International Journal of Research in Science and Technology http://www.ijrst.com/
(IJRST) 2012, Vol. No. 2, Issue No. III, July-Sept **ISSN: 2249-0604**

USE OF FUZZY LOGIC AND ITS IMPLEMENTATION IN SOFTWARE ENGINEERING

***Swati Garg, #Dr. V.K. Gaur**

**Research Scholar*

#Associate professor, Govt. Dungar College, Bikaner

ABSTRACT

***Fuzzy logic** is a form of many-valued logic; it deals with reasoning that is approximate rather than fixed and exact. In contrast with traditional logic theory, where binary sets have two-valued logic: true or false, fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific functions.*

Fuzzy logic began with the 1965 proposal of fuzzy set theory by Lotfi Zadeh. Though fuzzy logic has been applied to many fields, from control theory to artificial intelligence, it still remains controversial. Fuzzy logic and probabilistic logic are mathematically similar – both have truth values ranging between 0 and 1 – but conceptually distinct, due to different interpretations—see interpretations of probability theory. Fuzzy logic corresponds to "degrees of truth", while probabilistic logic corresponds to "probability, likelihood"; as these differ, fuzzy logic and probabilistic logic yield different models of the same real-world situations.

Both degrees of truth and probabilities range between 0 and 1 and hence may seem similar at first. For example, let a 100 ml glass contain 30 ml of water. Then we may consider two concepts: Empty and Full. The meaning of each of them can be represented by a certain fuzzy set. Then one might define the glass as being 0.7 empty and 0.3 full. Note that the concept of emptiness would be subjective and thus would depend on the observer or designer. Another designer might equally well design a set membership function where the glass would be considered full for all values down to 50 ml. It is essential to realize that fuzzy logic uses truth degrees as a mathematical model of the vagueness phenomenon while probability is a mathematical model of ignorance. The same could be achieved using probabilistic methods, by defining a binary variable "full" that depends on a continuous variable that describes how full the glass is. There is no consensus on which methodshould be preferred in a specific situation.

INTRODUCTION

"Everyone feels they understand it" [PRES97]. A software quality factor is a non- functional requirement for a software program, which is not called upon by the customer's contract, but is a desirable requirement, which enhances the quality of the software program. Some software quality factors are understandability, completeness, conciseness, portability, consistency, maintainability, testability, usability, reliability, structuredness, efficiency and security. Sun Sup So et al. [SUNS02] proposed a fuzzy logic based approach to predict error-prone modules using inspection rate and error density.

The cause of software errors have ranged from poorly designed user interfaces to direct programming errors. Easily maintainable software saves considerable costs in industries [BERN84]. Aggarwal et al. [AGGA03], describes software maintainability as a measure of characteristics of software, e.g., source code readability, documentation quality and cohesiveness among source code anddocuments.

Software productivity is usually expressed as work done per unit of time whereas work done on a software system after it becomes operational is regarded as maintenance productivity. Sommerville [SOMM92] asserted that the productivity factors concerning the environment of software maintenance could include characteristics such as module independence, programming language, programming style etc. Cyclomatic complexity density of software is considered as an effective metric for measuring productivity [GILL91].

It is noted that traditional estimation approaches may face serious difficulties when used on software engineering data that is usually scarce, incomplete, and imprecisely collected. Fuzzy logic being one of the important tools to model uncertainties, the emphasis is on quantitative estimation of various software attributes using fuzzy technique.

Even though effort has been done to propose, fuzzy based models, there is a vastscope for improvement. The existence of a large set of alternatives provides the option of selecting the best software measurement techniques for a particular application. The option of using a fuzzy logic technique opens up the opportunity of new ways to model uncertainties in software testing and debugging.

FUZZY LOGIC

In 1948, Alan Turing wrote a paper, which marked the beginning of a new era, the era of the intelligent machine. To allow computers to mimic the way humans think, the theories of fuzzy sets and fuzzy logic was created. Classical logic deals with crisp knowledge where statements can only be either true or false, while fuzzy logic deals with vaguely formulated or uncertain knowledge.

Dr. Lotfi Asker Zadeh first used the term fuzzy in the engineering journal, “proceedings of the IRE” in 1962. Fuzzy sets were introduced by Zadeh [ZADE65] as an extension of the classical notion of the set. Fuzzy sets are sets whose elements have degrees of membership. Fuzzy sets generalize classical sets, since the indicator functions of classical sets are special cases of the membership functions of fuzzy sets, if the latter only take values 0 or 1. In fuzzy set theory, classical sets are usually known as crisp sets. Fuzzy logic is a methodology, based on fuzzy set theory to solve problems, which are too complex, to be understood quantitatively [ZADE65]. Fuzzy logic is a superset of conventional logic that has been extended to handle the concept of partial truth. Fuzzy logic can be thought of as the application side of fuzzy set theory. It is an effective technique to solve uncertainties due to imprecise data.

Fuzzy Number

A fuzzy number is a quantity whose value is imprecise, rather than exact as in the case of ordinary single valued numbers. A fuzzy number is represented by a membership function, whose domain is a fuzzy set. The membership function associates a real number [0, 1] with each point in the fuzzy set, called degree of uncertainty or grade of membership. The membership μA (x) of an element x of a classical set A, as subset of the universe X, isdefined by:

$$\mu_A(x) = \begin{cases} 1 \; iff & x \in A \\ 0 \; iff & x \notin A \end{cases} \qquad 1.1$$

A triangular fuzzy number (TFN) is described by a triplet (□ , m, β), where m is the model value, and are the right and left boundary respectively (Figure 1.3). The membership function (x) for TFN is defined as:

$$\mu(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a \leq x \leq m \\ \frac{b-x}{b-m}, & m \leq x \leq b \\ 0, & x \geq b \end{cases} \qquad 1.2$$

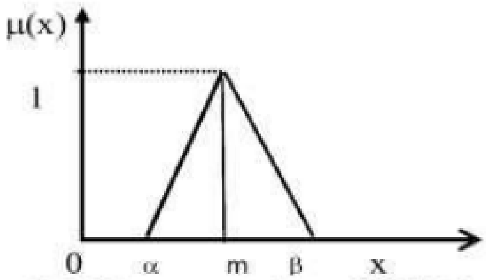

Figure 1.3: Representation of TFN (α, m,β)

Fuzziness

Fuzziness is explored as an alternative to randomness for describing uncertainty. Fuzziness relates to the un-sharp boundaries of the parameters of the model. Fuzziness of TFN (□ , m, β) is defined as[MUSI00]:

$$\text{Fuzzinessof TFN(F)} = \frac{\beta - \alpha}{2m}, \; 0 < F < 1 \qquad 1.3$$

The higher the value of fuzziness, the fuzzier is TFN. The value of fuzziness taken depends upon the confidence of the estimator. A confident estimator can take a smaller value of F. Let (m, 0) internally divide the base of the triangle in ratio k: 1, where k is real positive number. So that,

$$m = \frac{\alpha + k\beta}{k+1} \quad 1.4$$

As by definition of fuzziness

$$F = \frac{\beta - \alpha}{2m} \quad 1.5$$

so

$$\alpha = \left(1 - \frac{2kF}{k+1}\right) * m \, and \beta = \left(1 + \frac{2F}{K-1}\right) * m \quad 1.6$$

FUZZY LOGIC PROCESSES

The fuzzy logic process involves mainly following three sub-processes:

- Fuzzification
- Application of fuzzy logic
- Defuzzification

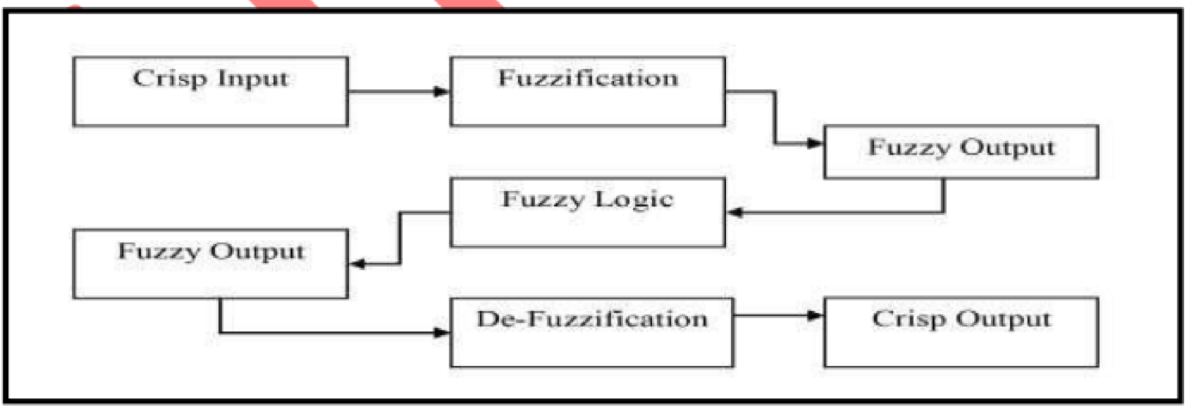

Figure 1.4 illustrates the process of getting crisp out using fuzzy logic process.

- **Fuzzification:** Fuzzification is a process whereby crisp values are converted to fuzzy values, represented by membership function. First step is to select appropriate linguistic/fuzzy system variables followed by defining fuzzy sets to represent concepts for each fuzzy variable. Assignment of fuzzy values may be done either by intuition or by some algorithm. Intuition involves contextual and semantic knowledge about an issue.

- **Application of fuzzy logic:** Identification of rules to relate inputs to outputs, selecting techniques for correlation of inputs to outputs and composition of rules.
- **Defuzzification:** Defuzzification is the process of converting fuzzy values to crisp values. Several techniques have been proposed for defuzzification. The simplest method is the Maximum method in which a fuzzy set with maximum membership function is selected. Some popular defuzzification methods are discusses below:

1. **Centroid:** Centroid defuzzification returns the center of area under the curve. If you think of the area as a plate of equal density, the centroid is the point, about which this shape would balance.
2. **Bisector:** The bisector is the vertical line that will divide the region into two sub-regions of equal area. It is sometimes, but not always coincident with the centroid line.
3. **Middle, Smallest, and Largest of Maximum:** MOM, SOM and LOM stand for middle, smallest, and largest of maximum, respectively. These three methods key off the maximum value assumed by the aggregate membership function. If the aggregate membership function has a unique maximum, then MOM, SOM and LOM all take on the same value.
4. **Miscellaneous criteria:** We can formulate criteria that are not directly related to any theoretical concepts or foundations, but that are of more practical importance [LEEK99].

There is however no simple answer to the question „which of these methods is the right one?" However, if you want to get started quickly the centroid method is the best option. Later you can always change your defuzzification method to see if another method works better. The fuzzy logic technique used in this work is based on formulating individual criteria for defuzzification based on miscellaneous criteria described above.

SOFTWARE ENGINEERING MEASUREMENTS

Measurement is fundamental to any engineering discipline for increasing quality and reducing cost. Software measurement enables us to characterize, evaluate, predict and improve various attributes of software and the commercial software production process providing data in useable formats that can be readily in the development process (Figure 1.1). We need to measure parameters of software like size, complexity, reliability, correctness and maintainability and programmer hours per kilo lines of code.

Many software development projects face problems leading to late delivery, cost overruns and often-dissatisfied customers. There is therefore a need to develop intelligent models to effectively plan and manage the development process of medium to large software projects. Before initiating any project, calculation of the time, cost and work force involved in the project must be made to ensure the success of the project. Software measurement consists of direct and indirect measures. Direct measures include cost, effort, line of code (LOC), execution speed, memory, and defects reported over a period. Indirect measures include functionality, quality, complexity, reliability, maintainability and other abilities.

Typical size oriented metrics are errors per KLOC, defects per KLOC, cost per KLOC and LOC produced per person month. Typical function-oriented metrics are function point (FP), errors per FP, defects per FP, cost per FP and FP per person month. Software cost estimation is the process of predicting the amount of effort required to build a software system and its duration. The surveys presented by Jorgensen et al. [JORG06] summarize the software estimation evolution. Alaa F. Sheta et al. [ALAA06, ALAA08], introduced KLOC based effort estimation models using Genetic Algorithms and soft computing techniques. Bingchiang Jenga et al. [BING06] modified popular Function Point Analysis (FPA) model, so that it is specific and incorporate more of an application ‟ s characteristics. In the context of software engineering, software quality measures how well software is designed (quality of design), and how well the software conforms to that design (quality of conformance). One of the challenges of software quality is that:

REVIEW OF LITERATURE

This chapter presents a brief review of the literature relevant to the areas of the present study concerning software effort estimation, estimation of software testing costs and risks, estimation of quality of software and methods to quantify maintainability of software using fuzzy logic approach.

SOFTWARE COST ESTIMATION

Software cost estimation is the process of predicting the effort required to develop a software system. In the early days of computing, software costs constituted a small percentage of the overall computer-based system cost. An order of magnitude error in estimates of software cost had relatively little impact. Today, software is the most expensive element of virtually all computer-based systems. A large cost estimation error can make the difference between profit and loss. Cost overruns can be disastrous for the developer. Of the three principal components of software cost i.e. hardware, travel and training, and effort costs, the effort cost is dominant. Software cost estimation starts at the proposal state and continues throughout the life of a project.

PROPOSED RESEARCH WORK

Fuzzy concepts may generate uncertainty (they do not provide a clear orientation for action or decision-making) and reducing fuzziness may generate more certainty. However, this is not necessarily always so, insofar as a concept, although it is not fuzzy at all and very exact, could equally well failed to capture the meaning of something adequately. A concept can be very precise, but not - or insufficiently - *applicable* or *relevant* in the situation to which it refers. A fuzzy concept may indeed provide *more* security, because it provides a meaning for something when an exact concept is unavailable - which is better than not being able to denote it at all. A concept such as God, although not easily definable, for instance can provide security to the believer.

REFERENCES

[AGGA03] Aggarwal, K.K., Singh, Y., Chhabra, J. K., A multiple parameter software complexity measure. J. CSI, 33: 22-30, 2003.

[AGGA05] Aggarwal, K. K., Singh, Y., & Puri, M. Measurement of Software Maintainability using Fuzzy model. Journal of Computer Science, 1 (4), 538- 542, 2005.

[AKIY71] Akiyama, F. An example of software system debugging, Proc. IFIP Congress, 353-358, 1971.

[ALAA06] Sheta, A. F., Estimation of the COCOMO Model Parameters Using Genetic Algorithms for NASA Software Projects, Journal of Computer Science 2(2):118-123, 2006.

[ALAA08] Sheta, A., Rine, D., Ayesh, A., Development of Software Effort and Schedule Estimation Models Using Soft Computing Techniques, IEEE World Congress on Computational Intelligence, 2008, Hong Kong, pp. 1283-1289,2008.

[ALBR79] Albrecht A.J., Measuring application development productivity, SHARE/GUIDE IBM Application development Symposium, Monterey, California, pp 83-92, 1979.

[ALBR84] Albrecht A.J., AD/M Productivity Measurement and Estimate Validation-Draft, IBM Corp. Information Systems and Administration, AD/M Improvement Program, Purchase, NY, May 1, 1984.

[ARIA96] ARIANE 5 Flight 501 Failure, Reported by the Inquiry Board http://www.esrin.esa.it/htdocs/tidc/Press/Press96/ariane5rep.html, Paris, 1996 July 19.

REFERENCES

Van Pelt, Miles (2008). Fuzzy Logic Applied to Daily Life. Seattle, WA: No No No NoPress. ISBN 0-252-16341-9.

Steeb, Willi-Hans (2008). The Nonlinear Workbook: Chaos, Fractals, Cellular Automata, Neural Networks, Genetic Algorithms, Gene Expression Programming, Support Vector Machine, Wavelets, Hidden Markov Models, Fuzzy Logic with C++, Java and Symbolic C++ Programs: 4edition. World Scientific. ISBN 981-281-852-9.

Pedrycz, Witold; Gomide, Fernando (2007). Fuzzy systems engineering: Toward Human- Centerd Computing. Hoboken: Wiley-Inter science. ISBN 978047178857-7.

"Lotfi A. Zadeh" faculty page from College of Engineering, Electrical Engineering and Computer Science, University of California at BerkeleyS

At this time, the Azerbaijan SSR was an independent republic, created by the Red Army. It would become part of the Transcaucasian Socialist Federative Soviet Republic in March 1922, and then part of the Soviet Union in December 1922.

McNeil & Freiberger, p.17

"Jews in Computer & Information Science" on the JINFO.org website Gale, Thomson. Lotfi Asker

Zadeh Biography World of Computer ScienceMcNeil & Freiberger, p.18

Blair, Betty. Interview with Lotfi Zadeh (December 1999) in "Famous People: Then and NowLotfi Zadeh, Creator of Fuzzy Logic (1921-)" Azerbaijan International (7.4) (Winter 1999)

McNeil & Freiberger, p.19

Blair, Betty. "Short Biographical Sketch". Azerbaijan International, Vol. 2:4 (Winter 1994),p. 49.

Blair, Betty. "Interview with Lotfi Zadeh, Creator of Fuzzy Logic". Azerbaijan International,Vol. 2:4 (Winter 1994), pp. 46 ff.

McNeil & Freiberger, passim

"IEEE James H. Mulligan, Jr. Education Medal Recipients". IEEE. http://www.ieee.org/documents/education_rl.pdf. Retrieved September 3, 2011.

"IEEE Richard W. Hamming Medal Recipients". IEEE.http://www.ieee.org/documents/hamming_rl.pdf. Retrieved May 29, 2011.

"IEEE Medal of Honor Recipients". IEEE. http://www.ieee.org/documents/moh_rl.pdf.Retrieved September 3, 2011.

"AI's Hall of Fame". IEEE Intelligent Systems (IEEE Computer Society) ***26*** *(4): 5–15. 2011. doi:10.1109/MIS.2011.64. http://www.computer.org/cms/Computer.org/ComputingNow/homepage/2011/0811/rW_IS_AIsHallofFame.pdf. edit*

"IEEE Computer Society Magazine Honors Artificial Intelligence Leaders". Digital Journal.com. August 24, 2011. http://www.digitaljournal.com/pr/399442. Retrieved September 18, 2011. Press release source: PRWeb (Vocus).