [sig c]).

B.W. Boehm [3] is the first researcher to look at software engineering from an economic point of view, and he came up with COCOMO dataset. The COCOMO dataset is published where the complete dataset is accessible. In designing the above model, we have used the COCOMO [16] dataset includes 63 historical projects with 17 effort drivers and one dependent variable of the software development effort. The software development effort is recorded in terms of unit of person-month. Cost drivers are measured using a rating scale of six linguistic values: 'very low', 'low', 'nominal', 'high', 'very high' and 'extra high'. The assignment of linguistic values to the cost drivers (or project attributes) uses conventional quantification where the values are intervals. For example, in the case of the DATA cost driver, we have defined a fuzzy set for each linguistic value with a Gaussian-shaped membership function shown in “Fig 3,”. We note that the fuzzy sets associated with the DATA cost driver satisfy the normal condition.

The evaluation consists in comparing the accuracy of the estimated effort with the actual effort. There are many evaluation criteria for software effort estimation introduced in the literature, among them we applied the most frequent evaluation criteria such as: Magnitude of Relative Error (MRE), Mean Magnitude of Relative Error (MMRE) [18], which is defined as in “(5),”.

$$MRE = \frac{|ActualEffort - EstimatedEffort|}{ActuralEffort} \times 100 \qquad (5)$$

The GMF that has been proposed in this work gives accurate effort than by using any other membership functions. When it uses trapezoidal function the peak value is linear but in Gaussian function it touches the peak at only one point. Hence, Gaussian function is better than trapezoidal function, as it demonstrates a smoother transition between its intervals. The results clearly indicate that such fuzzy set modeling approach affects significantly the estimation outcomes.

VI. Experimental Results

Experiments were done by taking original data from COCOMO dataset. The software development efforts obtained when using COCOMO and other membership functions were observed. After analyzing the results attained by means of applying COCOMO, trapezoidal and GMF models, it is observed that the effort estimation of the proposed model is giving more precise results than the other models. The effort estimated by means of fuzzifying cost drivers using GMF is yielding better estimate which is very nearer to the actual effort. Therefore, using fuzzy sets, cost drivers of a software project can be specified by distribution of its possible values, by means of which we can evaluate the associate imprecision residing with the final results of cost estimation.

Table I shows the sample results obtained for some of the data sets taken from COCOMO dataset, which includes the effort estimated using Constructive Cost Model and the effort obtained using TMF for the cost drivers, and the effort achieved using GMF for the cost drivers i.e. the proposed fuzzified model. It has been found the proposed model is performing better than ordinal COCOMO and Gaussian function is performing better than trapezoidal function, as it demonstrates a smoother transition in its intervals, and the achieved results were closer to the actual effort.

Figure 4 shows the bar chart representing comparative analysis of actual effort with that of the effort estimated using COCOMO, Trapezoidal and Gaussian membership functions. Effort in person months is scaled along with y-axis. Actual effort, COCOMO effort, and effort obtained using trapezoidal MF, and effort obtained using GMF for cost drivers, were represented for each sample projects, which were taken along with x-axis.

TABLE I.
Results And Comparison Of
Effort Estimation In Person Months

Project ID	Actual Effort	Effort in Person Months (PM)		
		COCOMO	Using Trapezoidal MF	Using Gaussian MF
1	61	45.63	48.49	51.35
2	237	214.10	227.52	241.16
3	599	539.60	573.44	607.8
4	603	553.43	588.12	623.36
5	702	1335.1	1129.64	1057.8
6	523	278.86	356.48	408.29
7	1075	661.30	794.28	876.42
8	2455	1945.4	2068.3	2121.3
9	958	408.33	535.72	599.57
10	1063	1275.9	1179.5	1123.5

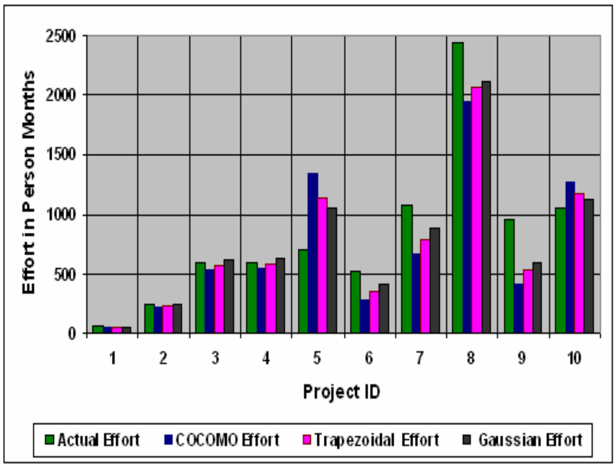

Figure 4. Chart representing the comparisons of effort estimation

The magnitude of relative errors was calculated using “(5)”. For example, the relative error calculated for project 1 for COCOMO, trapezoidal and for the proposed model is 25.20, 20.51 and 15.82 respectively. In the case of second project it is 9.66, 4.00 and 1.76. The Mean Magnitude of Relative Error (MMRE) is 32.65, 22.09 and

17.02 respectively. Figure 5 shows the chart representing relative errors which are represented along with y-axis against each project, which is taken along with x-axis. This clearly shows that there is a decrement in the relative error, so that the proposed model is more suitable for effort estimation.

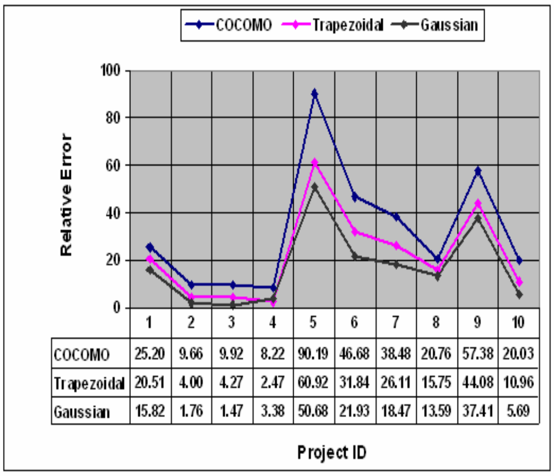

Figure 5. Assessments of Magnitude of Relative Errors

VII. Conclusions And Future Research

A crucial issue for project managers is the accurate and reliable estimates of the required software development effort, especially in the early stages of the software development life cycle. Software effort drivers usually have properties of uncertainty and vagueness when they are measured by human judgment. Cost drivers in algorithmic software cost estimation are often expressed through linguistic assessments and they usually represent high level concepts for which a single, precise measurement scale is not available. This motivates the use of fuzzy techniques to model estimation inputs and their assessment procedures. To date, fuzzy logic modeling techniques have been shown to be the most effective approximation method to handle imprecise data.

In this paper, it is projected an improved approach to estimate the software project effort by the use of fuzzy sets rather than classical intervals in the COCOMO model. For each cost driver and its associated linguistic values, corresponding fuzzy sets were defined. These fuzzy sets are represented by Gaussian-shaped membership functions. After considering the results attained by means of applying COCOMO, trapezoidal and GMF models, it is observed that the effort estimated by means of fuzzifying cost drivers using GMF is yielding better estimate which is very nearer to the actual effort. The relative error for COCOMO using Gaussian function is lower than that of the error obtained using TMF. From the experimental results, it is concluded that, by fuzzifying the cost drivers of the project using GMF, it can be proved that the resulting estimate impacts the effort. The effort generated using the proposed model gives more precise result than that of using the TMF. This illustrates that by using GMF, the accuracy of effort estimation can be improved and the estimated effort can be very close to the actual effort.

In conclusion, the success of any software project relies on accurate estimations and a soft-computing technique such as fuzzy logic is a feasible choice as an estimation model for improving estimation accuracies. At the same time, this study shows the applicability and the strength of using fuzzy logic in software development effort estimation. Reliable estimations by using fuzzy logic will ensure significantly higher probabilities of software project success rates.

An ongoing research is related to applying the fuzzy logic system proposed in this paper to other software development effort estimation models. There are other possible representations which can be tried with different forms of membership functions for a more realistic modeling. To define a convenient representation, we have to study the significance of the various linguistic values in the environment from which the COCOMO database was assembled. This work can be extended by integrating with neural networks to take the advantage of its features, such as learning ability and good interpretability. Thus, a promising line of future work is to extend to the neuro-fuzzy approach that allows the integration of numerical data and expert knowledge.

Appendix COCOMO Cost Drivers

Cost Drivers	Range	Description
RELY	0.82-1.26	Required Software Reliability
DATA	0.90-1.28	Database Size
CPLX	0.73-1.74	Product Complexity
RUSE	0.95-1.24	Developed for Reusability
DOCU	0.81-1.23	Documentation Match to Life-Cycle Needs
TIME	1.00-1.63	Execution Time Constraint
STOR	1.00-1.46	Main Storage Constraint
PVOL	0.87-1.30	Platform Volatility
ACAP	1.42-0.71	Analyst Capability
PCAP	1.34-0.76	Programmer Capability
PCON	1.29-0.81	Personnel Continuity
APEX	1.22-0.81	Applications Experience
PLEX	1.19-0.85	Platform Experience
LTEX	1.20-0.84	Language and Tool Experience
TOOL	1.17-0.78	Use of Software Tools
SITE	1.22-0.80	Multi site Development
SCED	1.43-1.00	Required Development Schedule

REFERENCES

[1] Ali Idri and Alain Abran, “COCOMO cost model using fuzzy logic,” *Proceedings of 7th International Conference on Fuzzy Theory and Technology, Atlantic City, New Jersey*, Feb 27-Mar 03,2000.

[2] Alberta’s., Gaffney, J.E., “Software function, Source lines of code, and development effort prediction: A Software science validation,” *IEEE Transactions on Software Engineering*, Vol.SE-9, No.6, Nov, 1983, pp.639-647.

[3] Boehm, B.W., *Software Engineering Economics*, Prentice-Hall, Englewood Cliffs, NJ (1981).

[4] Boehm, B.W., Royce, W.W., *Le COCOMO Ada*, Genie logiciel & Systemes experts, 1989.

[5] Boehm, B.W., et al., “Cost models for future software life cycle processes: COCOMO2.0,” *Annals of Software Engineering on Software Process and Product Measurement, Amsterdam*, 1995.

[6] Jones, C., *Programming Productivity*, McGraw-Hill, New York, 1986.

[7] Matson, J.E., Barrett, B.E., Mellichamp, J.M., “Software development cost estimation using function points,” *IEEE Transactions on Software Engineering*, Vol. 20, No. 4, Apr., 1994, pp.275-287.

[8] Putnam, L.H., “A general empirical solution to the macro software sizing and estimation problem,” *IEEE Transactions on Software Engineering*, No. 4, July, 1978.

[9] Walston, C.E., Felix, A.P, “A method of programming measurement and estimation,” *IBM Systems Journal*, Vol. 16, no.1, 1997.

[10] Musflek P, Pedrycz W., Succi G., Reformat M, 2000, “Software cost estimation with fuzzy models,” *Applied Computing Review*, Vol. 8, No. 2, pages 24-29.

[11] Jeffery R, Ruhe M,Wieczorek I, “Using public domain metrics to estimate software development effort,” *In Proceedings of the 7th International Symposium on Software Metrics*, (April 04 - 06, 2001), *IEEE Computer Society, Washington, DC*, pp 16–27.

[12] Jorgenson M, Sjoberg D.I.K., The impact of customer expectation on software development effort estimates. *International Journal of Project Management*, 22(4) : 317–325.

[13] Gray A. R., S. G. MacDonell, “Applications of fuzzy logic to software metric models for development effort estimation,” *Fuzzy Information Processing Society 1997 NAFIPS’ 97*, Annual Meeting of the North American, 21 – 24 September 1997, pp. 394 – 399.

[14] Braz, M., Vergilio S., “Using fuzzy theory for effort estimation of object-oriented software,” *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, ICTAI 2004.

[15] Ahmed, M. A., Omolade Saliu, M., Alghamdi, J., “Adaptive fuzzy logic-based framework for software development effort prediction,” *Elsevier Science, Information and Software Technology*; Volume 47, Issue 1, 1 January 2005, Pages 31-48.

[16] Marcel Korte, Dan Port, “Confidence in software cost estimation results based on MMRE and PRED,” *PROMISE'08, Leipzig, Germany*, May 12-13, 2008.

[17] Fei. Z, X. Liu., “f-COCOMO - Fuzzy constructive cost model in software engineering,” *Proceedings of IEEE International Conference on Fuzzy System*, pp. 331-337.

[18] Briand L.C., Emam K.E. Surmann D. and Wieczorek I., “An assessment and comparison of common software cost estimation modeling techniques,” *ISERN*-1998-27.

[19] Leonard J. Jowers A, James J. Buckley B, Kevin D. Reilly A, “Estimation of f-COCOMO model parameters using optimization techniques,” http://sunset.usc.edu /events/2006/CIIForum/pages/presentations/2006SEWorl d Jowers-Buckley-Reilly-c.pdf.

[20] Zadeh. L. A., *Fuzzy Sets, Information and Control*, Volume 8, 1965, pp. 338-353.

[21] Ryder. J., “Fuzzy modeling of software effort prediction,” *Proceedings of IEEE Information Technology Conference, Syracuse, NY*, 1-3 Sept.’1998, pp: 53-56.

[22] Ali Idri, Alain Abran, “ La Logique Floue Appliquee Aus Modeles d’Estimation d’Effort de Development de Logiciels Cas Du Modele COCOMO’81,” http://www.gelog.etsmtl.ca/publications/pdf/560.pdf.

[23] Chiu NH, Huang SJ, “The adjusted analogy-based software effort estimation based on similarity distances,” *Journal of Systems and Software*, Volume 80, Issue 4, April 2007, Pages 628-640.

[24] Kaczmarek J, Kucharski M, “Size and effort estimation for applications written in Java,” *Journal of Information and Software Technology*, Volume 46, Issue 9, 1 July 2004, Pages 589-601.

[25] Heiat A, “Comparison of artificial neural network and regression models for estimating software development effort,” *Journal of Information and Software Technology*, Volume 44, Issue 15, 1 December 2002, Pages 911-922.

[26] Huang SJ, Lin CY, Chiu NH, “Fuzzy decision tree approach for embedding risk assessment information into software cost estimation model,” *Journal of Information Science and Engineering*, Volume 22, Number 2, March 2006, Pages 297–313.

[27] Jeffery R, Ruhe M, Wieczorek I, “A comparative study of two software development cost modeling techniques using multi-organizational and company-specific data,” *Journal of Information and Software Technology*, Volume 42, Number 14, 15 November 2000 , pp. 1009-1016(8).

[28] Pedrycz W, Sosnowski ZA, “The design of decision trees in the framework of granular data and their application to software quality models,” *Journal of Fuzzy Sets and Systems*, Volume 123, Issue 3, 1 November 2001, Pages 271-290.

Ch. Satyananda Reddy obtained his B.E. Degree in Computer Science and M.Tech Degree in Computer Science with specialization in Software Engineering from Jawaharlal Nehru Technological University, Hyderabad, India. He is currently a Research Scholar and working as Assistant Professor in the Department of Computer Science and Systems Engineering at Andhra University College of Engineering, Visakhapatnam, India. His Research interests include Software Engineering like Software Metrics, Software Quality Assurance, Cost Estimation, Neural Networks and Fuzzy Systems.

KVSVN Raju holds a PhD degree in Computer Science and Technology from IIT, Kharagpur, and is presently working as Professor in the department of Computer Science and Systems Engineering at Andhra University Engineering College, Visakhapatnam. His research interests include Software Engineering Data Engineering and Data Security.