Estimate reliability of component-based software system using modified neuro fuzzy model

Ravi Kumar Sharm [1]*, Parul Ghandi [2]

[1] *Research Scholar, Manav Rachna International University*
[2] *Assistant Professor, Faculty of Computer Applications, Manav Rachna International University*
*Corresponding author E-mail: ravirasotra@yahoo.com

Abstract

There are many algorithms and techniques for estimating the reliability of Component Based Software Systems (CBSSs). Accurate estimation depends on two factors: component reliability and glue code reliability. Still much more research is expected to estimate reliability in a better way. A number of soft computing approaches for estimating CBSS reliability has been proposed. These techniques learnt from the past and capture existing patterns in data. In this paper, we proposed new model for estimating CBSS reliability known as Modified Neuro Fuzzy Inference System (MNFIS). This model is based on four factors Reusability, Operational, Component dependency, Fault Density. We analyze the proposed model for diffent data sets and also compare its performance with that of plain Fuzzy Inference System. Our experimental results show that, the proposed model gives better reliability as compare to FIS.

Keywords: *Component-Based Software Systems (CBSS); Fuzzy; Modified Neuro Fuzzy Inference System (MNFIS); Neuro Fuzzy; Reliability.*

1. Introduction

Component Based Software Engineering (CBSE) is a specialized form of software reuse. CBSE is concerned with building software from existing components [5]. It includes Commercial Off-The-Shelf (COTS) components, by assembling them together in an interoperable manner. It is very difficult task to achieve a highly reliable software application, even when high-quality, pretested, and trusted software components are combined. Many techniques have been developed to analyze the reliability of component based applications using fuzzy approach [3]. Neuro Fuzzy System can also be taken into consideration for the assessment of quality factors [7]. Software reliability management is defined in IEEE as "The process of optimizing the reliability of software through a program that emphasizes software error prevention, fault detection and removal, and use of measurements to maximize reliability in light of project constraints such as resources, schedule, and performance." Developer needs excellent support in the assessment of the reliability levels of the software that we are trying to develop [8]. Therefore, any reliable system must guarantee and take care of fault prevention, fault tolerance, fault removal, and fault forecasting. The most suitable models for reliability of MNFIS are the ones based on architecture. This paper presents an adaptive neuro fuzzy inference system (MNFIS) model for estimating CBSS reliability. However, this is a relatively time-consuming process. The advantage of an MNFIS over the FIS model is that it combines fuzzy logic with the learning capabilities of neural networks (NNs) to solve this problem. The aim of our proposed model is to integrate the best features of fuzzy logic and neural networks in a CBSS reliability estimation model. Our results demonstrate how this is an improvement over an FIS.

2. Literature review

Several approaches and models have been proposed for reliability estimation of CBSS: some are based on mathematical formulae, and others are based on soft-computing techniques. There are some models that are characterized as architecture based reliability models. We can divide the related work into three categories namely architectural models; mathematical models and soft computing based models. Yacoub Yacoub et al. (2004) proposed an approach to reliability analysis, which is called scenario-based reliability analysis [7]. The above approach introduces component dependency graphs (CDGs) which can be extended for complex distributed systems. We can use the same algorithm, for analyzing as a function of component reliabilities and link reliabilities. This approach is based on scenarios, which can be captured with sequence diagrams, which means that the approach can be automated. The limitation of this approach is that it does not consider failure dependencies among the components. also proposed a model based on a CDG. In this approach, a system's operational profile is given. It is assumed that control flow transits fromcomponent,t i to component j, and the reliability of component j is calculated as $T_{ij} = R_{ij} \cdot W_{ij}$, where T_{ij} = the transition probability fromcomponent,t i to component j, R_{ij} = reliability vector for each subdomain of component j, and W_{ij} = weight vector for each subdomain of component j in transition from component i to component j. Dong et al. (2008) proposed a new model for estimating CBSS reliability in which various complex component relationships are analyzed. The Markov model is used to solve these complicated relationships, which have a large impact on a system's reliability [4]. A new tool was developed to calculate soft-ware application reliability. Huang et al. (2008) proposed a technique based on algebra which provides a framework for describing the syntax and predicting the reliability of a CBSS, implemented on Seth et al. (2010) proposed a minimum spanning-tree-based ap-

proach to estimating CBSS reliability. Goswami and Acharya (2009) proposed an approach to CBSS reliability anal-ysis which takes the component usage ratio, which is the time allotted for a component's execution out of the application's overall execution time, into consideration. Mathematical formulae are used to calculate the usage ratio [1]. This approach can be used in real-time applications, because of its flexibility. Fiondella et al. (2013) proposed an approach based on correlated component failures (CO-COF). In this paper an efficient approach to access the reliability of a software application, considering the component reliabilities, correlation and application architecture is proposed. This proposed approach is based on an algorithm that transforms a Multivariate Burnoulli distribution (MVB) into a joint distribution of the component outcomes. Tyagi and Sharma (2012) proposed an approach based on fuzzy logic for estimating CBSS reliability [5]. In this approach, four critical factors were identified for estimating the reliability of a CBSS, and these were used to design an FIS for the estimation.

3. Soft computing techniques

Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth, and approximation. In effect, the role model for soft computing is the human mind. The guiding principle of soft computing is: Exploit the tolerance for imprecision, uncertainty, partial truth, and approximation to achieve tractability, robustness and low solution cost. The different techniques are used in soft computing are [6] below:-
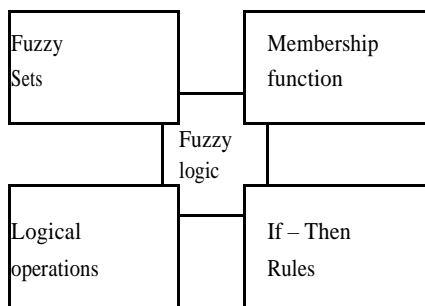
3.1. Fuzzy logic

Fig. 1: Working of Fuzzy System.

Fuzzy prototypes do not have knowledge capabilities, so for knowledge, they necessity adopt procedures from additional methods. It is consequently accepted to marry the two tools. Fuzzy logic is based on the mishmash of four Model/factors shown in Fig. 1.

3.2. Neural network

Neural network tool are software applications that are used to simulate the behavior of artificial or biological neural networks. They focus on one or a limited number of specific types of neural networks. They are typically stand-alone and not intended to produce general neural networks that can be integrated in other software. Simulators usually have some form of built-in visualization to monitor the training process. Some simulators also visualize the physical structure of the neural network. Neural network model also used to represent the complex non-linear associations (input-output relationship) and functions [2].
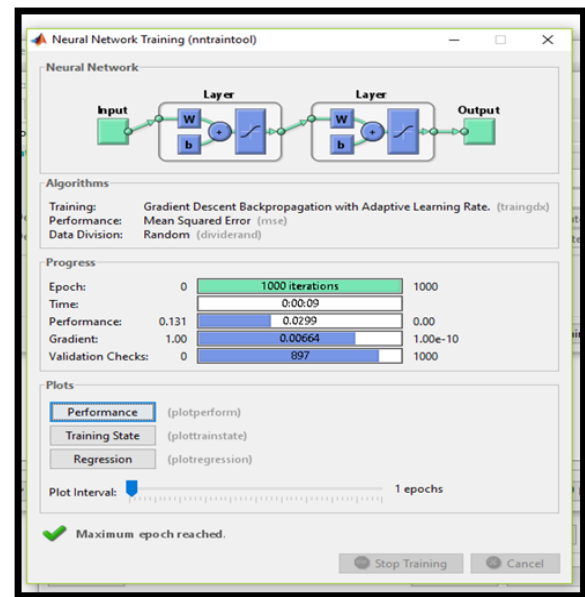

Fig. 2: NN Tool.

3.3. ANFIS

ANFIS were recommended in 1992. These are adaptive systems whose determination is like to FISs [5]. The goal of an ANFIS is to fit in the best structures of fuzzy structures and NNs. The benefits of an ANFIS are:

- An ANFIS can simulate and examine the charting relation among contribution and production data through a learning algorithm to improve the parameters of an agreed FIS.
- An FIS is fully in need of on its relationship functions. An ANFIS includes systems which combine nodes and indicator links, and some knowledge instructions are related with these systems. The systems learn relationships between contributions and productions.
- An ANFIS can be qualified without any essential for the skilled information usually compulsory for the normal fuzzy logic strategy.

4. Proposed model

In this paper, we have proposed Modified Neuro Fuzzy Inference System (MNFIS) for estimating CBSS reliability. In this model, we used four factors, which are. Various researchers worked on three factors for reliability estimation. In this paper, we introduced one more factor called fault density. The additional fourth factor used by us plays an important role to estimate the reliability of component based system, and our results show that the proposed model gives better results as compare to FIS.

1) Reusability of the module
2) Operational profile for the component
3) Component dependency
4) Fault Density

- Reusability of the module

Reusing existing software component is key features in increasing productivity [9], [7] .Reusability is unique among the greatest basic representations of component-based development (CBD). As the term recommends, reusability denotes to how often a component is used in different applications. We nominated reusability as an issue for valuing component reliability because it is held that components that have been used in numerous applications are extremely reliable. Hence the consistency of a factor is directly relative to its reusability: component reliability ∞ reusability.

- Operational profile for the component

The operational profile (OP), a measurable description of exactly how the software determination be used, is important in any software reliability industrial application. An OP is a whole set of

processes with their possibilities of existence. The reliability may be dissimilar for changed OPs. The OP for any factor describes the responses provided to the factor. Extra [9] factors may donate to component reliability, but these are the two issues that have the highest influence on a component's reliability. As the components are stage self-determining, the actions of these two issues will be the similar for all applications.

- Component dependency

In a CBSS, various mechanisms are related to one additional to form a greater application. The mechanisms are independent on each other to achieve their functions; that is, the production of one component may help as an contribution for additional component. This dependency plays an important role for approximating the reliability of the complete application. If components are highly helpless on one another, the reliability of the system will be low. There are numerous mechanisms for describing dependencies between components. For example, an adjacency matrix illustration of dependencies specifies which dependencies exist between components. However, this representation does not explain the exchanges between components. These exchanges play an important role in relation to the dependencies of components and so their reliability, and so we want a different way to characterize the interactions. [5] A planned dependency illustration which is based on linked lists and executed using Hash Map in Java. This illustration can store dependencies along with another information like delivered and required information, which can be used to evaluate several interface- and dependency-related issues.

- Fault density

The complication of any CBSS application can be exact in terms of the amount of mechanisms in that request and their inter connectivity. Are quest that has extra components is said to be more difficult and hence less dependable. Therefore, the difficulty of an application is inversely proportional to its reliability:

5. Implementation and results

5.1. FIS editor

Figure 3 shows the block diagram of the proposed model. The main objective of this model is to evaluate the reliability of component based system. We use four factors in the proposed model and through these factors we evaluated the actual reliability of components. In this section you should present the conclusion of the paper. Conclusions must focus on the novelty and exceptional results you acquired. Allow a sufficient space in the article for conclusions. Do not repeat the contents of Introduction or the Abstract. Focus on the essential things of your article.
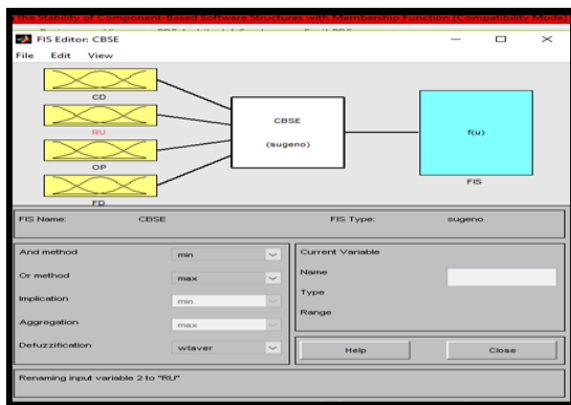
Fig. 3: FIS Editor.

5.2. Membership function for input and output parameter

We have used four inputs and one output parameter namely:- Reusability, Operational, Component dependency, Fault Density.

The membership function is defined as Low (L), Medium (M) and High (H). The range of membership function is 0.0119, 0.25 and 0.575 respectively. Fig 4, 5, 6, 7 and 8 shows fuzzy set taken for each input and output variables in Mamdani Model.
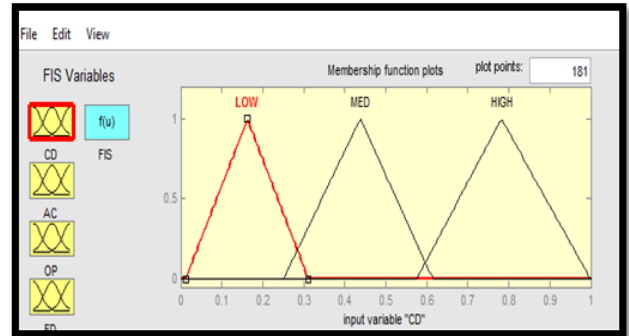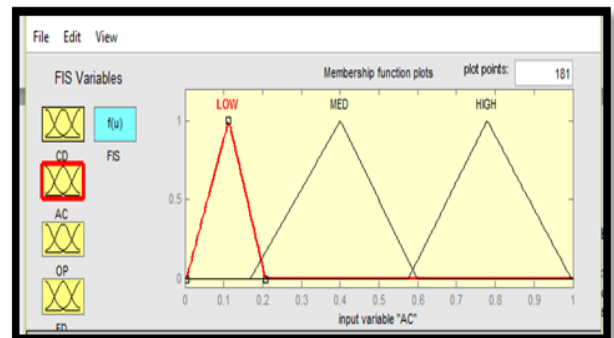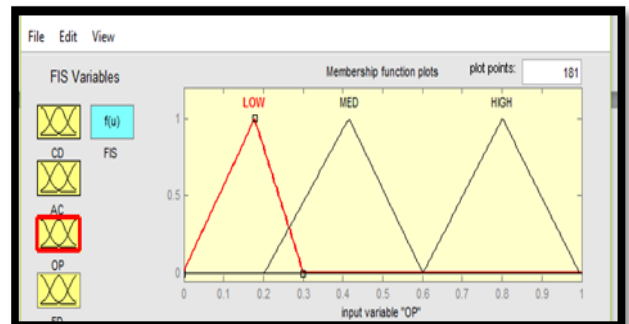
Fig. 4: Input Variable "CD".

Fig. 5: Input Variable "AC".
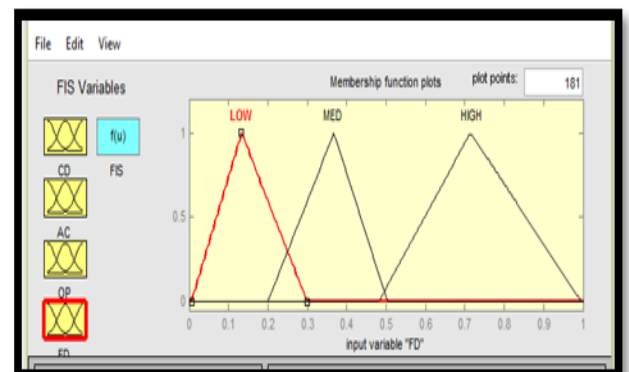
Fig. 6: Input Variable "OP".
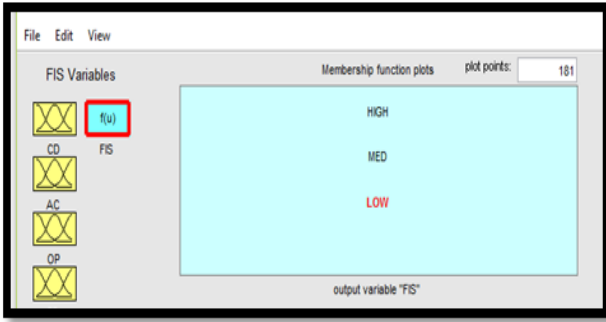
Fig. 7: Input Variable "FD".

Fig. 8: Output Variable "FIS".

5.3. Fuzzy rule

As the factors taken by us has a great impact on component reliability. In order to find effect of these factors we have used fuzzy logic and have designed various fuzzy rules. Figure 9 shows mamdani method for defining fuzzy rules.
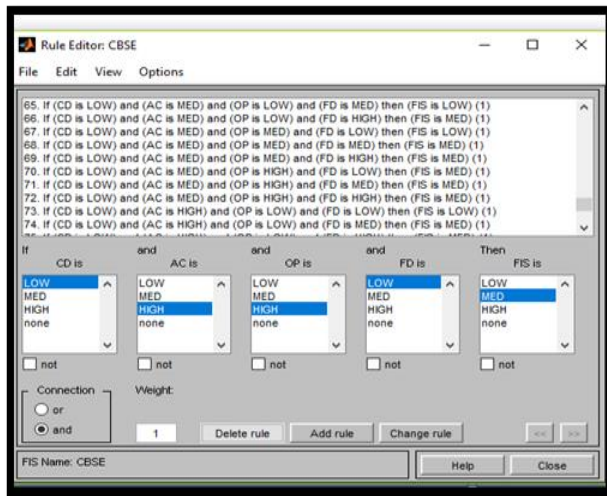

Fig. 9: Rule Editor.

5.4. Rule viewer

Figure 10 shows the proposed rule viewer of proposed model for the estimation of component reliability. Five columns have been used in the proposed rule viewer. The first four column present input variable and the fifth one present the membership function for output variable.
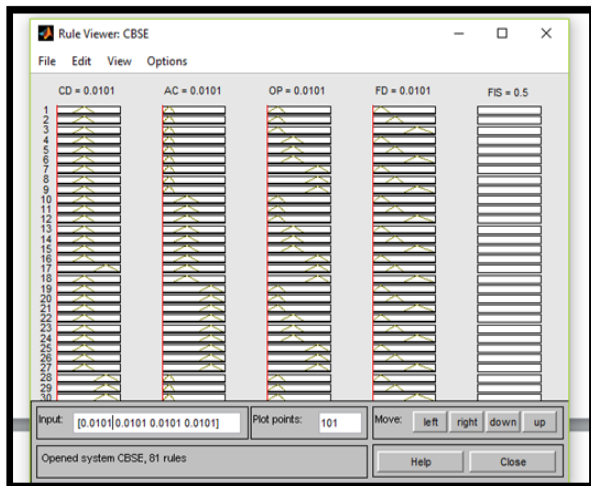

Fig. 10: Rule Viewer.

5.5. Modified neuro fuzzy inference system

First we have loaded previous generated FIS data using your chosen number of MFs and rules. Then we train FIS after setting optimization method, error tolerance, and number of epochs. We apply the data set to Neural Network tool. We have a training data set (NN) that contains desired input/output data pairs of the target system to be modelled. These training and checking data sets are collected based on observations of the target system Figure 11 and 12 shows the. Testing Error and training error mapping from Sugeno FIS to proposed model MNFIS.
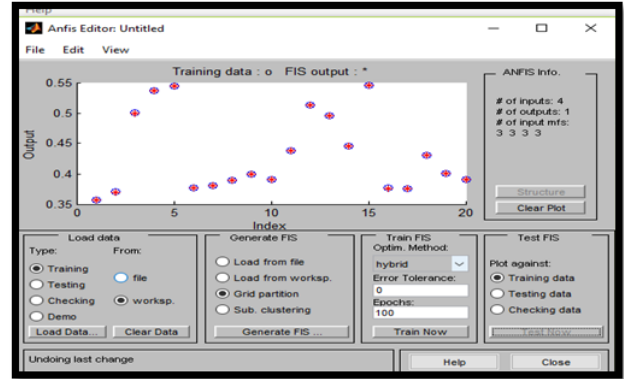

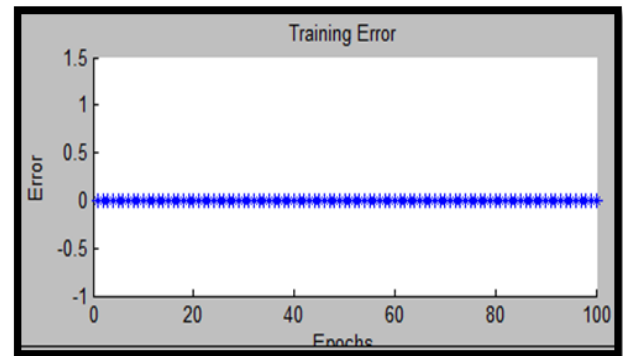**Fig. 11:** Testing Error Mapping Sugeno FIS to MNFIS.

Fig. 12: Training Error Mapping Sugeno FIS to MNFIS.

Table 1: Performance Analysis of FIS and MNFIS

CD	RU	OP	FD	Ori	FIS	MNFIS
0.2222	0.2222	0.2222	0.2222	0.500023	0.5227	0.4837
0.3232	0.3232	0.3232	0.3232	0.536421	0.4532	0.3228
0.3535	0.3535	0.3535	0.3535	0.543919	0.4483	0.339
0.687	0.687	0.687	0.687	0.377368	0.3762	0.2526
0.6666	0.6666	0.6666	0.6666	0.380438	0.3604	0.2541
0.6364	0.6364	0.6364	0.6364	0.388793	0.4772	0.4079
0.9697	0.9697	0.9697	0.9697	0.399926	0.5178	0.3896
0.9193	0.9193	0.9193	0.9193	0.390056	0.3013	0.389
0.1202	0.1202	0.1202	0.1202	0.437858	0.5734	0.4942
0.2525	0.2525	0.2525	0.2525	0.51312	0.571	0.4984

Table 1 shows the performance analysis of existing as well as proposed system using different data sets taken from online data repository. The results shows that MNFIS gives better results as compare to existing FIS.

6. Conclusion

Reliability plays an important role in selecting a component for software systems. It helps in better understanding and improving the efficiency for the system. This paper proposed a neuro fuzzy model for estimating cbss reliability called modified neuro fuzzy inference system. Many approaches to estimate cbss reliability have been proposed. The previous models proposed by various researchers depend on three factors. In our work, there is one additional factor named, fault density. The proposed model is based on four factors reusability, operational, component dependency, fault

density. We compare the proposed model with existing fis and our results give better reliability as compare to other approaches. Our work compares the sensitivity analysis of the two models and shows which one is better. Our work has concluded that the mnfis is a more stable than the fis model.

References

[1] Ashish Seth, Himanshu Agarwal and Ashim Raj Singla, 2014. Reliability Estimation of Services Oriented Systems Using Adaptive Neuro Fuzzy Inference System. Journal of Software Engineering and Applications, 2014, 7, 581-591. https://doi.org/10.4236/jsea.2014.77054.

[2] Charu Singh, Amrendra Pratap and Abhishek Singhal, 2014 .Estimation of Software Reusability for Component based System using Soft Computing Techniques, 2014 5th International Conference- Confluence The Next Generation Information Technology Summit (Confluence) 793.

[3] Gopal Prasad Jaiswal and Ram Nivas Giri, 2015. Software Reliability Estimation of Component based Software System using Fuzzy Logic. International Journal of Computer Applications (0975 – 8887).

[4] Jyoti Agarwal , Renuka Nagpal and Rajni Sehgal, 2014 Reliability of Component based Software System using Soft Computing Techniques – A Review, International Journal of Computer Applications (0975 – 8887) Volume 94 – No 2, May 2014.

[5] Kirti Tyagi and Arun Sharma, 2014. An adaptive neuro fuzzy model for estimating the reliability of component-based software system. Applied Computing and Informatics (2014) 10, 38–51. https://doi.org/10.1016/j.aci.2014.04.002.

[6] Kuldeep Singh Kaswan, Sunita Choudhary and Kapil Sharma, 2015. Software Reliability Modeling using Soft Computing Techniques: Critical Review, I.J. Information Technology and Computer Science, 2015, 07, 90-101. https://doi.org/10.5815/ijitcs.2015.07.10.

[7] Parul Gandhi, 2016. Assessment of Components Generality Using Fuzzy Approach to Optimize Software Development Cost. International Journal of Computer Science and Information Technology, PP. 647-654.

[8] Shalini Goel, 2014. Neuro Fuzzy based Approach to Predict Component's Reusability. International Journal of Computer Applications, PP. 0975-8887.

[9] Yogesh singh, pradeep kumar bhatia and omprakash sangwan, 2011. Software reusability assessment using soft computing techniques, ACM SIGSOFT Software Engineering Notes Page 1.