LC-JSTEM
Your Research Partner

ISSN: 2708-7123 | Volume-01, Issue Number-01 | March-2020
LC International Journal of STEM (Science, Technology, Engineering, and Math)
Email: editor@logicalcreations.org | Web: www.logicalcreations.org/stem | OJS: www.logicalcreations.org/ojs

IMPROVING SOFTWARE COST ESTIMATION WITH FUNCTION POINTS ANALYSIS USING FUZZY LOGIC METHOD

S. Z. Iqbal[1], Kashif Saghar[2]
Department of CS and Technology,
[1,2]Alhamd Islamic University, Islamabad, Pakistan

ABSTRACT— Function Points Analysis (FPA) is amongst the most generally used method to assess software cost estimation frameworks. This process speaks to the measurement of an undertaking, application, and function by its relative functional complexity. In general, it has numerous effective applications used in both industry and scholarly research. This is noticed that customized estimate technologies which can confront genuine challenges utilizing on programming building information is normally constrained, loosely gathered and deficient. To enquire these queries composite programming models, blend of information, fuzzy logic and master judgment is proposed. This is trusted that outcomes announced here will animate, renew investigation of fuzzy logic to genuine programming designing issues. In this research paper, we use Function Points and apply some new models to pick up a superior estimation of programming properties. The utilization of ideas and characteristics from the fuzzy set hypothesis to stretch out function points analysis to fuzzy function points analysis. Fuzzy hypothesis tries to construct formal quantitative arrangement equipped for imitating imprecision of the human information. With the function points created by Fuzzy FPA, an estimate value for example, expenses/cost and software development can be more correctly determined.

Keywords— Fuzzy Logic, function points analysis (FPA), softwares cost estimation, fuzzy function points.

I. INTRODUCTION

Softwares effort estimate plays a key role in successful software projects because this effects on the project's cost, duration, quality, and performance [1].

Software effort estimates helps in estimating the schedule and staffing requirements which in turn collectively form the basis for project bidding, budgeting, risk analysis, project planning, progress monitoring and control. Over past many decades different models have been proposed and used for estimation of effort but each of them has their own pros and cons [2].

One of the soft computing technique fuzzy logic is very popular in terms of prediction or estimation technique. There are a number of decision making applications available which employ fuzzy logic as a main tool. Fuzzy logic has been used for prediction in medical diagnosis, political predictions, sports and finance etc [3].

The paper deals, fuzzy logic is popular in software quality prediction or cost estimation technique by using the MATLAB Fuzzy tool. In this tool it implemented fuzzy logic membership function (MF) to predict or estimated the software products. The cost model and statistical methods foresee the potential cost minimizing and defects decrease anticipated. The nature of programming ought to have as not many imperfections as could reasonably be expected. It is acknowledges that defects will be infused and the goal is to convey programming with hardly any defects inside the evaluated financial plan [4].

Over past few years, it has been accurately measured that the software costs are basic to the two developers and clients. As demonstrated in [5], Software Cost Estimation, that the right software cost estimation could assume an indispensable job in creating proposals, contract arrangements, scheduling, observing and control for which underestimating (increase in the budget with poor quality and under developed software) and overestimating (wide number of resources utilized for the project with unsatisfactory results) could be a threat.

Such a significant number of research articles giving several models to assist in processing cost and effort for software projects, having the option to give precise cost and the effort estimate is as yet challenge for some causes [6]. They consist of:

- Uncertainty in gathered estimation;
- Estimation techniques utilized that may have numerous disadvantages;
- Cost drivers which accompanies different attributes dependent on the methodology of the improvement.

Software cost estimate is a strategy for evaluating cost for software development. Software cost estimate is a measure of cost in both person days and person hours fundamental for directing assessments. Most generally utilized strategies to discover software development cost and spending plan are Function Points Analysis (FPA) [7], Constructive Cost Model (COCOMO) [8], SEER SEM [6], Putnam model [7], Top-Down model [6] and Bottom-up model [6-8].

The correct cost estimation is necessary to categorize and prioritize the development projects, define what resources are needed, evaluate effect of changes and rethinking, to make projects simpler and to have accurate genuine development costs according to assessed costs. This estimate incorporates effort, project duration and cost [9].

A. *Function Points Analysis (FPA)*

Function Points Analysis (FPA) is amongst the most generally utilized technique to assess the software cost estimation frameworks. Amid the point counting process that addresses the estimation of an endeavor or application, each function is considered by its relative functional complexity. The FPA utilizes the quantity of functions software ought to satisfy. These functions are identified with sorts of information software utilizes and creates. It assesses fourteen general system characteristics (GSCs), the data input received from external input, external inquiry, external output and internal logical file (that is, the functionality demanded by and providing to the end consumer), of a software against the standard criteria.

Albrecht initially proposed five function kinds (inputs, inquiries, outputs, interfaces and files) with one lot of related loads. In 1983, work created in [10], proposed extension of function kind, a set of three weighting esteems (for example low, medium, high) known as complexity of software project as shown in Table-I.

TABLE.1 The whole of all function points is then develop the Total Unadjusted Function Points COMPLEXITY (TUFP).

Description	Low	Medium	High
Input	3	4	6
Output	4	5	7
Files	7	10	15
Interfaces	5	7	10
Queries	3	4	6

Fourteen (14) General System Characteristics (GSCs) were suggested known as processing difficulty of software project as given in the Table-II. The Unadjusted Function Points count (UFC) is given below:

TABLE.2 PROCESSING COMPLEXITY

S#	GSCs
1	Data communication
2	Heavily used configuration
3	Transactions rate
4	Complexity handling
5	Installation easiness
6	End Users Efficiency
7	Numerous sites
8	Performance
9	Distributed function
10	Online data entry
11	Extensibility
12	Online update
13	Reusability
14	Operational ease

These fourteen (14) GSCs elements which affect size of project effort and every tasks are graded from Zero (0) that means no effect on processing complexity to five (5) that means great effect on processing complexity. GSCs contains 14 factors from f1 to f14. The sum of entire factors known as Processing Complexity (PC) is then multiplied as given in

Equation-I that constitutes the Adjust Processing Complexity (APC).

$$APC = 0.65 + 0.01 * PC$$ □□□

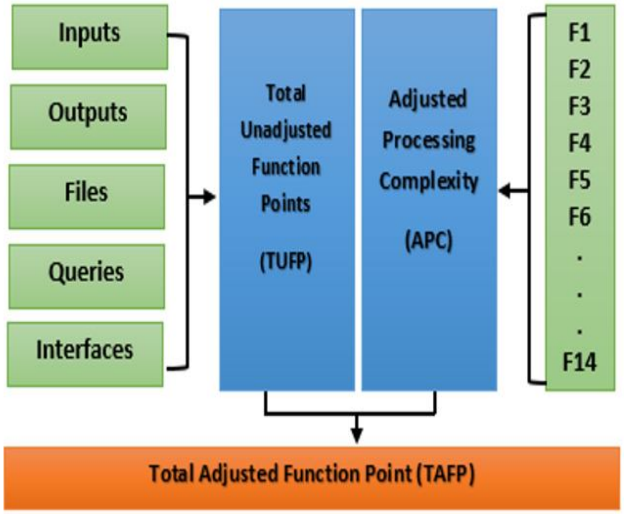

Inputs
Outputs
Files
Queries
Interfaces
Total Unadjusted Function Points (TUFP)
Adjusted Processing Complexity (APC)
F1
F2
F3
F4
F5
F6
.
.
.
F14
Total Adjusted Function Point (TAFP)

FIGURE.1 Function Point Calculation Method

Then, the Adjusted Processing Complexity (APC) is then multiplied by the Total Unadjust Function Points (TUFP) to generate Total Adjust Function Points (TAFP) count that is given in Equation-II.

$$TAFP = APC \ X \ TUFP$$ □□□

A diagram that shows method of calculating Total Adjust Function Points (TAFP) is given in Fig-1.

However, this model has very disadvantages. The advantages of FPA includes the availability from early requirement stage, applicability to the complete cycle, technology and language independent, provides better metrics, documentable and replicable lastly provides quantitative basis for earned value management. The disadvantages on the other hand are that accurate estimation can only be achieved with the full knowledge of the standards, this is largely manual process and variations are present that are not standardized

B. Fuzzy Logic

Whereas, according to, Fuzzy logic model is used to process vague and imprecise information that cannot be computed in any other estimation model. It is often used as machine learning tool that reads and computes human behavior as input, converts this crisp information in a process called fuzzification, \cite{IEEEhowto:ref13}. Fuzzy logic utilizes the fuzzy set philosophy that is the expansion of old style set theory where the participation of a component x of traditional set A, as a subset of universe X, is characterized as below:

$$\mu \ A \ (x) = 1 \ if \ x \ belongs \ to \ A$$ □□□

$$\mu \ \mu \ A(x) = 0 \ if \ x \ belongs \ to \ A$$ □□□

This framework has its own limitations. This method does not need special training, dependable estimation and its flexibility however is not simple to utilize rest its cost estimate of complex features is repetitive [2]. Figure-2 represents the Fuzzy Logic Model.

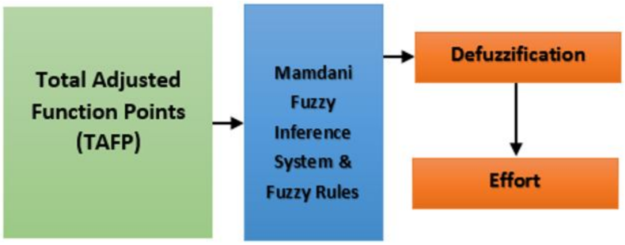

Total Adjusted Function Points (TAFP)
Mamdani Fuzzy Inference System & Fuzzy Rules
Defuzzification
Effort

FIGURE.2 Fuzzy Logic Model

II. LITERATURE REVIEW

For venture offering, planning and arranging are the premise and endeavors gauges by Software improvement, poor planning and arranging regularly has sensational results, and these are fundamental practices in the product business [14]. 15 percent of the product ventures are never finished because of the gross misestimating of improvement exertion and 60 percent of enormous tasks fundamentally overwhelm their evaluations in Boraso report [15]. For programming associations, conveying a product item on schedule, inside financial plan, and to a concurred degree of value is a simple concern. For better arranging, checking and control precise appraisals are significant [16]. In the writing new suggested cost estimate work execution is contrasted and related tasks.

To get most extreme execution, in the investigation same adjustment on the creation work is made all together [17]. Reason for the exploration is to suggest structure for developing product cost estimate model for the programming change venture dependent on social framework of an undertaking outline and programming cost utilizing relationship estimation technique and numerous relapse examination to gauge measure of programming alteration list and measure of exertion demanded for new task from information assortment of past undertaking [18].

In this examination for programming cost estimation a NN Constructive Cost Model (COCOMO) is proposed. Neural systems approach conveys a portion of the attractive highlights of this model, for example, great interpretability and, learning capacity while keeping up the benefits of the COCOMO model [19]. Back multiplication counterfeit neural frameworks in evaluating programming advancement exertion execution inspected the Wittig et al. Two unique investigations are performed first on mimicked information produced from measurement model (SPQR/2000) and the second organization of examinations contain real advancement information. In all cases, size of programming was estimated in venture exertion was supposed being as developed hours and capacity focuses (FPs) [20]. [21] Cerebellar Model Arithmetic Computer (CMAC), NN model to estimate of the exertion from programming code size (KDSI). Albus created CMAC is perceptron and the capacity approximator. NN model prepared on Boehm's COCOMO informational index so as to anticipate exertion from size, to fit line to informational index for the expectation drives relapse procedures are utilized. The consequences of straight relapse are produce better forecast performed on the informational collection [22].

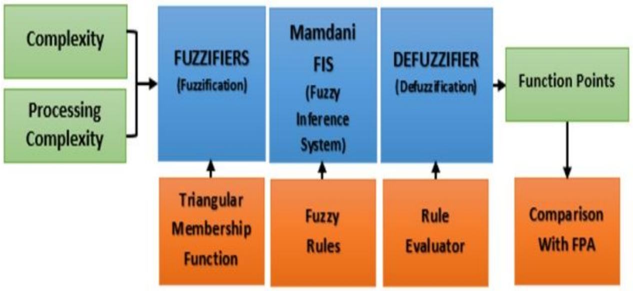
Over 33,000 distinct examinations directed by Boetticher and utilizing NN on investigational information gathered from isolated corporated areas. Analyses utilizes various factors (jargon, multifaceted nature, size and article) to programming the exertion, utilizing NN. [23] Later work of Boetticher, embraced base up a way to deal with lead arrangements of NN probes information from the two separate areas. In base up method utilizes information from the items instead of activities. A size based measurement, Source Lines of Code (SLOC) were removed from real program was utilized as main contribution for NN to anticipate venture exertion. Venkatachalam et al. [24-27] applied counterfeit neural system to cost estimation. Neural system can sum up from prepared informational collection. Over a lot of preparing information, neural system learning calculation builds mappings that fits information, and fits beforehand inconspicuous information in sensible way. [28] An investigation directed on imperfections database of Eclipse open source extends by Zimmermann et al. Zimmermann built up a numerical model for abandons forecasts as far as some multifaceted nature measurements related with the source code. Forecast is, sadly, past the point of no return as coding is a piece of the execution stage from venture arranging viewpoint. In the arranging stage Project directors for the most part need to foresee costs including the expense of deformities expulsion as right on time as could reasonably be expected. [29]

A contextual analysis led by Efe and Demirors to examining the effects of programming revamps on venture estimations. During testing stage 152 problems were originated bringing about extra 133 impromptu people days for absconds evacuation for this situation study. Utilizing venture procedures and the executives' apparatuses in their essential structures may not address the issues of the product ventures without considering the extraordinary moves identified with these product ventures. [30] That the current models ought to be adjusted to join the expense of the impromptu improves proposed by Khalid and Yeoh. [31] The relationship among number of programming deserts, profitability, programming size, and endeavors for programming improvement ventures portrayed by Bhardwaj and Rana. Bhardwaj and Rana utilized a different straight relapse strategy on a benchmarking authentic informational collection. This wound up with two straight relapse conditions depicting the online and non-electronic activities. Online activities are conveyed quicker, they proposed that they additionally contended that lower profitability will prompt less imperfections while higher efficiency brought about a higher number of deformities Bhardwaj and Rana recommended that executing non-electronic undertaking with experienced group require less an ideal opportunity for advancement and the vast majority of the deformity can be recognized during unit testing bringing about decreased adjust endeavors. Programming size has a lot of huge effect on complete number of deformity in contrast with endeavors additionally they presumed that.

III. METHODOLOGY

A. Problem Formulation

The information to gauge the product cost estimation in the underlying phases of venture if frequently fragmented, conflicting, unsure and indistinct. A fuzzy technique is progressively adoptable for investigation by regular methodology or when the accessible information is dubious, off base or unclear [2015 Maharishi SDEE utilizing fuzzy logic].


Complexity
Processing Complexity
FUZZIFIERS (Fuzzification)
Mamdani FIS (Fuzzy Inference System)
DEFUZZIFIER (Defuzzification)
Function Points
Triangular Membership Function
Fuzzy Rules
Rule Evaluator
Comparison With FPA

The upsides of fuzzy rationale are joined learning capacity and great speculation got. The principle advantage of this methodology is it has acceptable interpretability rules. The exertion anticipated utilizing nine principles will be contrasted and Albrecht Dataset.

FIGURE.3 FUZZY FUNCTIONAL POINT ANALYSIS USING FUZZY LOGIC

$$\mu_A(x) = \begin{cases} 0 & if\ x \le a \\ \frac{x-a}{b-a} & if\ a \le x \le b \\ \frac{c-x}{c-b} & if\ b \le x \le c \\ 0 & if\ x \ge c \end{cases}$$

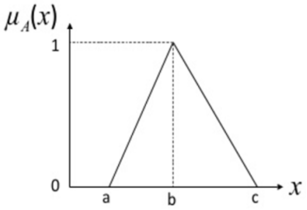


FIGURE.4 MEMBERSHIP FUNCTION (X)

A. Fuzzy Functional Point Analysis using Fuzzy Logic

To sort this problem a new method should be figured out by combining the concept of fuzzy knowledge with FPA. According to [32], in Fuzzy Functional Point Analysis (FFPA), the fuzzy logic technique is used to estimate size of software while the three-sided fuzzy figures are utilized to denote the language definitions in FPA [19]. This allows the classic intervals by fuzzy sets; this model helps to interpret linguistic values as done by human beings and with gradual transition from linguistic term to another adjacent in a more feasible manner. The stages include change of complexity matrices, fuzzification of the language variables and defuzzification. Fig-3 illustrates suggested model.

The linguistic variables are treated as three-sided fuzzy numbers (TFN) where TFN (a, m, β) and, a <m< β Fig-4 illustrates.

New language variable is created by TFN (a, m, k), high, very high, where k is +ive integer. When average and low are given, creates high variables and when case low, high and average are given, creates very high variable. The new linguistic variables help to deal superior with the larger systems [32]. Fuzzy based cost estimation technique estimates findings keys to the annoyed problems under the fuzzy set theory.

B. Proposed Technique

Soft Computing/Machine learning is a range of research that arrangements with genuine issues in a more powerful manner, along these lines giving more exact outcomes. This proposed work depends on utilizing Fuzzy Logic (FL) based method to foresee endeavors to be spent on a given programming advancement venture. Fig-3 demonstrates the proposed display utilized for estimation in light of FL. The fuzzy inference system that is suggested in this exploration work depends on Mamdani framework. The model requires two input parameters viz. Complexity (COM) and Processing Complexity (PCOM). There is one output parameters named Fuzzy Function Point (FuzzyFP) is used for the estimate of software project efforts. Table-III depicts the linguistic variables associated with various fuzzy input/output parameters.

TABLE.3 INPUT VARIABLES AND THEIR DESCRIPTORS

Input	Descriptor	Variable
Complexity	Low	Lo
	Medium	Me
	High	Hi
Processing Complexity	Simple	Si
	Average	Av
	Complex	Co

The output is project effort estimation measured as function points.

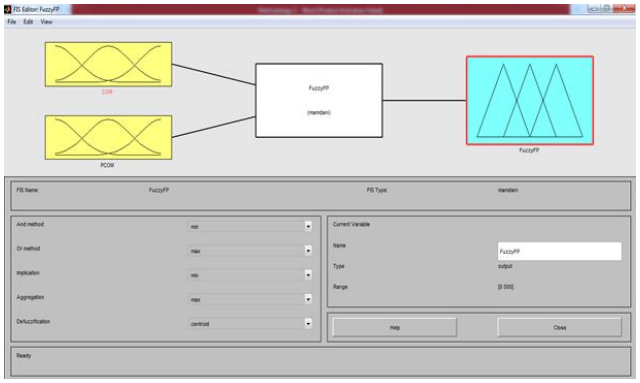
TABLE.4 OUTPUT VARIABLE AND ITS DESCRIPTOR

Output	Descriptor	Variable
	Very Large	VL
	Large	L
Fuzzy Function Points	Very Small	VS
	Small	S
	Medium	M

Based on above linguistic variables, these input parameters are applied with nine (09) fuzzy rules. The fuzzy rules are defined as below:

1. If (Processing Complexity is Simple) and (Complexity is Low) THEN (Effort is Very Small).
2. IF (Processing Complexity is Average) and (Complexity is Low) THEN (Effort is small).
3. IF (Complexity is Low) and (Processing Complexity is Complex) THEN (Effort is Medium).
4. IF (Processing Complexity is Simple) and (Complexity is Medium) THEN (Effort is small).
5. IF (Processing Complexity is Average) and (Complexity is Medium) THEN (Effort's Medium).
6. IF (Processing Complexity is Complex) and (Complexity is Medium) THEN (Effort is Large).
7. IF (Processing Complexity is Simple) and (Complexity is high) THEN (Effort is Medium).
8. IF (Processing Complexity is Average) and (Complexity is high) THEN (Effort is Large).
9. IF (Processing Complexity is Complex) and (Complexity is high) THEN (Effort is Very Large).

The model is capable of utilizing two input factors and apply predefined fuzzy rule base to get an accurate prediction of fuzzy function points and software efforts. The results thus produced are compared with Function Points. The trouble with FP is that when applied to the records from Albrecht dataset, it tends to misinterpret, both over as well as under. Whereas, the proposed model when applied to the same dataset produce results that are very much aligned with the actual results given with the records. Fig-5 shows a screenshot of proposed FIS developed using Matlab 2014a.



FIS Name: FuzzyFP
FIS Type: mamdani

And method: min
Or method: max
Implication: min
Aggregation: max
Defuzzification: centroid

Current Variable
Name: FuzzyFP
Type: output
Range: [0 500]

Ready

FIGURE.5 MAMDANI FIS DEVELOPED USING MATLAB.

IV. Experiments and Results

Extensive experimentation has been done to assert the suitability of the proposed methodology as compared to current methodologies found in the literature. Vast amount of data available through the dataset makes it relatively straightforward to experimentally analyze various available techniques. Table-5 shows the comparison of proposed technique using fuzzy logic and existing Function Point Analysis technique for various software projects chosen from the selected Albrecht dataset. The results are also depicted using table.

TABLE.5 COMPARISON OF EFFORT ESTIMATION RESULTS IN MRE

P#	COM	PCOM	FP	FuzzyFP	MRE
1	39	0.80	199	191.00	0.0402
2	53	0.80	205	187.00	0.0878
3	40	0.95	209	208.00	0.0048
4	63	0.95	224	210.00	0.0625
5	42	1.10	260	247.00	0.0500
6	55	1.00	283	264.00	0.0671
7	54	1.10	289	279.00	0.0346
8	77	1.10	400	361.50	0.0963
9	96	0.85	417	384.00	0.0791
10	109	0.80	428	398.00	0.0701
11	89	0.90	431	404.00	0.0626
12	79	1.05	500	456.00	0.0880
13	102	1.10	512	484.00	0.0547
14	99	1.05	512	480.00	0.0625
15	116	1.10	606	605.00	0.0017
16	139	0.95	680	672.00	0.0118
17	138	0.85	682	678.00	0.0059
18	140	1.00	694	669.00	0.0360
19	132	1.15	759	750.00	0.0119
20	135	1.15	794	777.00	0.0214
21	177	1.15	1235	1144.00	0.0737
22	241	1.20	1572	1475.00	0.0617
23	310	1.01	1750	1720.00	0.0171
24	397	1.01	1902	1890.00	0.0063

The proposed technique succeeds at giving accurate results Table-6.

TABLE.6 COMPARISON THE VALUES OF MODEL EVALUATION MEASURES

Mean	**0.0462**
Standard Deviation	**0.0302**
Standard Error	**0.0062**
MMRE	**0.0019**

The data evaluated using MRE, Mean, Standard Deviation, Standard Error and MMRE have verified to be more correct in estimating software cost. The experimental results are minimizing the estimated cost as compare to FPA. The testing coverage at this time is more effectively indicates the reliability analysis.

The experimental results are minimizing the estimated cost as compare to FPA. The testing coverage at this time is 99\% more effectively indicates the reliability analysis.

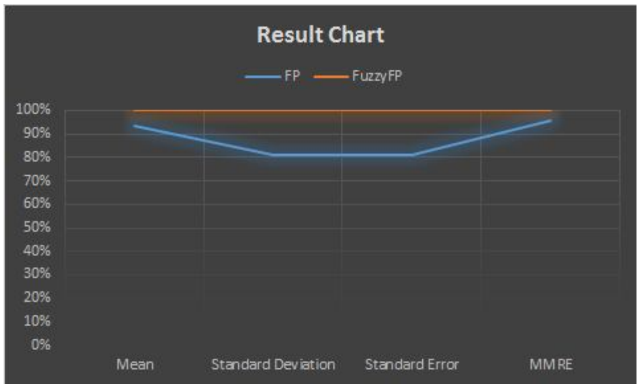


Figure.6 Graphical representation of Results in Mean, Standard Deviation, Standard Error and MMRE

V. DISCUSSION

Software development effort estimation remains an area of research since long. Even the well-established and widely used technique named Function Point fails to show acceptable accuracy. Results show that there is a need to enhance the techniques to achieve acceptable accuracy. Fuzzy logic based techniques show a great promise while calculating the estimates for efforts on software development. The result graphs show that the proposed technique result curve runs relatively very much closer to the actual curve. The results proposed technique encourage us to announce the suitability of the proposed technique for estimation of software development efforts in software companies.

The analysis of the results shows more accurate results using fuzzy logic with FPA in software development while compared to the earlier methodologies. Hence, the improvements to our model are actually valuable and given improved performance.

IV. CONCLUSION

By research, it has been proven that by combing these two methods a new method can be developed that would prove to be an effective method system.

ACKNOWLEDGMENT

Syed Zaffar Iqbal is working as Program Coordinaor at Alhamd Islamic University, Quetta, Pakistan and research scholor. His research areas consist of software quality assurance, software engineering, programming, development. He is supervisor of the final year students' projects at his university and ICT R&D sponsored projects. He is also writer of intermediate level computer science books. Email: zis.shah@gmail.com

REFERENCES

[1] Tülin Erçelebi Ayyıldız and Hasan Can Terzi, "Case Study on Software Effort Estimation", International Journal of Information and Electronics Engineering, Vol. 7, No. 3, May 2017.

[2] Mandeep Kaur, "A fuzzy logic approach to software development effort estimation", International Journal of Advanced Research and Development, Volume 3; Issue 1; Page No. 125-127; January 2018.

[3] Amardeep Singh and Ravneet Preet Singh Bedi, "Software Cost Estimation using Fuzzy Logic Technique", Indian Journal Of Science And Technology, Vol 10 (3) | Jan 2017.

[4] Arun Kumar Marandi and Danish Ali Khan, "Software Quality Improvement and Cost Estimation using Fuzzy Logic Technique", International Journal Of Applied Engineering Research, Volume 12, pp. 5433-5440, Number 16 (2017).

[5] Leung, H., and Fan, Z. (2015). Software Cost Estimation. Cost Estimation, 257-269. doi:10.1002/9781118802342.ch15

[6] Syed Zaffar Iqbal, Idrees, M., Sana, A. B., and Khan, N. (2017). Comparative Analysis of Common Software Cost Estimation Modelling Techniques. Mathematical Modelling and Applications,2(3), 33. Vol. 2, No. 3, 2017, pp. 33-39.

[7] Jeng, B., Yeh, D., Wang, D., Chu, S. L., and Chen, C. M. (2011). A Specific Effort Estimation Method Using Function Point. Journal of Information Science and Engineering, 27(4), pp. 1363 - 1376.

[8] Ali Moftah, Abu Baker (2014) Comparative analysis between FPA and Cocomo Techniques for software cost estimation. Masters thesis, Universiti Tun Hussein Onn Malaysia, July 2014.

[9] Basavaraj, M., and Shet, K. (n.d.). Software Estimation using Function Point Analysis: Difficulties and Research Challenges. Innovations and Advanced Techniques in Computer and Information Sciences and Engineering,111-116.

[10] Albrecht, Allan J., and John E. Gaffney. "Software function, source lines of code, and development effort prediction: a software science validation." IEEE transactions on software engineering 6 (1983): 639-648.

[11] Verner, J. (2002). Function Point Analysis. Encyclopedia of Software Engineering. doi:10.1002/0471028959.sof137

[12] Ahmed, M. A., Saliu, M. O., and Alghamdi, J. (2005). Adaptive fuzzy logic-based framework for software development effort prediction. Information and Software Technology,47(1), 31-48. doi:10.1016/j.infsof.2004.05.004

[13] Chandra, V. (2014). Software Effort Estimation: A Fuzzy Logic Approach. International Journal of Computer Applications,103(9), 39-43. doi:10.5120/18106-9210

[14] Grimstad, S., Jorgensen, M., Østvold, K.M.: Software Effort Estimation Terminology: The tower of Babel. Information and Software Technology 48, 302–310 (2006)

[15] Boraso, M., Montangero, C., Sedehi, H.: Software Cost Estimation: an experimental study of model performances, Technical Report: TR-96-22, University of Pisa, Italy

[16] Wieczorek, I., Ruhe, M.: How Valuable is company-specific Data Compared to multicompany Data for Software Cost Estimation? In: METRICS 2002. Proceedings of the Eighth IEEE Symposium on Software Metrics (2002)

[17] Adalier, Oktay, Aybars Uğur, Serdar Korukoğlu, and Kadir Ertaş. "A new regression based software cost estimation model using power values." In International Conference on Intelligent Data Engineering and Automated Learning, pp. 326-334. Springer, Berlin, Heidelberg, 2007.

[18] Suwanjang, Hathaichanok, and Nakornthip Prompoon. "Framework for Developing Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using an Analogy Estimation Method." International Journal of Computer and Communication Engineering 1, no. 2 (2012): 129.

[19] Attarzadeh, Siew Hock Ow and Iman. "Proposing a new software cost estimation model based on artificial neural networks." In Computer Engineering and Technology (ICCET), 2010 2nd International Conference on, vol. 3, pp. V3-487. IEEE, 2010.

[20] G. E. Wittig., "Using artificial neural Networks and Function Points to Estimate 4GL Software Development Effort", Australian Journal of Info. System, 1994.

[21] Samson, B., etal. "Software cost estimation using an Albus perceptron" Journal of Info and Software, 39, pp. 55-60, 1997.

[22] Boetticher, G. D., “An Assessment of Metric Contribution in the Construction of a Neural Network-Based Effort Estimator”, in Proceedings of 2nd International Workshop on Soft Computing Applied to Software Engineering, 2001.

[23] Venkatachalam, A. R., “Software cost estimation using artificial neural networks”, in Proceedings of the 1993 International Joint Conference on Neural Networks, pp. 987-990, 1993.

[24] Schofield C., Non Algorithmic Effort Estimation Techniques, Technical Reports, Department of Computing, Bournemouth University, England, March 1998.

[25] Jingzhou, L., Guenther, R., “Analysis of attribute weighting heuristics for analogy-based software effort estimation method AQUA+”, in Proceedings Of Empirical Software Engineering Journal (2008), Vol. 13, No. 1, pp. 63-96, Febuary. 2008.

[26] Liu. H , Yu, L., Toward Integrating Feature Selection Algorithms For Classification And Clustering, IEEE Transactions On Knowledge And Data Engineering, Vol. 17, No. 4, pp. 491-502, April. 2005.

[27] Chiu N. H., Huang S. J., “The adjusted analogy-based software effort estimation based on similarity distances”. Journal Of Systems And Software, Vol. 25, pp. 628-640, 2007.

[28] T. Zimmermann, R. Premraj and A. Zeller, "Predicting Defects for Eclipse," Predictor Models In Software Engineering, 2007. PROMISE' 07: ICSE Workshops 2007. International Workshop On, Minneapolis, MN, 2007, pp. 9_9. doi: 10.1109/PROMISE.2007.10

[29] P. Efe and O. Demirors, "Applying EVM in a Software Company: Benefits and Difficulties," 2013 39th Euromicro Conference On Software Engineering And Advanced Applications, Santander,2013,pp.333-340.doi:0.1109/SEAA.2013.55

[30] T. A. Khalid and E. T. Yeoh, "Controlling software cost using fuzzy Quality based EVM," Computing, Control, Networking, Electronics And Embedded Systems Engineering (ICCNEEE), 2015 International Conference On, Khartoum, 2015, pp. 275_280.

[31] M. Bhardwaj and A. Rana, “Estimation of Testing and Rework Efforts for Software Development Projects,” In Asian Journal Of Computer Science And Information Technology, vol.5,no.5,pp.33-37,May2015. doi:10.15520/ajcsit.v5i5.15.T

[32] Kumar, S., Rastogi, R., and Nag, R. (2018). Limitations Of Function Points Analysis In Multimedia Software/Application Estimation. Advances In Intelligent Systems And Computing Software Engineering, 383_392.