

## Research Article

# Bayesian and Fuzzy Approach to Assess and Predict the Maintainability of Software: A Comparative Study

**Kawal Jeet<sup>1</sup> and Renu Dhir<sup>2</sup>**

<sup>1</sup>Department of Computer Science, DAV College, Jalandhar 144008, India

<sup>2</sup>Department of Computer Science & Engineering, NIT, Jalandhar 144011, India

Correspondence should be addressed to Renu Dhir, dhirr@nitj.ac.in

Received 31 August 2011; Accepted 27 September 2011

Academic Editor: D. Testi

Copyright © 2012 K. Jeet and R. Dhir. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Quality has always been one of the major issues responsible for the success of software. Maintainability is one of the characteristics of software quality. A large number of techniques were developed for the assess and predication of this characteristic. Most of these techniques do not decompose it to an actual assessment level and thus fail to give a detailed account of the impact of specific criteria. These techniques thus constrain their use as the basis for analysis quantitatively. In this paper, we develop a system based on fuzzy inference approach to assess and predict maintainability in a quantitative manner. This system is an enhancement of Bayesian approach which is using activity-based quality model to deal with maintainability. We also compare the proposed fuzzy technique with an existing Bayesian approach to depict the improvement achieved due to the advantageous accuracy of fuzzy over crisp approach.

## 1. Introduction

The objective of software engineering is to produce a good quality and maintainable software keeping schedule and budget intact. Many software fail to meet cost, schedule deadline, or quality standards and hence get declared as failure. According to a survey [1], about 45% of the software fails due to the lack of quality. So it is one of the major aspects responsible for the success of software. According to O'Regan [2], quality is fitness to use software.

ISO 9126 standard for information technology provides the framework for the evaluation of software quality. According to this framework, there are six quality characteristics [3]. These include the following.

**Functionality:** It indicates the extent to which the required functions are available in the software.

**Reliability:** This characteristic indicates the extent to which the software is reliable.

**Usability:** This characteristic exhibits the extent to which the users found the software easy to use.

**Efficiency:** It indicates the efficiency of the software.

**Maintainability:** It depicts the extent to which the software product is easy to maintain and modify.

**Portability:** This characteristic indicates the ease with which the software could be transferred to a different environment.

The extent to which software exhibits these quality characteristics indicates the extent to which the software is rated as a quality software.

From the above list, it could be found that maintainability is one of the important characteristics that must be taken care of while making quality software [4]. In this paper, we study maintainability as it is a key quality attribute of large software systems. The desire for high maintainability is a desire for low-maintenance efforts. However, current approaches to assess and improve maintainability fail to explicitly take into account the cost factor that largely determines software maintenance efforts.

Existing approaches to model this attribute have not created a common understanding of the influencing factors and their interrelations. They do not decompose these attributes and criteria to a level that is suitable for an actual assessment. So these models cannot be used as the basis for analysis [5].

In the present work, we are assessing and predicting maintainability by using fuzzy approach which is found to be better than the previously available approaches such as Bayesian networks. Along with this, we are explicitly comparing already available Bayesian approach and our proposed fuzzy approach to justify the improvement made by our technique.

## 2. Related Work

**2.1. Activity-Based Quality Model.** Activity-based quality model breaks down the complex concept of quality into more concrete ones, such as facts about the system, process, and environment and their impact on activities performed on and with the system. Activity-based quality model (ABQM) is a two-dimensional quality model. It is a structured decomposition of maintainability. First dimension is activity broken down structure which is depicted in row form like concept location, impact analysis, coding, and so forth. Second dimension shows the facts, which are shown in column form such as skills, documentation, and tools. It associates system properties with the activities explicitly carried out during maintenance. This separation of facts and activities is the first step towards justified practice of maintainability. For actual evaluation, the facts are found to be very coarse in granularity. So these are broken down into atomic facts which can be assessed without further decomposition such as recursion, debugger, and concurrency cloning. This decomposition leads to breaking down of facts into entities and attributes. Entities are the objects we observe in the real world, and attributes are the properties that an entity possesses [6]. ABQM successfully defined quality but lacks quantitative approach needed for actual analysis.

**2.2. A Bayesian Approach to Assess and Predict Software Quality Using ABQM.** A systematic approach for using ABQM is developed by the authors of [7]. They used Bayesian network to predict the probability of the occurrence of facts and activities involved and ultimately assessing and predicting maintainability. This Bayesian network contains three types of nodes:

- (i) activity nodes that represent activities from the quality model;
- (ii) fact nodes that represent facts from quality model;
- (iii) indicator nodes that represent metrics for activities or facts.

The following four steps are used to derive these nodes from the information of the ABQM.

First, the relevant activities with indicators based on the prediction goal are identified (such as maintenance).

Second, influencing subactivities and facts are identified. Other factors that are related to the identified activities are obtained from ABQM. This step is repeated recursively for subactivities.

Third, suitable indicators for the facts are added. One of them is average change efforts.

Fourth, the node probability tables (NPTs) are defined to show the quantitative relationships. The most common val-

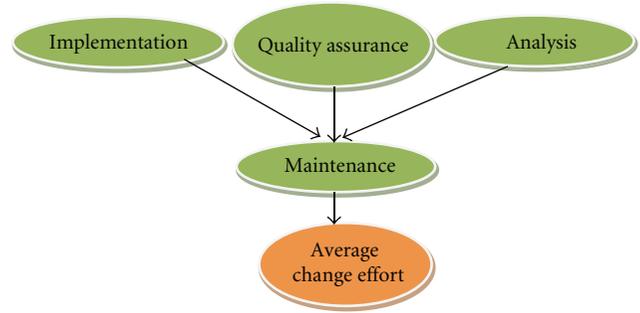


FIGURE 1: Partial Bayesian network to predict maintainability.

ues for the nodes are low, medium, or high. The partial Bayesian network developed using the above steps is shown in Figure 1.

Although Bayesian network is found to be much better than previous methods of evaluating maintainability, results are not very accurate due to crisp nature of input activities like implementation, quality assurance, analysis, and so forth.

**2.3. Proposed System: A Fuzzy Approach to Deal with ABQM.** Fuzzy inference systems (FISs) are found to be a feasible means for prediction of the systems based on the experience of experts. Fuzzy logic is used for the present system as the inputs (implementation, quality assurance, and analysis) could be fuzzy. For the present work, FIS is developed on the basis of dataset given by PROMISE Software Engineering Repository dataset [8]. We develop a system which is actually an enhancement of the Bayesian approach discussed above. We use fuzzy inference approach to overcome the shortcomings of Bayesian approach. The basic topology, on the basis of which this FIS is developed [7], is shown in Figure 1. From Figure 1, it is depicted that “average change effort” is used as an indicator of maintainability of a software project. A brief dataset, based on which this assess and prediction system is developed, is given in Table 1.

## 3. Basics of Fuzzy Logic

The term “fuzzy logic” emerged during the development of the theory of fuzzy sets is coined by Zadeh [9]. A fuzzy subset  $A$  of a (crisp) set  $X$  is characterized by assigning to each element  $x$  of  $X$  the degree of membership of  $x$  in  $A$ . Now, if  $X$  is a set of propositions, then its elements may be assigned their degree of truth, which may be “absolutely true,” “absolutely false,” or some intermediate truth degree. So, fuzzy logic can well define vague (imprecise) propositions of software project development domain. The point of fuzzy logic is to map an input space to an output space, and the primary mechanism for doing this is a list of if-then statements called rules. All rules are evaluated in parallel, and the order of the rules is unimportant. The rules themselves are useful because they refer to variables and the adjectives that describe those variables.

TABLE 1: Information regarding the nodes.

Name	Value	Details
Average change efforts	Numeric [effort required to make change in a software development project measured in person hours and expressed in numeric range]	Indicator of impact of activities on quality of the software. The more the average change efforts, the less maintainable the software, and hence, lower the quality
Maintenance, implementation, analysis, and quality assurance	Low, medium, and high	Activities performed on or within the system
Probability	Numeric [in the range of 0.0–1.0]	Probability of “average change efforts” to be in some specified range provided the occurrence of “maintenance” activity

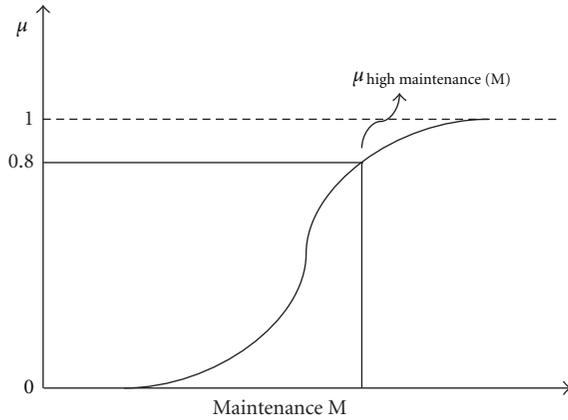


FIGURE 2: Membership function.

So fuzzy inference is a method that interprets the values in the input vector and, based on some set of rules, assigns values to the output vector.

**3.1. Membership Function.** The membership function ( $\mu$ ) of a fuzzy set represents the degree of truth and is mapped in  $[0, 1]$ . For instance, high maintenance is a fuzzy set, then given maintenance  $M$ , the membership function is defined as  $\mu$  high maintenance ( $M$ ) as shown in Figure 2.

**3.2. Implementation Details.** Fuzzy logic toolbox of MATLAB has been used to implement this system [10]. This toolbox provides all the necessary features that are required in a programmer friendly environment. Apart from development, it also provides tools for analyzing the results. Since the inputs for this system are fuzzy, the outputs are constant (in term of probability). So, the most appropriate inference system for this system is Sugeno, or Takagi-Sugeno-Kang, method of fuzzy inference.

For the system proposed above, two fuzzy inference systems (FISs) are developed. First, FIS indicates the relationship among “implementation,” “quality assurance,” “analysis,” and “maintenance.” Second, FIS depicts the relation between “maintenance” and “average change effort” as obtained from Figure 1. The output of first FIS is taken as an input to the second FIS. The partial dataset to be used to implement this fuzzy system is shown in Tables 2 and 3.

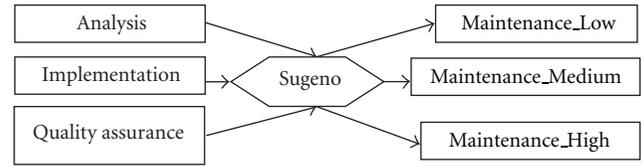


FIGURE 3: FIS between implementation, analysis, and quality assurance as input and maintenance as output.

**3.2.1. First FIS.** Fuzzy inference system among “implementation,” “analysis,” and “quality assurance” as antecedents and Maintenance\_Low, Maintenance\_Medium, and Maintenance\_High as consequents is shown in Figure 3. “Maintenance,” “implementation,” “analysis,” and “quality assurance” are attributes and are called activities. It is developed on the basis of Table 2. Each row of Table 2 indicates that probability of occurrence of “maintenance” is given or conditioned on occurrence of “implementation,” “analysis,” and “quality assurance,” that is,

$P$  (Maintenance|Implementation, Analysis, Quality Assurance).

As an example, row 1 indicates that the probability of “maintenance” to be “low” conditioned on occurrence of “implementation” = “low,” “analysis” = “low,” and “quality assurance” = “low” is 0.9930634. So, in this FIS, antecedents are fuzzy and consequents are constants (as indicated by Sugeno inference). Each antecedent is further having three possible fuzzy sets—low, medium, and high.

There are 27 rules being used in this FIS. The following are few rules from the total set of 27 rules.

If (implementation\_state is low) and (analysis\_state is low) and (quality\_assurance\_state is low) then (maintenance\_low is  $c_1$ ) (maintenance\_medium is  $c_1$ ) (maintenance\_high is  $c_1$ ) (1).

(9) If (implementation\_state is low) and (analysis\_state is high) and (quality\_assurance\_state is high) then (maintenance\_low is  $c_5$ ) (maintenance\_medium is  $c_3$ ) (maintenance\_high is  $c_4$ ) (1).

(27) If (implementation\_state is high) and (analysis\_state is high) and (quality\_assurance\_state is high) then (maintenance\_low is  $c_6$ ) (maintenance\_medium is  $c_1$ ) (maintenance\_high is  $c_6$ ) (1).

TABLE 2: Partial data about the probability of occurrence of maintenance when implementation, analysis, and quality assurance are given.

Maintenance	Implementation	Analysis	Quality assurance	Probability
Low	Low	Low	Low	0.9930634
Medium	Low	Low	Low	0.0069366
High	Low	Low	Low	0.0
Low	Low	Low	Medium	0.7724612
Medium	Low	Low	Medium	0.2275387
High	Low	Low	Medium	0.0
Low	Low	High	High	0.2275387
Medium	Low	Low	High	0.7724612
High	Low	Low	High	2.6977E-9
Low	Low	Medium	Low	0.7724612
Medium	Low	Medium	Low	0.7724612
High	Low	Medium	Low	0.2275387
High	Low	Medium	Low	0.0
Low	Low	Medium	Medium	0.2275387

TABLE 3: Partial data to show probability of average change effort when maintenance level is given.

Average change efforts	Maintenance	Probability
3.9–9.125	Low	0.31552193
3.9–9.125	Medium	0.013411271
3.9–9.125	High	2.3337046E
9.125–14.35	Low	0.31052786
9.125–14.35	Medium	0.062048152

The details of constants  $c_1$ ,  $c_2$ , and so forth are given in Table 4.

3.2.2. *Second FIS.* Another fuzzy inference system is between “maintenance” as fuzzy antecedent and various possible ranges of “average efforts” in person hours as constant consequents. The dataset used to build this inference system is shown in Table 3. This table indicates probability of “average efforts” to be in a range given by “maintenance,” that is,  $P(\text{Average Efforts}|\text{Maintenance})$ .

As an example, row 1 indicates the probability of “average efforts” to be “3.9–9.125” (person hours) conditioned on occurrence of “maintenance” = “low” is 0.31552193. In this FIS, the antecedents are fuzzy and consequents are constants, and so here also sugeno inference is used. The antecedent is further having three possible fuzzy sets—low, medium, and high. Gaussmf membership function is used to find the degree of membership of “maintenance” in fuzzy sets (as discussed above).

This FIS has three rules. These rules are as follows.

If (maintenance is low) then (3.9–9.125 is  $c_1$ ) (9.125–14.35 is  $c_1$ ) (14.35–19.575 is  $c_1$ ) (19.575–24.8 is  $c_1$ ) (24.8–30.025 is  $c_1$ ) (30.025–35.25 is  $c_1$ ) (35.25–40.475 is  $c_1$ ) (40.475–45.7 is  $c_1$ ) (45.7–50.925 is  $c_1$ ) (50.925–56.15 is  $c_1$ ) (56.15–61.375 is  $c_1$ ) (61.375–66.6 is  $c_1$ ) (1).

If (maintenance is medium) then (3.9–9.125 is  $c_2$ ) (9.125–14.35 is  $c_2$ ) (14.35–19.575 is  $c_2$ ) (19.575–24.8 is  $c_2$ ) (24.8–30.025 is  $c_2$ ) (30.025–35.25 is  $c_2$ ) (35.25–40.475 is  $c_2$ ) (40.475–45.7 is  $c_2$ ) (45.7–50.925 is  $c_2$ ) (50.925–56.15 is  $c_2$ ) (56.15–61.375 is  $c_2$ ) (61.375–66.6 is  $c_2$ ) (1).

If (maintenance is high) then (3.9–9.125 is  $c_3$ ) (9.125–14.35 is  $c_3$ ) (14.35–19.575 is  $c_3$ ) (19.575–24.8 is  $c_3$ ) (24.8–30.025 is  $c_3$ ) (30.025–35.25 is  $c_3$ ) (35.25–40.475 is  $c_3$ ) (40.475–45.7 is  $c_3$ ) (45.7–50.925 is  $c_3$ ) (50.925–56.15 is  $c_3$ ) (56.15–61.375 is  $c_3$ ) (61.375–66.6 is  $c_3$ ) (1).

In which  $c_1$ ,  $c_2$ ,  $c_3$ , and so forth are detailed in Table 5.

#### 4. Interface to Compare Bayesian and Fuzzy Approaches to Assess Maintainability Using ABQM

In order to compare and show the improvement of fuzzy approach over Bayesian approach, an interface shown in Figure 5 can be used. In this interface, input could be given for all the three activities under consideration. The inputs are to be entered in the range of 0 to 10. Further, three sets of inputs can be given in order to compare or show the difference between fuzzy and Bayesian approaches. On clicking button “Plot,” Figures 6, 7, and 8 are displayed. Each figure is used to show the output of both fuzzy and Bayesian inferences for all the three sets of inputs. The output is shown in the form of probability of average efforts required to make changes in a project to be in different range for all the three input sets. As an example, we can get probability of average change efforts to be in the range of 3.9–9.125, 9.125–14.35, 14.35–19.575, and 19.575–24.8 person hours for the three given input sets (see Figure 6). For example, the value entered for “implementation” attribute for input 1 is 3, for input 2 it is 2, and for 3 it is 1 (see Figure 5). Similarly, other inputs are also entered. The inputs are specially taken to be in the same

TABLE 4: Information about membership functions associated with the first FIS.

Crisp output variables	c1	c2	c3	c4	c5	c6
Maintenance_low	0.9930634	0.77246124	0.22753875	0.00688287	2,67163E-9	0.0
Maintenance_medium	0.006936614	0.22753875	0.77246124	0.9862223		
Maintenance_high	0.0	2.67163E-9	0.0068888287	0.22753875	0.77246124	0.993063

TABLE 5: Information about membership functions associated with the second FIS.

Output variables	c1	c2	c3
3.9-9.125	0.31552193	0.013411271	2.3337046E-7
9.125-14.35	0.31052786	0.062048152	1.659625E-5
14.35-19.575	0.2327199	0.14756572	2.007459E-4
19.575-24.8	0.11020119	0.21272083	0.0065820036
24.8-30.025	0.027585793	0.22646089	0.039424136
30.025-35.25	0.0032700424	0.18976566	0.11653791
35.25-40.475	1.7260591E-4	0.10799699	0.19554493
40.475-45.7	3.9161127E-6	0.034315377	0.22718826
45.7-50.925	3.7389842E-8	0.0053285165	0.20863116
50.925-56.15	1.4826267E-10	3.7498344E-4	0.13919194
56.15-61.375	2.4211362E-13	1.1459871E-5	0.05524505
61.375-66.6	1.5715541E-16	1.4830746E-7	0.011137037

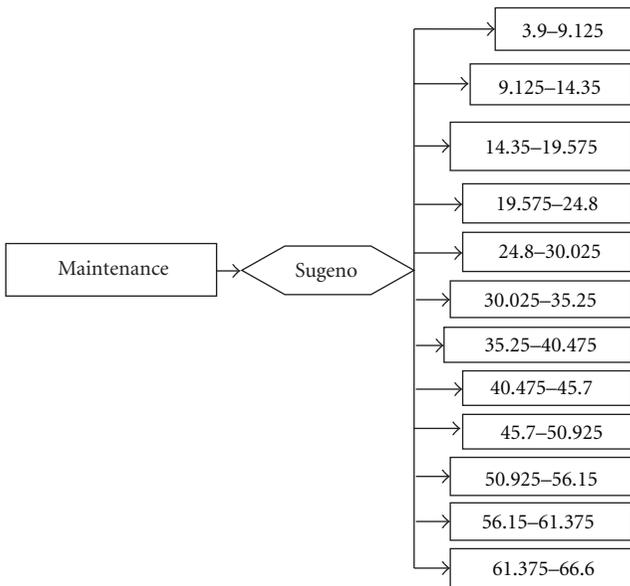


FIGURE 4: FIS between maintenance and various possible ranges of average change efforts.

range in order to depict the difference between Bayesian and fuzzy approaches.

For Bayesian approach, anything between 0 and 3.3333 is taken as low, between 3.3333 and 6.6666 is considered medium, and between 6.6666 and 10 is taken as high. As an example, for the input sets entered in Figure 5, values entered for “implementation” (3, 2 and 1) are considered low and give the same output (as shown in Figures 6, 7, and 8).

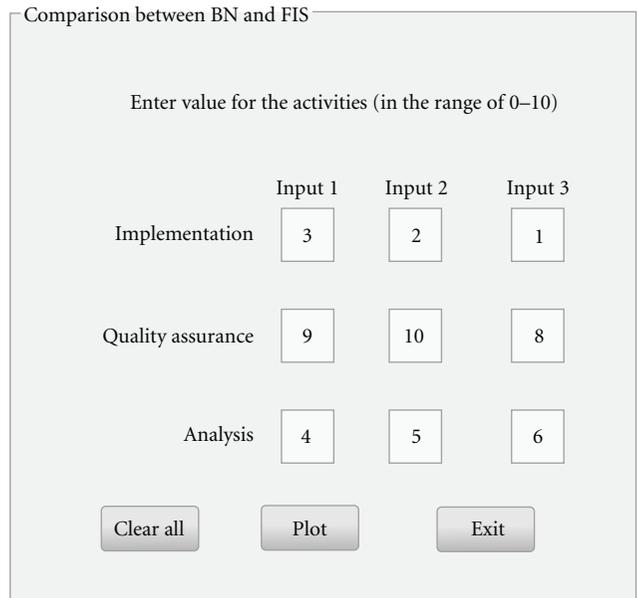


FIGURE 5: Interface to compare Bayesian and fuzzy approaches.

In case of proposed fuzzy approach, these inputs are separately fuzzified and taken as different value each time.

### 5. Working of Sugeno Inference System

A typical rule in a Sugeno fuzzy model has the following form.

If input 1 =  $x$  and input 2 =  $y$ , then output is  $z$  (linear or constant). For each rule  $i$ , the output level  $z_i$  of each rule is weighted by the firing strength  $w_i$  of the rule (as shown in Figure 9). For example, for an AND rule with input 1 =  $x$  and input 2 =  $y$ , the firing strength is

$$w_i = \text{And Method } (F1(x), F2(y)), \tag{1}$$

where  $F1, 2(\cdot)$  are the membership functions for inputs 1 and 2.

The final output of the system is the weighted average of outputs of all the rules, computed as

$$\text{Final Output} = \frac{\sum_{i=1}^N w_i * z_i}{\sum_{i=1}^N w_i}, \tag{2}$$

where  $N$  is the total number of rules.

In order to know internal working of our fuzzy approach, let us take one rule (rule 6) from FIS shown in Figure 4.

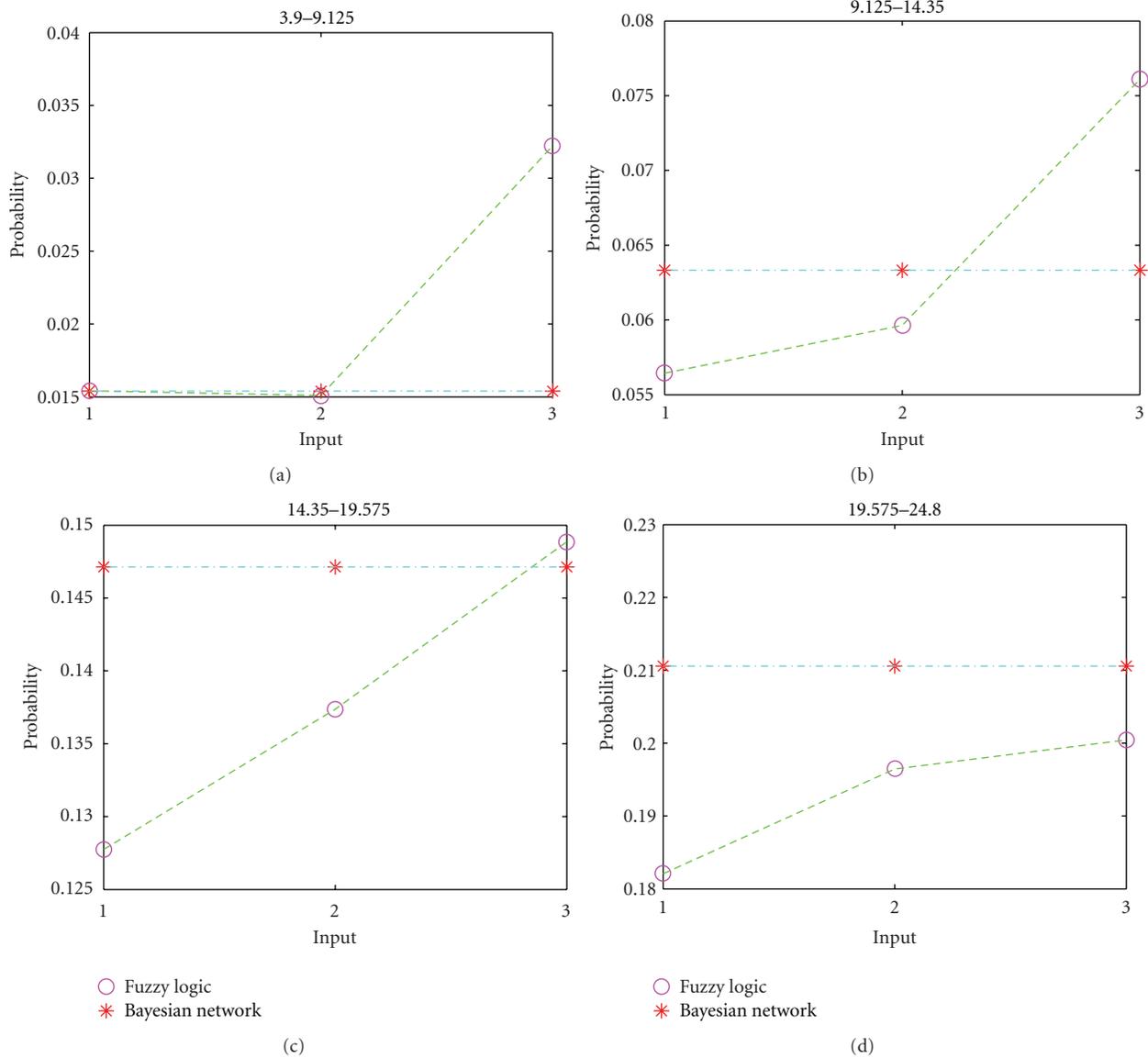


FIGURE 6: Probability of average change efforts to be in the range of 3.9–9.125, 9.125–14.35, 14.35–19.575, and 19.575–24.8 person hours for the three given input sets.

If (implementation\_state is low) and (analysis\_state is medium) and (quality\_assurance\_state is high) then (maintenance\_low is c4) (maintenance\_medium is c4) (maintenance\_high is c3) (1).

For this single rule, fuzzy Sugeno inference works as follows.

*Step 1 (fuzzify inputs).* The first step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. Since gauss is the membership function (a curve that defines how each point in the input space is mapped to a membership value) which is found to be most suitable for the dataset under study, this function finds the degree of membership by putting parameters  $\sigma$ ,  $c$  in the following (3) [10]:

$$f(x; \sigma, c) = e^{-(x-c)^2/2\sigma^2}, \quad (3)$$

TABLE 6: Parameter values for “implementation,” “analysis,” and “quality assurance.”

Fuzzy set	$c$	$\sigma$
Low	0	1.699
Medium	5	1.699
High	10	1.699

where  $x$  is the value whose degree of membership (in the fuzzy set under study) is to be calculated.

For example, in case of implementation, values of parameters  $c$  and  $\sigma$  are given in Table 6.

By putting the value of  $x$ ,  $c$ , and  $\sigma$  in (3), we can get the degree of membership of “implementation” in the fuzzy set

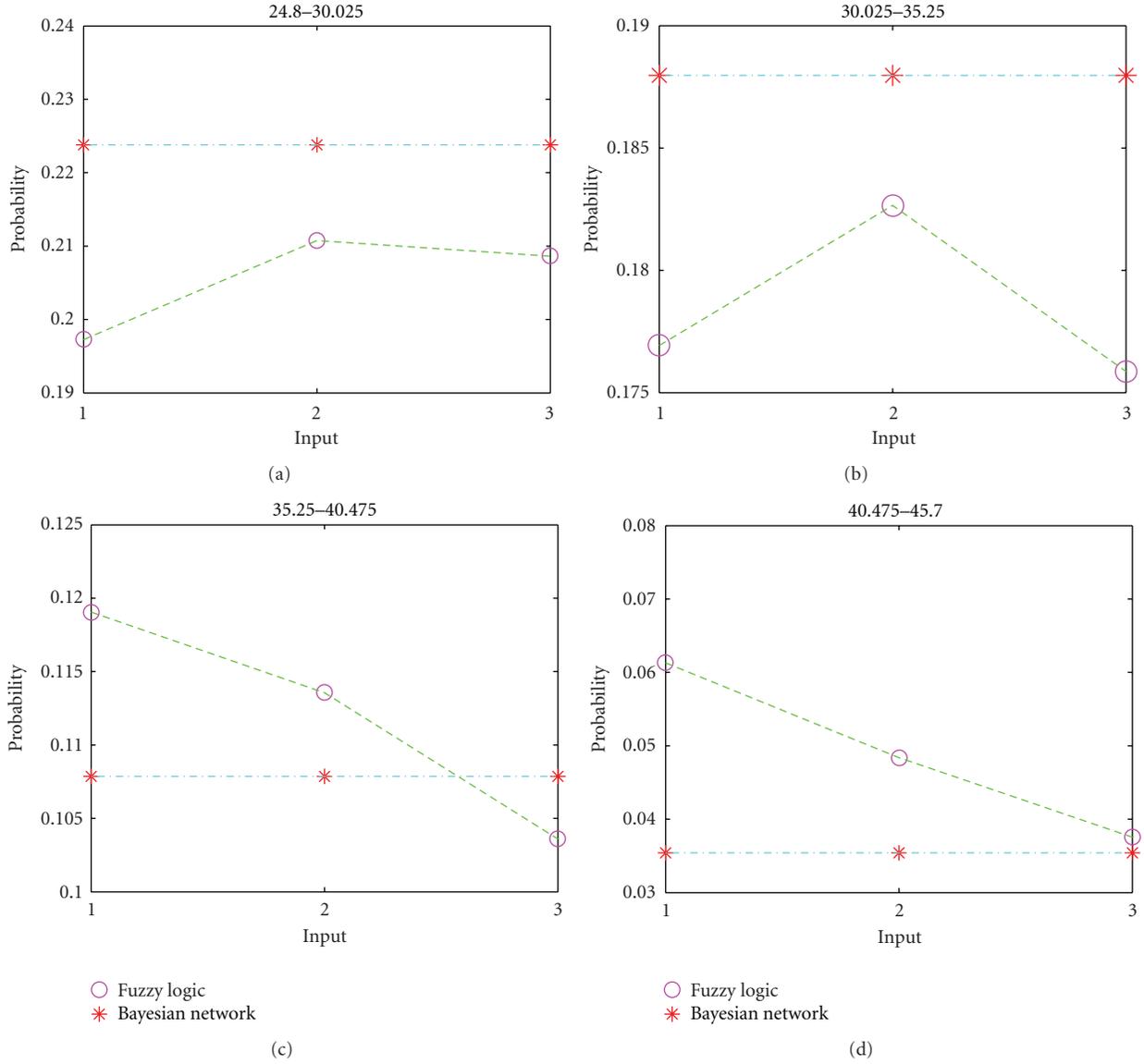


FIGURE 7: Probability of average change efforts to be in the range of 24.8–30.025, 30.025–35.25, 35.25–40.475, and 40.475–45.7 person hours for the three given input sets.

under consideration. Similarly, we can get the degree of membership for analysis and quality assurance.

In Fuzzy Logic Toolbox software, the input is always a crisp numerical value limited to the universe of discourse of the input variable (in this case the interval between 0 and 10). In the present case, it is entered by the interface (as shown in Figure 5) as implementation = 3, analysis = 9, and quality assurance = 4.

For “implementation = 3,” the degree of membership in fuzzy set “low” (as required for rule under consideration) is calculated by putting  $x = 3$  (as shown in Figure 5),  $c = 0$ , and  $\sigma = 1.699$  (using Table 6) in (3),

$$f(x; \sigma, c) = e^{-(3-0)^2/2*1.699^2} = 0.21, \quad (4)$$

as shown in Figure 9.

For “quality assurance = 9,” the degree of membership in fuzzy set “high” (as required for rule under consideration) is calculated by putting  $x = 9$  (as shown in Figure 5),  $c = 10$ , and  $\sigma = 1.699$  (using Table 6) in (3),

$$f(x; \sigma, c) = e^{-(9-10)^2/2*1.699^2} = 0.84, \quad (5)$$

as shown in Figure 10.

Similarly, for “analysis = 4,” the degree of membership in fuzzy set “medium” (as required for rule under consideration) is calculated by putting  $x = 4$  (as shown in Figure 5),  $c = 5$ , and  $\sigma = 1.699$  (using Table 6) in (3),

$$f(x; \sigma, c) = e^{-(4-5)^2/2*1.699^2} = 0.84. \quad (6)$$

*Step 2* (apply fuzzy operator). After the inputs are fuzzified, you know the degree to which each part of the antecedent

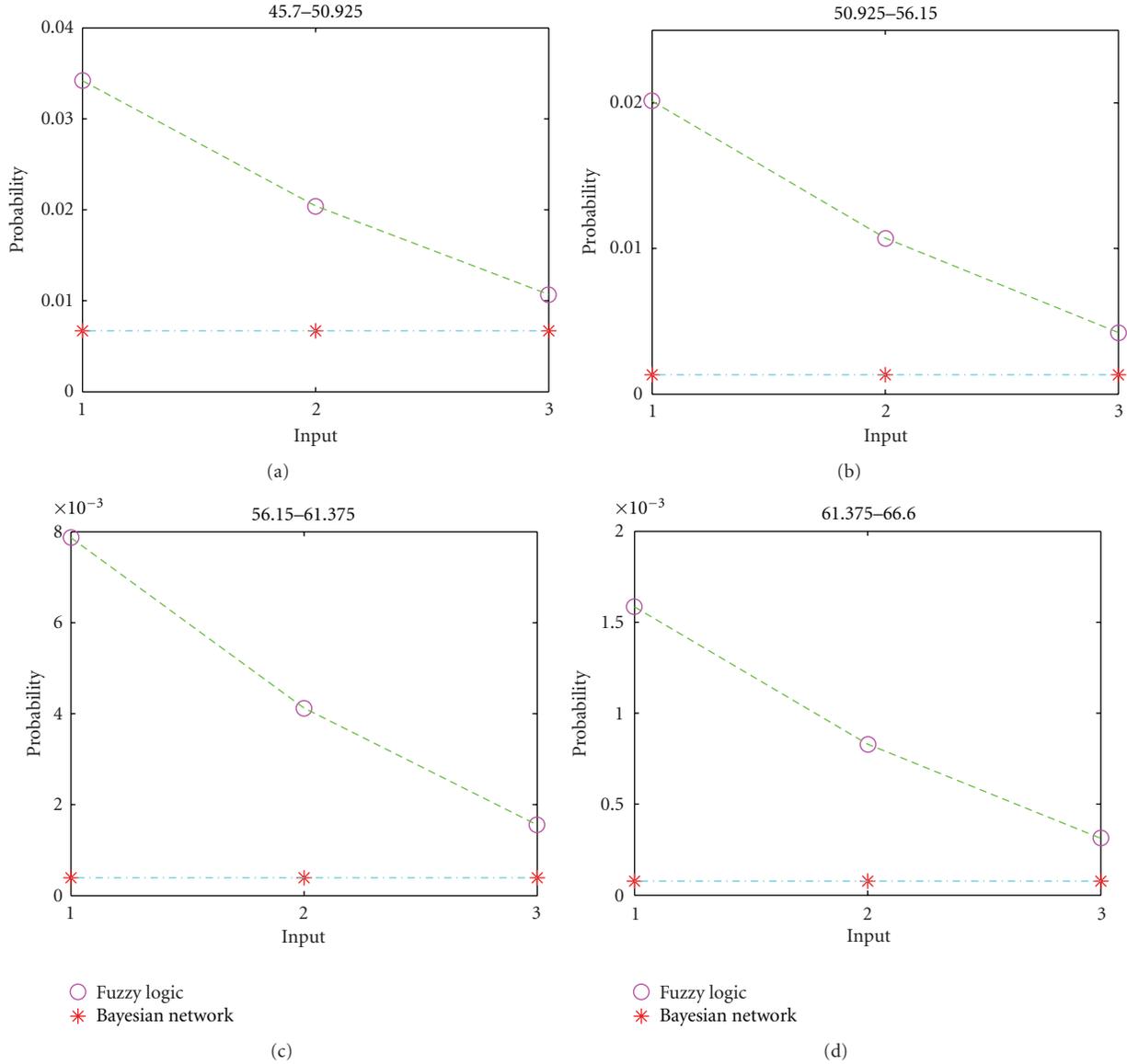


FIGURE 8: Probability of average change efforts to be in the range of 45.7–50.925, 50.925–56.15, 56.15–61.375, and 61.375–66.6 person hours for the three given input sets.

TABLE 7: To fuzzify inputs.

Input	$x$ (from Figure 5)	Fuzzy set under consideration (rule 6)	$c$ (from Table 6)	$\sigma$ (from Table 6)	Degree of membership = $f(x; \sigma, c) = e^{-(x-c)^2/2\sigma^2}$
Implementation	3	Low	0	1.699	$e^{-(3-0)^2/2*1.699^2} = 0.21$
Quality assurance	9	High	5	1.699	$e^{-(9-10)^2/2*1.699^2} = 0.84$
Analysis	4	Medium	10	1.699	$e^{-(4-5)^2/2*1.699^2} = 0.84$

is satisfied for each rule. In case the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number is then applied to the output function.

For the rule under consideration, the three different pieces of the antecedent (implementation is low, analysis is medium, and quality assurance is high) yielded the fuzzy membership values 0.21, 0.84, and 0.84, respectively, (as shown in Figures 9, 10, and 11). The fuzzy AND operator

TABLE 8: Outputs by Bayesian and fuzzy approaches.

Input	Output by Bayesian approach	Output by fuzzy approach
1	0.06333	0.056432
2	0.06333	0.059634
3	0.06333	0.076986

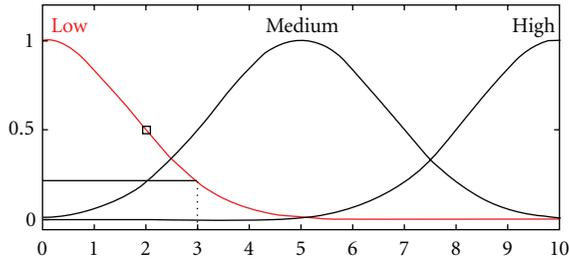


FIGURE 9: Degree of membership of implementation.

simply selects the minimum of the three values that is 0.21, and the fuzzy operation for rule 6 is complete. It gives the weight of the rule.

Step 3 (defuzzification). From the rule’s consequent, we can get Maintenance\_Medium to be having value  $z = c4$ . From Table 4, it is 0.9862223. So contribution of this rule for maintenance medium is  $0.21 \times 0.9862223$ . Similarly, contribution for all the 27 rules is calculated, and their weighted average is calculated as

$$\text{Final outcome} = \frac{\sum_{i=1}^{27} w_i * z_i}{\sum_{i=1}^{27} w_i} \quad (7)$$

(as discussed for (2) and shown in Figure 9).

So, for input 1, “implementation” is 3, and its membership in fuzzy set is 0.21 for rule taken above (as discussed above). Similarly, fuzzy membership for “implementation” for the same rule for input to be 2 is 0.5 and for 1 is 0.84 (as discussed in Table 7) which are obviously different for each input set. So, overall calculations take them differently and behave differently for each input set which is an advantage of fuzzy over Bayesian approach. For average change efforts to be in the range of 9.125–14.35, probability calculated with fuzzy and Bayesian approaches is shown in Table 8.

From this table, it can be easily seen that the output is the same for all the input sets in the same range with Bayesian approach, but it is different for each input set with fuzzy approach.

## 6. Conclusion

Obtaining high-quality software is an integral part of software development project, and maintainability is one of the major characteristics of quality. Many quality models especially activity-based quality model are found to be a major milestone in depicting maintainability, but they do not deco-

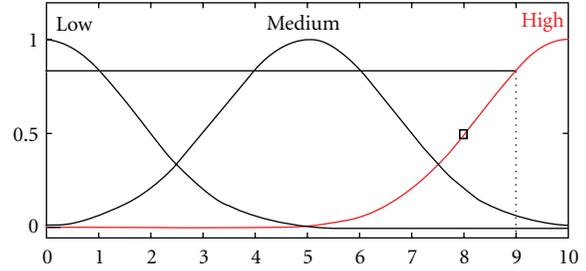


FIGURE 10: Degree of membership of quality assurance.

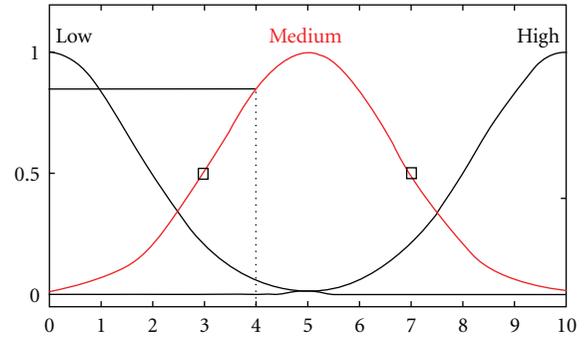


FIGURE 11: Degree of membership of analysis.

pose the attributes and criteria to an actual assessment level. Bayesian approach is developed as a systematic approach to use ABQM. Although it is found to be much better than previous methods of evaluation, results are not very accurate due to crisp nature of input. So fuzzy approach has been proposed which takes input in fuzzy form and predicts maintainability more accurately.

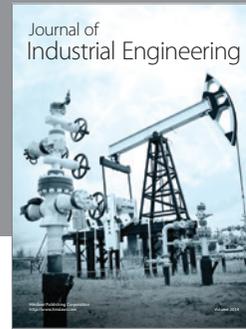
## Acknowledgments

The authors would like to thank University Grants Commission (UGC), New Delhi, India, for supporting this research work via their project number F. no. 8-2 (277)/11 (MRP/NRCB). We also feel immense pleasure to thank Mr. Harjit Singh, senior consultant, Tata Consultancy Services, New Delhi; Mr. Manish Snehi, technical lead, Infosys Technologies, Chandigarh, India; Pavneet Kaur, Component Design Engineer, Intel Corporation, USA for their able guidance and useful suggestions.

## References

- [1] The Standish Group Report Chaos—Project Smart, <http://www.projectsmart.co.uk/docs/chaos-report.pdf>.
- [2] C. G. O’Regan, *A Practical Approach to Software Quality*, Springer, New York, NY, USA, 2002.
- [3] H. W. Jung, S. G. Kim, and C. S. Chung, “Measuring software product quality: a survey of ISO/IEC 9126,” *IEEE Software*, vol. 21, no. 5, pp. 88–92, 2004.
- [4] J. Moses, “Should we try to measure software quality attributes directly?” *Software Quality Journal*, vol. 17, no. 2, pp. 203–213, 2009.

- [5] M. A. Côté, W. Suryn, and E. Georgiadou, “In search for a widely applicable and accepted software quality model for software quality engineering,” *Software Quality Journal*, vol. 15, no. 4, pp. 401–416, 2007.
- [6] F. Deissenboeck, S. Wagner, M. Pizka, S. Teuchert, and J. F. Girard, “An activity-based quality model for maintainability,” in *Proceedings of the 23rd IEEE International Conference on Software Maintenance (ICSM '07)*, pp. 184–193, IEEE Computer Society Press, 2007.
- [7] S. Wagner, “A Bayesian network approach to assess and predict software quality using activity-based quality models,” in *Proceedings of the ACM International Conference on Predictor Models in Software Engineering (PROMISE '09)*, ACM, New York, NY, USA, 2009.
- [8] NASA IV&V Facility. Metrics data program, [http://promise-data.org/repository/data/cm1/cm1\\_bn.arff](http://promise-data.org/repository/data/cm1/cm1_bn.arff).
- [9] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [10] Fuzzy Logic Toolbox for MATLAB and Simulink, <http://www.mathworks.com/products/fuzzylogic>.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

