A Study on Effort Estimation of Software Development Using Fuzzy Logic
Sana1 Shahnawaz Anwer2
1 M.Tech Scholar, Department of Computer Science & Engineering,
Al-Falah School of Engineering & Technology, Dhauj, Faridabad (Haryana), INDIA
2 Tata Consultancy Services, Kolkata, India

Abstract--- To develop any software project we have to face a lot of challenges to meet the competitive demand of the customer. Developer abilities, size, accuracy, complexity, reliability are the major challenges. The data for a software project is unclear, uncertain, inconsistent and often incomplete at the early stage of the development. Due to this reason the development process become more complicated. Overcome from all of these situations we suggest use of expert system because they give more accuracy than the formal estimation techniques. In this paper we study how fuzzy logic is beneficial to be used for the estimation of a software project development.

I. INTRODUCTION
For developing software project estimation need estimation techniques and estimation tools. It is vital aspects that guide the planning of a software project. The manager of the project must be able to analyze the problem and respond to the corresponding situation quickly. From the last 30 years several techniques were proposed for the effort and cost estimation.

II. ESTIMATION TECHNIQUES
Software effort estimation is divided into many techniques. Some of the general categories techniques are:
1) Expert judgment [EJ] which is based on brain Storming of one or more experts who have experience in similar projects [9,5,7].
2) Machine learning is used as a complement or alternative to the expert judgment and algorithmic model techniques [7].
3) Analogy is a technique based on comparing previous similar activities and analyzing the most relevant project.
4) Decomposition technique tries to make granular list of initially planned task. It is based on top-down estimation [9].
5) Statistical is a set of related mathematical equation to define alternative scenario by changing values of a set of fixed co-efficient.

Estimation strategy has criteria for software development effort, expert estimation is one of the best in this a person is appointed as an expert on the particular task [5]. Estimation is the oldest and more mature aspect of software metrics towards rigorous software management. There is mainly two types of estimation techniques Algorithmic and Non-algorithmic models [1,7].

A. Algorithmic Models
This model generally represents the relationship effort and characteristic between one or more project. Cost drivers are used in this model. The prediction of this model generally rely upon accurate estimate of either size of software in terms of LOC (line of code), number of pages, number of user screens, complexity, number of links, interfaces, etc. one of the best example of algorithmic model is COCOMO model [1,2,3,5,7]. Boehm was the first researcher who consider software engineering economically and came up with a model known as COCOMO-81 as it is discovered in the year 1981 [1,2,3]. This model was developed from the analysis of 63 software projects. But still there is some limitation in the model so, Bohem further projected three levels of model known as Basic COCOMO, Intermediate COCOMO, and Detailed COCOMO [1,2,3]. Algorithmic model sometimes not able to present suitable solution because these models are often unable to capture complex set of relationship [1]. These are some disadvantages in this model such as they are not flexible to adopt new environment, they cannot handle categorical data and the important one they have lack of reasoning ability. Due to this lack of characteristic a number of studies has explored non-algorithmic model also known as fuzzy logic.

B. Non-Algorithmic Model
The non-algorithmic techniques are based on soft computing these include artificial neural network (ANN), evolutionary computation, and fuzzy logic models, case based reasoning, combinational model, etc. this technique came up in 1990’s, and turns whole attention towards itself. This methodology can easily handle real life ambiguous situation due to having the characteristic of flexible processing. The ANN has ability to learn from previous data due to this characteristic it is used in effort estimation. It can easily handle the complex relationship between dependent (effort) and independent (cost driver) variables [2]. Jack Ryder investigated application of fuzzy logic model with most known model of software effort estimation for the effort prediction they are COCOMO and function point model [3]. As we know that Boehm has given three COCOMO model as it is widely used than the other two. The intermediate COCOMO model has relatively high estimation accuracy than the basic and detailed COCOMO version.

III. COCOMO WITH FUZZY LOGIC
Two researcher Fei and Lui [1,2] came with a model known as f-COCOMO. In this model they applied properties of fuzzy logic to COCOMO model for the software effort estimation process. But in their study there is no comparison of the between f-COCOMO and any other effort estimation model which led their study to an unknown stage [2]. Another researcher Roger [2] proposed a model called fuzzy COCOMO which has ability to adopt fuzzy logic method to
model the uncertainty of software effort driver. But this model also fails because the effectiveness of this model is not given. Later Idri [1,2] proposed a model with trapezoid shaped membership function for the fuzzy COCOMO model. Xue and Khoshgoftaar [2] in 2004 proposed a fuzzy identification effort estimation model technique which deals with linguistic effort drive and automatically generate membership functions and rules by upside database of COCOMO-81 [2].

IV. FUZZY IDENTIFICATION

A fuzzy model is used when the system are not suitable for analysis by conventional approach. We can also say that when the given data of a software project is unclear, uncertain and vague we use fuzzy model for the effort estimation of the software project. There are some rules for the effort estimation with the help of fuzzy logic. Rules are those which map an input space to an output space using a list of if-then statement [3]. For writing rules it is compulsory that the input and output of the system must be identical [2,3]. There are some steps to be followed:-

1) Select a sugeno type fuzzy interface system.
2) Define the input and output variable.
3) Set the type of the membership function for input variable.
4) Set the type of membership function as linear output variable.
5) The data is now translated into a set of if-then rules written in Rule editor.
6) A certain model structure is created, and parameters of input and output variables can be turned to get the desired output.

V. FUZZY EVALUATION CRITERIA

The concept of fuzzy logic was conceived by Lofti zadeh in 1986 [1], a professor at the California at Barkley and presented not as a control methodology, but as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership. It is a technique which deals with imprecision and information granularity [2]. The fuzzy reasoning has three main components are: fuzzification, interface from fuzzy rules and defuzzification. The main parameter for assessing the predictive power of effort prediction models are the following:

1) MRE (magnitude of relative error)
2) MMRE (mean magnitude of relative error)
3) MdMRE (medium magnitude of relative error)
4) Pred(n) (prediction at level n)

\[
\text{MRE}_i = \frac{|\text{Actual effort}_i - \text{predicted effort}_i|}{\text{Actual effort}_i}
\]

\[
\text{MMRE} = \frac{1}{n} \sum_{i=1}^{n} \text{MRE}_i
\]

VI. RESULTS

Some of the experiments are done by taking data from COCOMO datasheet [10].

<table>
<thead>
<tr>
<th>Model</th>
<th>VAF %</th>
<th>MARE %</th>
<th>VARE %</th>
<th>Mean BRE</th>
<th>Pred(25)%</th>
</tr>
</thead>
<tbody>
<tr>
<td>COCOMO</td>
<td>87.16</td>
<td>21.41</td>
<td>5.48</td>
<td>0.25</td>
<td>72</td>
</tr>
<tr>
<td>Triangular MF</td>
<td>95.83</td>
<td>18.63</td>
<td>4.35</td>
<td>0.23</td>
<td>68</td>
</tr>
<tr>
<td>GBell MF</td>
<td>92.25</td>
<td>20.35</td>
<td>4.24</td>
<td>0.26</td>
<td>62</td>
</tr>
</tbody>
</table>

Table 1: Effort Assessment Using Cocomo, Triangular And Gbell Membership Function

<table>
<thead>
<tr>
<th>Model</th>
<th>MRE</th>
<th>MMRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>COCOMO</td>
<td>25.20</td>
<td>32.65</td>
</tr>
<tr>
<td>FIS with 3MFs</td>
<td>64.26</td>
<td></td>
</tr>
<tr>
<td>FIS with 5MFs</td>
<td>41.06</td>
<td></td>
</tr>
<tr>
<td>FIS with 7MFs</td>
<td>38.38</td>
<td></td>
</tr>
<tr>
<td>FIS with 11MFs</td>
<td>34.15</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Effort Assessment Using Cocomo and Increasing Membership Functions

From the experiment in table I it is found that triangular membership function model is giving better result for estimation criteria in effort estimation than COCOMO and a Bell MF model. While from table ii we found with the increase in membership function the effort estimation also improved. While from table iii haussian membership function model gives better result than COCOMO and Trapezoidal MF models in effort estimation.

VII. EVALUATION

As we know that effort estimation of software is very important. So from the study I found that COCOMO model is the best one technique which is used foe estimation. But,
it is clear that we have not proper data at the early stage of software development, because of this we need fuzzy logic. Using fuzzy COCOMO for effort estimation is a better option because this model has ability to adopt fuzzy logic method, but still it fails sometimes. I found that using expert minds with fuzzy COCOMO will lead to a complete effort estimation technique for the software development. Because according to my point of view this model has complete features that is needed for effort estimation. The fuzzy model help in solving uncertainty, vaugness etc. While the expert judgment helps in solving the other entire problem because they have already worked on similar type of project. We can say that it provides transparency at the preliminary stage.

VIII. CONCLUSION

From the study we found that there is still uncertainty in prediction of techniques for different problem. There is a need to develop a single soft computing model which handles tolerance of imprecision in the input at the preliminary phases. The other one is expert estimation which is a powerful strategy for the effort estimation of a software development projects. We need a technique which allows total transparency in the prediction system by predicting the results through rules and other ways.

REFERENCES

[9] Luigi Buglione and Christof Ebert “Estimation Tools and Techniques” software technology
[12] A.Idri and A.Abran “Confidence in software cost estimation result based on MMRE and PRED”