Software Maintainability Prediction by Using Multiple Criteria Decision Making Technique with Fuzzy-AHP Methods

Rachna Jain\textsuperscript{1}, Karan Snotra\textsuperscript{2}, Sunil Kumar Khatri\textsuperscript{3}, Rana Majumdar\textsuperscript{4}
\textsuperscript{1,2,3,4} Amity Institute of Information Technology, Amity University Uttar Pradesh, Noida, India.
\textsuperscript{1}jain.rac16@gmail.com, \textsuperscript{2}karansnotra94@gmail.com \textsuperscript{3}sunilkkhatri@gmail.com, \textsuperscript{4}rana.majumdarwb@gmail.com

October 9, 2018

Abstract

Software maintainability prediction is one of the distinct challenges in the area software engineering. It is very difficult to predict the reliability of software. In literature, various approaches in the form of a genetic algorithm, automated testing, application of neural networks were used by researchers to predict reliability and hence maintainability of software. Multiple criteria decision-making technique is one of the emerging methods for predicting software maintainability. Truly Multi Criteria Decision Making is a selection process in which we find the most satisfying solution among the available solutions subjects to certain constraints. Fuzzy logic is again a process to find the degree of truth instead of its truthiness and falseness. Fuzzy logic includes 0 & 1 as extreme cases of truth but also use various stages of truth in between. In this work, we apply the combined approach of Fuzzy-MCDM to figure out the best suitable alternatives between all feasible alternatives to predict software maintainability.
1 Introduction

Software maintainability is a process of identifying and removing software bugs within specific time and by allocating specified resources. By predicting software maintainability one can increase the efficiency of software and may add new functionalities. Nowadays the common objective of any organization is to work on software maintainability for the purpose of sustainability and for having a greater impact on the market.

In general software maintainability is addressed by the following components, they are namely

- Testability
- Modifiability
- Understandability
- Stability
- Readability

These five factors are used by practitioners to influence software maintainability and software quality. Eventually, multiple criteria decision making (MCDM) involves Multi-Criteria Analysis for finding an appropriate solution among all existing feasible possibilities. Basically, Multi-Criteria Analysis requires a controller called decision maker to provide a qualitative assessment. Many small computer programs are available for this purpose and these Programmes are very simple to use and developed to support technical components of Multi-Criteria Analysis (MCA). From all available methods, Fuzzy Analytic Hierarchy Process (F-AHP) is a very effective method of tackling the problems related with Multiple-Criteria Analysis. The following paper is structured as: Literature Review has been discussed in Section 2. Section 3 gives the general concepts of software maintainability factors followed by the description of the proposed MCDM approach in Section 4 and research methodology in Section 5. The procedure for fuzzifying the software metrics have been discussed in Section 6. Section 7 explains the evaluation performed on the proposed model with the comparative analysis in section 8 and then the conclusion in section 9.
2 LITERATURE REVIEW

Software maintainability prediction always attracted researchers; here this section covers some of the work carried out by various researchers with their impact on the overall software maintainability prediction. According to Goel et al. (2012) factors affecting maintainability hierarchy are analysability, changeability, stability, and testability. These factors vary with design complexity metrics like NOC, CBO, DIT, WMC and RFC (Goel et al., 2012). Ghosh et al. (2012) discussed the sub-factors of maintainability namely size, complexity, coupling, and inheritance. NOC, CBO, NA, CBO, IUB, and CBO, U are taken as metrics for design description and change-ability prediction in Chauman et al. (2000). Kumar et al. (2010) developed a three factor model where complexity varies with complexity of attributes (CMPAt), complexity of operations (CMPOp), and complexity of nested components (CMPNested). Based on the fuzzy approach Goel et al. (2012), proposed a method of quantitative evaluation of object oriented languages while using fuzzy AHP approach. The model as illustrated in ISO/IEC-9126 was taken as the support model for maintainability. Dubey et al. (2012) suggested a framework where internal quality attributes of size, complexity, coupling, and inheritance are used as quality determining factors that involves object oriented metrics such as LOC, WMC, DIT, and CBO. Dubey et al. (2012), projected model for maintainability prediction of object-oriented software system where the attributes were complexity, class, number of children and size of code on which the software maintainability is tested with quality factors. Several attempts have been made in order to quantify the software quality criteria. Khatri et al. (2016) developed a model that quantifies the software quality whereas Rana et al. (2017) predicted software vulnerability through self-learning model which in turn utilizes the knowledge for resolving the problem of software maintainability. Cai et.al (2012) proposed a new software maintainability prediction model that actually used to realize maintainability related design issues and built for fuzzy based analysis design model. Chidamber et.al (1994), applied some methods on object oriented languages and system and proposed six factors that are used to estimate the maintainability of OO systems.
3 MAINTAINABILITY FACTORS

For predicting the maintainability of any software, many researchers proposed various factors and techniques that help to predict maintainability. Therefore following is the snapshot of various software maintainability factors that really affect the software prediction. There are five basic factors that affect the maintainability of software. Analyzability explains the efforts required for finding the errors, fault or causes of failure in a software system. Any alteration or modification of software program or code is described by Modifiability. The risk associated with an unforeseen effect of modification is referring to Stability. The ability to verify the alteration or modification is done by Testability. To relate the user attempts for identification of theory and logic and its applicability, Understandability is used. Figure 1 is showing the maintainability and its sub factors.

Modifiability
All the software system in this world is constantly changing and
requires to be modified so many times after their development. It is proven by market research and survey that 60-80% of the cost of software development is spent after the initial development. The reason of high cost of software maintenance is that the nature of modifiability of software system. During the life cycle of a software system the cost of software product is going to be high always. Therefore any change that took place in software environment requirements, functionality and other specifications is called software modifiability of a software system.

**Stability**
Software stability is also known as software reliability, it can be understand as an ability to remain unchanged over a time under predefine reasonable conditions of use and storage.

**Testability**
The degree to which a software module, software requirements or design documents are supports software testing is called Software testability. We can measure testability directly such as software size; also it is not an essential property of software artefact. Even software testability is an extrinsic property that allows a developer to test different goals, methods, and resources. If the result of testability is high then it is easy to find faults in the system by means of testing.

**Understandability**
Understandability means how software is user-friendly and how easy to operate. In other words, we can say that the ability of any software product that enables the end user to comprehend whether the underlying software product is appropriate for use or not, and how the system should operate.

**Analyzability**
By means of analyzability, the capability of software product to be diagnosed for testing the data and to be analyzed for detecting the errors and bugs in the software code or Programme, cause of failure and find all the necessary changes need to be done in the software product.
4 PROCESS ADOPTED

To predict the maintainability of software various researchers came with his research and methods but to predict the maintainability of real time software is very difficult. MCDM is one of the approaches that are used to solve problem associated with real time prediction. The distinguishing feature of AHP is its ability to measure inconsistency. Approaches like Multi-Attribute Utility Theory (MAUT), conjoint analysis (CA), goal programming applies linear programming to gain the desires situation to converting goals constrained via adding slack and different variables representing deviation from the aim. MAUT gives numbers that designate how much attributes are valued via constructing Multiattribute software features, scaling elements for every characteristic, and estimating probabilities of best-case, intermediate-case and worst-case effects attributable to the selections. AHP is a less difficult shape of MAUT where the paired comparisons are used to derive the software functions represented by means of the concern or weight vector from contributing standards and alternatives inside the hierarchy. Conjoint Analysis (CA) is an advertising and marketing method used to measure, observe, and forecast. In todays era, the most applicable approach is AHP approach proposed by Saaty, 1988. In this research paper software maintainability is achieved by using a Fuzzy logic approach with AHP method named as F-AHP also using Multi-Criteria Analysis (MA), this is a combination of similarity approach and fuzzy based comparison. MCDM methods can be categorized into two parts one is called Discrete MCDM and other is known as Continuous MODM.

Fuzzy set theory

It is proved that the theory of fuzzy number or fuzzy set has advanced in many ways and in many fields and disciplines. Many of the traditional tools of modeling are crisp and deterministic. The area of interest of this found in many discipline like, artificial intelligence, research methodologies, management science, Digital image processing, pattern recognition and off course in robotics. The main objective of using fuzzy logic model is to build software maintainability measurement model with almost no or very little data base or data set.
5 RESEARCH METHODOLOGY

This section try to explain the interest into the MCDM methods and provide state of the art review of the literature in respect to MCDM procedure and techniques, there are total 350 articles published in 130 journals since 1990. While designing and developing any software product there are number of measures that are to be taken into account. The quality of design should be count into the measurement of quality of software (Ghosh et al., 2012). For the economic success of any software product there are many components that accepted broadly like software reliability, software quality and software maintainability too. The main question for understanding is how one can collaborate various components to achieve software maintainability. Software Maintainability is known as the most expensive phase of development process. This is found that most of the software development companies are spending 45-75% of cost only on software maintainability phase. In todays science and technology era we have various techniques and methods for predicting software maintainability but unfortunately none of them is providing exact prediction. Therefore this paper presents new approaches to find software maintainability measurement with the help of fuzzy AHP approach.

6 METRICS USED FOR MEASURE SOFTWARE MAINTAINABILITY USING FUZZY-AHP MODEL

Diverse characteristics and sub characteristics affecting the software quality have been quantified by means of usage of metrics to evaluate the software quality. In this work software Maintainability assessment is performed by using fuzzy AHP layered approach. For achieving the same two metrics have been used one is Chidamber and other one is Kemerer (CK). These two metrics are used because of assessment of designing of system instead of implementation. The CK metric set consists of six design complexity metrics namely, WMC, DIT, NOC, CBO, RFC and LCOM, a majority of these metrics may be used as maintainability predictors. This is so due to the fact LCOM is uncorrelated with the maintainability of
the software. The CK metrics except for LCOM are in brief defined as follows:

**Weighted Methods per Class**
This metrics is used to measure the complexity of class and also the sum of all weighted methods. If we get high WMC that means the complexity is greater hence low.

**Depth of Inheritance Tree**
This gives the length of longest path from a specified class to the root class and should be measure by the number of ancestor classes. If the value of DIT is high then it is showing that the complexity design is greater and the fault proneness hence low.

**Number of Children**
The number of immediate child classes is equal to total number of children. If the value of NOC is high that level of reuse is very high otherwise low.

**Coupling**
Total number of classes which are coupled with each other is counted by Coupling between objects. Two classes are coupled if one class is uses methods of other class.

**Response for a class**
All methods that used directly or indirectly for conveying message to an object or classes through some common method are called response for a class. Higher RFC implies more effort is required for testing phase.

The calculation of maintainability of software product by using AHP method with fuzzy logic method following steps needs to implement as follows:

i. By using AHP find all weighted matrices for maintainability.
ii. Find out the fuzzy index matrices using fuzzy-AHP methods
iii. To find out maintainability repeat septs I and II.
7 MODEL IMPLEMENTATION

From Model implementations perspective the following parameters are considered as input
- Coupling
- Inheritance
- Class
- Complexity and
- Number of Children each class have

Fuzzy logic or fuzzy interface system uses fuzzy logic and match input values with output results. This process is called fuzzification and the reverse is called de-fuzzification process. De-fuzzification process gives a singleton value. The most commonly used method for de-fuzzification is called centroid method.

Figure 3. Proposed Fuzzy-AHP models.
Values for Metrics
The values of CK metrics was initiated using analyst4j standalone tool. In this method the complexity is related with WMC whereas class is related with RFC and coupling is concerned with CBO and inheritance can be calculated with DIT which is same as NOC is related with a number of children.

MATLAB tool is used to find out the value of matric and the value of maintainability as shown in Figure 4.

8 RESULTS
The Role of AHP is to validate the proposed model given by Saaty,1988. For calculation of maintainability we need to consider CK metrics and their factors that affect software maintainability in the form of square metrics as given below.

Table 1
Input taken for AHP
The factors considered here are complexity (Cx), class (Cl), coupling (Cp), inheritance (Inhe) and Number of children (NOC). The relation of CK metrics and software maintainability exists and detailed in the previous section. The priority column is the relative ranking of the criteria produced with the aid of dividing every element of the matrix with the sum of its column. Next, the average across the rows is computed. The sum of priority criteria vector is one. The criteria with the largest weight is assumed to be the most important criterion, which is inheritance having the value 0.62. In order to get the values of metrics, we take weight times analyst4j. This weight is obtained through AHP so as to get the maintainability of the entire project. As per the approach, the authors achieved maintainability of project as 25.65. Here we prefer Analyst 4J tool to carry out operations along with Visual Studio Code matrix power tool for generating the final results.

9 CONCLUSION

This paper considered fuzzy AHP model for calculating maintainability usage of a software system. The inputs considered for this approach are complexity, coupling, inheritance, number of children and class. Data is taken from the survey and considering expert advice. Rules are considering from knowledge system and experts knowledge base. The proposed system evaluated maintainability of a software system and results are validated by the fuzzy AHP technique.

References

[1] A. Kumar, R. Kumar, and P.S. Grover, A Fuzzy Logic Approach to Measure Complexity of generic Aspect-Oriented Sys-


