Contents lists available at GrowingScience

Decision Science Letters

homepage: www.GrowingScience.com/dsl

Application of fuzzy-MOORA method: Ranking of components for reliability estimation of component-based software systems

Zeeshan Ali Siddiqui*[*] and Kirti Tyagi

Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh, India

CHRONICLE

Article history:
Received April 16, 2015
Received in revised format:
June 12, 2015
Accepted June 21, 2015
Available online
June 25 2015

Keywords:
Fuzzy-MOORA
Reliability
CBSS
Ranking
Optimization
Multi criteria decision making

ABSTRACT

Component-based software system (CBSS) development technique is an emerging discipline that promises to take software development into a new era. As hardware systems are presently being constructed from kits of parts, software systems may also be assembled from components. It is more reliable to reuse software than to create. It is the glue code and individual components reliability that contribute to the reliability of the overall system. Every component contributes to overall system reliability according to the number of times it is being used, some components are of critical usage, known as usage frequency of component. The usage frequency decides the weight of each component. According to their weights, each component contributes to the overall reliability of the system. Therefore, ranking of components may be obtained by analyzing their reliability impacts on overall application. In this paper, we propose the application of fuzzy multi-objective optimization on the basis of ratio analysis, Fuzzy-MOORA. The method helps us find the best suitable alternative, software component, from a set of available feasible alternatives named software components. It is an accurate and easy to understand tool for solving multi-criteria decision making problems that have imprecise and vague evaluation data. By the use of ratio analysis, the proposed method determines the most suitable alternative among all possible alternatives, and dimensionless measurement will realize the job of ranking of components for estimating CBSS reliability in a non-subjective way. Finally, three case studies are shown to illustrate the use of the proposed technique.

1. Introduction

Component-based software system (CBSS) is an approach to improve the quality, maintainability and reliability of the software system. Component-Based Software Engineering (CBSE) is a beneficial development model for building software applications as it emphasises on reusability of software components. A software component is an executable entity that may be deployed independently and confirms to a component model. Components have interfaces through which they communicate with other components. CBSE assembled the Commercial Off-The-Shelf (COTS) components and existing

[*] Corresponding author.
E-mail address: zeealis@gmail.com (Z. A. Siddiqui)

doi: 10.5267/j.dsl.2015.6.005

components together in such a manner that they interact as intended. CBSS are being developed using commercial pre-tested, high-quality, trusted and robust software components even though ensuring a reliable software application as a whole is a very difficult task.

However, a software system has two critically acclaimed user requirements, one is reliability and the other is availability. ‘‘Software reliability is defined as the probability of the failure-free operation of a software system for a specified period of time in a specified environment’’ (IEEE, 1990). Reliability of software system makes sense if non-performance of the system has scathing impact, and availability of the software system makes sense if downtime has severe impact. Probabilistically, we may measure reliability as RS = (1 – R(t)) with R(t) = $exp\left[\int_0^t -\lambda(\tau)d\tau\right]$,

where, RS = System reliability,

τ = the variable of integration,

$\lambda(\tau)$ = the failure rate and

R(t) = the probability that a failure will occur at time t given that no failure has occurred before time t.

This traditional software reliability model considers only timing and failure rate but this approach is incomplete as the failure rate should be connected to application complexity and test effectiveness as well (Chu et al., 2007).

Software unreliability is the result of faulty design and architecture of the system, issues arises due to human failures. CBSS leads to a complex system if not managed properly. Hence, estimation of reliability of such systems is very difficult. However, conventional approaches of software reliability analysis use black-box models, but this model is not useful in case of CBSS because it ignores the reliability of individual components. Therefore, several techniques have been developed for estimating component-based software applications reliability (Goševa-Popstojanova & Trivedi, 2001). These reliability estimation techniques may fall into two buckets:

- Overall system reliability estimation considering the application as a whole,
- Application reliability estimation on the basis of individual component reliability.

In CBSS, individual components contribute to the overall reliability of the system. Therefore, the selection process for finding out the most satisfying component (alternative), that contributes most to the reliability estimation of overall application, among set of feasible components (alternatives) subject to certain constraints (criteria) should be achieved using one of the proven multi-criteria decision making (MCDM) methods.

Sometimes, decision makers have to consider multiple criteria at a time. When decision makers are required to perform quantitative as well as qualitative decisions to determine the performance of each feasible alternative with respect to available criterion, then this sort of problem is known as multi-criteria group decision making (MCGDM) problem. Multiple-criteria decision making (MCDM) is an approach of ranking of alternatives from a set of available alternatives. MCDM approach may broadly be divided into two categories (Wang & Lee, 2007):

- Classical MCDM (Feng & Wang, 2000): Classical MCDM decisions are made under those circumstances where performance rating and weight can be given precisely.
- Fuzzy MCDM (Wang et al., 2003): Fuzzy set theory is used to model the uncertainty, perception etc. of human judgments, so fuzzy MCDM decisions are made where performance rating and weight are not precise.

In this paper, we analysed the ranking of components within a system using MOORA, a fuzzy MCDM method. We took four very critical factors for reliability estimation of a CBSS Tyagi and Sharma (2012), as criteria for rank assessment. We illustrated the applicability of the proposed method for three CBSS applications namely; Security Management System, Distributed Medical Informatics System and Waiting Queues Simulator.

2. Literature Review

It is very difficult to assess the reliability of the system as it is a real time issue. It cannot be perceive as a binary notion also, where we cannot say that if a program produces desired output then its reliability will be one and if it does not produce desired output then reliability will be zero. Researchers have proposed several models for CBSS reliability estimation. These approaches may fall into three buckets: architecture based models, hard computing based models and soft computing technique based models.

2.1. Architecture Based Models

Software architecture describes the structure of software at an abstract level. Architectural style is a pattern that describes the configurations of components and their connectors. Three different styles may be recognized as state-based models, path-based models and additive models. State-based models estimate software reliability analytically and used control flow graph (CFG) to represent the software architecture. Path-based models compute the software reliability by considering all the feasible execution paths of the program. Additive models assume that reliability of each component may be modelled by the use of Non-Homogeneous Poisson Process. Architecture based reliability models common requirement include (1) component identification, (2) Software architecture blueprint description, (3) failure behaviour outlining, and (4) analysis of failure behaviour with the architecture. Architecture-based models allow insight into the sensitivity of the system with respect to each component. Therefore, it may be used to identify critical components from the reliability point of view.

An analysis of Goševa-Popstojanova and Trivedi (2001), Kaiyuan et al. (2003) and Gokhle (2007) reveals that state-based and path-based architecture models are more suitable for estimating CBSS reliability. As CBSS are made up of components, components may transform from one operational profile to another Koziolek et al. (2005). Control flow among components are also taken into consideration in the state-based models (Littlewood, 1979; Cheung, 1980; Gokhle et al., 1998; Wang et al., 2006). It assumed that components failed independently. State-based models may be represented as hierarchical or as composite models. Path-based models are proposed to consider all possible execution paths for the reliability estimation of an application (Shooman, 1976); (Krishnamurthy et al., 1997). A limitation of this type of model is that they provide only an approximate estimation of software reliability.

Dependency among consecutive software runs also taken into consideration in a model based on Markov renewal process (Goseva & Trivedi, 1999). This model was adaptable for both independent and dependent successive software runs. As per Markov behaviour model present behaviour is not dependent on its previous behaviour. A similar approach is proposed in Wang et al. (2008) in which components relationships were analysed for CBSS reliability estimation. A constraint of this method is that it assumes that transition probabilities and component reliability are present, but actually this is not the case always.

2.2 Hard Computing Based Models

Hard computing based reliability models requires a precisely stated analytical model based on crisp systems, numerical analysis, binary logic and crisp software. These models describe in terms of mathematical logic (combinatorial logic), component usage ratio and presented by graphs such as trees, component dependency graphs and block diagrams. The statistical information on the reliability of components of the system is used here. Researchers have proposed many mathematical models for estimating reliability of CBSS.

A path based reliability estimation approach based on Component Dependency Graph (CDG), known as Scenario-Based Reliability Analysis (SBRA) addressed in Yacub et al. (2004). This approach does not consider failure dependency among components, results in limited use of this approach. Another similar approach based on CDG is presented in Fan Zhang et al. (2009). This model may be used to check the behaviour of reliability when the operation profile changes, provided an operational profile of the system is given. A framework based on algebra for describing the syntax and predicting the CBSS reliability proposed in Huang et al. (2008) and Seth et al. (2010) presented a minimum spanning tree based approach for estimating the reliability.

A mathematical model of reliability estimation which takes the component usage ratio into consideration has been addressed in Goswami and Acharya (2009). This approach seems feasible to use in real-time application due to the flexibility of the component usage ratio. Component composition mechanism was also taken into consideration for reliability estimation Si et al. (2010). A procedure is used to estimate the overall application reliability based on component utilization frequencies and component composition mechanism once the reliability of each component is being calculated. Modified Adaptive Testing (MAT) approach allows to test case history to be used (Hu et al., 2013).

Rewrite logic (RABRL) developed by Wang et al. (2008) may be applied to simple CBSS whose specification is provided with an operational profile. However, it does not consider failure dependencies among components but can statistically analyse execution process of an application and uses this to estimate the transition probability, approximately, between components and the estimated number of visits to components. Hsu et al. (2011) presented an adaptive technique using path testing for reliability estimation of complex CBSS where sequence, branch and loop structures were proposed for estimation of path reliability. Using this approach, propitious estimation of application software reliability becomes possible when testing information is available. Correlated component failures (COCOF) approach considers the component reliability, correlation and application architecture to assess the reliability of software application (Fiondella et al., 2013).

2.3 Soft Computing Based Models

Soft computing based reliability models are based on fuzzy logic, probabilistic reasoning, genetic algorithms and neural networks. Fuzzy logic deals with imprecision, approximate reasoning and qualitative representation of aspects. Probabilistic reasoning such as Bayesian Belief Networks has the ability to update previous results by conditioning them with newly available evidence. Genetic algorithms approach is based on analogues of natural section, such as the optimization methods based on ant colonies. Neural networks focus on learning systems, self-organizing structures and implementation of models from the available data. In hard computing uncertainty and imprecision are undesirable properties but in soft computing the tolerance for uncertainty and imprecision is exploited to achieve lower cost, tractability and economy of communication. In CBSS, Reliability estimation factors cannot be quantified easily because of vagueness in information and fuzziness in human perception. Therefore, soft-computing based models represent an alternative tool to aid research in this area.

Several soft computing techniques are based on support vector machines (SVMs), fuzzy logic and genetic algorithms (GAs) (Lo, 2010). Parameters of reliability estimation for the SVM can be determined by the GA. This model is more precise in reliability prediction and is less dependent on the size of failure testing data in comparison with other models. A framework based on possibility theory and fuzzy logic for CBSS reliability estimation proposed by Dimov et al. (2010). A mathematical fuzzy model was proposed to predict the CBSS reliability based on necessity and possibility. This model is based on uncertainty so it does not require component failure data unlike other models. Though, a mechanism is needed to model the propagation of failure between failure behaviour and components.

A Bayesian reliability estimation model for prediction and assessment of application reliability using unified modelling language (UML) was proposed by Singh et al. (2001), which helps in analyzing the reliability at the design level itself. A limitation of this approach is that it generates separate operational profiles every time, whenever new paths are taken into account because the reliability prediction algorithm considers them to be an entirely new system. A rule based approach for CBSS reliability estimation based on fuzzy logic presented was presented by Tyagi and Sharma (2012). Four critical factors were identified in this approach for CBSS reliability estimation and apply these factors to create a rule base of the application. Fuzzy technique for order preference by similarity to ideal solution (TOPSIS) was employed for ranking of components for estimating CBSS reliability; readers may refer to Tyagi and Sharma (2014) for detailed implementation.

3. Material and Methods

Ranking of the components within a system is a multi-criteria decision making (MCDM) problem. MCDM is the selection process of finding out the most satisfying alternative among set of feasible alternatives subject to certain constraints. The selection is obtained by evaluating each choice (alternatives) on the set of objectives (criteria). Fuzzy MCDM decisions are made where performance rating and weight are not precise. Therefore, fuzzy MCDM method has a role to play in order to rank the components of CBSS, where decision factors (criteria) are conflicting to each other while evaluating the reliability of the application.

MOORA is a fuzzy MCDM method which is used to solve the problems related to decision and planning using multiple conflicting criteria. Among these conflicting criteria some of them have beneficial nature where maximum values are desired and the others have non-beneficial nature where minimum values are preferred. MOORA method considers both beneficial and non-beneficial criteria for ranking alternatives from a set of available feasible options. Ratio analysis enables MOORA to determine the most suitable alternative among all possible alternatives, and dimensionless measurement realises the job of ranking of components for estimating CBSS reliability in a non-subjective way. Fuzzy MCDM combines the concept of MCDM and the fuzzy set theory. Performance ratings of alternatives and weights of criteria are expressed in linguistic variables in fuzzy MCDM approach. Linguistic variables are then transformed into triangular fuzzy numbers.

3.1. Fuzzy Number

Fuzzy set theory may be employed to model systems that are hard to define precisely because of the vagueness in information and fuzziness in human perception (Zadeh, 1965). This fuzziness and uncertainty may be modelled in terms of fuzzy numbers. A fuzzy number has three types namely; triangular fuzzy number, trapezoidal fuzzy number and parabolic fuzzy number.

Let U be the universe of discourse, $\bar{G}$ be the fuzzy subset of U which is defined by a membership function $\mu_{\bar{G}}$(u) that maps each element of u in U to a real number in the interval [0,1]. The function value of $\mu_{\bar{G}}$(u) denotes the grade of membership of u in $\bar{G}$. A triangular fuzzy number $\bar{G}$ is a triplet composed of crisp numbers, (s, c, e), and the membership function is defined as.

$$\mu_{\bar{G}}(\mathrm{u}) = \begin{cases} \frac{u-s}{c-s}, & s \le u \ \le c \\ \frac{u-e}{c-e}, & \mathrm{c} \le u \ \le e \\ 0, & otherwise \end{cases}$$

3.2 Fuzzy-MOORA Method

Multi-objective optimization on the basis of ratio analysis (MOORA) is the process of optimizing two or more conflicting attributes simultaneously subject to certain constraints (Brauers & Zavadskas, 2006). Therefore, this technique seems to be an appropriate tool to assess the ranking or selecting best alternatives from a set of available feasible options. It has been successfully applied in many fields of engineering manufacturing (Chakraborty, 2011), construction engineering and management (Kracka et al., 2015; Brauers et al., 2008) and also in economics (Brauers & Ginevicious, 2009; Brauers & Zavadskas, 2010), due to its comprehensiveness and simplicity. The values of objectives are measured for every alternative, and this provides the basis for comparison of alternatives and consequently facilitates the selection of best (most satisfactory) option.

The main logic of MOORA method is to calculate the overall performance of each alternative as the difference between sum of its normalized performances of beneficial and non-beneficial attributes by using the following equation.

$$f_p = \sum_{q=1}^{r} u_{pq}^{*} - \sum_{q=r+1}^{n} u_{pq}^{*} \, , \tag{1}$$

where u_{pq}^{*} is a dimensionless number, belongs to the interval of [0, 1], and represents the normalized performance of the pth alternative on qth criteria, r is the attributes count to be maximised (beneficial criteria) and (n-r) is the attributes count to be minimised (non-beneficial criteria and f_p is the overall performance indices value of pth alternative with respect to all other attributes.

Priority weights may be considered to give relative importance of one criterion over the other, as it is generally observed that some criteria are more important than the others. When priority weights are taken into consideration, Eq. (1) can be written as:

$$f_p = \sum_{q=1}^{r} l_q u_{pq}^{*} - \sum_{q=r+1}^{n} l_q u_{pq}^{*} \qquad (q = 1,2,3, \dots n) \tag{2}$$

where, l_q is the weight of qth attribute which may be derived by entropy method or analytical hierarchical process (AHP). Final preferences may be find out by the ordinal ranking of f_p. Highest f_p value corresponds to the best alternative whereas lowest f_p value corresponds to worst alternative.

In this paper, fuzzy-MOORA is proposed for solving the MCDM problem related to ranking of components for CBSS reliability estimation. This approach was also successfully employed to find out best ERP system for manufacturing organizations (Karande & Chakraborty, 2012). Here, this method is reiterated in the following procedural steps.

Step 1: Firstly, develop a fuzzy decision matrix based on the opinions of key decision makers where each criterion is measured using triangular membership function.

$$U = \begin{bmatrix} [u_{11}^{s}, u_{11}^{c}, u_{11}^{e}] & [u_{12}^{s}, u_{12}^{c}, u_{12}^{e}] & \cdots & [u_{1j}^{s}, u_{1j}^{c}, u_{1j}^{e}] \\ [u_{21}^{s}, u_{21}^{c}, u_{21}^{e}] & [u_{22}^{s}, u_{22}^{c}, u_{22}^{e}] & \dots & [u_{2j}^{s}, u_{2j}^{c}, u_{2j}^{e}] \\ \dots & \dots & \dots & \dots \\ [u_{i1}^{s}, u_{i1}^{c}, u_{i1}^{e}] & [u_{i2}^{s}, u_{i2}^{c}, u_{i2}^{e}] & \dots & [u_{ij}^{s}, u_{ij}^{c}, u_{ij}^{e}] \end{bmatrix}, \tag{3}$$

where u_{pq}^{s}, u_{pq}^{c} and u_{pq}^{e} denote the lower, middle and upper values of a triangular membership function for the pth alternative with respect to qth criteria.

Step 2: Normalize the fuzzy decision matrix, created in step 1, using vector normalization procedure. Following equations are used for this purpose (Stanujkic et al., 2012).

$$g_{pq}^{s} = \frac{u_{pq}^{s}}{\sqrt{\sum_{p=1}^{i} \left[\left(u_{pq}^{s}\right)^2 + \left(u_{pq}^{c}\right)^2 + \left(u_{pq}^{e}\right)^2 \right]}} \tag{4}$$

$$g_{pq}^{c} = \frac{u_{pq}^{c}}{\sqrt{\sum_{p=1}^{i}\left[\left(u_{pq}^{s}\right)^{2} + \left(u_{pq}^{c}\right)^{2} + \left(u_{pq}^{e}\right)^{2}\right]}} \tag{5}$$

$$g_{pq}^{e} = \frac{u_{pq}^{e}}{\sqrt{\sum_{p=1}^{i}\left[\left(u_{pq}^{s}\right)^{2} + \left(u_{pq}^{c}\right)^{2} + \left(u_{pq}^{e}\right)^{2}\right]}} \tag{6}$$

Step 3: Weighted normalized fuzzy decision matrix is determine by the following equations,

$$h_{\mathrm{pq}}^{\mathrm{s}} = l_q g_{pq}^{s} \tag{7}$$

$$h_{\mathrm{pq}}^{\mathrm{c}} = l_q g_{pq}^{c} \tag{8}$$

$$h_{\mathrm{pq}}^{\mathrm{e}} = l_q g_{pq}^{e} \tag{9}$$

Step 4: Overall ratings of beneficial and non-beneficial criteria for each alternative is calculated is this step.

The overall rating of an alternative for lower, middle and upper values of the triangular function for beneficial criteria are determined by the following equations,

$$\mathrm{k}_{\mathrm{p}}^{+s} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{s}} \quad | \; q \in Q^{max} \tag{10}$$

$$\mathrm{k}_{\mathrm{p}}^{+c} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{c}} \quad | \; q \in Q^{max} \tag{11}$$

$$\mathrm{k}_{\mathrm{p}}^{+e} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{e}} \quad | \; q \in Q^{max} \tag{12}$$

The overall rating of an alternative for lower, middle and upper values of the triangular function for non-beneficial criteria are determined by the following equations,

$$\mathrm{k}_{\mathrm{p}}^{-s} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{s}} \quad | \; q \in Q^{min} \tag{13}$$

$$\mathrm{k}_{\mathrm{p}}^{-c} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{c}} \quad | \; q \in Q^{min} \tag{14}$$

$$\mathrm{k}_{\mathrm{p}}^{-e} = \sum_{q=1}^{n} h_{\mathrm{pq}}^{\mathrm{e}} \quad | \; q \in Q^{min} \tag{15}$$

Step 5: In this step, defuzzied the values of the overall ratings for beneficial and non-beneficial criteria for each alternative using the vertex method (Huiqun & Guang, 2012) to determine the overall performance index (Si).

$$K_p(k_p^+, k_p^-) = \sqrt{\frac{1}{3}\left[\left(k_p^{+s} - k_p^{-s}\right)^2 + \left(k_p^{+c} - k_p^{-c}\right)^2 + \left(k_p^{+e} - k_p^{-e}\right)^2\right]} \tag{16}$$

Step 6: Now, arrange the overall performance indices in the descending order and rank them from best to worst. The alternative with the highest overall performance index is the most favourable choice.

4. Application of fuzzy-MOORA for Component Ranking

To determine the ranking of components for estimating CBSS reliability applications using fuzzy-MOORA we selected four critically acclaimed factors (Application Complexity (C1), Operational Profile (C2), Reusability (C3) and Functionality (C4)) proposed by Tyagi and Sharma (2012) as criteria for rank assessment. A team of five decision makers, experts in their field, was formed to evaluate the CBSS proposed criterion. Linguistic variables and their fuzzy numbers are shown in Table 1. Weightage for each criterion, as shown in Table 2, is then collected by different decision makers.

Table 1
Fuzzy numbers for linguistic variables

Criterion weightage	Reliability ranking	Fuzzy number
VL	VP	(1,1,3)
L	P	(1,3,5)
M	F	(3,5,7)
H	G	(5,7,9)
VH	VG	(7,9,9)

Table 2
Weightage ranking assignment

Criteria	D1	D2	D3	D4	D5
C1	VH	VH	VH	H	H
C2	M	M	H	M	L
C3	VH	H	H	VH	VH
C4	L	M	M	H	H

5. Case Study

Three case studies are illustrated in this paper to analyse the applicability and potentiality of the proposed method.

5.1 Security Management System

In this case study, we considered a small component-based security management system (SMS) presented by Tyagi and Sharma (2014). This system has total eight components namely: (1) Login (A1), (2) Server (A2), (3) Time Management System (A3), (4) Alarm Management System (A4), (5) Access Management System (A5), (6) Calculation Script (A6), (7) Door Management System (A7) and (8) Devices (A8). These components are considered as alternatives for the ranking. Ranking of components for CBSS reliability estimation problem hierarchy is described in Fig. 1. The domain of SMS is the set of components and the execution paths from source node to destination node. The graphical representation of the system is described in Fig. 2.

By catering decision maker’s subjective judgement develop the fuzzy criteria and then use the linguistic variables to evaluate the ratings of alternative with respect to each criterion and corresponding weights as shown in Table 3.

Table 3
Reliability ranking assignment matrix for SMS

Criteria	Candidate	D1	D2	D3	D4	D5
Application complexity (C1)	Login (A1)	VP	P	P	F	F
	Server (A2)	G	VG	VP	P	P
	Time MS (A3)	F	G	G	VG	F
	Alarm MS (A4)	VG	F	G	G	G
	Access MS (A5)	P	F	P	P	P
	Calculation Script (A6)	G	G	G	G	G
	Door MS (A7)	F	F	G	G	G
	Devices (A8)	P	P	P	P	P
Operational profile (C2)	Login (A1)	P	P	F	P	P
	Server (A2)	P	F	G	G	F
	Time MS (A3)	G	G	G	VG	G
	Alarm MS (A4)	VG	G	G	G	G
	Access MS (A5)	VG	G	F	F	G
	Calculation Script (A6)	G	VG	G	P	VG
	Door MS (A7)	G	VG	G	G	VG
	Devices (A8)	P	F	P	G	F
Reusability (C3)	Login (A1)	P	VP	P	VP	P
	Server (A2)	P	VP	P	VP	VP
	Time MS (A3)	G	G	G	VG	G
	Alarm MS (A4)	G	F	P	G	G
	Access MS (A5)	P	P	VP	P	P
	Calculation Script (A6)	VG	VG	VG	G	G
	Door MS (A7)	G	G	F	G	P
	Devices (A8)	VP	P	P	F	G
Functionality (C4)	Login (A1)	VG	G	P	F	F
	Server (A2)	VG	G	P	F	G
	Time MS (A3)	G	G	F	G	VG
	Alarm MS (A4)	G	G	F	G	G
	Access MS (A5)	G	G	F	G	G
	Calculation Script (A6)	F	G	G	G	VG
	Door MS (A7)	VG	VG	G	VG	G
	Devices (A8)	VG	G	G	VG	G

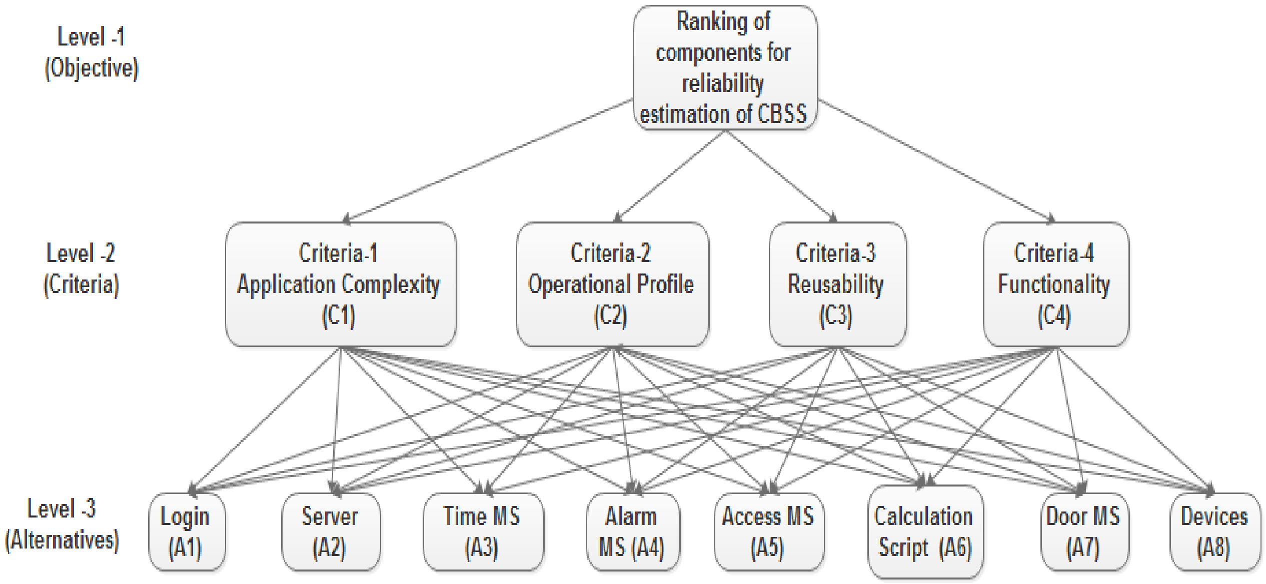
Level -1
(Objective)
Ranking of components for reliability estimation of CBSS
Level -2
(Criteria)
Criteria-1 Application Complexity (C1)
Criteria-2 Operational Profile (C2)
Criteria-3 Reusability (C3)
Criteria-4 Functionality (C4)
Level -3
(Alternatives)
Login (A1)
Server (A2)
Time MS (A3)
Alarm MS (A4)
Access MS (A5)
Calculation Script (A6)
Door MS (A7)
Devices (A8)

Fig. 1. SMS reliability estimation problem hierarchy

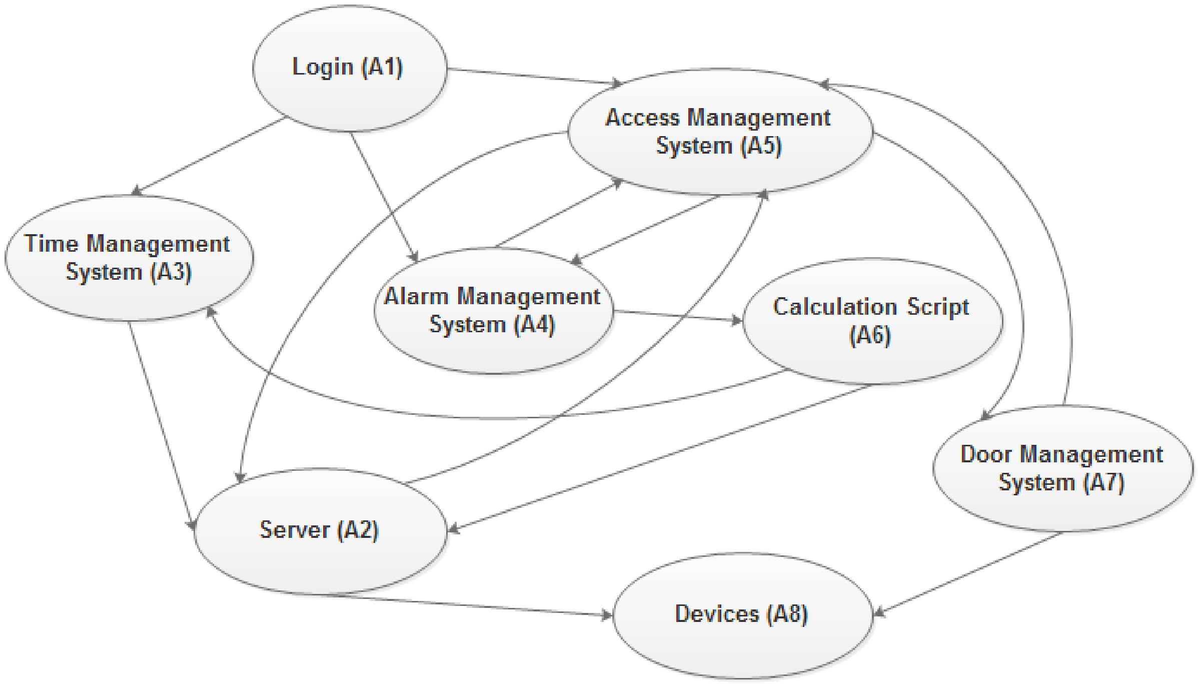
Login (A1)
Access Management System (A5)
Time Management System (A3)
Alarm Management System (A4)
Calculation Script (A6)
Door Management System (A7)
Server (A2)
Devices (A8)

Fig. 2. Graphical representation of SMS

The fuzzy decision matrix for this system was developed using triangular fuzzy membership function, as given in Table 4 and fuzzy TOPSIS method was then applied to determine the best alternative as time MS, whereas, login was the worst chosen component. We, here, implemented more stable and robust fuzzy-MOORA method for evaluation and assessment of ranking of component for reliability estimation of SMS (a component-based system).

5.1.1 Results

Firstly, determine the weights of four criteria, application complexity (C1), operational profile (C2), reusability (C3) and functionality (C4), from the fuzzy numbers of weights given in Table 4 using AHP method as $L_{c1} = 0.2629$, $L_{c2} = 0.2280$, $L_{c3} = 0.2629$ and $L_{c4} = 0.2462$.

Step 1: Design fuzzy decision matrix using triangular fuzzy membership function as shown in Table 4.

Table 4
Fuzzy decision matrix for SMS

	C1	C2	C3	C4
A1	(1.8,3.4,5.4)	(1.4,3.4,5.4)	(1.0,2.2,4.2)	(3.8,5.8,7.4)
A2	(3.0,4.6,6.2)	(3.4,5.8,7.4)	(1.0,1.8,3.8)	(4.2,6.2,7.8)
A3	(4.6,6.6,8.2)	(5.4,7.4,9.0)	(5.4,7.4,9.0)	(5.0,7.0,8.6)
A4	(5.0,7.0,8.6)	(5.4,7.4,9.0)	(3.8,5.8,7.8)	(4.6,6.6,8.6)
A5	(1.4,3.4,5.4)	(4.6,6.6,8.2)	(1.0,2.6,4.6)	(4.6,6.6,8.6)
A6	(5.0,7.0,9.0)	(5.0,7.0,8.2)	(6.2,8.2,9.0)	(5.0,7.0,8.6)
A7	(4.2,6.2,8.2)	(5.8,7.8,9.0)	(3.8,5.8,7.8)	(6.2,8.2,9.0)
A8	(1.0,3.0,5.0)	(2.6,4.6,6.6)	(2.2,3.8,5.8)	(5.8,7.8,9.0)
Weight	(6.2,8.2,9.0)	(3.0,5.0,7.0)	(6.2,8.2,9.0)	(3.4,5.4,7.4)

Step 2: Determine the normalized fuzzy decision matrix using Eqs. (4-6). It is essential for an MCDM method to normalize the decision matrix to make its elements dimensionless and comparable. Normalized fuzzy decision matrix is shown in Table 5.

Table 5
Normalized fuzzy decision matrix for SMS

	C1	C2	C3	C4
A1	(0.066,0.124,0.197)	(0.044,0.108,0.171)	(0.038,0.084,0.159)	(0.112,0.171,0.218)
A2	(0.110,0.168,0.227)	(0.108,0.184,0.235)	(0.038,0.068,0.144)	(0.124,0.182,0.229)
A3	(0.168,0.241,0.300)	(0.171,0.235,0.286)	(0.205,0.281,0.342)	(0.147,0.206,0.253)
A4	(0.182,0.256,0.314)	(0.171,0.235,0.286)	(0.144,0.220,0.296)	(0.135,0.194,0.253)
A5	(0.051,0.124,0.197)	(0.146,0.209,0.260)	(0.038,0.099,0.175)	(0.135,0.194,0.253)
A6	(0.168,0.256,0.329)	(0.159,0.222,0.260)	(0.235,0.311,0.342)	(0.147,0.206,0.253)
A7	(0.154,0.227,0.300)	(0.184,0.248,0.286)	(0.144,0.220,0.296)	(0.182,0.241,0.265)
A8	(0.037,0.110,0.182)	(0.083,0.146,0.209)	(0.084,0.144,0.220)	(0.171,0.229,0.265)

Step 3: Design the weighted normalized fuzzy decision matrix by multiplying the normalized fuzzy criteria values with the corresponding values of crisp criteria weights (L_{c1}, L_{c2}, L_{c3}, L_{c4}) using Eq. (7), Eq. (8) and Eq. (9). Weighted normalized fuzzy decision matrix is shown in Table 6.

Table 6
Weighted normalized fuzzy decision matrix for SMS

	C1	C2	C3	C4
A1	(0.0174,0.0326,0.051	(0.0100,0.0246,0.039	(0.0100,0.0221,0.041	(0.0276,0.0421,0.053
A2	(0.0289,0.0442,0.059	(0.0246,0.0420,0.053	(0.0100,0.0179,0.037	(0.0305,0.0448,0.056
A3	(0.0442,0.0634,0.078	(0.0390,0.0536,0.065	(0.0539,0.0739,0.089	(0.0362,0.0507,0.062
A4	(0.0478,0.0673,0.082	(0.0390,0.0536,0.065	(0.0379,0.0578,0.077	(0.0332,0.0478,0.062
A5	(0.0134,0.0326,0.051	(0.0333,0.0477,0.059	(0.0100,0.0260,0.046	(0.0332,0.0478,0.062
A6	(0.0442,0.0673,0.086	(0.0363,0.0506,0.059	(0.0618,0.0818,0.089	(0.0362,0.0507,0.062
A7	(0.0405,0.0597,0.078	(0.0420,0.0565,0.065	(0.0379,0.0578,0.077	(0.0448,0.0593,0.065
A8	(0.0097,0.0289,0.047	(0.0189,0.0333,0.047	(0.0221,0.0379,0.057	(0.0421,0.0564,0.065

Step 4: Among these four criteria application complexity (C1) is the only non-beneficial attribute, where lower value is preferred, whereas higher values are desired for operational profile (C2), reusability (C3) and functionality (C4). The overall rating of beneficial criteria is calculated using Eqs. (10-12) and the overall rating of non-beneficial criteria is calculated using Eqs. (13-15).The overall ratings for beneficial and non-beneficial criteria for the SMS is shown in Table 7.

Table 7
Overall ratings for beneficial and non-beneficial criteria for the SMS

	S+			S-		
	l	m	u	l	m	u
A1	0.0476	0.0888	0.1345	0.0174	0.0326	0.0518
A2	0.0651	0.1047	0.1479	0.0289	0.0442	0.0597
A3	0.1291	0.1782	0.2174	0.0442	0.0634	0.0789
A4	0.1101	0.1592	0.2053	0.0478	0.0673	0.0826
A5	0.0765	0.1215	0.1676	0.0134	0.0326	0.0518
A6	0.1343	0.1831	0.2115	0.0442	0.0673	0.0865
A7	0.1247	0.1736	0.2082	0.0405	0.0597	0.0789
A8	0.0831	0.1276	0.1707	0.0097	0.0289	0.0478

Step 5: Derive the values of overall performance index for all the components of SMS by using the vertex method as shown in Eq. (16) to defuzzify the overall ratings for both beneficial and non-beneficial criteria as illustrated in Table 8.

Table 8
Overall performance index values and the rank assignment of all the components for the SMS

	S		Rank
A1	S1	0.0603	8
A2	S2	0.0652	7
A3	S3	0.1148	1
A4	S4	0.0955	5
A5	S5	0.0918	6
A6	S6	0.1113	2
A7	S7	0.1107	3
A8	S8	0.1004	4

Step 6: The ranking of SMS components is shown in Table 8, therefore alternative A3 (Time MS) will contribute the most and alternative A1 (Login) will contribute the least to the reliability estimation of overall application of SMS. Components rank are as follows (starting with the best alternative): A3 ≻ A6 ≻ A7 ≻ A8 ≻ A4 ≻ A5 ≻ A2 ≻ A1. It is worthwhile to mention here that Tyagi and Sharma (2014) also derived the same the best and the worst alternatives but the ranking of the intermediate components were found different. Comparative ranking of SMS components obtained by fuzzy-MOORA and fuzzy TOPSIS is shown in Fig. 3.

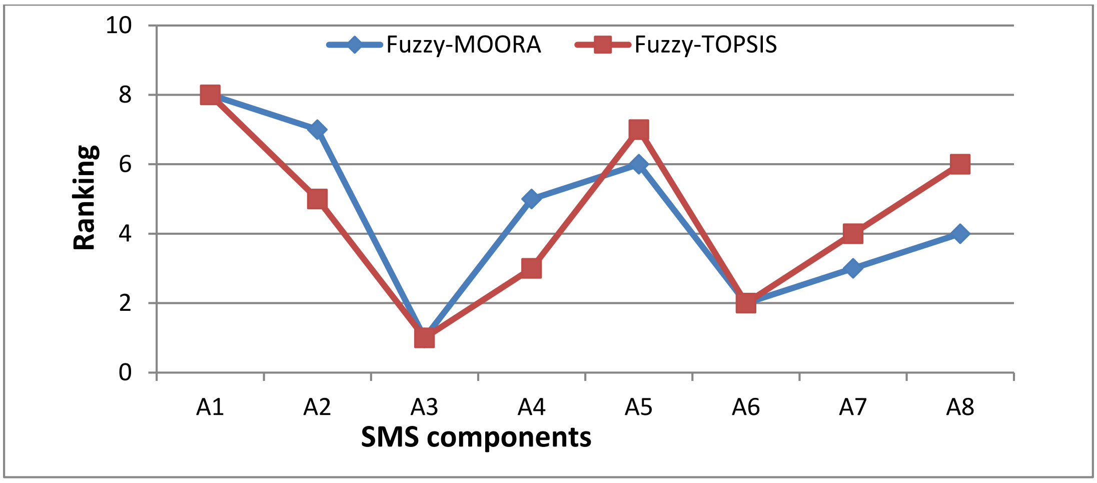

Fig. 3. Comparative ranking of SMS components obtained by fuzzy-MOORA and fuzzy TOPSIS

The disparities among the intermediate rankings of the components may be due to the diverse opinion given by the decision makers.

5.2. Waiting Queues Simulator

In this case study, we considered a small component-based waiting queues simulator (WQS) presented by Yacoub et al. (2004). WQS mimics the behaviour of customer waiting queues lined up at checkout counters at immigration posts, airport, railway station, supermarkets etc. It was developed by combining the components. This system has total six components namely: (1) Generator (A1), (2) Simulator Driver (A2), (3) Service Facility (A3), (4) Event List (A4), (5) Queuing Facility (A5) and (6) Measurement (A6). These components are considered as alternatives for the ranking. Ranking of components for CBSS reliability estimation problem hierarchy is described in Fig. 4. Components available in the application are taken as alternatives. The architecture of the WQS is centred around a dynamic Event List component as the communication vehicle of events. The graphical representation of the system is described in Fig. 5.

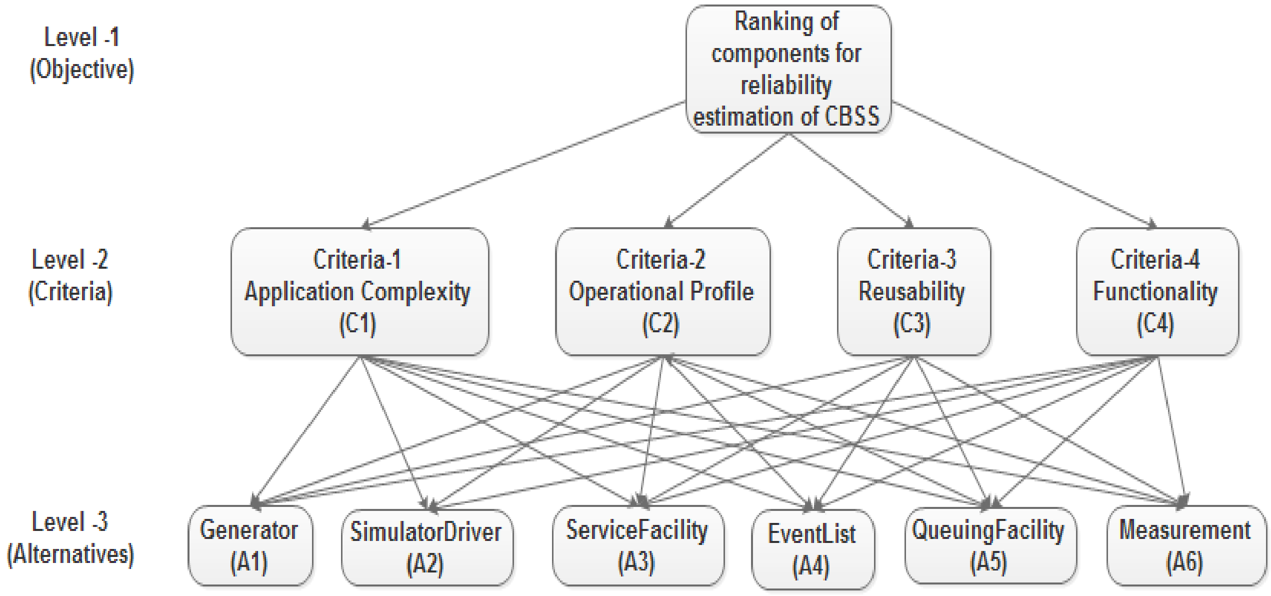

Fig. 4. WQS reliability estimation problem hierarchy

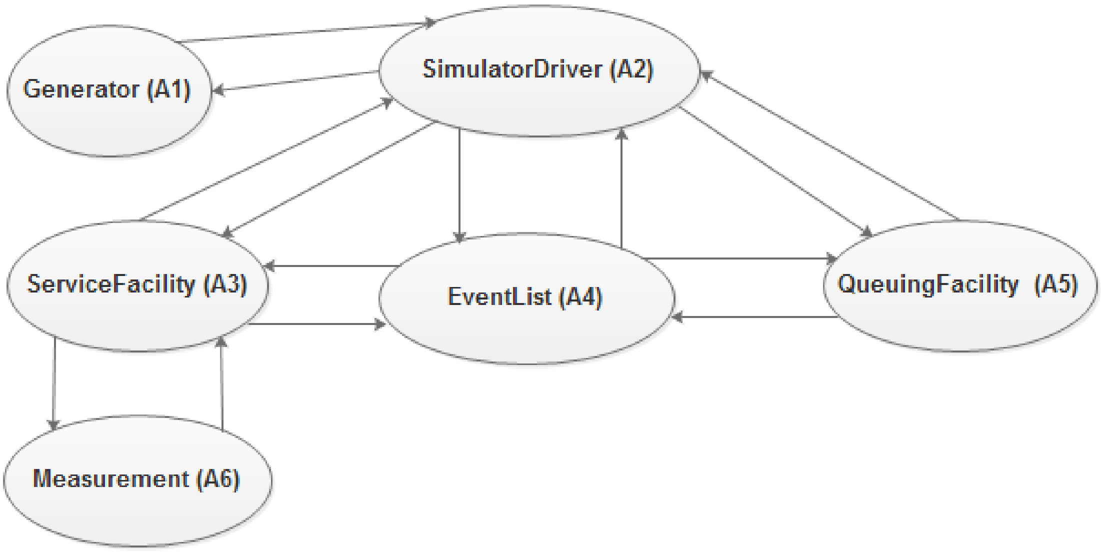

Fig. 5. Graphical representation of WQS

We, here, applied the fuzzy-MOORA method for assessment of ranking of component for reliability estimation of WQS (a component-based system).

5.2.1. Results

Set the linguistic ratings for all alternatives with respect to criteria. Linguistic variables and their fuzzy numbers are shown in Table 1. Weightage for each criteria is then collected by decision makers, as shown in Table 2. A team of five decision makers, D1, D2, D3, D4 and D5, was formed to evaluate the WQS. Now, by catering decision maker’s subjective judgement develop the fuzzy criteria and then use the linguistic variables to evaluate the ratings of alternative with respect to each criteria as shown in Table 9.

Table 9

Reliability ranking assignment matrix for WQS

Criteria	Candidate	D1	D2	D3	D4	D5
Application complexity (C1)	Generator (A1)	F	F	F	P	P
	Simulator Driver (A2)	G	G	F	F	P
	Service Facility (A3)	VG	VG	G	VG	G
	Event List (A4)	F	F	G	F	F
	Queuing Facility (A5)	VG	VG	G	G	VG
	Measurement (A6)	P	P	P	VP	VP
Operational profile (C2)	Generator (A1)	P	P	VP	F	P
	Simulator Driver (A2)	VG	VG	VG	G	G
	Service Facility (A3)	F	F	F	G	G
	Event List (A4)	F	G	G	G	F
	Queuing Facility (A5)	VG	VG	G	VG	VG
	Measurement (A6)	P	P	VP	VP	VP
Reusability (C3)	Generator (A1)	VG	VG	VG	G	G
	Simulator Driver (A2)	VP	P	VP	P	P
	Service Facility (A3)	F	VP	F	P	P
	Event List (A4)	VP	VP	P	P	VP
	Queuing Facility (A5)	G	G	F	F	F
	Measurement (A6)	VG	VG	VG	VG	G
Functionality (C4)	Generator (A1)	VP	P	VP	VP	P
	Simulator Driver (A2)	VG	VG	VG	G	G
	Service Facility (A3)	F	F	F	G	G
	Event List (A4)	F	F	G	G	F
	Queuing Facility (A5)	G	G	F	F	P
	Measurement (A6)	VG	VG	F	F	G

Weights of four criteria (application complexity (C1), operational profile (C2), reusability (C3) and functionality (C4)), from the fuzzy numbers of weights given in Table 10 using AHP method, can be determined as $L_{c1} = 0.2629$, $L_{c2} = 0.2280$, $L_{c3} = 0.2629$ and $L^{c4} = 0.2462$.

Step 1: Design fuzzy decision matrix using triangular fuzzy membership function as shown in Table 10.

Table 10

Fuzzy decision matrix for WQS

	C1	C2	C3	C4
A1	(2.2,4.2,6.2)	(1.4,3.0,5.0)	(6.2,8.2,9.0)	(1.0,1.8,3.8)
A2	(3.4,5.4,7.4)	(6.2,8.2,9.0)	(1.0,2.2,4.2)	(6.2,8.2,9.0)
A3	(6.2,8.2,9.0)	(3.8,5.8,7.8)	(1.8,3.4,5.4)	(3.8,5.8,7.8)
A4	(3.4,5.4,7.4)	(4.2,6.2,8.2)	(1.0,1.8,3.8)	(3.8,5.8,7.8)
A5	(6.2,8.2,9.0)	(6.6,8.6,9.0)	(3.8,5.8,7.8)	(3.4,5.4,7.4)
A6	(1.0,2.2,4.2)	(1.0,1.8,3.8)	(6.6,8.6,9.0)	(5.0,7.0,8.2)
Weight	(6.2,8.2,9.0)	(3.0,5.0,7.0)	(6.2,8.2,9.0)	(3.4,5.4,7.4)

Step 2: Determine the normalized fuzzy decision matrix using Eqs. (4-6). Normalized fuzzy decision matrix is shown in Table 11.

Table 11

Normalized fuzzy decision matrix for WQS

	C1	C2	C3	C4
A1	(0.086,0.165,0.243)	(0.054,0.116,0.193)	(0.257,0.340,0.374)	(0.039,0.070,0.148)
A2	(0.133,0.212,0.291)	(0.239,0.316,0.347)	(0.042,0.091,0.174)	(0.241,0.319,0.350)
A3	(0.243,0.322,0.353)	(0.147,0.224,0.301)	(0.075,0.141,0.224)	(0.148,0.226,0.304)
A4	(0.133,0.212,0.291)	(0.162,0.239,0.316)	(0.042,0.075,0.158)	(0.148,0.226,0.304)
A5	(0.243,0.322,0.353)	(0.255,0.332,0.347)	(0.158,0.241,0.324)	(0.132,0.210,0.288)
A6	(0.039,0.086,0.165)	(0.039,0.069,0.147)	(0.274,0.357,0.374)	(0.195,0.272,0.319)

Step 3: Design the weighted normalized fuzzy decision matrix by multiplying the normalized fuzzy criteria values with the corresponding values of crisp criteria weights (Lc1, Lc2, Lc3, Lc4) using Eqs. (7-9). Weighted normalized fuzzy decision matrix is shown in Table 12.

Table 12
Weighted normalized fuzzy decision matrix for WQS

	C1	C2	C3	C4
A1	(0.0226,0.0434,0.0639)	(0.0123,0.0264,0.0440)	(0.0676,0.0894,0.0983)	(0.0096,0.0172,0.0364)
A2	(0.0350,0.0557,0.0765)	(0.0545,0.0720,0.0791)	(0.0110,0.0239,0.0457)	(0.0593,0.0785,0.0862)
A3	(0.0639,0.0847,0.0928)	(0.0335,0.0511,0.0686)	(0.0197,0.0371,0.0589)	(0.0364,0.0556,0.0748)
A4	(0.0350,0.0557,0.0765)	(0.0369,0.0545,0.0720)	(0.0110,0.0197,0.0415)	(0.0364,0.0556,0.0748)
A5	(0.0639,0.0847,0.0928)	(0.0581,0.0757,0.0791)	(0.0415,0.0634,0.0852)	(0.0325,0.0517,0.0709)
A6	(0.0103,0.0226,0.0434)	(0.0089,0.0157,0.0335)	(0.0720,0.0939,0.0983)	(0.0480,0.0670,0.0785)

Step 4: Among these four criteria application complexity (C1) is the only non-beneficial attribute, where lower value is preferred and higher values are desired for operational profile (C2), reusability (C3) and functionality (C4). The overall rating of beneficial criteria is calculated using Eqs. (10-12) and the overall rating of non-beneficial criteria is calculated using Eqs. (13-14) and (15).The overall ratings for beneficial and non-beneficial criteria for the WQS is shown in Table 13.

Table 13
Overall ratings for beneficial and non-beneficial criteria for the WQS

	S+			**S-**		
	l	m	u	l	m	u
A1	0.0895	0.1331	0.1788	0.0226	0.0434	0.0639
A2	0.1249	0.1745	0.2110	0.0350	0.0557	0.0765
A3	0.0897	0.1438	0.2024	0.0639	0.0847	0.0928
A4	0.0844	0.1299	0.1884	0.0350	0.0557	0.0765
A5	0.1322	0.1908	0.2352	0.0639	0.0847	0.0928
A6	0.1289	0.1766	0.2104	0.0103	0.0226	0.0434

Step 5: Derive the values of overall performance index of all the components of WQS by Eq. (16) to defuzzify the overall ratings for both beneficial and non-beneficial criteria, as illustrated in Table 14.

Table 14
Overall performance index values and the rank assignment of all the components for the WQS

	S		Rank
A1	S1	0.0926	4
A2	S2	0.1159	2
A3	S3	0.0734	6
A4	S4	0.0826	5
A5	S5	0.1098	3
A6	S6	0.1479	1

Step 6: The ranking of WQS components is shown in Table 14, therefore alternative A6 (Measurement) will contribute the most and alternative A3 (Service Facility) will contribute the least to the reliability estimation of overall application of WQS. Components ranks as follows (starting with the best alternative): A6 $\succ$ A2 $\succ$ A5 $\succ$ A1 $\succ$ A4 $\succ$ A3.

5.3 Distributed Medical Informatics System

In this case study, we considered a small component-based distributed medical informatics system (MIS) originally presented by Yacoub et al. (2004). This system has total five components namely: (1) AE Client (A1), (2) AE Server (A2), (3) DICOM Upper Layer Client (A3), (4) DICOM Upper Layer Server (A4) and (5) Network (A5). Ranking of components for CBSS reliability estimation problem hierarchy is described in Fig. 6. Components available in the application are taken as alternatives. The graphical representation of the MIS is described in Fig. 7. Here, AE Client and AE Server subsystems (application entities) are connected via a Network subsystem.

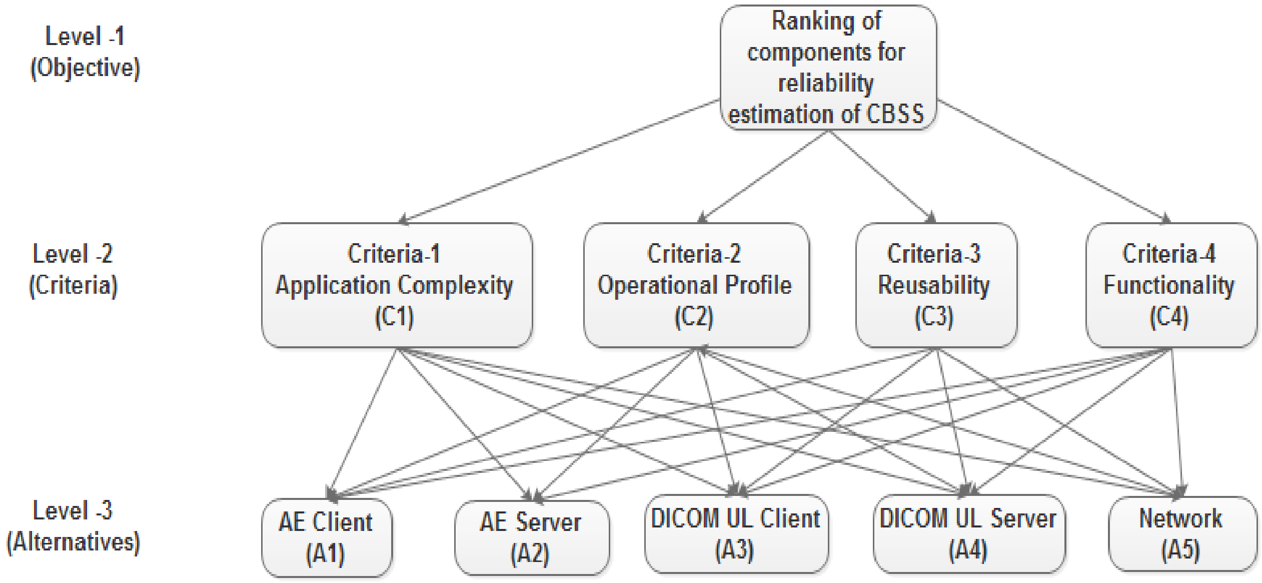
Level -1
(Objective)
Ranking of components for reliability estimation of CBSS
Level -2
(Criteria)
Criteria-1 Application Complexity (C1)
Criteria-2 Operational Profile (C2)
Criteria-3 Reusability (C3)
Criteria-4 Functionality (C4)
Level -3
(Alternatives)
AE Client (A1)
AE Server (A2)
DICOM UL Client (A3)
DICOM UL Server (A4)
Network (A5)

Fig. 6. MIS reliability estimation problem hierarchy

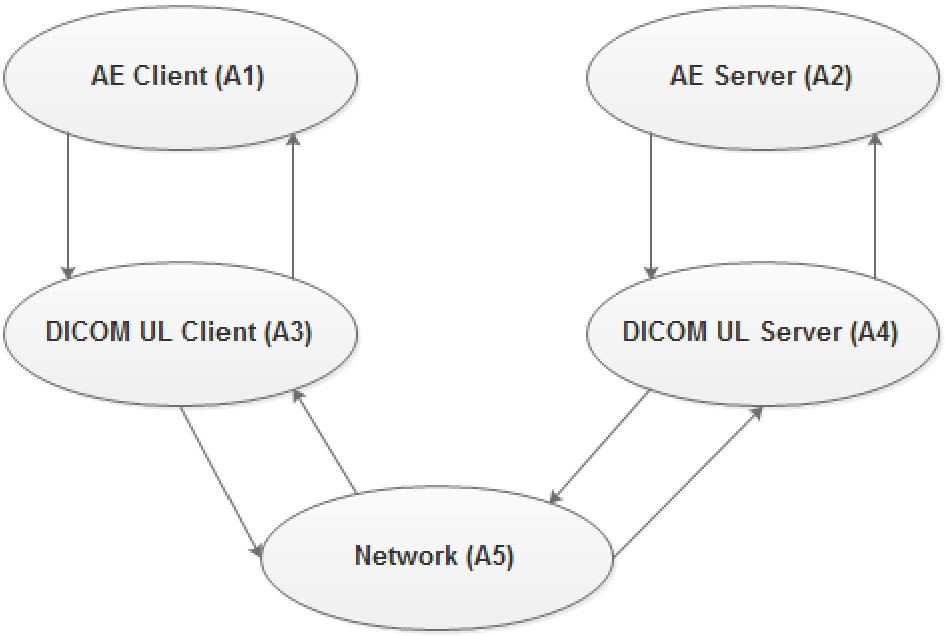

Fig. 7. Graphical representation of MIS

The following describes the fuzzy-MOORA approach analyses we conducted for MIS.

5.3.1. Results

Firstly, we set the linguistic ratings for all alternatives with respect to criteria. Linguistic variables and their fuzzy numbers are shown in Table 1. Weightage for each criteria is then collected by decision makers, as shown in Table 2. A team of five decision makers, D1, D2, D3, D4 and D5, was formed to evaluate the MIS. Now, by catering decision maker’s subjective judgement we develop the fuzzy criteria and then we use the linguistic variables to evaluate the ratings of alternative with respect to each criteria as shown in Table 15.

Table 15
Reliability ranking assignment matrix for MIS

Criteria	Candidate	D1	D2	D3	D4	D5
Application complexity (C1)	AE Client (A1)	F	F	P	P	P
	AE Server (A2)	G	G	VG	G	VG
	DICOM UL Client (A3)	F	G	F	G	G
	DICOM UL Server (A4)	G	G	F	G	P
	Network (A5)	VG	VG	VG	G	G
Operational profile (C2)	AE Client (A1)	P	F	F	P	P
	AE Server (A2)	VP	VP	VP	F	F
	DICOM UL Client (A3)	F	VG	G	G	VG
	DICOM UL Server (A4)	G	G	G	F	F
	Network (A5)	VG	VG	G	G	G
Reusability (C3)	AE Client (A1)	VP	VP	P	P	VP
	AE Server (A2)	F	F	F	G	G
	DICOM UL Client (A3)	VP	VP	VP	P	VP
	DICOM UL Server (A4)	G	G	G	F	F
	Network (A5)	VG	VG	VG	G	VG
Functionality (C4)	AE Client (A1)	P	VP	VP	P	VP
	AE Server (A2)	VP	P	F	F	VP
	DICOM UL Client (A3)	G	G	P	F	F
	DICOM UL Server (A4)	G	G	G	F	F
	Network (A5)	VG	VG	VG	G	G

Determine the weights of four criteria, application complexity (C1), operational profile (C2), reusability (C3) and functionality (C4), from the fuzzy numbers of weights given in Table 16 using AHP method as $L_{c1} = 0.2629$, $L_{c2} = 0.2280$, $L_{c3} = 0.2629$ and $L_{c4} = 0.2462$.

Step 1: Design fuzzy decision matrix using triangular fuzzy membership function as shown in Table 16.

Table 16
Fuzzy decision matrix for MIS

	C1	C2	C3	C4
A1	(1.8,3.8,5.8)	(1.8,3.8,5.8)	(1.0,1.8,3.8)	(1.0,1.8,3.8)
A2	(5.8,7.8,9.0)	(1.8,2.6,4.6)	(3.8,5.8,7.8)	(1.8,3.0,5.0)
A3	(4.2,6.2,8.2)	(5.4,7.4,8.6)	(1.0,1.4,3.4)	(3.4,5.4,7.4)
A4	(3.8,5.8,7.8)	(4.2,6.2,8.2)	(4.2,6.2,8.2)	(4.2,6.2,8.2)
A5	(6.2,8.2,9.0)	(5.8,7.8,9.0)	(6.6,8.6,9.0)	(6.2,8.2,9.0)
Weight	(6.2,8.2,9.0)	(3.0,5.0,7.0)	(6.2,8.2,9.0)	(3.4,5.4,7.4)

Step 2: Determine the normalized fuzzy decision matrix using Eq. (4), (5) and (6). Normalized fuzzy decision matrix is shown in Table 17.

Table 17
Normalized fuzzy decision matrix for MIS

	C1	C2	C3	C4
A1	(0.071.0.150,0.228)	(0.078,0.164,0.250)	(0.046,0.084,0.176)	(0.047,0.084,0.177)
A2	(0.228,0.307,0.354)	(0.078,0.112,0.198)	(0.176,0.269,0.362)	(0.084,0.140,0.233)
A3	(0.165,0.244,0.323)	(0.233,0.319,0.370)	(0.046,0.065,0.158)	(0.158,0.251,0.345)
A4	(0.150,0.228,0.307)	(0.181,0.267,0.353)	(0.195,0.288,0.381)	(0.196,0.289,0.382)
A5	(0.244,0.323,0.354)	(0.250,0.336,0.388)	(0.306,0.399,0.418)	(0.289,0.382,0.419)

Step 3: Design the weighted normalized fuzzy decision matrix by multiplying the normalized fuzzy criteria values with the corresponding values of crisp criteria weights (Lc1, Lc2, Lc3, Lc4) using Eq. (7), (8) and (9). Weighted normalized fuzzy decision matrix is shown in Table 18.

Table 18
Weighted normalized fuzzy decision matrix for MIS

	C1	C2	C3	C4
A1	(0.0187,0.0394,0.0599)	(0.0178,0.0374,0.0570)	(0.0121,0.0221,0.0463)	(0.0116,0.0207,0.0436)
A2	(0.0599,0.0807,0.0931)	(0.0178,0.0255,0.0451)	(0.0463,0.0707,0.0952)	(0.0207,0.0345,0.0574)
A3	(0.0434,0.0641,0.0849)	(0.0531,0.0727,0.0844)	(0.0121,0.0171,0.0415)	(0.0389,0.0618,0.0849)
A4	(0.0394,0.0599,0.0807)	(0.0413,0.0609,0.0805)	(0.0513,0.0757,0.1002)	(0.0483,0.0712,0.0940)
A5	(0.0641,0.0849,0.0931)	(0.0570,0.0766,0.0885)	(0.0804,0.1049,0.1099)	(0.0712,0.0940,0.1032)

Step 4: Among these four criteria application complexity (C1) is the only non-beneficial attribute, where lower value is preferred, whereas higher values are desired for operational profile (C2), reusability (C3) and functionality (C4). The overall rating of beneficial criteria is calculated using Eqs. (10-12) and the overall rating of non-beneficial criteria are calculated using Eqs. (13-15).The overall ratings for beneficial and non-beneficial criteria for the WQS is shown in Table 19.

Table 19
Overall ratings for beneficial and non-beneficial criteria for the MIS

	S+			S-		
	l	m	u	l	m	u
A1	0.0414	0.0802	0.1468	0.0187	0.0394	0.0599
A2	0.0847	0.1307	0.1977	0.0599	0.0807	0.0931
A3	0.1041	0.1516	0.2108	0.0434	0.0641	0.0849
A4	0.1408	0.2077	0.2747	0.0394	0.0599	0.0807
A5	0.2086	0.2756	0.3015	0.0641	0.0849	0.0931

Step 5: Derive the values of overall performance index of all the components of MIS by using Eq. (16) to defuzzify the overall ratings for both beneficial and non-beneficial criteria as illustrated in Table 20.

Table 20
Overall performance index values and the rank assignment of all the components for the MIS

	S		Rank
A1	S1	0.0569	5
A2	S2	0.0684	4
A3	S3	0.0952	3
A4	S4	0.1525	2
A5	S5	0.1832	1

Step 6: The ranking of MIS components is shown in Table 20, therefore alternative A5 (Network) will contribute the most and alternative A1 (AE Client) will contribute the least to the reliability estimation of overall application of MIS. Components ranks are as follows (starting with the best alternative): A5 $\succ$ A4 $\succ$ A3 $\succ$ A2 $\succ$ A1. These results clearly suggest the usefulness, accuracy, feasibility and scalability of fuzzy-MOORA method in ranking the components of CBSS for reliability estimation.

6. Conclusions

Multi-criteria decision problems include both qualitative and quantitative factors. Generally, these factors cannot be quantified easily. This is because of vagueness in information, fuzziness in human perception, unavailability of reliable and complete information. Fuzzy-MOORA plays a vital role in such environment. The proposed approach is very simple and comprehensible which may handle large number of selection criteria. It gives the ranking of components as well as the usage frequency due to resulted performance index based on chosen criteria as suggested by subject matter experts (SMEs) and this ranking will help to estimate the overall reliability of application. Ranking of components of any CBSS may be calculated using this study. This method is also applicable to the problems with large numbers of scenarios and objectives. However, a limitation of this approach is that it is a time consuming process for complex software application and needs an automated tool to do this job also rules are made by SMEs which may take time and may be of diverse opinion. In this study, we have selected four most critical criteria but there may be some more good criteria according to other SMEs, also a criterion depends upon the application itself.

References

Brauers, W. K. M., & Zavadskas, E. K. (2006). The MOORA method and its application to privatization n a transition economy. *Control and Cybernetics,35*(2), 445.

Brauers, W. K. M., Zavadskas, E. K., Turskis, Z., & Vilutiene, T. (2008). Multi-objective contractor's ranking by applying the Moora method. *Journal of Business Economics and Management, 9*(4), 245-255.

Brauers, W. K. M., & Ginevičius, R. (2009). Robustness in regional development studies. The case of Lithuania. *Journal of Business Economics and Management, 10*(2), 121-140.

Brauers, W. K. M., & Zavadskas, E. K. (2010). Project management by MULTIMOORA as an instrument for transition economies. *Technological and Economic Development of Economy, 1,* 5-24.

Chakraborty, S. (2011). Applications of the MOORA method for decision making in manufacturing environment. *The International Journal of Advanced Manufacturing Technology, 54*(9-12), 1155-1166.

Cheung, R. C. (1980). A user-oriented software reliability model. *Software Engineering, IEEE Transactions on, 2,* 118-125.

Chu, Y., & Xu, S. (2007). Exploration of complexity in software reliability. *Tsinghua Science & Technology, 12*, 266-269.

Dimov, A., & Punnekkat, S. (2010, September). *Fuzzy reliability model for component-based software systems*. In Software Engineering and Advanced Applications (SEAA), 2010 36th EUROMICRO Conference on (pp. 39-46). IEEE.

EEE, I. (1990). *Standard Glossary of Software Engineering Terminology*. IEEE Software Engineering Standards & collection. IEEE, 610-12.

Feng, C. M., & Wang, R. T. (2000). Performance evaluation for airlines including the consideration of financial ratios. *Journal of Air Transport Management, 6*(3), 133-142.

Fiondella, L., Rajasekaran, S., & Gokhale, S. S. (2013). Efficient software reliability analysis with correlated component failures. *IEEE Transactions on Reliability, 62*(1), 244-255.

Gokhale, S. S., Wong, W. E., Trivedi, K. S., & Horgan, J. R. (1998, September). *An analytical approach to architecture-based software reliability prediction*. In Computer Performance and Dependability Symposium, 1998. IPDS'98. Proceedings. IEEE International (pp. 13-22). IEEE.

Gokhale, S. S. (2007). Architecture-based software reliability analysis: Overview and limitations. *IEEE Transactions on Dependable and Secure Computing, 1*), 32-40.

Goševa-Popstojanova, K., & Trivedi, K. (1999). *Failure correlation in software reliability models. In Software Reliability Engineering, 1999. Proceedings*. 10th International Symposium on (pp. 232-241). IEEE.

Goševa-Popstojanova, K., & Trivedi, K. S. (2001). Architecture-based approach to reliability assessment of software systems. *Performance Evaluation, 45*(2), 179-204.

Goswami, V., & Acharya, Y. B. (2009). *Method for Reliability Estimation of COTS Components based Software Systems International*. In Symposium on Software Reliability Engineering (ISSRE 2009).

Hsu, C. J., & Huang, C. Y. (2011). An adaptive reliability analysis using path testing for complex component-based software systems. *IEEE Transactions on Reliability, 60*(1), 158-170.

Hu, H., Jiang, C. H., Cai, K. Y., Wong, W. E., & Mathur, A. P. (2013). Enhancing software reliability estimates using modified adaptive testing. *Information and Software Technology, 55*(2), 288-300.

Huang, N., Wang, D., & Jia, X. (2008, November). *An algebra-based reliability prediction approach for composite web services*. In Software Reliability Engineering, 2008. ISSRE 2008. 19th International Symposium on (pp. 285-286). IEEE.

Huiqun, H., & Guang, S. (2012). ERP software selection using the rough set and TPOSIS methods under fuzzy environment. *Advances in Information Sciences & Service Sciences, 4*(3), 111-118.

Kaiyuan, C., Chenggang, B., & Xiaojun, Z. (2003). Introduction to reliability models of component based software system. *Journal of Xian Jiaotong University*, 37(6), 551-554.

Karande, P., & Chakraborty, S. (2012). A Fuzzy-MOORA approach for ERP system selection. *Decision Science Letters*, 1(1), 11-21.

Koziolek, H., & Becker, S. (2005, July). *Transforming operational profiles of software components for quality of service predictions*. In Proc. of Workshop on Component Oriented Programming.

Kracka, M., Brauers, W. K. M., & Zavadskas, E. K. (2015). Ranking heating losses in a building by applying the MULTIMOORA. *Engineering Economics,21*(4), 352-259.

Krishnamurthy, S., & Mathur, A. P. (1997, November). On the estimation of reliability of a software system using reliabilities of its components. In *Software Reliability Engineering, 1997. Proceedings., The Eighth International Symposium on* (pp. 146-155). IEEE.

Littlewood, B. (1979). Software reliability model for modular program structure. *IEEE Transactions on Reliability, 28*(3), 241-246.

Lo, J. H. (2010, June). Early software reliability prediction based on support vector machines with genetic algorithms. In *Industrial Electronics and Applications (ICIEA), 2010 the 5th IEEE Conference on* (pp. 2221-2226). IEEE.

Seth, K., Sharma, A., & Seth, A. (2010). Minimum spanning tree-based approach for reliability estimation of COTS-based software applications. *The IUP Journal of Computer Sciences*, *4*(4), 13-21.

Shooman, M. L. (1976, October). *Structural models for software reliability prediction. In Proceedings of the 2nd international conference on Software engineering (pp. 268-280)*. IEEE Computer Society Press.

Si, Y., Yang, X., Wang, X., Huang, C., & Kavs, A. J. (2010, April). An architecture-based reliability estimation framework through component composition mechanisms. In *Computer Engineering and Technology (ICCET), 2010 2nd International Conference on* (Vol. 2, pp. V2-165). IEEE.

Singh, H., Cortellessa, V., Cukic, B., Gunel, E., & Bharadwaj, V. (2001). *A bayesian approach to reliability prediction and assessment of component based systems*. In Software Reliability Engineering, 2001. ISSRE 2001. Proceedings. 12th International Symposium on (pp. 12-21).

Stanujkic, D., Magdalinovic, N., Stojanovic, S., & Jovanovic, R. (2012). Extension of ratio system part of MOORA method for solving decision-making problems with interval data. *Informatica, 23*(1), 141-154.

Tyagi, K., & Sharma, A. (2012). A rule-based approach for estimating the reliability of component based systems. *Advances in Engineering Software, 54*, 24-29.

Tyagi, K., & Sharma, A. (2014). Ranking of components for reliability estimation of CBSS using fuzzy TOPSIS. *International Journal of System Assurance Engineering and Management*, 1-9.

Wang, D., & Huang, N. (2008, October). *Reliability analysis of component-based software based on rewrite logic. In Future Trends of Distributed Computing Systems, 2008*. FTDCS'08. 12th IEEE International Workshop on(pp. 126-132). IEEE.

Wang, W. L., Pan, D., & Chen, M. H. (2006). Architecture-based software reliability modeling. *Journal of Systems and Software, 79*(1), 132-146.

Wang, Y. J., Lee, H. S., & Lin, K. (2003). Fuzzy TOPSIS for multi-criteria decision making. International Mathematical Journal, *3*, 367-379.

Wang, Y. J., & Lee, H. S. (2007). Generalizing TOPSIS for fuzzy multiple-criteria group decision making. *Computers & Mathematics with Applications*, *53*(11), 1762-1772.

Yacoub, S., Cukic, B., & Ammar, H. H. (2004). A scenario-based reliability analysis approach for component-based software. *IEEE Transactions on Reliability, 53*(4), 465-480.

Zadeh, L. A. (1965). Fuzzy sets. *Information and control, 8*(3), 338-353.

Zhang, F., Zhou, X., Dong, Y., & Chen, J. (2009, May). *Consider of fault propagation in architecture-based software reliability analysis*. In Computer Systems and Applications, 2009. AICCSA 2009. IEEE/ACS International Conference on (pp. 783-786). IEEE.