2008 IEEE International Conference on Research, Innovation and Vision for the Future in Computing & Communication Technologies
University of Science – Vietnam National University, Ho Chi Minh City, July 13-17, 2008

Innovative project resource allocation: estimating by analogy based on linguistic values and fuzzy inference systems

Dr.S. Mohammadi
Department of Industrial Engineering, IT grou.
K N Toosi University of Technology
Tehran, Iran
Smohammadi40@yahoo.com

K. K. Dehkordy
Department of Industrial Engineering, IT grou.
K N Toosi University of Technology
Tehran, Iran
kimiJava@yahoo.com

Abstract— Estimation models in software engineering are used to predict some important attributes of future entities such as software development cost, project resource allocation, and software reliability and programmers productivity. Among these models, those estimating software effort have motivated considerable research in recent years. The prediction procedure used by these software-effort models can be based on a mathematical function or other techniques such as analogy based reasoning, neural networks, regression trees, and rule induction models. Estimation by analogy is one of the most attractive techniques in the resource allocation estimation field. However, the procedure used in estimation by analogy is not yet able to handle correctly linguistic values (categorical data) such as 'very low', 'low' and 'high'. In this paper, we propose a new approach based on reasoning by analogy, fuzzy logic and linguistic quantifiers to estimate project resource allocation when it is described either by numerical or linguistic values; this approach is referred to as Fuzzy Analogy. This paper also presents an empirical validation of our approach based on the COCOMO'81 dataset.

Keywords- resource allocation, Fuzzy, estimation by analogy, COCOMO'8

I. INTRODUCTION

What is project resource allocation? A wide variety of engineering and business activities are structured as projects: they have tasks, they require resources of various types, and they are constrained in both time and budget. Many types of projects are also subject to considerable uncertainty – uncertainty in time to complete specific tasks, in the resource requirements of those tasks, and in whether or not the resource will produce an outcome judged to be "successful." Programs often have many such projects, and the program managers face critical decisions about what projects to pursue, how much time and money to invest in each one, and how to reach decisions to terminate individual projects (or parts of projects) if they do not seem promising.[4]

The resource allocation problem in its generic form has two variants. The first is to obtain best quality software for a given total resource. That is, the total resource to be spent on the entire Quality Control (QC) process is given, and the goal is to allocate this resource among the different QC tasks in the process such that maximum number of defects is detected by the QC process. [3]

Resource allocation problems are widely encountered in industrial as well as academic practices. Owing to its combinatorial complexity, large-scale resource allocation often proves mathematically intractable. [1]

Developing computing systems that are self-configuring and self-optimizing in unpredictable environments is becoming the central concern of industrial giants such as IBM, HP, and Sun. For example, the recently popular "utility computing paradigm" describes the environment where system components (agents) "purchase" resources from each other to respond to local spikes in demand. [2]

While this vision is very promising, only small steps have been made so far toward achieving it. Developing fuzzy based estimation by analogy is the key issue in well allocating the project resources.

For the quality of the final software we will use the commonly used measure of delivered defect density - the number of defects present in the final product normalized by the size of the product. One of the main objectives of a project is to achieve the desired quality goal with least amount of resources. [3]

Non-linear increasing functions is been used to represent the resources, and use them to derive expressions for the resource estimate of each (QC) stage, once the defect injection rates are known. The problem of allocating resources then reduces to a optimal resource allocation problem, which we solve using sequential quadratic programming. [3]

We are about to discuss the problem of resource allocation in IT projects by using fuzzy techniques as the most convenient tool. An estimating method based of fuzzy analogy is been proven to do the problem as our entire work. And the model also is tried to be validated using the COCOMO'81 dataset as the work carried out by Ali I., et al., [5].

The paper is organized as follows. A review of the literature work on the subject will be referred to in section II. The section III consists of brief information about the fuzzy

ISSN 1242-5125 ENST S (Paris) (France 1983-9999)

logic and FIS, so that; the usage of them in our work will be clearly announced. In section IV, proposition of the problem is mentioned. Problem solution, fuzzy based estimation by analogy is discussed in section V in detail. In section VI validation of the model using the COCOMO'81 dataset is presented. And finally our conclusion is lugged in section VII as the main result of the work.

II. LITERATURE REVIEW

The "management-oriented" literature contains considerable evidence of concern with uncertainty and risk in managing projects.

Much of this literature is very qualitative, focusing on the

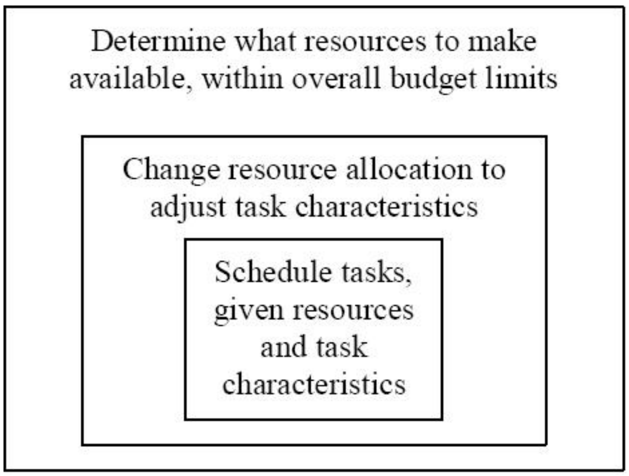

Figure 1 - A perspective on problem levels" within

process of managing to ameliorate risks. On the other hand, the operations research literature is replete with articles on project scheduling, but very few of these articles deal with uncertainty. We believe this "disconnect" is the result of different collections of researchers defining problems at different levels of a hierarchy in a way that obscures their relationship to one another. Figure 1 portrays a nested view of the problem that highlights the potential connections between project scheduling and the larger issues of resource allocation and risk management. The innermost box represents the scheduling problem, where it is assumed that the available resources are fixed and specified, and the characteristics of individual tasks (duration and resource use) are given. With those inputs specified, the scheduling problem is to determine a likely operational schedule for the tasks that "fits" within the available resources. The project scheduling literature focuses on this "inner" problem and generally does not deal with uncertainty.

PERT, developed in the 1950's, represented the first consideration of uncertainty in project scheduling, focusing on uncertain task durations. This technique allowed an estimate of the overall duration of a project to be constructed. However, PERT has major weaknesses. It does not consider constraints on available resources and it assumes that all tasks will be completed successfully. Using a PERT framework, Valadares Tavares, et al. (1998) also consider uncertainty in the resource requirements of individual tasks and the resulting effect on overall project cost, but they do not incorporate resource constraints. [4]

Experience has shown that there does not exist a 'best' prediction technique outperforming all the others in every situation. Indeed, Shepperd et al., Niessink and Van Vliet found that estimation by analogy generated better results than stepwise regression [9, 10, and 11]. However, Briand et al., Stensrud and Myrtveit reported opposite results [7, 8]. Recent research has been initiated to explain the relationship between different properties of historical projects dataset (size, number of attributes, presence of outliers…) and the accuracy of a prediction system [12]. This work deals with an important limitation of all estimation techniques, which arises when software projects are described using categorical data (nominal or ordinal scale) such as 'very low', 'low' and 'high'. These qualifications are called "linguistic values" in fuzzy logic terminology. Building cost estimation models based on linguistic values is a serious challenge for the software cost estimation community. Recently, Angelis et al. [6] were the first to propose the use of categorical regression procedure (CATREG) to build cost estimation models when software projects are described by categorical data. This procedure quantifies categorical attributes by assigning numerical values to their categories in order to produce an optimal linear regression equation for the transformed variables. This approach has the following limitations: It replaces each linguistic value by one numerical value. This is based on the assumption that a linguistic value can always be defined without vagueness, imprecision and uncertainty. Unfortunately, this is not often the case. Indeed, linguistic values come from human judgements that are often vague, imprecise and uncertain. For example, let us assume that the experience of programmers is measured by three linguistic values: 'low', 'nominal' and 'high'. Most often the meaning of these values is not defined precisely and consequently we cannot represent them by individual numerical values. There is no natural interpretation of the numerical values assigned by this approach. It assigns numerical quantities to linguistic values in order to produce an optimal linear regression equation whereas the initial relation between resource and cost drivers may be non-linear. A more comprehensive approach to deal with linguistic values is by using "fuzzy set" theory. Consequently, the purpose of this paper is to provide a new approach based on analogy and fuzzy logic to estimate resource when software projects are described either by numerical or linguistic values. [5]

Pankaj Jalote and Bijendra Vishal (2003) proposed a model for the cost of QC process and then view the resource allocation among different QC stages as an optimization problem. We solve this optimization problem using non-linear optimization technique of Sequential Quadratic Programming. They have used defect density as the metric for measuring the quality of software at any stage and have used the defect injection and removal stages as the basic model. We assume that the cost of defect removal increases with the latency between injection and removal and also with the defect removal efficiency. Using these two assumptions we have modeled the problem as an optimal resource allocation

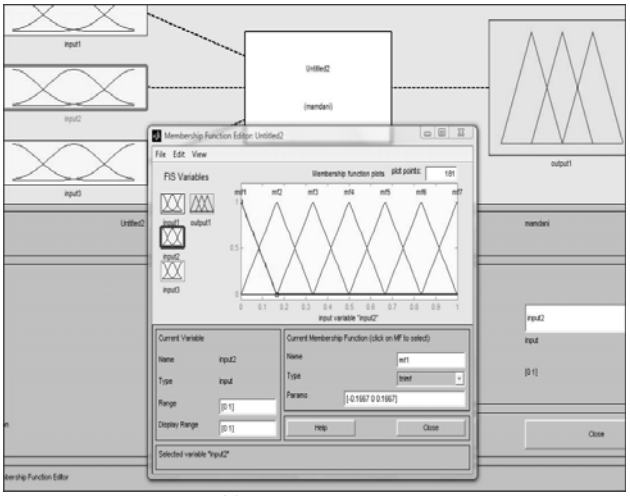

Figure 2 – representation of fuzzy numbers as inputs and output, and application of the fuzzy operation on fuzzy input values.

problem. They have also developed the software to determine the optimal resource allocation among different QC stage. [15]

Tzy, et al., (2006) proposed a method based on the fuzzy multi-criteria decision model for the evaluation of new IT/IS investments. We compare prior studies to compile a list of evaluation criteria. The final list of criteria is obtained using a Delphi approach. This model provides the flexibility to change the weights of the criteria, and uses those weights to reflect special concerns about IT/IS, as well as organizational characteristics such as cultural, and contextual issues. It can also be used as an analytical tool to improve the project's contract to attain to the ideal goal. [17]

III. FUZZY LOGIC AND FIS

In this section, some basic notions of the area of fuzzy theory that have been defined by Kaufmann and Gupta (1985) and Zimermann (1996) are introduced. As defined by Zadeh [25], Soft Computing is composed of three criteria part of human nature: tolerance of imprecision, learning, and uncertainty.

A. Definitions

Definition1: Let R be the space of real numbers. A fuzzy set Ã is a set of ordered pairs $\{(x, \mu_{\tilde{A}}(x)) \mid x \in R\}$, where $\mu_{\tilde{A}}(x) : R \to [0,1]$ and is upper semi continuous. Function $\mu_{\tilde{A}}(x)$ is called membership function of the fuzzy set.

Definition2: A convex fuzzy set, Ã, is a fuzzy set in which:

$$\forall\, x, y \in R,\ \forall \lambda \in [0,1],$$

$$\mu_{\tilde{A}}(\lambda x + (1-\lambda) y) \geq \min[\,\mu_{\tilde{A}}(x), \mu_{\tilde{A}}(y)\,]$$

Definition3: A fuzzy set Ã is called positive if its membership function is such that

$$\mu_{\tilde{A}}(x) = 0\ , \forall x \leq 0$$

Definition4: Trapezoidal Fuzzy Number (TFN) is a convex fuzzy set which is defined as: [18, 19]

$$\tilde{A} = (x, \mu(x)) \text{ where}$$

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & x \leq a \\ \dfrac{x-a}{b-a} & a < x \leq b \\ 1 & b < x \leq c \\ \dfrac{x-d}{c-d} & c < x \leq d \\ 0 & x < d \end{cases}$$

B. Operations on TFNs

Chen and Hwang (1992) and Dubois and Prade (1988) have been defined a number of operations can be performed on TFNs. The following are employed operations in the development of the proposed method: [20, 21]

Let $\tilde{A} = (a_1, b_1, c_1, d_1)$ and $\tilde{B} = (a_2, b_2, c_2, d_2)$ be any two TFNs, then:

$$\tilde{A} \oplus \tilde{B} = (a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2)$$

$$\tilde{A} \ominus \tilde{B} = (a_1 - d_2, b_1 - c_2, c_1 - b_2, d_1 - a_2)$$

$$M\tilde{A}X(\tilde{A}, \tilde{B}) = (\max(a_1, a_2), \max(b_1, b_2), \max(c_1, c_2), \max(d_1, d_2))$$

$$M\tilde{I}N(\tilde{A}, \tilde{B}) = (\min(a_1, a_2), \min(b_1, b_2), \min(c_1, c_2), \min(d_1, d_2))$$

C. Fuzzy inference systems (FIS)

According to Stefan, S. et al., (2007) the usage of fuzzy inference systems in modern technologies specially IT researches is been a complex structure of an agent acting based on what it percepts (its input) and its embedded "rule-base" dictionary of fuzzy rules, and produces the fuzzy (or somewhere non-fuzzy) output as the result of the FIS. The structure of a formal FIS is shown in figure 2.

Stefan, S., et al., use the FIS as the reasoning machine for their work on trust evaluation in multi agent systems. Using FIS the input values to the system should be in form of fuzzy sets, so the input values must be fuzzified. The FIS acts with fuzzy numbers as well the fuzzy operations will be used by such systems. Figure 3 shows the way fuzzified values are represented and how the fuzzy operations are applied to them. [22]

Discussion about the Fuzzy Rule Base used by the FIS, leads to the pure definition of a rule-base agent that is: "an agent who acts dependent on what its knowledge-base of specific rules said about the specific input data". More details can be found in [22]. A fuzzy rule base is on rule-base wiz contains the rules as fuzzy rules.

Main usage of FIS is to reasoning situations in controlling systems. They are also useful to be used as perfect decision makers acting with a number of vague parameters and constraints.

Our approach to use FIS in order to manage the resource allocation of an IT project is based on their ability to act perfectly with the imprecise nature of the problem, "the task of allocating the project resources is based on human abilities and his complex process of making decisions".

IV. THE PROBLEM

There have been many detailed definitions for the problem of project resource allocation. It's been meant to be Allocating the project resources to several tasks so that it would be the most perfect allocation" as it's mentioned in this paper. But, what kind of resource allocation method is perfect? And; what is the most perfect one. Resources utilized in a project completion progress are also costy either non-resumption able. Thus, there should be a reasonable method covering the resource allocation process so that the resources don't go short in performance. As its clear enough, our problem is to develop a respective well defined model to allocate the resources of an IT project regarding to the vagueness and uncertainty surrounding the situation, and the limitation of the existing resources.

With this approach, we outlined the detail of the problem as finding the best resource allocation wiz ever applied to a project the same as ours. This is the basic rule of the Estimating by Analogy method.

V. THE SOLUTION: FUZZY BASED ESTIMATION BY ANALOGY

A. Overview

Estimation by analogy technique bases on comparing the existing projects which are completed, to initiating project which's resources are to be allocated. And then, when the most similar project is found, we can imitate the way the allocated their resources to allocate our resources.

Estimation by analogy is essentially a form of Case-Based reasoning which has four steps:

1) *Retrieve the most similar case or cases.*
2) *Reuse the information and knowledge in that case to solve the problem.*
3) *Revise the proposed solution.*
4) *Retain the parts of this experience likely to be useful for future problem solving.*

Fuzzy Analogy is a fuzzification of the classical analogy procedure. It is also composed of three steps: identification of cases, retrieval of similar cases and cases adaptation. Each step is a fuzzification of its equivalent in the classical analogy procedure. In the following subsections, each step will be further detailed.

5) *identification of cases*
6) *retrieval of similar cases*
7) *case adaptation*

B. Identification of cases

The goal of this step is the characterization of all software projects by a set of attributes. Selecting attributes describing accurately software projects is a complex task in the analogy procedure. Indeed, the selection of attributes depends on the objective of the CBR system. In our case, the objective is to estimate the software project resource. Consequently, the attributes must be relevant for the resource estimation task. The problem is to detect the attributes exhibiting a significant relationship with the resource in a given environment. The solution adopted by cost estimation researchers and practitioners is to test the correlation between the resource and all the attributes for which data in the studied environment are available. This solution does not take into account attributes that can affect largely the resource, if they have not yet recorded data. The objective of our Fuzzy Analogy approach is to deal with linguistic values. In the identification step, each software project is described by a set of selected attributes that can be measured by numerical or linguistic values.

The use of fuzzy sets to represent categorical data, such as 'very low' and 'low' mimics in the way in which humans interpret these values and consequently it allows us to deal with vagueness, imprecision and uncertainty in the cases identification step. Another advantage of the Fuzzy Analogy approach is that it takes into account the importance of each selected attribute in the cases identification step.

C. Retrieval of similar cases

This step is based on the choice of a software project similarity measure. This choice is very important since it will influence which analogies are found. In literature, most researchers have used the Euclidean distance when projects are described by numerical data and the equality distance when they are described by linguistic values (categorical data) [24].

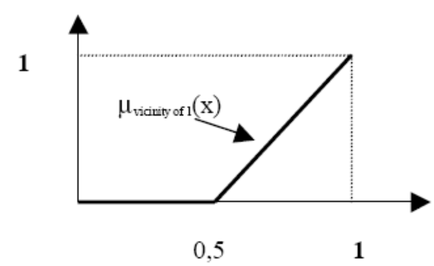

Figure 4 – A possible definition of the value 'vicinity of 1'

These two measures are not suitable when linguistic values are represented by fuzzy sets. Consequently, we have proposed a set of candidate measures for software project similarity to avoid this limitation [12]. These measures evaluate the overall

Similarity of two projects P_1 and P_2, $d\ (P_1,\ P2)$, by combining the individual similarities of P_1 and P_2 associated with the various linguistic variables (attributes) (V_j) describing P_1 and P_2, $d_{vj}(P_1, P_2)$. After an axiomatic validation of some proposed candidate measures for the individual distances d_{v_j} (P_1, P_2), we have retained two measures [13]:

$$d_{v_j}(P_1, P_2) = \begin{cases} \left.\begin{matrix} \max\limits_k \min(\mu_{A_k^j}(P_1), \mu_{A_k^j}(P_2)) \\ max-min\ aggregation \end{matrix}\right\} (1.1) \\ \left.\begin{matrix} \sum\limits_k \mu_{A_k^j}(P_1) \times \mu_{A_k^j}(P_2) \\ sum-product\ aggregation \end{matrix}\right\} (1.2) \end{cases}$$

where V_j are the linguistic variables describing projects P_1 and P_2, A_k^j are the fuzzy sets associated with V_j, and $\mu_{A_k^j}$ are the membership functions representing fuzzy sets A_k^j. To evaluate the overall distance of P_1 and P_2, the individual distances d_{v_j} (P_1, P_2) are aggregated using Regular Increasing Monotone (RIM) linguistic quantifiers such as '*all*', '*most*', '*many*', '*at most* 〈' or '*there exists*'. The choice of the appropriate RIM linguistic quantifier, *Q*, depends on the characteristics and the needs of each environment. *Q* indicates the proportion of individual distances that we feel is necessary for a good evaluation of the overall distance. The overall similarity of P_1 and P_2, $d(P_1, P_2)$ is given by one of the following formulas:

$$d(P_1,P_2)=\begin{cases} all\ of\ (d_{v_j}(P_1,P_2)) \\ most\ of\ (d_{v_j}(P_1,P_2)) \\ many\ of\ (d_{v_j}(P_1,P_2)) \\ \dots \\ there\ exists\ of\ (d_{v_j}(P_1,P_2)) \end{cases}$$

The overall distance, $d(P_1, P_2)$, is calculated by means of the following formula:

$$d(P_1,P_2)=\sum_{j=1}^{M}\left(Q\left(\frac{\sum_{k=1}^{j}u_k}{T}\right)-Q\left(\frac{\sum_{k=1}^{j-1}u_k}{T}\right)\right)\times d_{v_j}(P_1,P_2)$$

D. Case adaptation

The objective of this step is to derive an estimate for the new project by using the known effort values of similar projects. There are two problems here. First, the choice of how many similar projects should be used in the adaptation; Second, how to adapt the chosen analogies in order to generate an estimate for the new project.

Briand et al. have used a single analogy [7]. Angelis and Stamelos have tested a number of analogies in the range of 1 to 10 when studying the calibration of the analogy procedure for the Albrecht's dataset [6]. Fixing the number of analogies to be considered in the case adaptation step will not be considered here neither as a requirement nor as a constraint. The principle of this approach is to take only the first k projects that are similar to the new project.

In Fuzzy Analogy, strategy to select projects that will be used in the adaptation step is proposed. This strategy is based on the distances $d(P, P_i)$ and the definition adopted in the studied environment for the proposition 'P_i is closely similar project to *P*'. Intuitively, P_i is closely similar to *P* if $d(P,P_i)$ is in the approximately of 1 (0 in the case of Euclidean distance). A better way to represent the value 'vicinity of 1' is by using a fuzzy set defined in the unit interval [0, 1]. Indeed, this fuzzy set defines the 'closely similar' qualification adopted in the environment. Figure 4 shows a possible representation for the value 'vicinity of 1'. In this example all projects that have $d(P,P_i)$ higher than 0.5 contribute to the estimated cost of *P*.

The second problem in this step is to adapt the chosen analogies in order to generate an estimate for the new project. The most common solutions use the (weighted) mean or the median of the *k* chosen analogies. In the case of weighted mean, the weights can be the similarity distances or the ranks of the projects. Ali I. et al., (2002) used the weighted mean of all known effort projects in the dataset for their fuzzy analogy approach. The weights are the values of the membership function defining the fuzzy set 'vicinity of 1'. The formula is then: [5]

$$cost(P)=\frac{\sum_{i=1}^{N}\mu_{vicinity\ of\ 1}(d(P,P_i))\times Effort(P_i)}{\sum_{i=1}^{N}\mu_{vicinity\ of\ 1}(d(P,P_i))}$$

Our suggestion is to use the analogy approach to estimate the best prior project fitting in the current project to use its way of resource allocation as a pattern. The main idea is that "*similar software projects consume similar resources*". So that by finding the best project(s) fitting in the initiating project we can out come the problem of resource allocation by making use of their resource allocation plan to make our plan. This approach is been validated using the COCOMO'81 dataset by [5]. We can, as well, refer to it as a proof to the mentioned model solving the problem.

VI. Model validation

Main goal of the validation section is to show how the represented model can be proved. The following section presents and discusses the results obtained when applying the Fuzzy Analogy approach on the COCOMO'81 dataset. The results were compared with those of three other models: classical analogy, the original intermediate COCOMO'81, and 'fuzzy' intermediate COCOMO'81 [23, 24].

Ali I. et al. [5] compared the results of the Fuzzy Analogy with the other three techniques in regards to two criteria: the type of the technique and whether or not the technique uses fuzzy logic in its estimation process. Our findings were the following:

- Two advantages were found when using fuzzy logic with the estimation by analogy. First, it tolerates imprecision and uncertainty in its inputs (cost drivers) and consequently it generates gradual outputs (cost). This is why Fuzzy Analogy gives closer results for the three datasets while classical analogy generates the same or significantly different outputs when the inputs are different (this is the same case between 'fuzzy' and classical intermediate COCOMO'81, see [11] for more details, Table 2). Second, it improves the accuracy of the estimates because our similarity measures are more appropriate than those used in the literature.

- Intermediate COCOMO'81 generates more accurate results than classical analogy but when integrating fuzzy logic in the estimation by analogy procedure, the Fuzzy Analogy performs better than intermediate COCOMO'81. This illustrates that fuzzy logic is an appropriate tool to deal with linguistic values rather than the classical logic (Aristote logic) used in the original version of the COCOMO'81.

Taking into account these results, we suggest the following ranking of the four techniques in terms of accuracy and adequacy to deal with linguistic values:

1) *Fuzzy Analogy*

2) *Fuzzy intermediate COCOMO'81*
3) *Classical intermediate COCOMO'81*
4) *Classical analogy.*

VII. CONCLUSION

In this paper, we have proposed a new approach to estimate the allocation of project resources based on the method proposed to make the estimation of software effort mentioned by Ali I. et al. [5]. This approach is based on reasoning by analogy, fuzzy logic and linguistic quantifiers. Such an approach can be used when the software projects are described by linguistic and/or numerical values. Thus, our approach improves the classical analogy procedure that does not take into account linguistic values. In the Fuzzy Analogy approach, both linguistic and numerical data are represented by fuzzy sets. The advantage of this is to handle correctly the imprecision and the uncertainty when describing a software project. Also, by using RIM linguistic quantifier to guide the aggregation of the individual similarities between two projects, the Fuzzy Analogy approach can be easily adapted and configured according to the needs specifications of each environment. Also, we have tried to validate the Fuzzy Analogy by using the COCOMO'81 dataset. The results of this validation were compared to those of the classical analogy approach, 'fuzzy' intermediate COCOMO'81 and original intermediate COCOMO'81. We can conclude that fuzzy logic improves the estimation process and consequently generates more accurate estimates.

By using fuzzy logic in its estimation process, this approach satisfies the first criterion of the concept Soft Computing, which is the tolerance of imprecision.

VIII. REFERENCES

[1] Lily Rachmawati, Dipti Srinivasan, "A Hybrid Fuzzy Evolutionary Algorithm for A Multi-Objective Resource Allocation Problem", Proceedings of the Fifth International Conference on Hybrid Intelligent Systems (HIS'05), 2005.

[2] David Vengerov, Nikolai Iakovlev, "A Reinforcement Learning Framework for Dynamic Resource Allocation: First Results", Proceedings of the Second International Conference on Autonomic Computing (ICAC'05), 2005.

[3] Pankaj Jalote, Bijendra Vishal, "Optimal Resource Allocation for the Quality Control Process", Proceedings of the 14th International Symposium on Software Reliability Engineering (ISSRE'03), 2003.

[4] Mark A. Turnquist, Linda K. Nozick, "Allocating Time and Resources in Project Management under Uncertainty", Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS'03), 2002.

[5] Ali Idri, Alain Abran, Taghi M. Khoshgoftaar, "Estimating Software Project Effort by Analogy Based on Linguistic Values", Proceedings of the Eighth IEEE Symposium on Software Metrics (METRICS.02), 2002.

[6] L. Angelis, I. Stamelos, M. Morisio, "Building a Software Cost Estimation Model Based on Categorical Data", 7th Int, Soft. Metrics Symp., IEEE Computer Society, London, April, pp. 4-15, 2001.

[7] L. Briand, T. Langley, I. Wieczorek., "Using the European Space Agency data set: A replicated assessment and comparison of common software cost modeling", In Proc 22th IEEE International Conference on Software engineering, Limerik, Ireland, pp. 377-386, 2000.

[8] I. Myrtveit, E. Stensrud, "A Controlled Experiment to Assess the Benefits of Estimating with Analogy and Regression Models", IEEE Transaction on Software Engineering, 25, 4, July/August, pp. 510-525, 1999.

[9] F. Niessink, H. Van Vliet "Predicting Maintenance Effort with Function Points", in Proc Inter. Conf. On Soft Maintenance, Bari, Italy, IEEE Computer Society, 1997.

[10] M. Shepperd, C. Schofield, and B. Kitchenham, "Effort Estimation using Analogy", ICSE-18, Berlin, pp. 170-178, 1996.

[11] M. Shepperd, and C. Schofield, "Estimating Software Project Effort Using Analogies", IEEE Trans. on Software Engineering, Vol. 23, no. 12, November, pp. 736-743, 1997.

[12] S. Shepperd, G. Kadoda, "Using simulation to evaluate predictions systems", 7th International Symposium on Software Metrics, IEEE Computer Society, 4-6 April, England. pp. 349-358, 2001.

[13] Lily Rachmawati, Dipti Srinivasan, "A Hybrid Fuzzy Evolutionary Algorithm for A Multi-Objective Resource Allocation Problem", Proceedings of the Fifth International Conference on Hybrid Intelligent Systems (HIS'05), 2005.

[14] Pankaj Jalote, Bijendra Vishal, "Optimal Resource Allocation for the Quality Control Process", Proceedings of the 14th International Symposium on Software Reliability Engineering (ISSRE'03), 2003.

[15] Tzy-Yuan Chou, Seng-cho T. Chou, Gwo-Hshiung Tzeng, "Evaluating IT/IS investments: A fuzzy multi-criteria decision model approach", European Journal of Operational Research 173, 1026–1046, 2006.

[16] Kaufmann A., Gupta M., Introduction to fuzzy arithmetic theory and applications; Van Nostrand Reinhold; New York, 1985.

[17] Zimermann H.J., Fuzzy set theory-and its applications; Third Edition, Kluwer Academic Publishers; Boston, 1996.

[18] Chen S. J., Hwang C. L. (1992), Fuzzy multiple attribute decision making: methods and applications; Lecture notes in economics and mathematical systems, Springer-Verlag; Berlin, Germany.

[19] Dubois D., Prade H. (1988), Possibility theory: an approach to computerized processing of uncertainly; Plenum Press; New York.

[20] Stefan Schmidt, Robert Steele, Tharam S. Dillon, Elizabeth Chang, "Fuzzy trust evaluation and credibility development in multi-agent systems", Applied Soft Computing 7, 492–505, 2007.

[21] A. Idri, L. Kjiri, and A. Abran, "COCOMO Cost Model Using Fuzzy Logic", 7th International Conference on Fuzzy Theory & Technology, Atlantic City, NJ, February. pp. 219-223, 2000.

[22] L.A. Zadeh, "Fuzzy Logic, Neural Networks, and Soft Computing", Comm. ACM, Vol. 37, no. 3, March, pp.77- 84, 1994.

ISSN 1242-5125 ENST S (Paris) (France 1983-9999)