International Journal of Innovative Computing, Information and Control
Volume **11**, Number **3**, June **2015**
ICIC International ©2015 ISSN 1349-4198
pp. 1045–1058

MULTI-CRITERIA SELECTION OF SOFTWARE COMPONENTS USING FUZZY-AHP APPROACH

Shrddha Sagar[1], Pratistha Mathur[1] and Arun Sharma[2]

[1]AIM & ACT Department
Banasthali University
Rajasthan 304022, India
sagarshraddha@gmail.com; mathurprati@yahoo.com

[2]Department of Information Technology
Indira Gandhi Delhi Technical University for Women
Delhi 110006, India
arunsharma2303@gmail.com

Received May 2014; revised September 2014

Abstract. *Object-oriented programming is not enough influential to handle the changing requirements. Component-based software development is a paradigm, which develops software systems that are integrated with the existing software component as plug-ins, which can be reused again and again. Hence, reusability of a component is more important than other quality factors of a software system. Selection of the most reusable software component is one of the critical activity for developing a quality system. Reusability has few types of sub-quality factors, which have considerable impact on reusability either directly or indirectly. In this paper Fuzzy-Analytic Hierarchy Process (Fuzzy-AHP) model is proposed for component selection using reusability as an important quality factor of a software component. Proposed model selects components with concurrent consideration of multiple criteria and ranks the components according to their reusability values. An empirical analysis has also been carried out on six industrial components, which are used for cleaning the system or improving system performance. A code is developed in Java for the proposed model, which successfully assigns the rank to each component based on overall performance index value. The result shows that the proposed approach is suitable for identifying the best software component in efficient and simple manner.*
Keywords: Component-based software development, Reusability, Fuzzy AHP

1. **Introduction.** To improve the business performance, one requires improving software development performance. That is why one enforces the application developers and research scholars to give a thought for the adoption of the advanced technologies and new development approaches. In 1990's, Object-Oriented Approach (OOA) was introduced as a new paradigm in software development area with some important features such as inheritance, data abstraction and polymorphism [1]. Though, there are many advantages of OOA but programmer has to write the code line by line and every time it is required to write the code from scratch and it cannot be used in other software product. Also, the cost and effort required for developing the software is very high. To overcome these problems faced in OOA, component based software approach is used to develop the high quality and low cost software product. Standard and existing components are used to develop application software and it is known as Component Based Software Development (CBSD). CBSD is now most widely accepted cost effective approach for software development and more emphasis is on the design and development of software system using reusable components [1]. Researchers have identified numbers of sub-quality factors such

as adaptability, price, documentation, customizability, which are influencing reusability [2]. Reusability of a software component is defined as the ability of software component to build it once and use it in different applications. Freeman [3] defined "Reuse is the use of any information which a developer may need in the software creation process". Krueger [4] defined "Reuse is the process of creating software systems from existing software rather than building them from scratch".

Kumar et al. [5] have done critical analysis of metrics for various quality aspects of the software component-based system. Authors have identified four main software quality factors: complexity, dependency, reusability, and maintainability through systematic review. They evaluated several research proposals on various key factors: metrics dentition, implementation technique, validation, usability, data source, comparative analysis, practicability, and extendibility. They pointed out that reusability is one of the most important quality factor for the selection of the software components.

In this paper we have proposed a Fuzzy Analytical Hierarchical Process (Fuzzy AHP) approach to evaluate the reusability of software components. The proposed methodology will help application developer to develop the software product.

Paper is organized into 6 sections. Section 2 describes the work related to software metrics for CBSD. Sections 3 discusses the proposed methodology for multi-criteria component selection which includes factors identification and fuzzy-AHP. Section 4 includes empirical study. Section 5 is the result discussion and Section 6 is the conclusion.

2. **Related Work.** Application developer has to identify the most reusable software components while developing software product. Several researchers have proposed various metrics for estimating reusability of object-oriented systems and some of them can be used in CBSD [6-8]. The concept of software reuse is to use existing software to develop a new software product. Portability, document quality, customizability, understandability, interface complexity, etc., are few reusable factors that improve software reusability. As reusability is the degree to which a component can be reused and reduces the software development cost by enabling less writing and more assembly [9]. Cho et al. [10] defined a set of metrics called Component Reusability (CR) and Component Reusability Level (CLR). These metrics are based on line of codes (LOC) and can only be used at design time for components. Boxall and Araban [11] proposed few metrics for better understanding of the components interfaces by considering the size of the interface, argument count, argument repetition scale and others. It does not consider the other aspect in the interface like return types and complexities of arguments.

Washizaki et al. [9] proposed Component Reusability Model (CRM) for black-box components from the user's perspective and the application point of view. The model has identified factors like portability, adaptability and understandability. Rotaru et al. [12] have taken adaptability, complexity and composeability of a component to estimate its reusability. Sharma et al. [13] defined interface complexity metrics for software components and considered interface methods and their associated properties, argument types and return types. Gill and Balkishan [14], introduced a set of component based metrics like Component Dependency Metrics (CDM) and Component Interaction Density Metrics (CIDM), which measures the dependency and coupling aspect of the software components. Sharma et al. [15] have considered four factors namely customizability, portability, understandability and interface complexity of component and used fuzzy logic based technique to evaluate reusability of software components. Hristov et al. [16] have pointed out that there is no clear framework that describes the reusability of software components and dependency between the metrics. They have proposed software reusability measurement models and metrics like availability, documentation, complexity, quality, maintainability, in paper but paper still lacked empirical validation.

Ioana and Doru [17] have proposed a fuzzy logic based solution for the specification based software component retrieval of software components. Cangussu et al. [18] have done empirical study on Analytic Hierarchy Process (AHP) and used AHP to rank the components. They have concluded that AHP is well suited for task of selecting components when several criteria are considered. Authors had limited their study to select the component on multiple non-functional criteria but have not included multiple functional criteria. Nerurkar et al. [19] have discussed reusability in relation to CBD and proposed a reusability metrics for black box components. Fuzzy logic based approach is used to estimate reusability and results found on real time applications which are quite satisfactory.

We have implemented Fuzzy AHP approach for estimating reusability by considering sub-factors of reusability.

3. Proposed Methodology for Multi-criteria Component Selection.

Component selection problem is a multi-criteria, complex and imprecise nature problem. All the factors have inter-dependency among them. They have some dependency structure. Hence, there is a need to identify interdependency between the multiple criteria and to propose a soft computing based solution to identify the more reusable software component. Due to fuzziness in decision making, the range driven fuzzy approach is used for selecting software component.

3.1. Factors identification.

Direct estimation of software quality is not always possible; therefore, one needs to establish the relationship between the factors in relation with reusability to achieve the goal. As we know that reusability is one of the most important factor of quality of software component, of which developer has to keep in mind while developing a software product. Various factors that are affecting reusability measurement are as follows: frequency of reuse, adaptability, price, maintainability, complexity, documentation quality, availability, portability, configurability, compatibility, customizability, interface complexity, commonality, etc. From literature review we have selected following five sub-factors for estimation of reusability which are as follows: adaptability, availability, interface complexity, customizability and understandability [15,19-23].

Rotaru et al. [12] have defined adaptability of a component as, a software component is adaptable only if it can encompass changes in its environment, irrespective of their types. Hence, component should be strong enough to handle modification of the environment without any external intervention and be able to provide adjustable behavior. According to Hristov et al. [16], availability can determine how easily and efficiently it can retrieve it back, but should not be mixed with operational availability, as developer of any application software will use the component through its interface only. Hence interface is one of the important sources for understanding, use and implementation and at last maintenance of the component. Therefore, interface complexity plays a vital role for evaluating overall complexity of the component [12]. As component is used in multiple applications, so it should be customized as per new requirement [8]. A good quality document should be there, so that application developer can easily understand the components [22,24].

In this paper, we have considered above mentioned reusability quality factors for estimating reusability of a software component which can further be used for component selection.

Assuming we want to estimate reusability of six different components, which are developed for the system performance improvement. Figure 1 shows hierarchical structure of

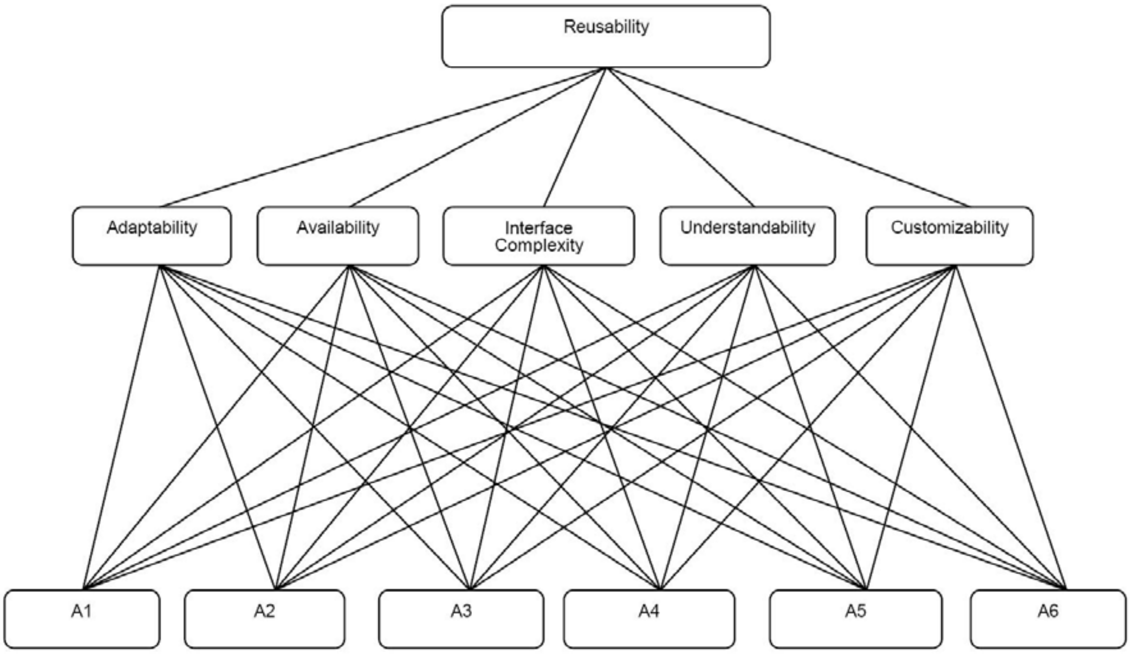

FIGURE 1. Hierarchical structure of component selection problem

reusability with five factors: adaptability, availability, interface complexity, understandability, and customizability and components named from A1 to A6.

3.2. **Fuzzy analytic hierarchy process (Fuzzy-AHP).** Analytic Hierarchy Process (AHP) as defined by Saaty is a very analytical tool that is used for modeling unstructured problems in various domains like health, management, social and economic [23,24]. Processing of subjective and individual preferences of a person or group in making decision, AHP provides objective mathematics to solve multi-criteria decision making. Hence, AHP can be used as multi-criteria decision method which represents a problem through hierarchical structure and on the basis of users judgments develop the priorities of the alternatives [25]. The problems faced by various researchers for solving complex problem using AHP are as follows [26,27].

- It is not always possible to obtain solution for linear equation.
- Only triangular fuzzy numbers are allowed to solve it.
- The number of pairwise comparison increases as the number of levels in hierarchy enhances.

The problem of subjectiveness and imprecision in the pairwise comparison method of AHP are resolved in fuzzy-analytical process (Fuzzy-AHP). Fuzzy-AHP uses a wide range of values rather than a single crisp value. With these range of values, decision makers can select any value that is reflecting confidence by which they can define their attitudes like optimistic, pessimistic or moderate [28]. The ratio provided by the decision maker is a fuzzy member defined by a membership function in the fuzzy set theory. Decision maker (DM) making estimation on alternatives with respect to every criterion is shown by triangular fuzzy number. To solve fuzzy reciprocal matrix, fuzzy extent analysis is used. Converting fuzzy performance matrix that represents overall performance of all alternatives corresponding to each criterion into an interval performance matrix is done. Thereafter, alpha cut analysis is used, so that unreliable and complex process of comparing utilities can be avoided. Hence, fuzzy-AHP has ability to solve multi-criteria problem [27].

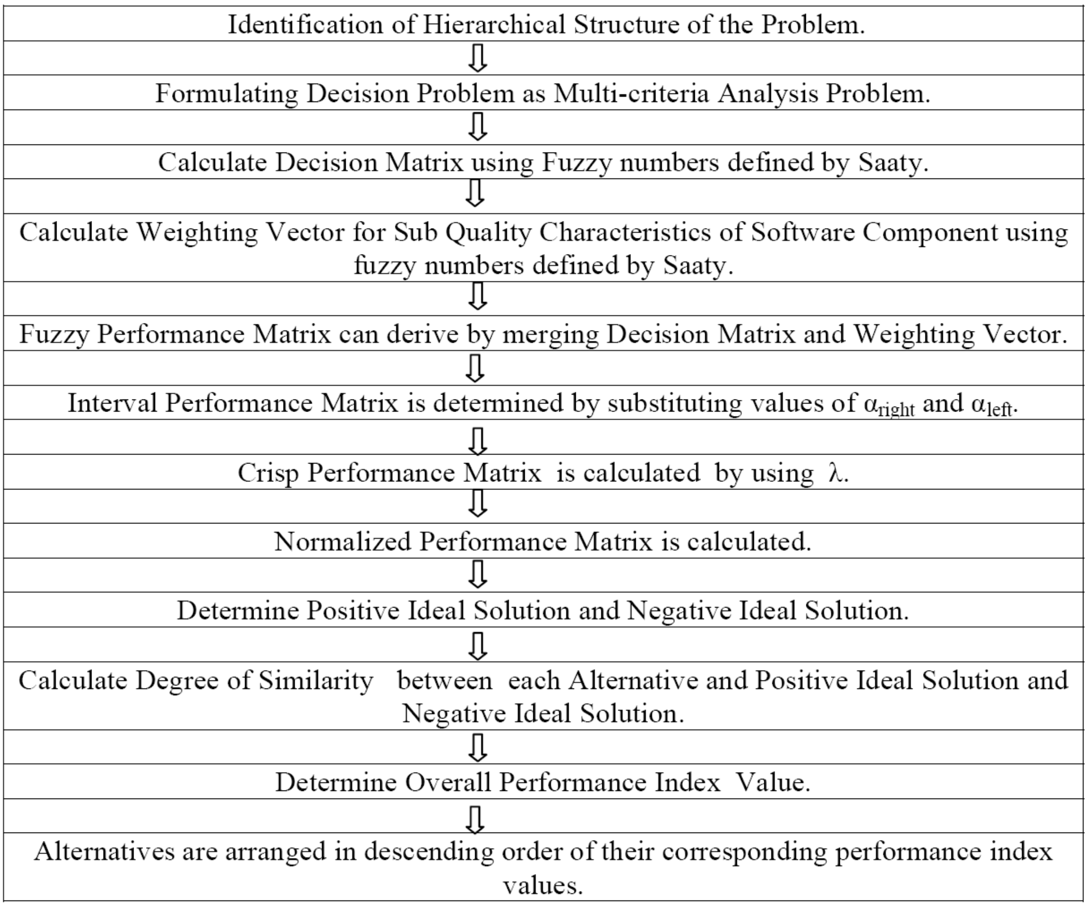

Figure 2. Process diagram of proposed methodology

Table 1. Scales for pair-wise comparison (Saaty, 1980 [29])

Intensity of value	*Interpretation*
1	*Requirements i and j have equal cost.*
3	*Requirement i has a slightly higher cost than j.*
5	*Requirement i has a strongly higher cost than j.*
7	*Requirement i has a very strongly higher cost than j.*
9	*Requirement i has an absolutely higher cost than j.*
2, 4, 6, 8	*These are intermediate scales between two adjacent judgments.*
Reciprocals	*If requirement i has a lower cost than j.*

Figure 2 shows the detailed steps of the proposed methodology. For comparison, Saaty table Table 1 is used. According to the preference level numerical value is selected. x_{ij} represents the performance rating of i^{th} component on the j^{th} criteria.

In this paper, adaptability (C1), availability (C2), interface complexity (C3), understandability (C4) and customizability (C5) are considered for evaluating the performance of software components namely A1, A2, A3, A4, A5 and A6. The description of these components is given in Table 2.

Deng [27] has proposed multi-criteria analysis with fuzzy pairwise comparison for tender selection. It has been implemented to our fuzzy problem to solve the problem of component selection.

- In this paper, we have used fuzzy triangular numbers to estimate the degree of belongings. It is defined as in Equation (1).

$$\mu_A(X) = \begin{cases} \frac{(x-a_1)}{(a_2-a_1)}, & a_1 \leq x \leq a_2, \\ \frac{(a_3-x)}{(a_3-a_2)}, & a_2 \leq x \leq a_3, \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where, a_1 is lower bound, a_2 most possible and a_3 upper bound.

- Fuzzy Reciprocal Judgment Matrix for criteria importance (W) or alternative performance with the help of fuzzy numbers defined in Equation (1), with respect to a specific criterion can be determined as

$$Cj \text{ or } W = \begin{bmatrix} \overline{a_{11}} & \overline{a_{12}} & \cdots & \overline{a_{1k}} \\ \overline{a_{21}} & \overline{a_{22}} & \cdots & \overline{a_{2k}} \\ \cdots & \cdots & \cdots & \cdots \\ \overline{a_{k1}} & \overline{a_{k2}} & \cdots & \overline{a_{kk}} \end{bmatrix} \tag{2}$$

where,

$$\begin{matrix} \bar{1}, \bar{3}, \bar{5}, \bar{9}, & l < s & \\ 1, & l = s & l, s = 1, 2, \ldots, k, \quad k = m \text{ or } n, \\ \frac{1}{\overline{a_{sl}}}, & l > s. & \end{matrix} \tag{3}$$

Then fuzzy extent analysis is carried out on (2) Criteria weight (w_j) or alternate performance rating (x_{ij}) using (1) and (4):

$$(x_{ij}) \text{ or } (w_j) = \frac{\sum_{s=1}^{k} \overline{a_{ls}}}{\sum_{i=1}^{k} \sum_{s=1}^{k} \overline{a_{ls}}} \tag{4}$$

- The resultant decision matrix (X) and weight vector (W) for component selection problem are given as (5) and (6) respectively:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \tag{5}$$

$$W = (w_1, w_2, \cdots, \cdots, w_m) \tag{6}$$

- A fuzzy performance matrix (Z) is computed by applying product operation on X and W as shown in (7):

$$Z = \begin{bmatrix} (w_1, x_{11}) & (w_2, x_{12}) & \cdots & (w_m, x_{1m}) \\ (w_1, x_{21}) & (w_2, x_{22}) & \cdots & (w_m, x_{2m}) \\ \cdots & \cdots & \cdots & \cdots \\ (w_1, x_{n1}) & (w_2, x_{n2}) & \cdots & (w_m, x_{nm}) \end{bmatrix} \tag{7}$$

- An interval performance matrix ($Z_\propto$) is obtained as given in (8) by using α-cut analysis (where $0 \leq \alpha \leq 1$) on the performance matrix (Z) as given in (7),

$$Z_\propto = \begin{bmatrix} [Z_{11l}^\propto, Z_{11r}^\propto] & [Z_{12l}^\propto, Z_{12r}^\propto] & \cdots & [Z_{1ml}^\propto, Z_{1mr}^\propto] \\ [Z_{21l}^\propto, Z_{21r}^\propto] & [Z_{22l}^\propto, Z_{22r}^\propto] & \cdots & [Z_{2ml}^\propto, Z_{2mr}^\propto] \\ \cdots & \cdots & \cdots & \cdots \\ [Z_{n1l}^\propto, Z_{n1r}^\propto] & [Z_{n2l}^\propto, Z_{n2r}^\propto] & \cdots & [Z_{nml}^\propto, Z_{nmr}^\propto] \end{bmatrix} \tag{8}$$

- An overall crisp performance matrix ($Z_{ij\propto}^{\lambda'}$) is computed using Equation (9) to know the attitude towards the risk of DM used optimum index λ,

$$Z_{ij\propto}^{\lambda'} = \lambda Z_{ijr}^\propto + (1-\lambda) Z_{ijl}^\propto \tag{9}$$

where, $\lambda \in [0, 1]$

$$Z^{\lambda'}_{ij\propto} = \begin{bmatrix} Z^{\lambda'}_{11\propto} & Z^{\lambda'}_{12\propto} & \cdots & Z^{\lambda'}_{1m\propto} \\ Z^{\lambda'}_{21\propto} & Z^{\lambda'}_{22\propto} & \cdots & Z^{\lambda'}_{2m\propto} \\ \cdots & \cdots & \cdots & \cdots \\ Z^{\lambda'}_{n1\propto} & Z^{\lambda'}_{n2\propto} & \cdots & Z^{\lambda'}_{nm\propto} \end{bmatrix} \tag{10}$$

- Normalized performance matrix $(Z^{\lambda}_{\propto})$ normalization process in regard to each criterion is applied on (10) by using (11)

$$Z^{\lambda}_{ij\propto} = \frac{Z^{\lambda'}_{ij\propto}}{\sqrt{\sum_{i=1}^{n}(Z^{\lambda'}_{ij\propto})^2}} \tag{11}$$

$$Z^{\lambda}_{\propto} = \begin{bmatrix} Z^{\lambda}_{11\propto} & Z^{\lambda}_{12\propto} & \cdots & Z^{\lambda}_{1m\propto} \\ Z^{\lambda}_{21\propto} & Z^{\lambda}_{22\propto} & \cdots & Z^{\lambda}_{2m\propto} \\ \cdots & \cdots & \cdots & \cdots \\ Z^{\lambda}_{n1\propto} & Z^{\lambda}_{n2\propto} & \cdots & Z^{\lambda}_{nm\propto} \end{bmatrix} \tag{12}$$

- To avoid the worst decision outcome this concept is further extended as per algorithm proposed by Hwang and Yoon [30]. Selecting the maximum value and the minimum value across all the alternatives with respect to each criterion (13), the positive ideal solution $A^{\lambda+}_{\propto}$ and the negative ideal solution $A^{\lambda-}_{\propto}$ are:

$$\begin{cases} A^{\lambda+}_{\propto} = (Z^{\lambda+}_{1\propto}, Z^{\lambda+}_{2\propto}, \cdots, \cdots, Z^{\lambda+}_{m\propto}) \\ A^{\lambda-}_{\propto} = (Z^{\lambda-}_{1\propto}, Z^{\lambda-}_{2\propto}, \cdots, \cdots, Z^{\lambda-}_{m\propto}) \end{cases} \tag{13}$$

where,

$$\begin{cases} Z^{\lambda+}_{\propto} = \max(Z^{\lambda+}_{1\propto}, Z^{\lambda+}_{2\propto}, \cdots, \cdots, Z^{\lambda+}_{m\propto}) \\ Z^{\lambda-}_{\propto} = \min(Z^{\lambda-}_{1\propto}, Z^{\lambda-}_{2\propto}, \cdots, \cdots, Z^{\lambda-}_{m\propto}) \end{cases} \tag{14}$$

- By applying the vector matching function, the degree of similarity between each alternative, and the positive ideal solution and the negative ideal solution is calculated as:

$$S^{\lambda+}_{i\propto} = \frac{A^{\lambda}_{i\propto} A^{\lambda+}_{\propto}}{\max(A^{\lambda}_{i\propto} A^{\lambda}_{i\propto}, A^{\lambda+}_{\propto} A^{\lambda+}_{\propto})}, \tag{15}$$

$$S^{\lambda-}_{i\propto} = \frac{A^{\lambda}_{i\propto} A^{\lambda-}_{\propto}}{\max(A^{\lambda}_{i\propto} A^{\lambda}_{i\propto}, A^{\lambda-}_{\propto} A^{\lambda-}_{\propto})}, \tag{16}$$

where $A^{\lambda}_{i\propto} = (Z^{\lambda}_{1i\propto}, Z^{\lambda}_{2i\propto}, \cdots, \cdots, Z^{\lambda}_{mi\propto})$, is the i^{th} row of the overall performance matrix as given in (12).

Overall performance index for each alternative can be determined by using Equation (17):

$$S^{\lambda}_{\propto i} = \frac{S^{\lambda+}_{i\propto}}{S^{\lambda+}_{i\propto} + S^{\lambda-}_{i\propto}}, \quad i = 1, 2, \ldots, n. \tag{17}$$

According to Deng [27] larger the index value is the more preferred the alternative.

4. **Empirical Study.** Selection of the most reusable component from the existing components is the most crucial decision-making process. For assessing the overall performance of the system, performance of individual component and quality assessment are considered as criteria for selecting the more reusable component. Quality assessment is imprecise and vague, so we have considered fuzzy approach to handle such type of decision making problems.

Survey

In this work a survey was carried out for identifying the weight values of fitness parameters and performance rating of software components. 20 experts from industry and academia have participated in this survey. These experts have sound knowledge of operating system.

Questionnaire

Experts have given their opinion for sub-characteristics – adaptability, availability, interface complexity, understandability and customizability for evaluating reusability of these components.

Components Used

In this empirical study we have used six software components used for system cleaning. Details of them are given in Table 2.

WinASO Registry Optimizer is an advanced registry cleaner and optimizer for Windows that allows you to safely clean and repair registry problems with a few simple mouse clicks. This component is designed to fix common problems like denied access to missing drives and disks and illegally modified Internet Explorer pages. Advanced SystemCare Free is a comprehensive PC care utility that takes one-click approach to help protect, repair and optimize your computer. Cleans the unnecessary records from your registry that might cause trouble and slow downs. Wise Registry Cleaner is one of the safest Registry cleaning tool that scans the engine thoroughly, safely and fast. It can scan the Windows registry and the system can work better more quickly. Avast organizes its environment and prioritizes it in a sensible manner. It is used for cleaning and monitoring security. AVG AntiVirus Free 2014 works as anti-virus as well as internet security for both front-end and back-end. It is useful for personal data management and privacy utilities that will prevent uninvited access to the files. CCleaner is a free registry and junk-clearing utility for keeping the system clean and optimized. It removes unused files from the system and allows windows to run faster and frees up valuable hard disk space.

Table 2. Description of software component used

S.No.	Name of Component	Description
1	WinASO Registry Optimizer (A4)	Advanced registry cleaner and Optimizer for Windows
2	Advanced SystemCare (A2)	Comprehensive PC care utility that helps to protect, repair and optimize the computer.
3	Wise Registry Cleaner (A3)	The safest Registry cleaning tool that scans the engine thoroughly, safely and fast.
4	AVG AntiVirus Free 2014 (A6)	Works as anti-virus as well as internet security for both front-end and back-end.
5	Avast Free Antivirus 2014 (A5)	It is used for cleaning and monitoring security.
6	CCleaner (A1)	It is a free registry and junk-clearing utility for keeping the system clean and optimized.

Environment of Experiment

Reusability was evaluated by using fuzzy-AHP technique which was proposed by Deng [27]. The algorithm was implemented in Java programming language using JavaTM Standard Edition, Version 7 Updated 55 (build 1.7.0_55-b14). It was tested on processor (Intel® Core(TM) i3-2310M @ 2.10 GHz 2.10 GHa, RAM (4.00 GB) and operating system (Windows 7 Professional Service pack 1).

Fuzzy Reciprocal Judgment Matrix alternative performance (C1, C2, C3, C4, C5) with the help of fuzzy numbers are shown below:

$$C1 = \begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \\ A6 \end{array} \begin{array}{c} \begin{array}{cccccc} A1 & A2 & A3 & A4 & A5 & A6 \end{array} \\ \begin{bmatrix} 1 & 3 & 5 & 1/7 & 7 & 1/9 \\ 1/3 & 1 & 7 & 1/5 & 9 & 5 \\ 1/5 & 1/7 & 1 & 3 & 1/7 & 5 \\ 7 & 5 & 1/3 & 1 & 1/3 & 9 \\ 1/7 & 1/9 & 7 & 3 & 1 & 1/3 \\ 9 & 1/5 & 1/5 & 1/9 & 3 & 1 \end{bmatrix} \end{array}$$

$$C2 = \begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \\ A6 \end{array} \begin{array}{c} \begin{array}{cccccc} A1 & A2 & A3 & A4 & A5 & A6 \end{array} \\ \begin{bmatrix} 1 & 5 & 7 & 9 & 1/3 & 1/7 \\ 1/5 & 1 & 3 & 1/7 & 1/9 & 9 \\ 1/7 & 1/3 & 1 & 1/5 & 3 & 5 \\ 1/9 & 7 & 5 & 1 & 7 & 1/9 \\ 3 & 9 & 1/3 & 1/7 & 1 & 5 \\ 7 & 1/9 & 1/5 & 9 & 1/5 & 1 \end{bmatrix} \end{array}$$

$$C3 = \begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \\ A6 \end{array} \begin{array}{c} \begin{array}{cccccc} A1 & A2 & A3 & A4 & A5 & A6 \end{array} \\ \begin{bmatrix} 1 & 1/3 & 1/5 & 7 & 9 & 1/5 \\ 3 & 1 & 1/9 & 5 & 1/7 & 3 \\ 5 & 9 & 1 & 1/7 & 3 & 1/9 \\ 1/7 & 1/5 & 7 & 1 & 9 & 1/3 \\ 1/9 & 7 & 1/3 & 1/9 & 1 & 9 \\ 5 & 1/3 & 9 & 3 & 1/9 & 1 \end{bmatrix} \end{array}$$

$$C4 = \begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \\ A6 \end{array} \begin{array}{c} \begin{array}{cccccc} A1 & A2 & A3 & A4 & A5 & A6 \end{array} \\ \begin{bmatrix} 1 & 7 & 9 & 5 & 1/5 & 1/9 \\ 1/7 & 1 & 3 & 1/5 & 7 & 5 \\ 1/9 & 1/3 & 1 & 9 & 1/9 & 1/3 \\ 1/5 & 5 & 1/9 & 1 & 7 & 3 \\ 5 & 1/7 & 9 & 1/7 & 1 & 1/9 \\ 9 & 1/5 & 3 & 1/3 & 9 & 1 \end{bmatrix} \end{array}$$

$$C5 = \begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \\ A6 \end{array} \begin{array}{c} \begin{array}{cccccc} A1 & A2 & A3 & A4 & A5 & A6 \end{array} \\ \begin{bmatrix} 1 & 1/5 & 1/3 & 7 & 5 & 1/9 \\ 5 & 1 & 9 & 1/5 & 1/3 & 7 \\ 3 & 1/9 & 1 & 3 & 9 & 1/3 \\ 1/7 & 5 & 1/3 & 1 & 1/7 & 5 \\ 1/5 & 3 & 1/9 & 7 & 1 & 5 \\ 9 & 1/7 & 3 & 1/5 & 1/5 & 1 \end{bmatrix} \end{array}$$

Fuzzy Reciprocal Judgment Matrix for criteria importance (W) with the help of fuzzy numbers is as follows:

$$W = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ C5 \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & C5 \end{array} \\ \begin{bmatrix} 1 & 3 & 7 & 1/7 & 9 \\ 1/3 & 1 & 5 & 1/3 & 7 \\ 1/7 & 1/5 & 1 & 5 & 1/7 \\ 7 & 3 & 1/5 & 1 & 1/3 \\ 1/9 & 1/7 & 7 & 3 & 1 \end{bmatrix} \end{array}$$

The resultant decision matrix (X) and weight vector (W) is:

$$X = \begin{bmatrix} (.0827, .1692, .3429) & (.1287, .2207, .3828) & (.1087, .1739, .2901) & (.1283, .2149, .3624) & (.0777, .1450, .2924) \\ (.1323, .2346, .4195) & (.0739, .1321, .2206) & (.0497, .1202, .2574) & (.0809, .1575, .3084) & (.1343, .2395, .4326) \\ (.0434, .0988, .2108) & (.04304, .0950, .2045) & (.0975, .1791, .3135) & (.0681, .1049, .1682) & (.0847, .1748, .3346) \\ (.1327, .2360, .4298) & (.1122, .1986, .3699) & (.1074, .1734, .2882) & (.0809, .1571, .3076) & (.0609, .1235, .2595) \\ (.0601, .1206, .2509) & (.0972, .1815, .3265) & (.1071, .1723, .2847) & (.0895, .1484, .2401) & (.0823, .1733, .3557) \\ (.0760, .1407, .2427) & (.1057, .1719, .2788) & (.0982, .1809, .3248) & (.1290, .2171, .3467) & (.0772, .1439, .2511) \end{bmatrix}$$

$$W = \begin{bmatrix} (.1927, .3193, .5445) \\ (.1124, .2166, .4275) \\ (.0522, .10281, .1965) \\ (.0878, .1828, .3675) \\ (.0864, .1784, .3452) \end{bmatrix}$$

A fuzzy performance matrix (Z) is:

$$Z = \begin{bmatrix} (.0159, .0540, .1867) & (.0145, .0478, .1636) & (.0057, .0179, .0569) & (.0113, .0393, .1331) & (.0067, .0258, .1009) \\ (.0255, .7492, .2285) & (.0083, .0286, .0943) & (.0026, .0124, .0506) & (.0071, .0288, .1134) & (.0116, .0427, .1493) \\ (.0084, .0315, .1148) & (.0048, .0206, .0874) & (.0051, .0184, .0616) & (.0059, .0192, .0618) & (.0073, .0312, .1155) \\ (.0255, .0754, .2340) & (.0126, .0430, .1581) & (.0056, .0178, .0566) & (.0071, .0287, .1131) & (.0053, .0220, .0896) \\ (.0116, .0385, .1366) & (.0109, .0393, .1395) & (.0056, .0771, .0559) & (.0078, .0271, .0882) & (.0072, .0309, .1228) \\ (.0146, .0449, .1321) & (.0119, .0373, .1192) & (.0051, .0186, .0638) & (.0113, .0396, .1274) & (.0067, .0256, .0867) \end{bmatrix}$$

An interval performance matrix ($Z_\propto$) is:

$$Z_\propto = \begin{bmatrix} (.0349, .1204) & (.0311, .1057) & (.0118, .0374) & (.0253, .0862) & (.0163, .0634) \\ (.0502, .1516) & (.0185, .0615) & (.0075, .0315) & (.0179, .0711) & (.0272, .0960) \\ (.0199, .0732) & (.0127, .0540) & (.0117, .0400) & (.0126, .0405) & (.0192, .0734) \\ (.0505, .1547) & (.0278, .1006) & (.0117, .0372) & (.0179, .0709) & (.0136, .0558) \\ (.0251, .0876) & (.0251, .0894) & (.0117, .0368) & (.0175, .0577) & (.0191, .0769) \\ (.0298, .0885) & (.0246, .0782) & (.01109, .0412) & (.0255, .0836) & (.0162, .0562) \end{bmatrix}$$

An overall crisp performance matrix ($Z_\propto^{\lambda'}$) is:

$$Z_\alpha^{\lambda'} = \begin{bmatrix} .0777 & .0684 & .0246 & .0558 & .0398 \\ .1009 & .0399 & .0195 & .0445 & .0616 \\ .0466 & .0334 & .0259 & .0265 & .0463 \\ .1026 & .0642 & .0245 & .0444 & .0347 \\ .0563 & .0573 & .0242 & .0376 & .0479 \\ .0596 & .0514 & .2654 & .0545 & .0362 \end{bmatrix}$$

Normalized performance matrix ($Z_\propto^{\lambda}$) is:

$$Z_\propto^{\lambda} = \begin{bmatrix} .4118 & .5183 & .4133 & .5058 & .3587 \\ .5351 & .3026 & .3271 & .4037 & .5544 \\ .2468 & .2527 & .4347 & .2407 & .4167 \\ .5438 & .4862 & .4109 & .4028 & .3126 \\ .2985 & .4338 & .4070 & .3409 & .4318 \\ .3136 & .3892 & .4457 & .4947 & .3256 \end{bmatrix}$$

TABLE 3. Overall performance index value of system clean components

Component → / λ and α ↓		$A1$	$A2$	$A3$	$A4$	$A5$	$A6$
$\lambda.0$	$\alpha(.1)$	0.6144	0.6036	0.6033	0.5749	0.5286	0.4205
	$\alpha(.3)$	0.6062	0.5968	0.5942	0.5666	0.5238	0.4241
	$\alpha(.5)$	0.6015	0.5930	0.5890	0.5617	0.5211	0.4260
	$\alpha(.6)$	0.5998	0.5916	0.5870	0.5598	0.5201	0.4267
	$\alpha(.9)$	0.5998	0.5916	0.5870	0.5598	0.5201	0.4267
$\lambda.5$	$\alpha(.1)$	0.5805	0.5766	0.5718	0.5141	0.5046	0.4214
	$\alpha(.3)$	0.5822	0.5770	0.5733	0.5196	0.5059	0.4218
	$\alpha(.5)$	0.5848	0.5780	0.5760	0.5267	0.5082	0.4231
	$\alpha(.6)$	0.5864	0.5786	0.5777	0.5310	0.5096	0.4238
	$\alpha(.9)$	0.5927	0.5848	0.5811	0.5478	0.5152	0.4271
$\lambda1$	$\alpha(.1)$	0.5778	0.5746	0.5703	0.5081	0.5034	0.4232
	$\alpha(.3)$	0.5787	0.5747	0.5707	0.5115	0.5038	0.4227
	$\alpha(.5)$	0.5802	0.5751	0.5718	0.5166	0.5047	0.4225
	$\alpha(.6)$	0.5817	0.5757	0.5732	0.5205	0.5059	0.4230
	$\alpha(.9)$	0.5900	0.5819	0.5796	0.5416	0.5130	0.4263

The *degree of similarity* between each alternative, and *the positive ideal solution* and *the negative ideal solution* are:

$$\begin{array}{ll} S_{1\alpha}^{\lambda+} = .8532 & S_{1\alpha}^{\lambda-} = .6055 \\ S_{2\alpha}^{\lambda+} = .8334 & S_{2\alpha}^{\lambda-} = .6134 \\ S_{3\alpha}^{\lambda+} = .6120 & S_{3\alpha}^{\lambda-} = .8346 \\ S_{4\alpha}^{\lambda+} = .8353 & S_{4\alpha}^{\lambda-} = .6099 \\ S_{5\alpha}^{\lambda+} = .7393 & S_{5\alpha}^{\lambda-} = .7154 \\ S_{6\alpha}^{\lambda+} = .7553 & S_{6\alpha}^{\lambda-} = .6786 \end{array}$$

The overall performance index value at $\alpha = 0.5$ and $\lambda = 0.5$ clearly shows that component $C1$ is ranked 1$^{\text{st}}$, component $C2$ is ranked 2$^{\text{nd}}$, component $C3$ is ranked 3$^{\text{rd}}$ and component $C6$ has lowest rank. And component (C1 AVG AntiVirus Free 2014) is the most reusable component and can be selected for improving the system performance. Simultaneously, we have also considered other values of α =.1, .3, .6 .9 and λ =.0 for a pessimistic decision maker and $\lambda = 1$ for an optimistic decision maker. Then we calculate overall performance index of each six system clean component and determine its corresponding ranking respectively.

5. **Result Discussion.** In this paper, we impemented our Fuzzy-AHP approach on six software components. The performance index value of each component was calculated by using Fuzzy-AHP. Table 3 clearly shows that component $C1$ is ranked 1$^{\text{st}}$, component $C2$ is ranked 2$^{\text{nd}}$, component $C3$ is ranked 3$^{\text{rd}}$ and component $C6$ has lowest rank.

Table 3 shows the overall performance index value of system cleaning components using moderate, pessimistic and optimistic DM. From Table 3, it can be concluded that component 1 is more reusable then component 2 and so on.

Figure 3 to Figure 5 show the overall performance index values of all six components at various value of alpha for pessimistic, moderate and optimistic decision makers. The figures show clearly that component (AVG AntiVirus Free 2014) is the most reusable component and can be selected for improving the system performance. It is also clear from above figures that component ($C1$) is most suitable component under any degree of

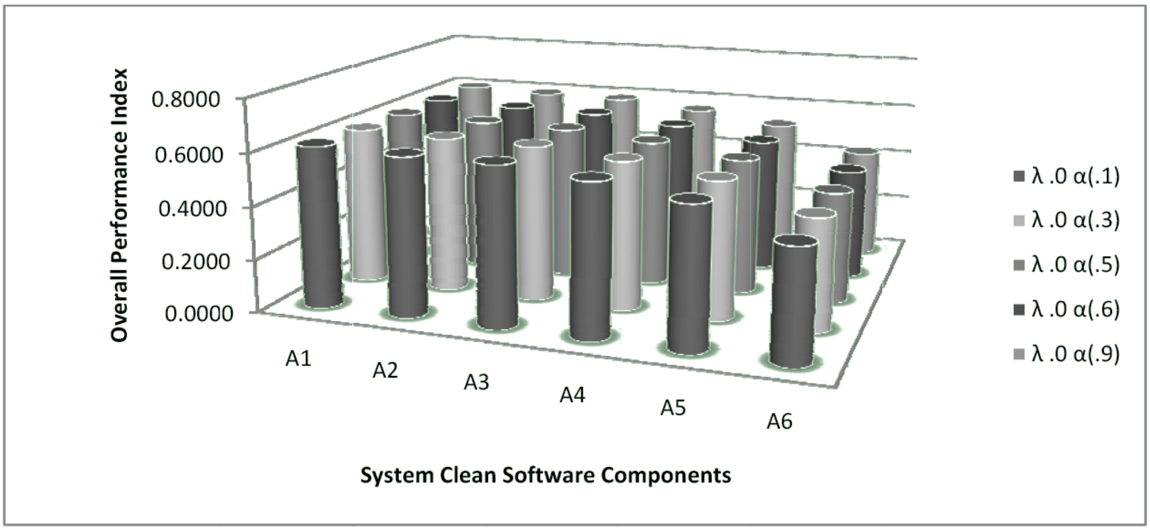

Figure 3. Performance index and ranking of system clean component for pessimistic DM

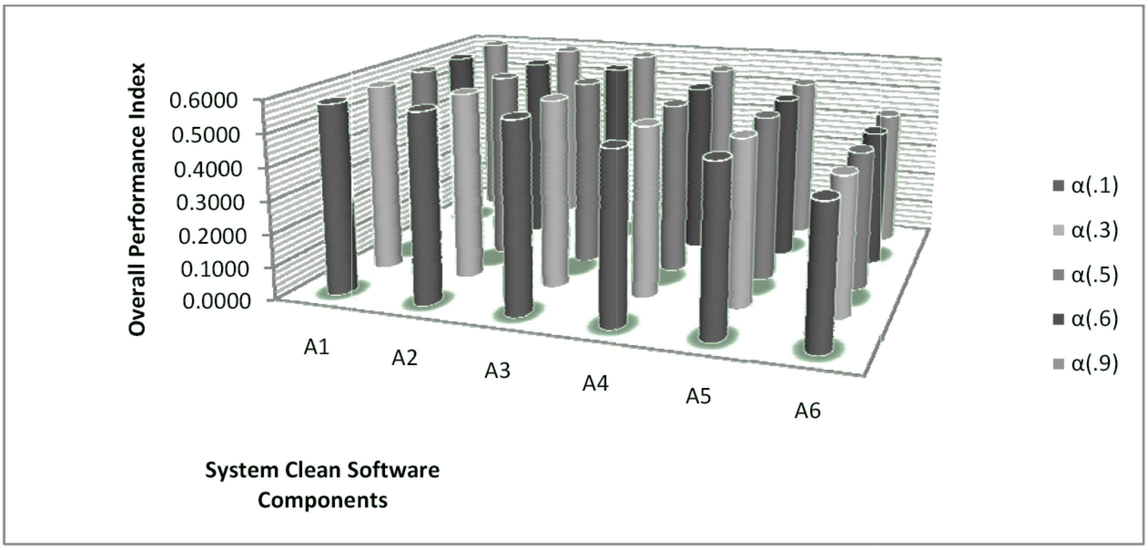

Figure 4. Performance index and ranking of system clean component for moderate DM

confidence of the DM with various attitudes towards risk. By using fuzzy-AHP algorithm the uncertainty and imprecision associated with DM's subjective judgment in human thinking is reflected very clearly. This tool provides DM a better understanding of the problem and his decision behavior.

6. **Conclusion and Future Work.** For developing an overall quality product, the selection of the best quality component is of prime concern. For selecting the best suitable reusable component, the important factors are identified that will help the software developer in the selection of the component. For the selection of appropriate and better quality component, sufficient amount of efforts must be taken. A reusable component helps in better understanding and low maintenance effort for application. In our present work, we have used FAHP approach to evaluate the reusability of component. This approach is validated on six system clean component. The result of the proposed methodology shows that it may be used to predict the relative reusability of the components with an

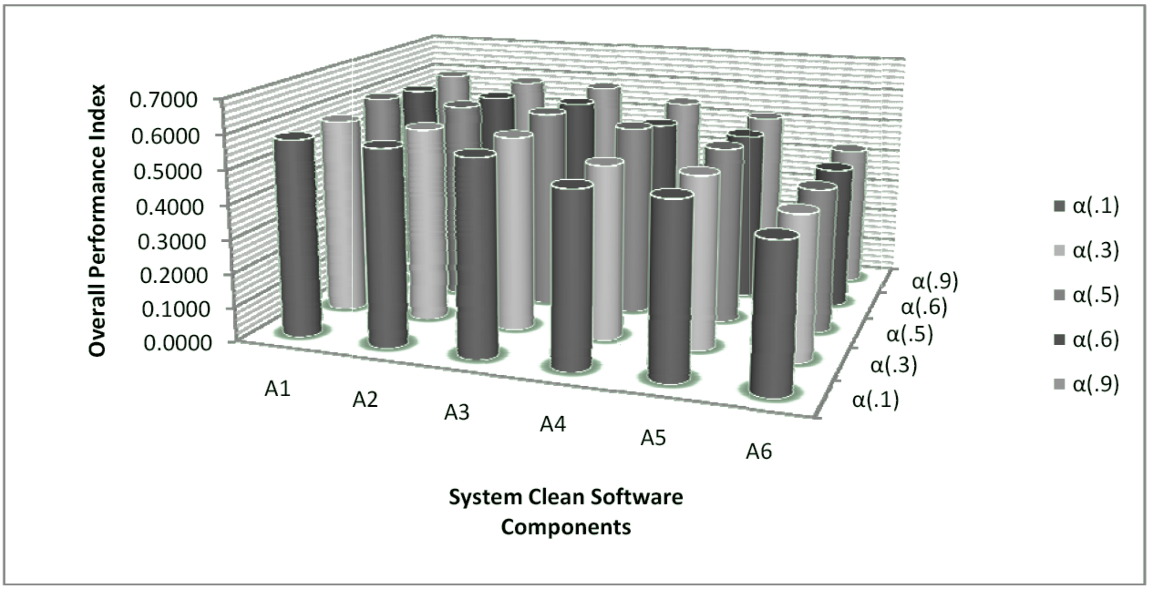

FIGURE 5. Performance index and ranking of system clean component for optimistic DM

acceptable accuracy. This approach can also be used in several domains like for ranking the educational organization, ranking of hotels, ranking of tourist places.

REFERENCES

[1] C. Szyperski, *Component Software – Beyond Object-Oriented Programming*, 2nd Edition, Addison-Wesley, 1998.

[2] A. Jatain and D. Gaur, Estimation of component reusability by identifying quality attributes of component: A fuzzy approach, *Proc. of the 2nd International Conference on Computational Science, Engineering and Information Technology*, pp.738-742, 2012.

[3] P. Freeman, Reusable software engineering concepts and research directions, *Proc. of the Workshop on Reusability in Programming*, pp.10-23, 1983.

[4] C. W. Krueger, Software reuse, *ACM Computing Surveys*, vol.24, no.2, pp.131-183, 1992.

[5] V. Kumar, A. Sharma, R. Kumar and P. S. Grover, Quality aspects for component-based systems: A metrics based approach, *Software: Practice and Experience*, vol.42, no.12, pp.1531-1548, 2012.

[6] R. D. Banker, R. J. Kauffman, C. Wright and D. Zweig, Automating output size and reusability metrics in an object-based computer aided software engineering (case) environment, *NYU Working Paper No. IS-91-25*, 1991.

[7] T. Kamiya, S. Kusumoto, K. Inoue and Y. Mohri, Empirical evaluation of reuse sensitiveness of complexity metrics, *Information and Software Technology*, vol.41, no.5, pp.297-305, 1999.

[8] K. K. Aggarwal, Y. Singh, A. Kaur and R. Malhotra, Software reuse metrics for object-oriented systems, *Proc. of the 3rd ACIS Int'l Conference on Software Engineering Research, Management and Applications*, pp.48-55, 2005.

[9] H. Washizaki, H. Yamamoto and Y. Fukazawa, A metrics suite for measuring reusability of software components, *Proc. of the 9th International Symposium on Software Metrics September*, 2003.

[10] E. S. Cho, M. S. Kim and S. D. Kim, Component metrics to measure component quality, *The 8th Asia-Pacific Software Engineering Conference*, Macau, pp.419-426, 2001.

[11] M. A. S. Boxall and S. Araban, Interface metrics for reusability analysis of components, *Proc. of the Australian Software Engineering Conference*, Melbourne, Australia, pp.40-46, 2004.

[12] O. P. Rotaru, M. Dobre and M. Petrescu, Reusability metrics for software components, *Proc. of the 3rd ACS/IEEE International Conference on Computer Systems and Applications*, Cairo, Egypt, pp.24-29, 2005.

[13] A. Sharma, R. Kumar and P. S. Grover, Empirical evaluation and validation of interface complexity metrics for student components, *International Journal of Software Engineering and Knowledge Engineering*, vol.18, no.7, pp.919-931, 2008.

[14] N. S. Gill and Balkishan, Dependency and interaction oriented complexity metrics of component-based systems, *ACM SIGSOFT Software Engineering Notes*, vol.33, no.2, pp.1-5, 2008.
[15] A. Sharma, P. S. Grover and R. Kumar, Reusability assessment for software components, *ACM SIGSOFT Software Engineering Notes*, vol.34, no.2, pp.1-6, 2009.
[16] D. Hristov et al., Structuring software reusability metrics for component-based software development, *The 7th International Conference on Software Engineering Advances*, Lisbon, Portugal, pp.421-429, 2012.
[17] S. Ioana and T. Doru, Specification-based retrieval of software components through fuzzy inference, *Acta Polytechnica Hungarica*, vol.3, no.3, pp.123-135, 2006.
[18] J. W. Cangussu et al., Multi criteria selection of components using the analytic hierarchy process, *Component-Based Software Engineering*, pp.67-81, 2006.
[19] N. W. Nerurkar, A. Sharma and S. Sagar, A soft computing based approach to estimate reusability of software components, *ACM SIGSOFT Software Engineering Notes*, vol.35, no.4, 2010.
[20] V. Bhardwaj, *Estimating Reusability of Software Components Using Fuzzy Logic*, Ph.D. Thesis, School of Mathematics and Computer Applications, Thapar University, 2010.
[21] A. Sharma, P. S. Grover and R. Kumar, Investigation of reusability, complexity and customizability metrics for component based systems, *ICFAI Journal of Information Technology*, vol.1, no.2, pp.6-11, 2006.
[22] T. H. Hsu and F. F. Pan, Application of Monte Carlo AHP in ranking dental quality attributes, *Expert Systems with Applications*, vol.36, no.2, pp.2310-2316, 2009.
[23] S. K. Lee, G. Mogi and J. W. Kim, The competitiveness of Korea as a developer of hydrogen energy technology: The AHP approach, *Energy Policy*, vol.36, no.4, pp.1284-1291, 2008.
[24] H. Kunz and S. Thorsten, General and specific formalization approach for a balanced scorecard: An expert system with application in health care, *Expert Systems with Applications*, vol.38, no.3, pp.1947-1955, 2011.
[25] S. Sittikruear and A. Bangviwat, Energy efficiency improvement in community – Scale whisky factories of thailand by various multi-criteria decision making methods, *Energy Procedia*, 2013.
[26] S. Oğuztimur, *Why Fuzzy Analytic Hierarchy Process Approach for Transport Problems?* http://www-sre.wu.ac.at/ersa/ersaconfs/ersa11/e110830aFinal00438.pdf, 2012.
[27] H. Deng, Multicriteria analysis with fuzzy pairwise comparison, *International Journal of Approximate Reasoning*, vol.21, pp.215-231, 1999.
[28] L. H. Jie, M. C. Meng and C. W. Cheong, Web based fuzzy multicriteria decision making tool, *International Journal of the Computer, the Internet and Management*, vol.14, 2006.
[29] T. L. Saaty, Decision making with the analytic hierarchy process, *International Journal of Services Sciences*, vol.1, no.1, pp.83-98, 2008.
[30] C. L. Hwang and K. S. Yoon, *Multiple Attribute Decision Making: Methods and Applications*, Springer, Berlin, 1981.