A min-max Approach for Improving the Accuracy of Effort Estimation of COCOMO

H. S. Hota, Ramesh Pratap Singh

Abstract— **This study proposed to extend the Constructive Cost Model (COCOMO) by incorporating the concept of min-max approach to estimation. Formal effort estimation models like Constructive Cost Model (COCOMO) are limited by their inability to manage uncertainties and imprecision surrounding software projects early in the development life cycle. A min-max approach is suggested to rectify data uncertainties and modeling errors. The proposed method of min-max is used to improve the accuracy of effort estimation of COCOMO and its result have been compared with the gradient descent, robust fuzzy clustering, k-mean clustering methods of estimation. It has been observed that the proposed method have lowest and steady state absolute estimate error AE(k) and mean absolute estimate error MAE(k) for different value of k(time series) and different step-size s.**

Index Terms— **Sugeno fuzzy inference system, min-max method, Constructive Cost Model (COCOMO), effort estimation, Absolute Estimate Error, Mean Absolute Estimate Error.**

I. Introduction

The precision and reliability of the effort estimation is very important for software industry because both overestimates and underestimates of the software effort are harmful to software companies. As highlighted by McConnell in [1], several surveys have found that about two-thirds of all projects substantially overrun their estimates. In reality estimating software development effort remains a complex problem attracting considerable research attention. It is very important to find novel method to handle vague and uncertain information for improving the accuracy of such estimates. A min-max approach [2] based cost estimation model is more appropriate when vague and imprecise information is to be accounted for.

The best known technique using LOC (Lines of Code) is the COCOMO (Constructive COst MOdel), developed by Boehm[3]. This model performs estimation using LOC, and other factors such as development environment, product attributes and hardware limitations. These factors provide one or more adjustment factors which adjust the estimate of the effort needed. In COCOMO's case, there are fourteen such factors derived by Boehm. But Constructive Cost Model (COCOMO) is limited by their inability to manage uncertainties and imprecision surrounding software projects early in the development life cycle [4]. Assumptions make estimates more accurate. Fuzzy logic-based cost estimation models are more appropriate when vague, imprecise and uncertain information is to be accounted [5] for suggested the use of fuzzy sets in improving the accuracy of effort estimation through fuzzy sets. But the amount of uncertainty involve in the estimation process, the min-max method is propose here to use in effort estimation of software development.

This study for improving the accuracy of effort estimation of COCOMO model assuming as physical process y=f(x1,….,x2), is concerned with the fuzzy partitioning of n-dimensional input space into K different clusters then estimating the process behavior $\hat{y} = f(\hat{x})$ for a given input $\hat{x} = (\hat{x}_{1,} .... \hat{x}_n) \in X$ and then fuzzy approximation of the process with uncertain input-output data $\{x(k) \pm \delta x_{k,} y(k) \pm y_k\}_{k=1,...}$ using Sugeno type fuzzy inference system. The size of the project in COCOMO is represented by fixed numerical values. In min-max based cost estimation models, this size is represented with fuzzy partition space. So, the transition from one partition space to an adjacent partition space is abrupt rather than gradual.

II. Related Literature Review

Estimation accuracy is largely affected by modeling accuracy [6]. Finding good models for software estimation is very critical for software engineering in bidding and planning. In the recent years many software estimation models have been developed [7, 8, 9, 10].

Gray and MacDonell compared function point analysis regression techniques, feed forward neural network and fuzzy logic in software effort estimation [11]. Their results showed that fuzzy logic model achieved good performance, being out performed in terms of accuracy only by neural network model with considerably more input variables [12]. The first realization of the fuzziness of several aspects of COCOMO was that of [13]. They observed that an accurate estimate of delivered source instruction [14] cannot be made before starting a project and it is unreasonable to assign a determinate number for it.

The data driven construction of fuzzy models [15] has become an important topic of research with a wide range of real-world applications. A common practice is to choose the type of fuzzy model i.e. Sugeno type [16], that is linear in consequent parameters and therefore standard least-squares linear techniques can be apply for their estimation. The constraints in the estimation of antecedent parameters may arise in order to preserve the interpretability of the fuzzy model [17]. The derivative based estimations methods like gradient-descent offer the advantage of fast convergence in comparison to the derivative-free methods like genetic algorithms [18], but they tend to converge to a local minima.

Manuscript received June 11, 2011.

Ramesh Pratap Singh, Department of Computer Science, D. P. Vipra College, Bilaspur, India, M:9300486756, (e-mail: singhramesh30@gamil.com).

H. S. Hota, Department of CS & IT, Guru Ghasidas Vishwavidyalaya (A central university), Bilaspur, India, M:9425222658, (e-mail: hota_hari@rediffmail.com).

However, if the assumptions are not met due to modeling errors, they perform poorly. This motivates the study of robust methods for fuzzy clustering, estimation and identification of unknown physical process.

In this research paper it is projected to unified min-max approach to fuzzy clustering, estimation and identification with uncertain data such that worst case effect of regression vector uncertainty, model output uncertainty, and modeling errors on estimation performance is minimized without making any assumption and requiring a priori knowledge of uncertainties. The Sugeno-type fuzzy inference systems are considered appropriate models, since they ideally combine simplicity with good analytical properties [19]. Moreover the data-driven construction of Sugeno fuzzy systems allows qualitative insight into relationships [20]. Therefore we consider the identification of a Sugeno type fuzzy model with uncertain data that partitions the input space into different clusters and approximate the input-output mappings in an optimal manner.

III. Formulation Of The Problem

Let us consider a Sugeno-type fuzzy inference system i.e. Fs:X→Y, mapping n-dimensional input space ($X=X_1 \times X_2 \times X_3 \times \ldots \times X_n$) to one dimensional real line, consisting of K different following rules:

If x belongs to a cluster with Centre C_1 THEN $y=\alpha_1$

If x belongs to a cluster with Centre C_2 THEN $y=\alpha_2$

.

.

If x belongs to a cluster with Centre C_k THEN $y=\alpha_k$

Where $x \in R^n$ is n-dimensional input vector, $C_i \in R^n$ is the centre of i^{th} cluster, and the values $\alpha_1, \alpha_2 \ldots \alpha_k$ are real numbers. In COCOMO effort is expressed as Person Month (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 drivers known as Scale Factors (SF) and Effort-Multipliers (EM) as shown below:

$$PM= M(size)^{1.01+\sum_{i=1}^{s} SF_i} \prod_{i=1}^{n} EM_i \qquad \ldots(1)$$

Where, M is a multiplier constant and the set of Scale Factors (SF) and Effort Multipliers (EM) are defined the model. Here n=15 EM and s=5 SF. The standards numeric values of the cost drivers are given in Table 3 (see Appendix).

Let $\mu_i(x)$ be a multivariable membership function μ_i :X→[0,1] that represents the degree of membership of input vector $x \in X$ to the i^{th} cluster (in place of fuzzy interval values). Now the different rules [21] can be aggregated as

$$F_s(x)=\frac{\sum_{i=1}^{k}\alpha_i\mu_i(x)}{\sum_{i=1}^{k}\mu_i(x)} \qquad \ldots(2)$$

The optional shape of membership function $\mu_i(x)$ can ensure by the method of fuzzy C-means (FCM) as below

$$\sum_{x\in X}\sum_{i=1}^{k}\mu_i^m(x)\left\|x-c_i\right\|^2 \rightarrow \text{Minimum}$$

$$\sum_{i=1}^{k}\mu_i(x)=1 \qquad \ldots(3)$$

Where m>1 is the fuzzifier and || . || denotes the Euclidean norm. The solution of previously constrained optimization problem is given as:

$$\mu_i(x,C_1,\ldots,C_k)=\begin{cases} \dfrac{1}{\sum_{j=1}^{k}\left(\dfrac{\left\|x-c_j\right\|^2}{\left\|x-c_j\right\|^2}\right)^{\frac{1}{m-1}}} & for\, x\in X\setminus\{c_j\}_{j=1,2,\ldots,k} \\ 1 & for \quad x=c_j \\ 0 & for\, x\in\{c_j\}_{j=1,2,\ldots,k}\setminus\{c_j\} \end{cases} \qquad \ldots(4)$$

How ever, this clustering criterion suffers more than a few drawbacks like:

(1) In presence of uncertainty in the data, it may be difficult sometimes to explicitly express how to reconstruct the data from a cluster solution.

(2) The membership functions for the clusters are non convex and for generation of fuzzy sets, the membership functions should be convex.

To overcome the above drawbacks another clustering criterion is adopted assuming that there is a noise cluster outside each data cluster. Following clustering criterion is chosen:

$$J_c(\mu_i(x),C_1,\ldots,C_k)=\sum_{i=1}^{k}\sum_{x\in X}\left[\mu_i(x)\left\|x-c_i\right\|^2+\{1+\mu_i(x)\log\mu_i(x)-\mu_i(x)\}\delta_i\right] \ldots(5)$$

where the second term in the objective function is intended as a noise cluster. The term $\{1+\mu_i(x)\log\mu_i(x)-\mu_i(x)\}$ may be interpreted as the degree to which x does not belong to the i^{th} cluster and thus the membership of x to the noise cluster.

The criterion to set δ_i equal to the distance of nearest cluster center from c_i is:

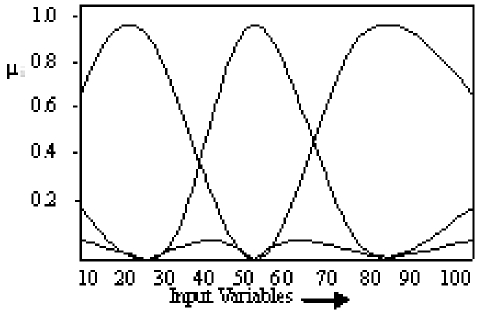
$$\delta_i = \min_j \left\|c_j-c_i\right\|^2 \qquad \ldots(6)$$

To minimize $J_c(.)$ of equations:

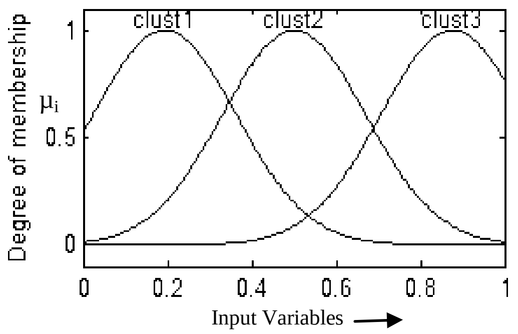
$$\frac{\partial}{\partial\mu_i(x)}J_c(\mu_i(x),C_1,\ldots C_k)=0 \qquad \ldots(7)$$

This result in the following expression for optimal membership function

$$\mu_i(x,C_1,\ldots C_k)=\exp\left(-\frac{\left\|x-c_i\right\|^2}{\delta_i}\right) \qquad \ldots(8)$$



(a) Membership functions for eq^n (4) before rectification



(b) Membership functions for eq^n (8) after rectification

Figure 1(a & b) shows the membership functions before (equation 4) and after (equation 8) the optimization and rectifying the noise clusters respectively for three different clusters with centres at 20, 50 and 80.Thus it can be seen (fig 1(b)) that the shape of membership functions being convex can be used to generate fuzzy sets. The output of Sugeno-type fuzzy inference system for this shape of membership function (fig 1(b)) is given by:

$$F_s(x)=\frac{\sum_{i=1}^{k}\alpha_i \exp\left(-\frac{\|x-c_i\|^2}{\delta_i}\right)}{\sum_{i=1}^{k}\exp\left(-\frac{\|x-c_i\|^2}{\delta_i}\right)} \quad \ldots(9)$$

$$\sigma_i(x, c_1,\cdots,c_k)=\frac{\exp\left(-\frac{\|x-c_i\|^2}{\delta_i}\right)}{\sum_{i=1}^{k}\exp\left(-\frac{\|x-c_i\|^2}{\delta_i}\right)} \quad \ldots(10)$$

then

$$F_s(x)=\sum_{i=1}^{k}\alpha_i\sigma_i(x, c_1,\ldots, c_k) \quad \ldots(11)$$

Let us introduce the notations,

$\alpha=[\alpha_i]_{i=1\ldots k}\in R^k$

$\theta=\left[c_{1,\ldots}^T c_k^T\right]^T\in R^{k.n}$ and

$\sigma(x,\theta)=[\sigma_i(x,\theta)]_{i=1\ldots k}\in R^k$

The equation (11) can be written as

$$F_s(x)=\sigma^T(x,\theta)\alpha \quad \ldots(12)$$

Let us consider the fuzzy approximation [22] of software effort estimation process described by time series equation as given below:

$$y(k)= f(x(k))+\delta x_k+v_k \quad \ldots(13)$$

where δx_k is uncertainty present in input vector $x(k)$ and v_k is the corresponding uncertainty in $y(k)$. Let (α^*,θ^*) be some true value of fuzzy model parameters, which approximate the software effort estimation process then

$$y(k)=\sigma^T(x(k)+\delta x_{k,}\theta^*)\alpha^*+v_k \quad \ldots(14)$$

or $$y(k)=\left[\sigma(x(k),\theta^*)+\delta\sigma_k\right]^T+v_k \quad \ldots(15)$$

Where $\delta\sigma_k=\delta\sigma_k(x(k),\delta x_k,\theta^*)$ is the uncertainty in regression vector due to noise δx_k in $x(k)$. For $y=f(x)$ software effort estimation process, a unified approach to robust fuzzy clustering, estimation and identification is to estimate recursively the parameters (α^*,θ^*), say $\{\alpha_k,\theta_k\}$ at k^{th} time index, with the measurements $\{x(k),y(k)\}$ in presence of uncertainties $(\delta x_k, v_k)$, without making any assumption and requiring priori knowledge of upper bounds, statistics and distribution of data uncertainties and modeling errors, such that

(i) For a known $x'\in X$

$$\left|\sigma^T(x',\theta^k)\alpha^k - f(x')\right|\rightarrow\text{minimum} \quad \ldots(16)$$

(ii) For all $x\in X$

$$\left|\sigma^T(x,\theta^*)\alpha^* - f(x)\right|\rightarrow\text{minimum}$$

where $$(\alpha^*,\theta^*)=\lim_{k\rightarrow\infty}(\alpha^k,\theta^k) \quad \ldots(17)$$

IV. MIN-MAX APPROACH TO SOFTWARE EFFORT ESTIMATION

Now, we present a min-max approach to the robust estimation of cluster centers vector θ^* and fuzzy consequents α^*. The approach consists of solving a local min-max estimation problem where the worse-case effect of data uncertainties and modeling error on estimation performance is minimized. The software effort estimation process can be modeled as:

$$y(k)=\sigma^T(x(k),\theta_k)\alpha^*+v(k) \quad \ldots(18)$$

where $v(k)$ includes not only the data uncertainties, but also the error resulting from a difference between θ_k and θ^*. We define the following error measures for our problem. $\tilde{\alpha}_k$ denotes the difference between α^* and estimates α_k i.e.

$$\tilde{\alpha}_k=\alpha^*-\alpha_k \quad \ldots(19)$$

and $e_a(k)$ denotes the priori estimation error i.e.

$$e_a(k)=\sigma^T(x(k),\theta_k)\tilde{\alpha}_{k-1} \quad \ldots(20)$$

To measures the estimation performance, we define instantaneous absolute estimation error (AE) at time interval k as:

$$AE(k)=\frac{1}{100}\sum_{j=1}^{100}\left|f(x_i^j)-\sigma^T\left(\left[x_i^j\right]^T,\theta_k\right)\alpha_k\right|^2 \quad \ldots(21)$$

where the points $\left\{x_i^j\right|^T\Big\}_{j=1,\ldots,100}$ are uniform distributed in the i-dimensional input space, If the simulations run from k=1 to k=T, then the mean absolute estimation error (MAE) can be defined as:

$$MAE=\frac{1}{T}\sum_{k=1}^{T}AE(k) \quad \ldots(22)$$

We run over simulation for T=10000 for a fair comparison, the same step-size has been chosen for Gradient-descent, Robust fuzzy clustering, K-means clustering and min-max methods. Since the different methods may have their best performance at different step-size, therefore we perform simulations at different values of step-size ranging from S=0.01 to S=0.1.

Table 1: Describes the identified value base using min-max approach at k=10000

Rules	Antecedent	Consequent
R_1	If x belongs to the cluster having centre $[-0.51,0.04]^T$ then y is equal to	7.21
R_2	If x belongs to the cluster having centre $[-0.214,0.676]^T$ then y is equal to	-4.024
R_3	If x belongs to the cluster having centre $[-0.54,2.5]^T$ then y is equal to	-4.2
R_4	If x belongs to the cluster having centre $[1.55,-0.67]^T$ then y is equal to	-2.8
R_5	If x belongs to the cluster having centre $[0.41,0.83]^T$ then y is equal to	1.15
R_6	If x belongs to the cluster having centre $[1.15,2.5]^T$ then y is equal to	3.08
R_7	If x belongs to the cluster having centre $[2.5,-0.37]^T$ then y is equal to	-3.67
R_8	If x belongs to the cluster having centre $[2.5,1.04]^T$ then y is equal to	1.65
R_9	If x belongs to the cluster having centre $[2.5,2.5]^T$ then y is equal to	3.22

V. SIMULATION OF PROBLEM

Now we provide simulation studies to compare gradient-descent, robust fuzzy clustering and K-means clustering with the proposed min-max approach, for this purpose, consider a software effort estimation process for two inputs as:

$$y = f(x_1,x_2) = \frac{20(x_1-0.06)(x_2-0.06)}{1+x_1^2+x_2^2} \qquad \ldots(23)$$

where $x_1 \in [-0.6,2.6]$ and $x_2 \in [-0.6,2.6]$

The fuzzy model divides the two dimensional input space into nine clusters with initial guess about cluster centers as (equally spaced).

$$[c_1,\ldots,c_9] = \begin{bmatrix} -0.6 & -0.6 & 0.6 & 1 & 1 & 1 & 2.6 & 2.6 & 2.6 \\ -0.6 & 1 & 2.6 & -0.6 & 1 & 2.6 & -0.6 & 1 & 2.6 \end{bmatrix} \qquad \ldots(24)$$

The initial guess α_0 is taken equal to a zero vector. The nonlinear system was simulated by choosing x_1 and x_2 from a uniform distribution on the interval (-0.6, 2.6). The uncertain input-output identification data is generated by the sequence:

$$\left\{\begin{bmatrix} (1+\delta x_1 k) & x_1 k \\ (1+\delta x_2 k) & x_2 k \end{bmatrix}, (1+\delta y(k))y(k)\right\} \qquad \ldots(25)$$

where $\delta x_1 k$, $\delta x_2 k$ and δy_k are random entries, chosen from a normal distribution with mean zero and variance 0.01.

Fig2 and Fig3 summarizes the simulation results by comparing the final value of absolute estimation error (i.e. AE(10000)) and mean absolute estimation error (i.e. MAE) of all four approaches at different values of step sizes S. A remarkable good performance of proposed min-max approach in comparison to others can be easily seen in Fig 2 and Fig 3. This verifies the robustness properties of min-max estimation scheme in presence of data uncertainties and modeling errors. The better performance of min-max estimation is due to the fact that no only linear parameters (consequents), but also the non-linear parameters (cluster centre's) attempt to match the output of the fuzzy model to unknown process output in an optimal manner.

Experiments were done by taking original data from COCOMO dataset [23]. The software development efforts obtained when using COCOMO [3, 24] and other membership functions observed. After analyzing the results attained by means of applying COCOMO model using Gradient-descent, Robust fuzzy clustering, K-means clustering and min-max methods, 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 fuzzyfying step-size using bell shaped membership function is yielding better estimates, which is very nearer to the actual effort. Therefore, using fuzzy sets (membership functions) size of a software project can be specified by distribution of its possible values, by means of which we can evaluate the associated imprecision residing within the final results of cost estimation.

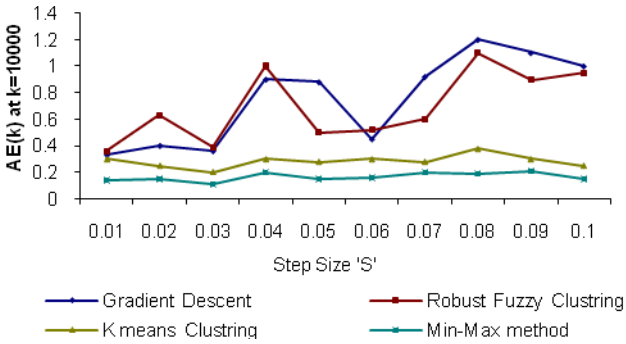


Fig. 2: Comparison of AE(k) for different methods for different values of step-size 'S'

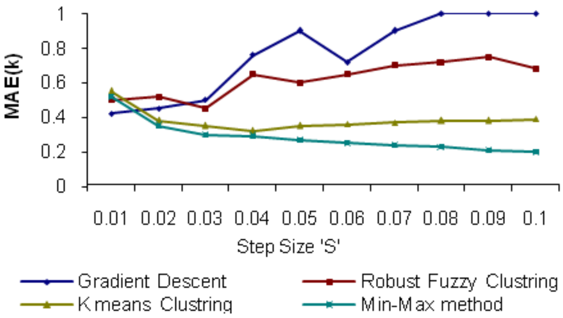


Fig 3: Comparison of MAE(k) for different methods for different values of step-size 'S'.

VI. DISCUSSION

From the above, it is observed that by fuzzyfying the some step-size of the project dataset using belshaped curved after rectification, it can be proved that the resulting AE(k) and

MAE(k) for effort estimation of COCOMO model for the proposed min-max method is found better in performance. This illustrates that by fuzzyfying step size using belshaped membership function after its correction, the accuracy of effort estimation can be improved by using proposed min-max method and estimated effort can be very close to the actual effort. This concept open a new conceptual dimension to the models of software cost estimation.

The COCOMO-81 consists of 63 projects dataset. The quantification of COCOMO multipliers of the dataset is a task that must be done by an expert based on his experience. Therefore, there is an uncertainty lying in the identification data (uncertain measurements of parameters and comments of expert).

VII. CONCLUSION

This study has outlined a new min-max approach to the fuzzy clustering, estimation and identification with uncertain data. The proposed approach minimizes the worst-case effect of data uncertainties and modeling errors on estimation performance of effort estimation of COCOMO model, without making and statistical assumption and requiring a priori knowledge of uncertainties. Simulation studies have been provided to show the better performance of proposed method in comparison to the standard techniques applying on effort estimation of COCOMO model. The performance of proposed method is found to be better in comparison to the standard techniques.

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] Steve McConnell. *Rapid development: taming wild software schedules*. Microsoft Press, 1996.
[2] Mohit Kumar et al, A Min-Max Approach to fuzzy clustering Estimation and Identification, IEEE Transaction on fuzzy system, vol. 14, No. 2, PP 248-262, 2006.
[3] B.W. Boehm, Software Engineering Economics, Englewood Cliffs, NJ, Prentice Hall, 1981.
[4] Iman Attarzadeh, Siew Hock Ow, Improving the Accuracy of Software Cost Estimation Model Based on a New Fuzzy Logic Model, World Applied Science, Vol. 8(2), pp. 177-184, 2010.
[5] C.S. Reddy and KVSVN Rao, Improving the accuracy of effort estimation through fuzzy set representation of size, Journal of Computer Science, Vol 5, pp 451-455, 2009.
[6] M. Jorgensen, B. Faugli, T. Gruschke, Characteristics of software engineers with optimistic prediction, Journal of Systems and Software, Vol. 80, Issue. 9, 2007, pp. 1472-1482.
[7] C.L. Martin, J.L. Pasquier, M.C. Yanez, T.A. Gutierrez, Software Development Effort Estimation Using Fuzzy Logic: A Case Study, IEEE Proceedings of the Sixth Mexican International Conference on Computer Science (ENC'05), 2005, pp. 113-120.
[8] M.T. Su, T.C.Ling, K.K.Phang, C.S.Liew, P.Y.Man, Enhanced Software Development Effort and Cost Estimation Using Fuzzy Logic Model, Malaysian Journal of Computer Science, Vol. 20, No. 2, 2007, pp. 199-207.
[9] A. Heiat, Comparison of artificial neural network and regression models for estimating software development effort, Information and Software Technology, Vol. 44, Issue 15, 2002, pp. 911-922.
[10] X. Huang, Danny Ho, J. Ren, L.F. Capretz, "Improving the COCOMO model using a neuro-fuzzy approach", Applied Soft Computing , Vol.7, Issue 1, 2007, pp. 29-40.
[11] A. R. Gray, 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.
[12] P. Thrift, Fuzzy logic synthesis with genetic algorithms, in proc., 4th Int. Conf. on Genetic Algorithms, pp 509-513, 1991.
[13] Fei Z, Liu X (1992) f-COCOMO: fuzzy constructive cost model in software engineering. In: IEEE international conference on fuzzy systems, pp 331 – 337.
[14] D. Simon, Design and rule base reduction of a fuzzy filter for the estimation of Motor currents, Int. Jou. of Approx. Reason, Vol. 25, pp 145-167, 2000.
[15] J. A. Roubos and M. Setnes, Compact fuzzy models through complexity reduction and evolutionary optimization, in proc. IEEE Int. Conf. fuzzy systems, San Antonio, TX, pp 762-767, 2000.
[16] J. S. R. Jang, ANFIS: Adaptive-network-based fuzzy inference systems IEEE Trans. Syst., Man, Cybernetics, Vol. 23, No. 3, pp 665-685, 1993.
[17] M. Setnes et al, Rule-based modeling precision and transparency, IEEE Trans. Syst. Man. Cybern. C. Apple. Rev., Vol. 28, No. 1, pp 165-169, 1998.
[18] J. S. R. Jang, C. T. Sun and E. Mizutani, Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence, Upper Saddle River, NJ: Prentice-Hall 1997.
[19] M. Kumar et al, Robust Adaptive Fuzzy Identification of Time-Varying Processes with Uncertain Data. Handling Uncertainties in the Physical Fitness Fuzzy Approximation with Real World Medical Data: An Application, Fuzzy Optimization and Decision making, Vol. 2, No.3, pp 243-259, 2003.
[20] J. C. Bezdeck, Pattern Recognition with fuzzy objective function algorithms, New York: Plenum, 1981.
[21] L. X. Wang and J. M. Mendel, Generating fuzzy rules by learning from examples, IEEE Trans. Syst., Man, Cybernetics, Vol. 22, No. 6, pp 1414-1427, 1992.
[22] D. Nanck and R. Kruse, A Neuro-fuzzy approach to obtain interpretable fuzzy systems for function approximation, in proc. IEEE Int. Conf. fuzzy systems 1998 (FUZZ-IEEE'98), Anchorage, AK, pp 1106-1111, May 1998.
[23] Menzies, T., 2005. Promise software engineering repository. http://promise.site.uottawa.ca/SERepository/
[24] B. Boehm, E. Horowitz, R. Madachy, D. Reifer, B. K. Clark, B. Steece, A. W. Brown, S. Chulani, and C. Abts, Software Cost Estimation with Cocomo II. Prentice Hall, 2000.

H.S. Hota: received MCA degree from the Bhoj Open University, Bhopal (M.P.), India in 2001, and the Ph.D. degree in Computer Science from the Guru Ghasidas University, Bilaspur (Chhattisgarh), India in 2006.

Presently I am working as Assistant Professor, Department of Computer Science and Information Tech., Guru Ghasidas University, Bilaspur (Chhattisgarh), India. My research interests include Software Engineering and Soft Computing tools.

Ramesh Pratap Singh: received M.Sc. degree in Information Technology from the Guru Ghasidas University, Bilaspur (Chhattisgarh), India in 2002, the M. Phil degree in Computer Science from Alagappa University, Karaikudi (T.N.), India in 2009.

Presently I am working as Assistant Professor & Head, Department of Computer Science, D. P. Vipra College, Bilaspur (Chhattisgarh), India. My research interests include Cost and Effort estimation in Software Engineering and Soft Computing tools.