Improved Software Reliability Prediction through Fuzzy Logic Modeling

Sultan Aljahdali
College of Business Administration
Jeddah, Saudi Arabia
aljahdali@cba.edu.sa

Narayan C. Debnath
Winona State University
Winona, MN, USA
ndebnath@winona.edu

Abstract: *Software Reliability is a key concern of many users and developers of softwares. Demand for high software reliability requires robust modeling techniques for software quality prediction. This paper presents a new approach to software reliability assessment by using Fuzzy logic. The Series of Fuzzy logic modeling associated with different time segments can be directly used as a piecewise linear model for reliability assessment and Problem identification, which can produce meaningful results early in the testing process.*
The model has been applied to three different applications using fuzzy logic and normalized root mean of the square of error as an evaluation criterion. Results show that the fuzzy model adopted has good predictive capability.

Keywords: Software Quality, Software Reliability, Piecewise Linear Model, Fuzzy logic.

1. Introduction

As Software reliability is becoming more and more important in software industry various techniques are required to discover faults in the development of software. However as reliability of a software is measured in terms of failure it is impossible to measure reliability before the software is developed completely. Software reliability is the most extensively studied quality among all the quality attributes [9].

In the past few years a number of software reliability assessment models have been developed to solve software reliability models. These software models have been developed in response to the urgent need for software engineers, system engineers and managers to quantify the concept of software quality prediction. These models were useful in cases like managing reliability, managing project changes and monitoring test programs. Some of the models that have been developed for software quality prediction are: exponential order statistical model [15], logic regression [11]], Case based reasoning [10], Artificial Neural Networks [13], and Optimal Set reduction [8] The main objective of these models are to help predict which modules are error prone which in turn can help developer to focus on many aspects of maintenance cycle [14].

In this paper we propose a new approach towards software reliability assessment using fuzzy logic and Normalized Root of Mean of the Square of Error *(NRMSE)* criterion as an evaluation criterion. The rest of the paper is organized as follows: After a brief examination of the existing techniques for software reliability using fuzzy logic in section 2 a new approach using fuzzy logic is presented. Section 3 discusses the evaluation criterion used for the model. Prediction of the Software reliability for the data set and estimation of the parameters for the fuzzy model is presented in section 4. Section 5 presents the results of the prediction of software reliability model using fuzzy logic for 3 different projects. Finally a summary of the work done and future research directions for the proposed strategy are discussed in section 6.

2. Fuzzy logic for software reliability

A software system is generally characterized by its size, usually exceeding several thousand lines of code, high complexity, and diverse functionality, many components developed over long period, a large user population and diverse usage environments. Assuring software to very high levels of reliability is one of the most difficult yet important challenges confronting the software industry.
As seen above many models have been developed and much research has been devoted towards assessment of software reliability. However against this back ground only few papers have addressed the use of fuzzy models for assessing software reliability which is surprising given their rapid adoption into other areas such as control, classification and decision making. This paper addresses the problem of software reliability prediction using fuzzy logic.
Yuan et. Al in [19] used fuzzy subtractive clustering integrated with module order modeling for software quality prediction. First Fuzzy Subtractive clustering is used to predict the number of faults then module order modeling is used to predict whether modules are fault prone or not. Xu et al [18] introduced the fuzzy nonlinear regression (FNR) modeling technique as a method for predicting fault ranges in software modules. A case study of full scale industrial software

system was used to illustrate the usefulness of FNR Modeling. Jeff Tian in [17.] assessed software reliability by grouping data into clusters. The series of data clusters associated with different time segments are used directly as a piecewise linear model for reliability assessment and problem identification. The model is evaluated in the testing of two large software systems from IBM. Adnan et al in [1] explored the potential of prediction techniques which have been used for assessing software reliability.

This research is a continuation of work done for analyzing reliability of software systems using various methods, which include Genetic algorithms [5], RBF Neural Networks [6] and Parametric and Non Parametric Methods [7, 2]. Based on the work done we propose a new software reliability assessment model that is based on fuzzy logic and Normalized root of mean of the square of error (NRMSE) evaluation criterion. This new model has been successfully implemented for the projects Real Time and Control applications, Military applications and Operating System applications and shows promising predictive capability.

3. Evaluation Criteria

The models performance were measured in terms of the *NRMSE*

$$\frac{1}{n-1}\sqrt{\frac{\sum_{k=1}^{n}(y(k)-\hat{y}(k))^2}{\sum_{k=1}^{n}(y(k))^2}}$$

Where *y (k)* is the actual accumulated faults and $\hat{y}$ and k is the estimated accumulated fault.

4. Prediction using fuzzy logic

4-1. Data set

The DACS Services at the Department of Defense (DoD) Software Information Clearinghouse provides an authoritative source for the state of the art software information, supplying technical support for the software community. John Musa of Bell Telephone Laboratories compiled a software reliability database. His objective was to collect failure interval data to assist software managers in monitoring test status, predicting schedules and to assist software researchers in validating software reliability models. These models are applied in the discipline of Software Reliability Engineering. The dataset consists of software failure data on 16 projects. Careful controls were employed during data collection to ensure that the data would be of high quality. The data was collected throughout the mid 1970s. It represents projects from a variety of applications including real time command and control, word processing, commercial, and military applications.

4-2. Fuzzy Logic Structure

Fuzzy logic is a form of logic used in systems where variables can have degree of truthfulness or falsehood ness. With fuzzy logic, the outcome of an operation can be expressed imprecisely rather than as a certainly. A fuzzy model is a mapping between linguistic terms, attached to variables. Therefore the input to and output from a fuzzy model can be either numerical or linguistic.

The model structure was based on Takagi-Sugeno (TS) fuzzy model, which is suitable for large class of nonlinear systems. In a non-linear discrete time system, the relationship between system input u(k) and output y(k) at time k is given by:

$$y(k)= f\,(u(k - 1), y(k - 1))$$

The function f is a static function and fuzzy models of different types can be used to approximate this relationship function f. One of the commonly model is the NARX (Nonlinear Auto-Regressive with eXogenous input) model, represented by:

$$y(k) = f\,(y(k - 1), y(k - 2),..., y(k - n + 1), u(k - 1), u(k -2)..., u(k - m + 1)).$$

where $u(k - 1) \ldots u(k - m + 1)$ and $y(k - 1) \ldots y(k - n + 1)$ represents the past model inputs and outputs, respectively. Another model which is special case of the NARX is the NAR (Nonlinear Auto-Regressive) model which, can be represented as:

$$y(k) = f\,(y(k - 1), y(k - 2),..., y(k - n + 1)).$$

The assessment was done for three projects, Real Time and Control applications, Military applications and Operating System applications.

Once the model structure is selected the next step will be to estimate the parameters of the fuzzy model. These parameters include the antecedent membership functions and the consequence polynomials. An additional parameter needs to be selected, that is the number of rules (clusters), which need to be specified by the user. The methodology to build a fuzzy model can be described in the following steps:

- Using the flow sequence of measurements *y (k - 1), y (k - 2), y (k - 3), y (k - 4)* and the user defined parameters we form the nonlinear regression problem to find y(k).
- Compute the antecedent membership function from the cluster parameters.

- Given the antecedent membership functions, estimate the consequence parameters by the least-square method.

5. Experiment and Evaluation

The results of the experiment are shown in Figures 1 to 6, which shows the prediction of the software reliability using Fuzzy logic for the projects Military applications, Operating Systems applications and Real-time and Control applications and the membership function for each project based on the normalized root mean of the square of the error *NRMSE* as an evaluation criterion. The results obtained in Figures 1 to 6 are based on training and testing of sample data. Training was done on 70% of the sample data and testing was done on the entire data (100%). Figures 1 to 6 shows the actual and estimated responses using fuzzy logic for the projects Real-time and Control applications, Military applications, and Operating Systems applications respectively. The Fuzzy logic model performs very well through out all the experiments. Also, fuzzy logic model has the good predictive capability in all the projects for example the projects, military applications #40, and operating system #ss1c, the *NRMSE* are 1.1081, 1.2901, but in the real time control project # 1, the *NRMSE* is 0.9358, as shown in the Table 1, which is the lowest when compared between the two projects. In addition table 2 shows the result of the variance accounted for (VAF) for each project during the testing part and the training part, also the NRMSE during the training and testing part.

Military # 40	Real-time Control #1	Op. sys. ss1c
1.1081	0.9358	1.2901

Table 1: NRMSE values for the projects Military #40, Real-time Control # 1 and Operating System ss1c.

Project Name	Tr-VAF	Ts-VAF	Tr-NRMSE	Ts-NRMSE
Real-time	99.8854	95.2851	0.3257	1.1092
Military	99.9668	98.8370	0.4507	41.7954
Operating	99.9885	99.9056	0.5612	4.1335

Table 2: Results for the developed Fuzzy logic showing the variance accounted for (VAF) for each project

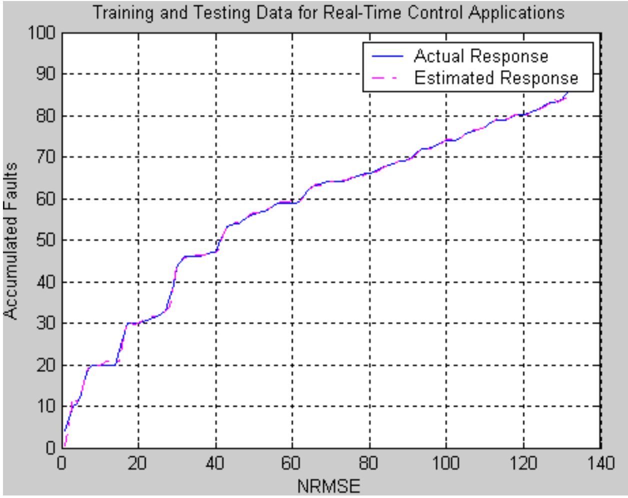

Figure 1: Actual and Estimated responses using fuzzy logic model: Real-time Control Applications.

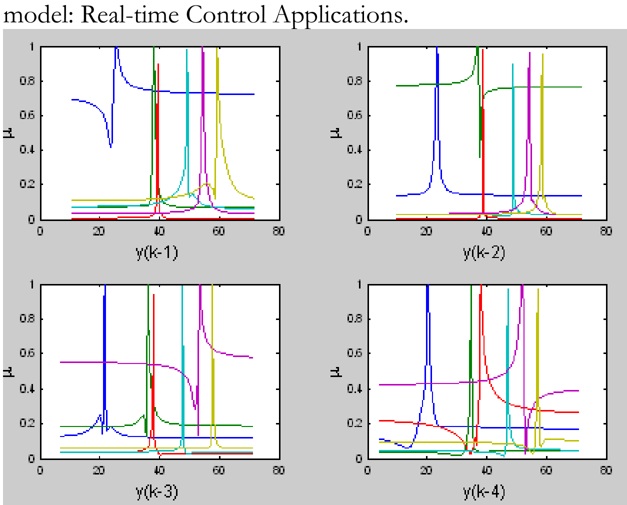

Figure 2: Membership Function for Real-time Control Application.

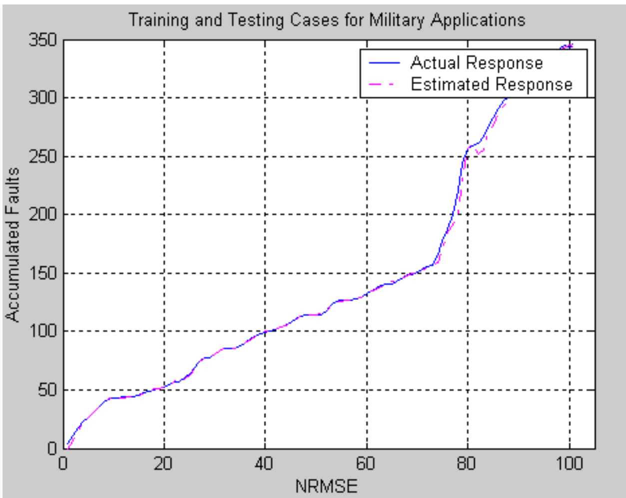

Figure 3: Actual and Estimated responses using fuzzy logic model: Military Applications.

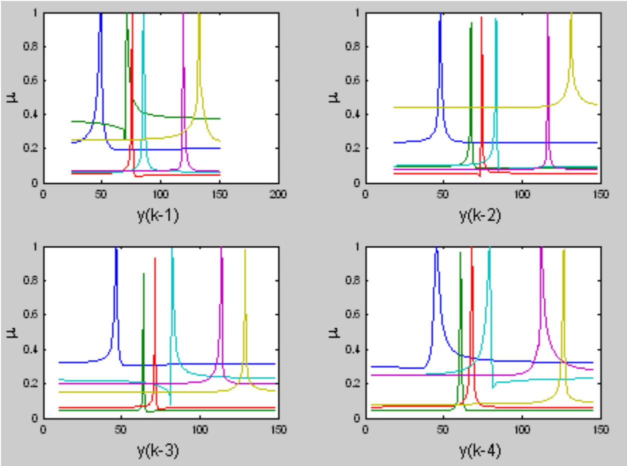

Figure 4: Membership function for Military Application project.

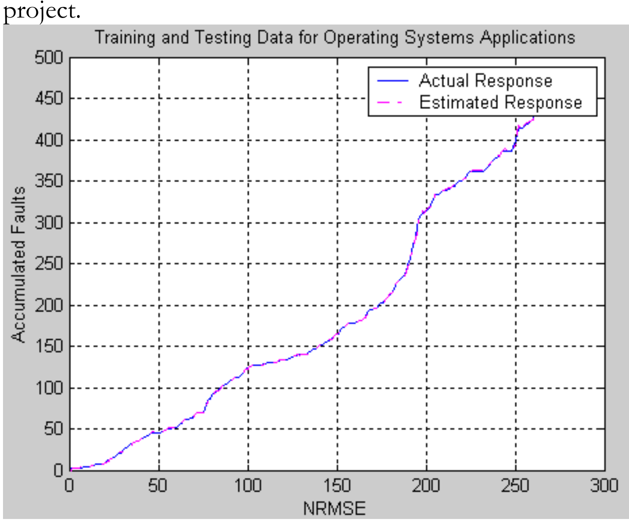

Figure 5: Actual and Estimated responses using fuzzy logic model: Operating Systems Applications.

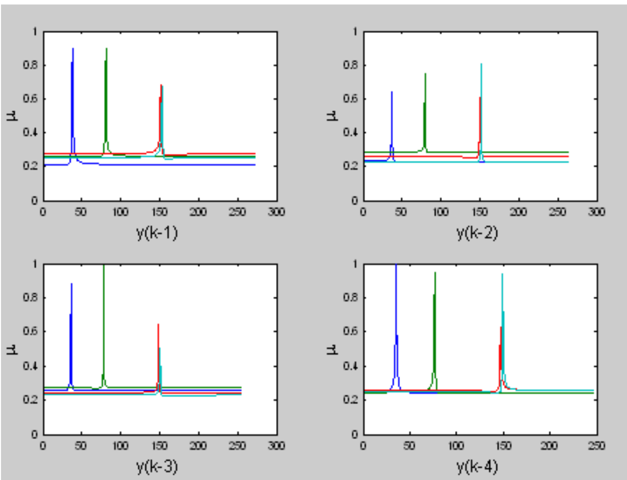

Figure 6: Membership function for Operating Systems project.

6. Conclusion and Future Work

We have shown that fuzzy logic can be used for building software reliability growth models. Fuzzy logic was able to provide models with small NRMSE. The entire system of software reliability research is considered useful for software development and testing industry. At the present we are investigating the use of genetic programming to solve the software reliability growth modeling problem.

7. Acknowledgment:

We would like to thank College of Business Administration for supporting this research (CBA).

8. Bibliography

1. Adnan, W,A, Mashkuri, H,Y, "An Integrated Neural Fuzzy System of Software Reliability Prediction", IEEE Software.

2. Aljahdali, S., Alduhaiman, K., and Francis, M. "Parameter Significance for Email Attack Detection", Proceedings of the CSITeA'03 International Conference on Computer Science, Software Engineering, Information Technology, e-Business, and Applications, June 5-7, 2003, pp.

3. Aljahdali, S., Sheta, A., and Habib, M. "Software Reliability Analysis Using Parametric and Non-Parametric Methods" , Proceedings of the ISCA 18th International Conference on Computers and their Application, March 26-28, 2003, pp. 63-66.

4. Aljahdali, S., Abdullah, I., and Sibley, E. "Avoiding Redundancies and Conflicts in Security Protocols" Proceedings of the ISCA 18th International Conference on Computers and their Application, March 26-28, 2003. pp. 321-324.

5. Aljahdali, S., Sheta, A., "Modeling of Nonlinear Systems Using Genetic Algorithms", Proceedings of the IASTED International Conference on Artificial Intelligence and Application, 4-7, September 2001, pp.220-224.

6. Aljahdali, S., Sheta, A., and Rine, D., "Predicting Accumulated Faults in Software Using Radial Basis Function Network", Proceedings of the ISCA 17th International Conference on Computers and their Application, 4-6, April 2002, pp. 26-29.

7. Aljahdali, S., Sheta, A., and Rine, D., "Prediction of Software Reliability: A Comparison between regression and neural network non-parametric

Models", Proceeding of the IEEE/ACS Conference, 25-29, June 2001,pp.470-471.

8. Briand, L.C, Basili, V.R, Hetmanski , C.J. , " Developing Interpretable models with optimized set reduction for identifying high-risk software components", IEEE Transactions on Software Engineering, Nov. 1993, 19(11):1028-1044.

9. Fenton, N.E., Pfleeger, S.L., "Sofware Merics : A Rigorous & Pracical Approach", Second Edition, PWS Publishing Company, 1997.

10. Ganesan, K , Khoshgoftaar ,T.M., Allen, E.B. ," Case based Software quality prediction", International Journal of Software Engineeering and Knowledge Engineering, 1999, 9(6), Forthcoming.

11. Khoshgoftaar, T.M., Allen, E.B., "Logistic Regression Modeling of Software Quality", International Journal of Reliability, Quality and Safety Engineering, Dec. 1999, 6(4), Forthcoming.

12. Khoshgoftaar , T.M.,Allen, E.B, Jones, W.D., Hudepohl, J.P., " Which Software modules have faults that will be discovered by Customers?", Journal of Software Maintenance: Research and Practice, , Jan 1996,11(1):1-18.

13. Khoshgoftaar, T.M. , Pandya A.S., Lanning, D.L. ,"Application of Neural Networks for Predicting Faults", Annals of Software Engineering, 1995,1:141-154.

14. Khoshgoftaar , T.M.,Allen, E.B, Kalaichelvan, K.S, Goel, N, "Early Quality Prediction: A case study in Telecommunications", IEEE Software, Jan 1996, 13(1): 65-71.

15. Miller, D.R., "Exponential order statistic models of software reliability growth," IEEE Transactions on software Engineering, SE-12(1), pp. 12-24, 1986.

16. Schneidewind, N.F., "Methodology for validating Software Metrics", IEEE Transactions on Software Engineering, May 1992, 18(5):410-422,.

17. Tian, J, " Better Reliability Assessment and Prediction through Data Clustering", IEEE Transactions on Software Engineering, October 2002, Vol. 28, No. 10.

18. Xu, Z, Allen, E, B, "Prediction of Software Faults Using Fuzzy Nonlinear Regression Modeling", IEEE Software, 2000, 281-290.

19. Yuan, X, Ganesan, K, "An application of Fuzzy clustering to software quality prediction", IEEE Software, 2000.