

DEVELOPMENT OF A MATHEMATICAL MODEL FOR PREDICTIVE ANALYSIS TO AVOID SOFTWARE PROJECT FAILURE

Abieva Akerke Zhandoskyzy

Student, International IT University, Kazakhstan, Almaty

Mohamed Ahmed Hamada

научный руководитель, PhD, International IT University, Kazakhstan, Almaty

Abstract. Software engineering is the process of developing software by utilizing applications of computer engineering. In the present day, predicting the reliability of the software system become a recent issue and an attractive issue for the research area in the field of software engineering. Different techniques have been applied to estimate and predict the reliability of a system.

To make new software from the beginning is a difficult task. Component-Based Software Engineering (CBSE) helps in minimizing these efforts in making new software because it utilizes factors like reusability, component dependency, and component interaction that results in decreasing complexity of the system. Soft computing may be applied to estimate reliability. A new model is proposed to estimate the reliability of Component-based Software (CBS) using series and parallel reliability models and later on, the proposed component-based software reliability model is evaluated using two soft computing techniques- Fuzzy Logic and PSO.

Keywords: predictive analysis, software project, a mathematical model.

The global digitalization of the world economy, which is taking place before our eyes, is fundamentally changing markets, socio-economic relations, ways, and possibilities of doing business.

This is most noticeable in high-tech industries (IT, telecommunications, finance, trade), but in the foreseeable future, the processes of the so-called digital transformation will become total.

Recent technological trends (Internet of Things, big data analytics, augmented reality, cognitive and geospatial technologies, etc.) have quickly become a reality to be reckoned with. Some experts (for example, the international consulting company Arthur D. Little) consider the digital transformation of the business to be the basis for the coming years.

A similar opinion is shared by other analysts of global companies that study the global IT market: Gartner, Accenture, IBM, Deloitte, and others. According to Accenture, the use of digital technologies in 2020 can add \$1.36 trillion to the GDP of the world's leading economies, and IDC predicts a 16.8% annual growth in global spending on digital transformation technologies [1].

According to a 2016 study by Gartner, by 2020, 100 million consumers will be shopping in virtual reality, 30% of online service interactions will be screenless, and for every dollar enterprise invest in innovation, they will have to spend an additional 7 dollars to support the core business [2].

As a result, as shown in [3], the success of the development of most modern companies will largely

depend on the effectiveness of mastering digital business tools, the ability of companies to transform and adapt their own strategy, products, and business processes to new economic and technological conditions.

How a corporation needs to build its management information system and how analytical models and applications should be used to support management decision-making so that they meet the requirements of digital transformation, we will try to answer in this article.

A software reliability model is a mathematical function that relates reliability and other metrics with time, software failure, error content, testing effort, and error propagation. Around 300 models were developed by eminent researchers and software engineers from 1970s. But the model applicability is limited, and no model is universally suitable for all projects and situations.

The non-homogenous Poisson process (NHPP) models are special class of software reliability models, and finds applications in many software projects. Goel-Okumoto (G-O) model and Delayed S-Shaped model are prominent models in the NHPP category. G-O model was proposed by Goel and Okumoto in the year 1979. Yamada proposed delayed s-shaped model in the year 1984. Parameter estimation is one of the important tasks in software reliability modeling and estimation. Probability plotting, and curve fitting are traditional parameter estimation techniques used before the evolution of computers. After computer evolution techniques like Maximum Likelihood Estimation (MLE) and Least Squares Estimation (LSE) were employed for estimation. Many heuristics and algorithms were developed by researchers to get the optimal parameter values. After estimation, optimization is used to arrive at the likelihood of the real value. Evolutionary optimization techniques like Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC) optimization, and firefly search algorithm have given promising results.

There are various methods of reliability prediction such as architecture based models, Gokhale model [3], Laprie model [4], Shooman model [5], Yacoub model [1], Everett model [6], etc. These models are based on state, path, and behavioral addition. The common parameters used in these models were availability, errors in arithmetic algorithms, mean repair times, component reliabilities, transition probabilities, components dependencies, operation profile, transition probabilities, failure behavior of components and interfaces, constant failure rate, number of faults, execution of a set of components, series and parallel combination of components factors used, average execution time of component etc. The reliability predicting models are related to the factors like effort, Kilo Delivered Lines of Code (KDLOC), fault density, reusability, availability, performance, serviceability, capability, maintainability, interface complexity, adaptability, fitness value and computational time, average execution time, reliability, probability, failure rate, fitness function, ants, etc.

Predictive analytics implies a set of data mining methods aimed at forecasting. For forecasting, data on the characteristics of the equipment, its use, external influences, and the state of the operating environment are used. The process of creating predictive analytics is based on the automated processing of large amounts of data. Processing is aimed at identifying arrays of patterns and dependencies, with the help of which predictive models are built. For this, statistical methods are mainly used. An example of implementing a predictive analytics method is fault tree analysis. The failure tree (accidents, incidents, consequences, undesirable events) is a multi-level graphological structure of causal relationships obtained as a result of tracing hazardous situations in reverse order in order to find possible causes of their occurrence.

The value of the fault tree method is as follows:

- allows showing unreliable places in an explicit form;
- makes it possible to perform qualitative or quantitative analysis
- system reliability;
- allows specialists to focus on individual tasks in turn
- specific system failures;
- provides deep insight into system behavior and penetration
- in the process of her work.

The disadvantages of a fault tree are as follows:

- implementation of the method requires a significant expenditure of funds and time;
- a fault tree is a Boolean logic diagram which show only two states: working and failed;
- it is difficult to consider the state of partial failure of elements;
- requires reliability specialists to have a deep understanding of the system and specifically consider only one specific failure each time;
- the fault tree describes the system at a certain point in time (usually steady-state) and sequences of events can be shown with great difficulty, and sometimes it turns out to be impossible.

The fault tree analysis method contributes to a thorough analysis of the causes of failures in technical systems and the development of measures that are most effective for their elimination. Such an analysis is carried out for each period of operation, each part or system as a whole [7].

The main advantage of predictive analytics is that everything can be predicted. The disadvantages include the fact that integration and the right approach to predictive analytics requires a lot of effort and energy, and is also very expensive. Therefore, it is quite difficult to implement all this, but it can be used almost everywhere.

In the previous section, five different factors of CFSR have been discussed that helps in assessing the reliability of components.

A new model is proposed with the integration of these factors for predicting reliability. The reliability can be estimated directly by the proposed model. The proposed approach uses fuzzy logic and PSO for the assessment of the proposed model. The model uses five factors of CBS i.e. component interaction, component dependency, complexity, reusability and failure rates. Fuzzy logic shows better outcomes than PSO. The mathematical equation is built by assessing the relationship between reliability and other factors [16],[20], [15], [26].

$$\text{Reliability} = (\text{Reusability} \times \text{Component Interaction}) / (\text{Component Dependency} \times \text{Complexity} \times \text{failure}) \quad (1)$$

Using series and parallel method of calculating the reliability of a system, the mathematical model can be expressed as shown in figure3. The reliability is shown in Eq. 1 can be expressed as:

$$\text{Reliability} = \text{Reu} * \text{Ci} * (1 - (1 - \text{Com})(1 - \text{Cd})(1 - f)) \quad (2)$$

In equation 4, Reu is the probability of occurrence of reusability, Ci is the probability of occurrence of Component Interaction, Com is the probability of occurrence of complexity, Cd is the probability of occurrence of Component Dependency, f is the probability of occurrence of failure rate. Equation 2 is formulated so that the two parameters reusability and component interaction effect directly and remaining factors affect less on overall reliability.

It is difficult to make new product/software from a new stage. Identification and parameter consideration is also a hard task. Many reliability models have been made using different considerations of factors. In the present day, soft computing becomes popular in the field of estimating and predicting software reliability. In this paper, a new mathematical model is proposed to guesstimate the reliability of Component-based Software (CBS) using series and parallel reliability models approach. The output of the proposed model is compared with the outputs of soft computing techniques PSO and Fuzzy logic to compare the best value of reliability.

The result shows that Fuzzy logic is more compatible for predicting reliability as compared to PSO. It is observed that the proposed reliability model has a lower error rate in predicting CBSE reliability as compared to reliability prediction utilizing fuzzy logic and PSO. Other factors may be added to enhance the proposed model for future work.

Adding more factors in the proposed model results in an increase in complexity due to the formation of a large combination of parameters.

References:

1. Song KY, Chang IH, Pham H. NHPP Software Reliability Model with Inflection Factor of the Fault Detection Rate Considering the Uncertainty of Software Operating Environments and Predictive Analysis. *Symmetry*. 2019; 11(4):521. <https://doi.org/10.3390/sym11040521>
2. C. Leboucher, H.-S. Shin, R. Chelouah, M. S. Le, P. Siarry, and M. Formoso, "An enhanced particle swarm optimization method integrated with evolutionary game theory," *IEEE Trans. Games*, vol. 10, no. 2, pp. 221–230, Jan. 2018.
3. C. Diwaker and P. Tomar, "Evaluation of swarm optimization techniques using CBSE reusability metrics," *Int. J. Control Theory Appl.*, vol. 2, pp. 189–197, Jan. 2016. [15] G. P. Jaiswal and R. N. Giri, "Software reliability estimation of component based software system using fuzzy logic," *Int. J. Comput. Appl.*, vol. 127, no. 7, pp. 16–20, 2015.
4. K. Tyagi and A. Sharma, "An adaptive neuro fuzzy model for estimating the reliability of component-based software systems," *Appl. Comput. Inform.*, vol. 10, nos. 1–2, pp. 38–51, 2014.
5. H. Singh and V. K. Toora, "Neuro fuzzy logic model for component based software engineering," *Int. J. Eng. Sci.*, vol. 1, pp. 303–314, Jul. 2011.