

## A METHOD BASED ON FUZZY SYSTEM FOR ASSESSING THE RELIABILITY OF SOFTWARE BASED ASPECTS

Mohammad Zavvar<sup>1</sup>, Farhad Ramezani<sup>1</sup>

<sup>1</sup> Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran, e-mail: zavvar.developer@gmail.com; ramezani.farhad@iausari.ac.ir

Received: 2015.07.08  
Accepted: 2015.08.05  
Published: 2015.09.01

### ABSTRACT

In fact, reliability as the qualities metric is the probability success or the probability that a system or a set of tasks will work without failure for a specified constraints of time and space, as specified in the design and operating conditions specified temperature, humidity, vibration and action. A relatively new methodologies for developing complex software systems engineering is an aspect oriented software systems, that provides the new methods for the separation of concerns multiple module configuration or intervention and automatic integration them with a system. In this paper, a method using fuzzy logic to measure software reliability based on the above aspects is presented. The proposed approach regarding the use of appropriate metrics and low errors in the estimation of reliability has a better performance than other methods.

**Keywords:** measure, metric, fuzzy logic, reliability, aspect oriented software development.

### INTRODUCTION

Aspect-Oriented Software Engineering provides new methods for the separation of concerns of multiple module configuration or intervention and their automatic integration with a system that leads to the production and delivery of software product quality and reliable results [1].

Software reliability as the probability of failure-free software operation for a specified time interval is defined in a specific environment [2]. Today, providing products with better features and higher reliability for producers is essential. Consumers expect the software required to work for a longer period of time. Therefore, to guarantee the reliability of products is one of the most interesting features in more detail, the evaluation of the reliability of the product, is a fundamental pillar. Software reliability is an important factor to determine the quantity of software, And with respect to the use of software applications in a variety of industries evaluation of reliability to reduce health and financial risk [3]. Reliability Assessment Software enables the developers of the

software that identify the Software quality factors that could improve the design and coding.

This is an urgent need for software designers to know how reliable it is, and this fact highlights the importance of addressing the reliability factor.

Hence, in this article provides a method for evaluating the reliability of using fuzzy logic-based applications. To do this effectively we need metrics to assess the reliability of software-based aspects, this metric is then transferred to the phase space and the implementation of our proposed system is based on this metric. The final output of the system, the amount estimated for the reliability of the system is based on specific aspects.

### RELATED WORKS

The number of works and research in the field of software reliability assessment based on aspects of performance is very limited. This section examines some aspects of the work and research that has been oriented and aspect-oriented software quality criteria evaluation carried out are discussed.

Figueiredo et al proposed a set of metrics and a set of rules for the assessment of products based on the aspects [4]. This aspect-oriented framework that allows developers to evaluate their products in design and implementation phases. The metric used for educational concerns include: Three metrics CDC, CDO and CDLOC for separation of concerns, as well as the CBC and DIT metric for adhesion or attachment or coupling and cohesion LCOO for. Finally, a tool for the proposed method was produced.

Some researchers [5] provide a method for evaluating the capability of storing and re-using aspect-oriented software. This framework includes a set of metrics of software features and a qualitative model. In this study, the method includes metrics cohesion metrics, metrics and metrics depend on the system size.

Dadhich and Mathur also suggest that aspect-oriented programming is a programming technique that secondary operations and support business logic distinguishes the main program. A part of this program is added to the maintenance of the facility to be constructed. Apart from the operating conditions, in the development of software, we should be careful about operating conditions in addition to other factors such as reliability, compatibility and performance. They determine the quantitative aspect-oriented software reliability model ISO / IEC 9126 used.

Due to the unpredictable nature of software quality features, for improving the application used fuzzy method with multi-criteria [6].

### STATEMENT OF CONCEPTS

In this section we review the basic concepts of fuzzy logic metrics and evaluation of software reliability based on the aspects described above.

#### Fuzzy logic

Fuzzy concept first was introduced by Professor Lotfi Aliasgharzadeh, an Iranian professor at UC Berkeley in 1965 in an article called *Fuzzy sets* [7].

Inability to express the quantity of classical and imprecise concepts like small, large, cheap, expensive, young, old and in each system has specific meaning led to the theory of fuzzy sets with a right step. This theory is a new mathematical framework for formulating and analyzing the concepts and attributes.

In fact, the theory of fuzzy sets theory is a generalization and extension of conventional nature which agrees with the language and understanding of human nature as well.

**Definition 1.** Suppose that  $X$  is an arbitrary set of reference, the common characteristic of a subset  $A$  of  $X$ , is defined as follows:

$$\mu_A(x) = \begin{cases} 1 & : x \in A \\ 0 & : x \notin A \end{cases} \quad (1)$$

According to the above definition, for each  $x \in X$ , only one of the values 0 or 1 will be.

**Definition 2.** If the range of the function  $\mu_A$  of the  $[1, 0]$  to the interval  $[1, 0]$  expand, we have a function to every member of  $X$ , the number in the range  $[1, 0]$  assigns. The other is a set  $A$  is not normal, but what it is called a fuzzy set ( $A$  is a fuzzy subset of  $X$ ).

In the above definition, if  $\mu_A(x) \in (0, 1)$ , then membership of  $x$  in response to  $A$ , we face the uncertainty. In fact, here is an extended concept of membership of an element.

It also represents the membership degree of  $\mu_A(x)$  membership in a fuzzy set is the element  $x$ . if the degree of membership of an element is set to zero, the member is fully withdrawn.[8] And if the degree of membership of a member is set to one, the member is quite a collection. If the degree of membership of a member is between zero and one, the number indicates the degree of membership is gradual. Figure 1 is an example of the membership function of a fuzzy set.

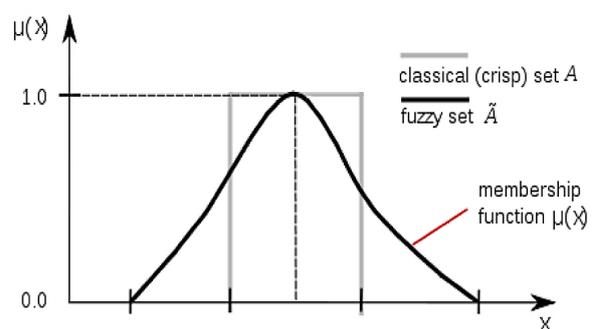


Fig. 1. Membership function of a fuzzy set

#### Effective metrics softwares based on reliability aspects

Metrics of software reliability based on aspects can be categorized the level of concern in

**Table 1.** The metric of the concerns

Metrics	Abbreviation
Program Element Contribution	CONT
Concern Diffusion over Components	CDC
Concern Diffusion over Operations	CDO
Degree of Scattering across Classes	DOSC
Degree of Scattering across Methods	DOSM
Lines of Concern Code	LOCC

**Table 2.** Structural metrics

Metrics	Abbreviation
Weighted Operations per Module	WOM
Depth of Inheritance Tree	DIT
Number of Children	NOC
Coupling on Field Access	CFA
Coupling on Method Calls	CMC
Coupling between Modules	CBM
Crosscutting Degree of an Aspect	CDA
Coupling on Advice Execution	CAE
Response for Module	RFM
Lack of Cohesion of Operations	LCO

both metric and metric structure. Metrics of concern, the lowest level of aggregation is a software system being considered or functional modules that are responsible for the implementation of a concern are grouped together.

The concerned metrics are collected and analyzed. The metrics considered by Ayadi, and colleagues Sanata have been introduced below.[5]

This set of metrics to make the implementation plan needs an assessor. Various techniques have been proposed to identify the source of the concern from the resource code. Methods and techniques widely accepted in general and not for concerns. Tools available to measure the metrics of concern are that researchers need to manually select the code that is linked to a concern. Also, the metric structure of aspect-oriented programming [9] that Ceccato and Tonella extract from known metrics CK [10].

While the level of concern for the analysis of metrics need helping of developers, structural metric based on the modules are already degraded.

The two sets of metrics have been already mentioned. In this article we are going to evaluate the safety aspect-oriented software metrics using the first group, the metric of the concerns, And the metrics listed, four metric CDO, DOSC, DOSM and LOCC due to greater use of the selected channel.

### The proposed approach

Next, we present the proposed method for evaluating software reliability aspect-oriented techniques. To do this you must first transfer a metric used to phase space and then we implement the proposed system based on this metric.

In the proposed model, we use a Gaussian membership function. A Gaussian membership function can be demonstrated by the following equation:

$$\mu_{A_i}(x) = \exp\left(-\frac{(c_i - x)^2}{2\sigma_i^2}\right) \quad (2)$$

In Table 3, we can see the metric and the language of each of the variables. In Table 4, the output of linguistic variables is shown.

**Table 3.** The range of each metric and linguistic variable

Abbreviation	Range
CDO	DOWN = [0.2124 – 0] MIDDLE = [0.2124 – 0.5] UP = [0.2124 – 1.003]
DOSC	
DOSM	
LOCC	

**Table 4.** The output linguistic variables

Output	Range
Reliability	VERYDOWN = [0.1062 – 0]
	DOWN = [0.1062 – 0.25]
	MIDDLE = [0.1062 – 0.5]
	UP = [0.1062 – 0.7489]
	VERYUP = [0.1062 – 1]

Next, exponential membership functions of input and output variables of the model is displayed (Figs. 2–6).

General model of the proposed system is shown in Figure 7. The input variables affecting the reliability metrics are the same and the outputs value of reliability is visible.

For defuzzification we used 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 along the x axis about which this shape would balance.

Given that the proposed model uses four metrics, and each of these metrics is also three linguistic variables, therefore, the proposed model law is 81.

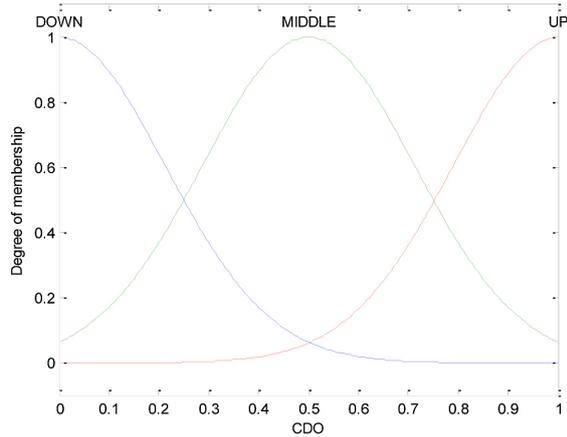


Fig. 2. The membership function metric CDO

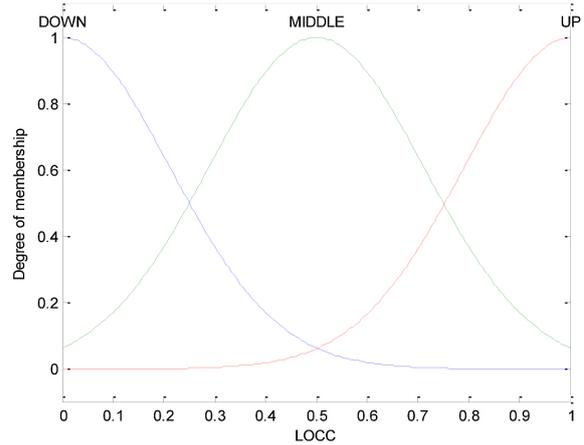


Fig. 5. The membership function metric LOCC

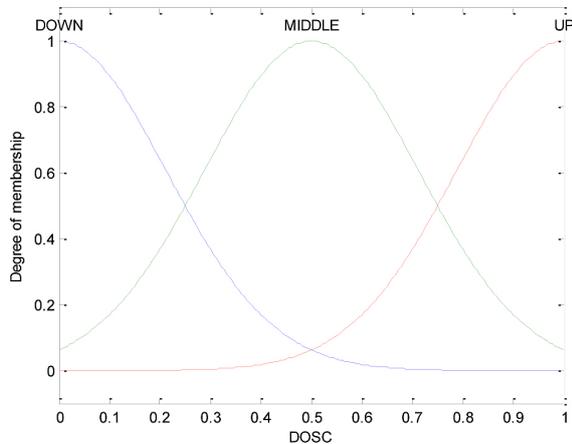


Fig. 3. Membership function metric DOSC

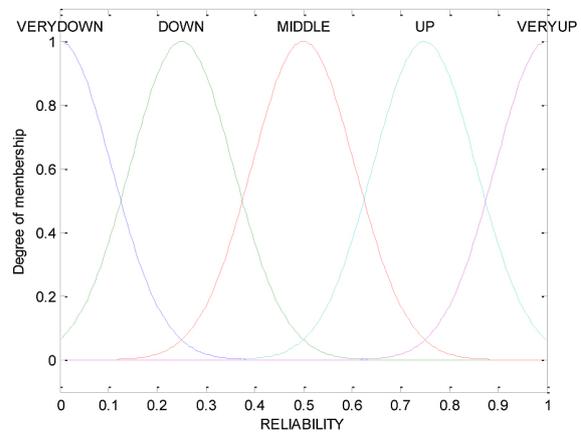


Fig. 6. The output membership function

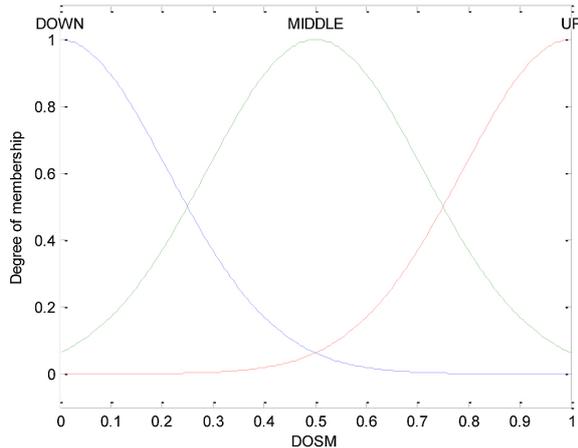


Fig. 4. Membership function metric DOSM

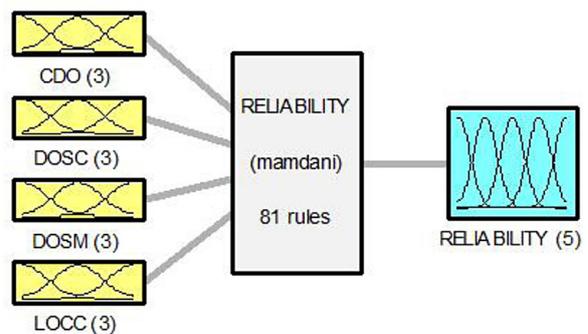


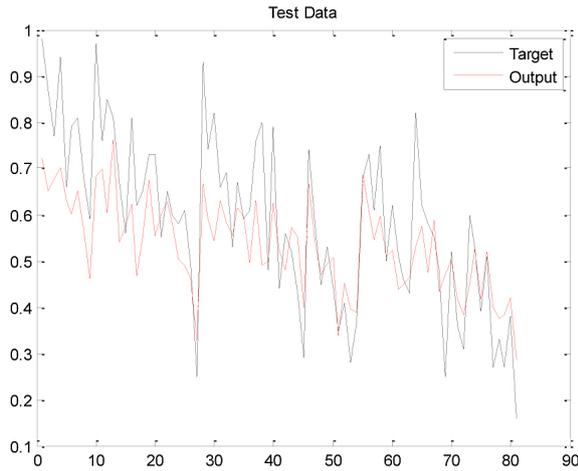
Fig. 7. General model of the proposed system

## ASSESSMENT

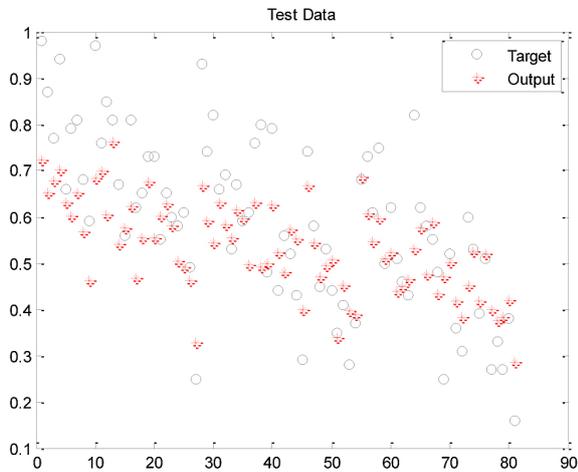
The model proposed in this paper, based on data extracted from the aspect-oriented software projects that have been developed methodology was tested. The diagram in Figure 8 and 9 of the proposed model-based software reliability aspect is displayed.

One of the criteria used to evaluate the models, root mean square error between the predicted and measured. Root mean square error is a suitable tool for comparing the prediction errors by a set of data [11]. This value is calculated using the following equation:

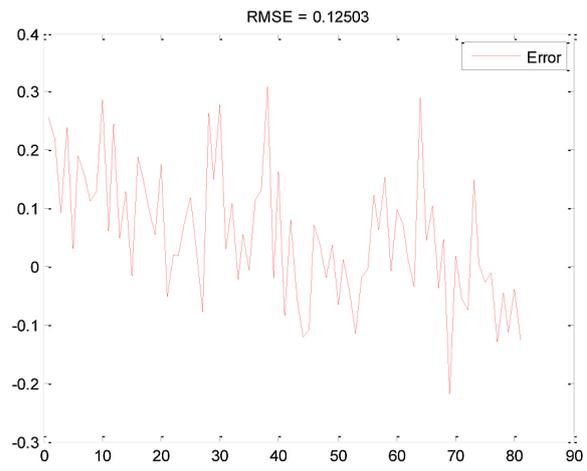
$$RMSE = \sqrt{\frac{\sum_{i=1}^N (y_i - \hat{y})^2}{N}} \quad (3)$$



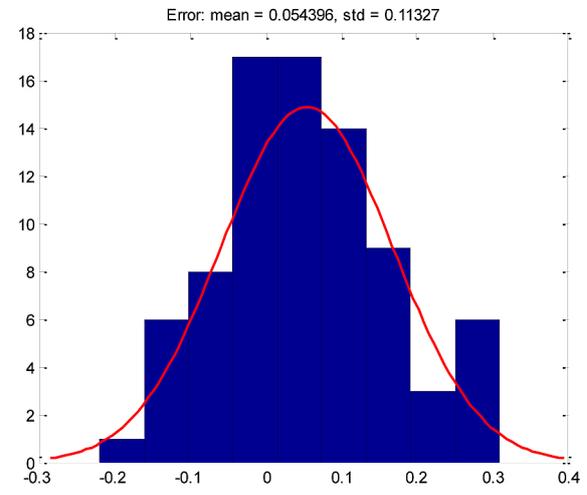
**Fig. 8.** The performance of the proposed model of reliability evaluation based on the aspect oriented softwares



**Fig. 9.** The performance of proposed model of reliability evaluation based on the aspect oriented software



**Fig. 10.** The root mean square error for the model



**Fig. 11.** The mean value and standard deviation error of the proposed model with the diagram histfit

The first indicator of individual calculate the differences observed values with the values predicted by the model and have two power. The square root of the mean and the mean number is the same as RMSE. This index is a measure of the accuracy of the model. It is generally better to match the amount of data if it is lower.

Figure 10 has shown the diagram of root mean square error. As can be seen, the square root of the sum of squared errors of the proposed method is equal to 0.12503, which is a good value for the criterion. In Figure 11, the mean value and standard deviation error of the proposed model with histogram diagram is displayed.

**CONCLUSION**

In this paper, after reviewing the evaluation of software reliability metrics and effective ex-

traction, To provide a method for assessing the reliability of oriented software using fuzzy logic-based applications. The fuzzy system has four inputs and one output. To measure the output value of 81 laws is used. In this system, the root mean square error was equal to 0.12503. Given the low error and using appropriate metrics, this method's accuracy is better than in other methods.

**REFERENCES**

1. Seyster J., et al. Aspect-oriented instrumentation with GCC. In: Runtime Verification, Springer, 2010.
2. Chiu K.-C., Huang Y.-S., Lee T.-Z., A study of software reliability growth from the perspective of learning effects. Reliability Engineering & System Safety, 93(10), 2008, 1410–1421.
3. Kiran N.R. and Ravi V., Software reliability pre-

- diction by soft computing techniques. *Journal of Systems and Software*, 81(4), 2008, 576–583.
4. Figueiredo E., et al, Assessing aspect-oriented artifacts: Towards a tool-supported quantitative method. 2005.
  5. Sant'Anna C., et al. On the reuse and maintenance of aspect-oriented software: An assessment framework. In: *Proceedings of Brazilian Symposium on Software Engineering*, 2003.
  6. Reena Dadhich B.M., Measuring reliability of an aspect oriented software using fuzzy logic approach. *International Journal of Engineering and Advanced Technology (IJEAT)*, June 2012.
  7. Zadeh L.A., Fu K.-S., Tanaka K., Fuzzy sets and their applications to cognitive and decision processes. *Proceedings of the US–Japan Seminar on Fuzzy Sets and Their Applications*. Academic Press University of California, Berkeley, California, July 1-4, 2014.
  8. Ross T.J., *Fuzzy logic with engineering applications*. Wiley, Vol. 761, 2013.
  9. Ceccato M. and Tonell P., Measuring the effects of software aspectization. *Workshop on Aspect Reverse Engineering*, 2004.
  10. Kemerer S.R.C.A.C.F., Suite for object oriented design. *IEEE Trans. on Software Engineering*, 20, 1994.
  11. Chai T. and Draxler R., Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, 7, 2014, 1525–1534.