View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Universiti Teknologi Malaysia Institutional Repository

Jurnal Teknologi

Full Paper

USING THE FUZZY MULTI-CRITERIA DECISION MAKING APPROACH FOR SOFTWARE REQUIREMENTS PRIORITIZATION

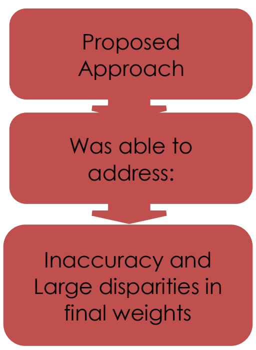
Article history
Received
15 April 2015
Received in revised form
29 September 2015
Accepted
12 November 2015

Philip Achimugu, Ali Selamat*, Roliana Ibrahim

Soft Engineering Research Group (SERG), Faculty of Computing, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor Malaysia

*Corresponding author
aselamat@utm.my

Graphical abstract



Abstract

To avoid breach of agreement or contract in software development projects, stakeholders converge to prioritize specified requirements. This is due to the fact that, not all the specified requirements can be implemented in a single release. Therefore, prioritization is the act of rating requirements according to their relative importance by project stakeholders in order to plan for software release phases. The problem of existing prioritization techniques includes computational complexities, ranking inaccuracy and large disparities between final ranks among others. Consequently, this paper presents an improved approach for prioritizing requirements for software projects requirements with stakeholders based on the limitations of existing prioritization techniques using fuzzy multi-criteria decision-making (FMCDM) approach.

Keywords: FMCDM, Fuzzy theory, Software requirements prioritization, stakeholders, software

© 2015 Penerbit UTM Press. All rights reserved

1.0 INTRODUCTION

During requirements elicitation, there are more prospective requirements specified for implementation by stakeholders with limited time and resources. Therefore, a meticulously selected set of requirements must be considered for implementation and planning for software releases with respect to available resources. This process is referred to as requirements prioritization. It is considered to be a complex multi-criteria decision making process [1].

There are many advantages of prioritizing requirements before architecture design or coding. Prioritization aids the implementation of a software system with preferential requirements of stakeholders [2, 3]. Also, the challenges associated with software development such as limited resources, inadequate budget, insufficient skilled programmers among others makes requirements prioritization really important [4]. It can help in planning for software releases since not all the elicited requirements can be implemented in a single release [5, 6]. It also enhances budget control and scheduling [1]. Therefore, determining which, among a pool of requirements to be implemented first and the order of implementation is necessary to avoid breach of contract or agreement during software development. Furthermore, software products that are developed based on prioritized requirements can be expected to have a lower probability of being rejected. To prioritize requirements, stakeholders will have to compare them in order to determine their relative importance through a weighting or scoring scale which is eventually used to compute the ranks [7]. These comparisons becomes complex with increase in the number of requirements [8].

Software system's acceptability level is mostly determined by how well the developed system has met or satisfied the specified requirements. Hence, eliciting and prioritizing the appropriate requirements and scheduling right releases with the correct functionalities are a critical success factor for building formidable software systems. In other words, when vague or imprecise requirements are implemented, the resulting system will fall short of users' or stakeholders' expectations. Many software development projects have enormous prospective requirements that may be practically impossible to deliver within the expected time frame and budget [1, 9]. It therefore becomes highly necessary to source for appropriate measures for planning and rating requirements in an efficient way.

Many techniques have been proposed in the literature by authors and scholars, yet many areas of improvement have also been identified to optimize the prioritization processes. With the advent of Internet and quest for software that can service distributed organizations, the number of stakeholders in large-scale projects have drastically increased with requirements possessing the attributes of being changed due to innovation, technological advancement or business growth. However, whatever prioritization technique is been proposed, its essence is to generate an ordered list of requirements based on the relative weights or scores provided by the relevant project stakeholders.

The rest of the manuscripts are organized as follows: Section 2 discusses the existing requirement prioritization techniques; Section 3 describes the existing FMCDM approaches while Section 4 presented the proposed approach. The experimental execution and dataset used to validate the proposed approach is described in Section 5 while the results are discussed in Section 6. The conclusion and future work are enumerated in Section 7.

2.0 RELATED WORK

Many requirements prioritization techniques exist in the literature. All of these techniques utilize a ranking process to prioritize candidate requirements. The ranking process is usually executed by assigning weights across requirements based on pre-defined criteria, such as value of the requirement perceived by relevant stakeholders or the cost of implementing each requirement. From the literature; analytic hierarchy process (AHP) is the most prominently used technique. However, this technique suffers bad scalability. This is due to the fact that, AHP executes ranking by considering the criteria that are defined through an assessment of the relative priorities between pairs of requirements. This becomes impracticable as the number of requirements increases. It also does not support requirements evolution or rank reversals but provide efficient or reliable results [10, 11]. Also, most techniques suffer from rank reversals problems. This term refers to the inability of a technique to update rank status of ordered requirements whenever a requirement is added or deleted from the list. Prominent techniques that suffer from this limitation are case base ranking [1]; interactive genetic algorithm prioritization technique [9]; Binary search tree [10]; cost value approach [6] and EVOLVE [12]. Furthermore, existing techniques are prone to computational errors [13] probably due to lack of robust algorithms. Karlsson *et al*. [10]; conducted some researches where certain prioritization techniques were empirically evaluated. From their research, they reported that, most of the prioritization techniques apart from AHP and bubble sorts produce unreliable or misleading results while AHP and bubble sorts were also time consuming. The authors then posited that; techniques like hierarchy AHP, spanning tree, binary search tree, priority groups produce unreliable results and are difficult to implement. Babar *et al*. [11] were also of the opinion that, techniques like requirement triage, value intelligent prioritization and fuzzy logic based techniques are also error prone due to their reliance on experts and are time consuming too. Planning game has a better variance of numerical computation but suffer from rank reversals problem. Wieger's method and requirement triage are relatively acceptable and adoptable by practitioners but these techniques do not support rank updates in the event of requirements evolution as well. The value of a requirement is expressed as its relative importance with respect to the other requirements in the set.

3.0 FMCDM CONCEPT

Multi-criteria decision making (MCDM) is the process of selecting or ranking prime alternatives from a pool of finite set through weighting or scoring system or scale. This usually involves decision makers whose aims are to rank given alternatives based on some pre-defined criteria or attributes. There are various techniques used for solving multiple criteria decision making problems. These include multiplicative exponential weighting (MEW), simple additive weighting (SAW), technique for ordering preference by similarity to ideal solution (TOPSIS) and analytic hierarchy process (AHP) among others. It is impractical to allot a crisp value for a subjective judgment, particularly when the information is ambiguous or imprecise. Therefore, this study attempts to explore the theories of fuzzy concept to address uncertainties associated with real world data. Zadeh [12, 13] initially invented a practical tool known as ''fuzzy sets theory" to model subjective decision making processes. Subsequently, Bellman and Zadeh, [14] extended these decision making processes into fuzzy environments where uncertain and vague problems or data were dealt with, by utilizing fuzzy sets theory. Fuzzy multiple attribute decision making (FMADM) analysis has been widely used to solve problems associated with many attributes or criteria in ambiguous situations. FMADM basically consist of two major phases [15]. The first phase deals with the aggregation of the performance scores of each alternative with respect to the defined criteria or attributes while the second phase deals with the ranking of the alternatives with respect to the synthetic or utility values obtained from the decision makers from the first phase. Xu and Chen [16] presented in their work, a fuzzy multiple attribute group decision making technique for determining the type of air conditioning systems that should be installed in a library. Similarly, Cheng *et al*. [17] weighed access strategy for future broadband service using fuzzy MCDM. In a related work, Li [18] presented a fuzzy multi-attribute decision making approach to address subjective judgments and objective information under an ambiguous situation. Also, Weng [19] used fuzzy multi-criteria decision making approach to assess financial performances of local airlines in Taiwan. Then, Chou [20] employed fuzzy MCDM technique to address the quality of marine transhipment container port problems. Furthermore, fuzzy measures and integrals were used to evaluate strategies in games in the study executed by Narukawa and Torra [21]. The evaluation of IT/IS investments has been documented in [22] where, fuzzy multi-criteria decision model was used. Meanwhile, Ding and Liang [23] used the MCDM concept to choose partners of strategic alliances for liner shipping in a fuzzy environment. In Lin *et al*. [24], the authors used the FMCDM concept to determine the planning process for HIV/AIDS treatment within a sample population. Jiang *et al*. [25] developed a method known as "fuzzy multi-granularity linguistic assessment information" for making decisions in a fuzzy environment while Royes and Bastos [26] used FMCDM to measure the ambiguities inherent in a typical political voting exercises. Chang and Wang [27] employed FMCDM to measure the possibility of successful knowledge management. Finally, in Chang *et al*. [28]; the authors proposed a technique for selecting supply chain partners at various phases of the product life cycle using fuzzy linguistic quantifier. With reference to all these literature, FMCDM was mostly utilized in either selecting, evaluating or ranking alternatives using pre-defined criteria or attributes in order to predict or forecast a situation. This also serves as motivation for this study. FMCDM is used to prioritize software requirements during the elicitation phase of the system development life cycle (SDLC) phases.

3.1 Concepts of Fuzzy Set Theories

A fuzzy number consist of precise fuzzy set $F = \{(x, \mu_F(x)), x \in R\}$ where $\mu_F(x)$ is a continuous mapping from the real line *R* to the closed interval [0, 1]. Meanwhile, a triangular fuzzy number denoted as $\tilde{M} = (a,b,c)$ or $\left(\frac{a}{b}, \frac{b}{c}\right)$, where $a \le b \le c$ has the following triangular-type membership functions (Eq. 1):

$$\mu_M(x) = \begin{cases} \frac{x-a}{b-a}, \text{if } a \le x \le b \\ \frac{c-x}{c-b}, \text{if } b \le x \le c \\ 0, \quad \text{otherwise} \end{cases} \quad (1)$$

Where *b* is the maximum value of the fuzzy set $\mu(x)$; a, and c are the lower and upper bounds, which contains all the linguistic variables in the fuzzy set. The triangular fuzzy numbers (TFNs) are depicted in Table 1.

The TFNs are used to score requirements in order to enhance relative comparisons of all the specified requirements; after which, a fuzzy judgment matrix is constructed in line with Equation 2.

$$A = \begin{Bmatrix} 1 & a_{12} & \dots & \dots & a_{1n} \\ a_{21} & 1 & \dots & \dots & a_{2n} \\ . & . & . & . & . \\ . & . & . & . & . \\ a_{n1} & a_{n2} & \dots & \dots & 1 \end{Bmatrix} \quad (2)$$

Table 1 Linguistic terms for importance weights of requirements

Terms	Numeric	Triangular fuzzy numbers
Extremely high (EH)	5	(0.9,1.0,1.0)
Very high (VH)	4	(0.7, 0.9, 1.0)
High (H)	3	(0.5, 0.7, 0.9)
Fair (F)	2	(0.3, 0.5, 0.7)
Low (L)	1	(0.1, 0.3, 0.5)

According to Zadeh [12, 13], the algebraic operations of triangular fuzzy numbers are executed as follows (addition, subtraction, multiplication and division respectively):

$$(a_1,b_1,c_1) \oplus (a_2,b_2,c_2) = (a_1+a_2, b_1+b_2, c_1+c_2) \quad (3)$$
$$(a_1,b_1,c_1) - (a_2,b_2,c_2) = (a_1-a_2, b_1-b_2, c_1-c_2) \quad (4)$$
$$(a_1,b_1,c_1) \otimes (a_2,b_2,c_2) = (a_1a_2, b_1b_2, c_1c_2) \quad (5)$$
$$(a_1,b_1,c_1) \div (a_2,b_2,c_2) = (a_1 \div a_2, b_1 \div b_2, c_1 \div c_2) \quad (6)$$

The membership function describes the level of membership of the elements *x* to the fundamental set *X*. Therefore, any element tending towards 0 connotes that the member is not included in the given set while the ones tending towards 1 connotes a fully included member. Values strictly between 0 and 1 characterize the fuzzy members.

4.0 PROPOSED TECHNIQUE

In this section, we summarize the process of identifying prime requirements by project stakeholders using simple linguistic variables parameterized with triangular fuzzy numbers (TFNs) to reflect the preference weights of requirements. Therefore, the processes involved in determining prime requirements are outlined below:

a. *Generating requirements: The elicitor or architect articulates the description of the project's problem to the stakeholders both in written and verbal form. They now lead the stakeholders to express* *their thoughts in brief phrases or statements. Each person quietly documents requirements.*

b. *Recording requirements: Stakeholders engage in a round-robin feedback meeting to precisely elicit requirements (without deliberations at this point). The architect or elicitor then collates these requirements from all the stakeholders.*

c. *Discussing requirements: The documented requirements are then deliberated upon to determine clarity and relevance. For each requirement, the architect or elicitor asks for comments, questions or constructive criticism. This will allow stakeholders to express themselves and have a thorough understanding of the requirements to undergo prioritization.*

d. *Rating requirements: These requirements are parameterized as R_i, where i = 1,2,3,...,n (total number of requirements) and rating confidence C_j Where j = 1,2,3,4,5.*

The ranking takes place as follows:

1. **Step 1:** Obtains a numeric weights of preference for the requirements.
2. **Step 2:** The numeric weights are then converted to their TFNs equivalence and a decision matrix is constructed (Table 1).
3. **Step 3:** The TFNs are summed using Equation 3 (Local weights).
4. **Step 4:** The sum is divided by the number of stakeholders (Global weights)
5. **Step 5:** The square root of the output in Step 4 is determined to reflect the final ranks.

5.0 EXPERIMENTAL SET UP

5.1 Dataset

The Greer's dataset was used to evaluate the performance of the proposed approach. The dataset contains 20 requirements with preference weights from 5 stakeholders. Greer's dataset [29] is shown in Table 2. The first half of the table contains the numeric weights of Greer's dataset and the second half of the table shows their TFNs equivalence. The results of Steps 2-5 are displayed in Tables 3. It reflects the local, global and final weights. Weights for these requirements were obtained from 5 stakeholders.

6.0 RESULTS AND DISCUSSION

The experiments were carried out on a computer with a 2.4 GHz processor and 4 GB RAM. It was observed that the proposed approach was able to accurately prioritize requirements based on the preferential weights of stakeholders. Subsequently, In the Greer and Ruhe's dataset, there is a ground truth of requirements that contains actual preferences of the project requirements. To calculate the accuracy of the proposed approach, aggregated weights of the ground truth requirements were compared to the final weight generated by the proposed approach. Figure 1 shows the final results generated by the proposed approach while Figure 2 shows the results for the aggregated ground truth weights of the dataset. A close look at the figures suggest that the final results generated by the proposed approach completely tallied with the aggregated ground truth weights of requirements in the dataset. This means that the proposed approach is accurate, hence completely reliable. In terms of the discrepancies in the final weights of requirements, the proposed approach provided exciting attribute. The entire requirements' final ranks had an average weight rank of 1.4 which shows good aggregation of the computed weights across the stakeholders.

7.0 CONCLUSION AND FUTURE WORK

The aim of this research was to identify the limitations of existing prioritization techniques with the aim of improving them. It was eventually discovered that existing techniques actually suffer from mainly inaccuracy, large disparity or disagreement between ranked weights as well as complexities. These were addressed during the course of undertaking this research. The method utilized in this research consisted of FMADM approach. Various formulae and model were formulated to enhance the viability of the proposed approach. The evaluation of the proposed approach was executed with relevant project datasets. The proposed approach have addressed important limitations of existing prioritization approaches. On the overall, the proposed approach performed better with respect to the above parameters. This will help software engineers determined the difference between the most valued and least valued requirements from relevant stakeholders which will help plan for software release planning in order to avoid breach of contract, trust or agreement during software development process. Based on the presented results, it will be appropriate to consider this research as an improvement in the field of multi-attribute decision making field. In the future, we hope to develop a parallel hybridization of FMADM and evolutionary algorithms to solve requirements prioritization problem.

Table 2 Weighted Requirements: Greer and Ruhe Dataset (1st half)/TFNs Equivalence (2nd half)

	S1	S2	S3	S4	S5
R_1	4	4	5	4	5
R_2	2	4	3	5	4
R_3	1	2	3	2	2
R_4	2	2	3	3	4
R_5	5	4	4	3	5
R_6	5	5	5	4	4
R_7	2	1	2	2	2
R_8	4	4	4	4	4
R_9	4	4	4	2	5
R_{10}	4	5	4	3	2
R_{11}	2	2	2	5	4
R_{12}	3	3	4	2	5
R_{13}	4	2	1	3	3
R_{14}	2	4	5	2	4
R_{15}	4	4	4	4	4
R_{16}	4	2	1	3	1
R_{17}	4	3	2	5	1
R_{18}	1	2	3	4	2
R_{19}	3	3	3	3	4
R_{20}	2	1	2	2	1
	S1	**S2**	**S3**	**S4**	**S5**
R1	0.7,0.9, 1.0	0.7,0.9, 1.0	0.9,1.0,1.0	0.7,0.9, 1.0	0.9,1.0,1.0
R2	0.3, 0.5, 0.7	0.7,0.9, 1.0	0.5, 0.7, 0.9	0.9,1.0,1.0	0.7,0.9, 1.0
R3	0.1, 0.3, 0.5	0.3, 0.5, 0.7	0.5, 0.7, 0.9	0.3, 0.5, 0.7	0.3, 0.5, 0.7
R4	0.3, 0.5, 0.7	0.3, 0.5, 0.7	0.5, 0.7, 0.9	0.5, 0.7, 0.9	0.7,0.9, 1.0
R5	0.9,1.0,1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.5, 0.7, 0.9	0.9,1.0,1.0
R6	0.9,1.0,1.0	0.9,1.0,1.0	0.9,1.0,1.0	0.7,0.9, 1.0	0.7,0.9, 1.0
R7	0.3, 0.5, 0.7	0.1, 0.3, 0.5	0.3, 0.5, 0.7	0.3, 0.5, 0.7	0.3, 0.5, 0.7
R8	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0
R9	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.3, 0.5, 0.7	0.9,1.0,1.0
R10	0.7,0.9, 1.0	0.9,1.0,1.0	0.7,0.9, 1.0	0.5, 0.7, 0.9	0.3, 0.5, 0.7
R11	0.3, 0.5, 0.7	0.3, 0.5, 0.7	0.3, 0.5, 0.7	0.9,1.0,1.0	0.7,0.9, 1.0
R12	0.5, 0.7, 0.9	0.5, 0.7, 0.9	0.7,0.9, 1.0	0.3, 0.5, 0.7	0.9,1.0,1.0
R13	0.7,0.9, 1.0	0.3, 0.5, 0.7	0.1, 0.3, 0.5	0.5, 0.7, 0.9	0.5, 0.7, 0.9
R14	0.3, 0.5, 0.7	0.7,0.9, 1.0	0.9,1.0,1.0	0.3, 0.5, 0.7	0.7,0.9, 1.0
R15	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0	0.7,0.9, 1.0
R16	0.7,0.9, 1.0	0.3, 0.5, 0.7	0.1, 0.3, 0.5	0.5, 0.7, 0.9	0.1, 0.3, 0.5
R17	0.7,0.9, 1.0	0.5, 0.7, 0.9	0.3, 0.5, 0.7	0.9,1.0,1.0	0.1, 0.3, 0.5
R18	0.1, 0.3, 0.5	0.3, 0.5, 0.7	0.5, 0.7, 0.9	0.7,0.9, 1.0	0.3, 0.5, 0.7
R19	0.5, 0.7, 0.9	0.5, 0.7, 0.9	0.5, 0.7, 0.9	0.5, 0.7, 0.9	0.7,0.9, 1.0
R20	0.3, 0.5, 0.7	0.1, 0.3, 0.5	0.3, 0.5, 0.7	0.3, 0.5, 0.7	0.1, 0.3, 0.5

Table 3 Local, Global and Final Ranks of the Greer and Ruhe Dataset

	S1	S2	S3	S4	S5	Local	Global	Final
R1	2.6	2.6	2.9	2.9	2.9	13.6	2.72	1.65
R2	1.5	2.6	2.1	2.9	2.6	11.7	2.34	1.53
R3	0.9	1.5	2.1	1.5	1.5	7.5	1.50	1.22
R4	1.5	1.5	2.1	2.1	2.6	9.8	1.96	1.40
R5	2.9	2.6	2.6	2.1	2.9	13.1	2.62	1.62
R6	2.9	2.9	2.9	2.6	2.6	13.9	2.78	1.67
R7	1.5	0.9	1.5	1.5	1.5	6.9	1.38	1.17
R8	2.6	2.6	2.6	2.6	2.6	13	2.60	1.61
R9	2.6	2.6	2.6	1.5	2.9	12.2	2.44	1.56
R10	2.6	2.9	2.6	2.1	1.5	11.7	2.34	1.53
R11	1.5	1.5	1.5	2.9	2.6	10	2.00	1.41
R12	2.1	2.1	2.6	1.5	2.9	11.2	2.24	1.50
R13	2.6	1.5	0.9	2.1	2.1	9.2	1.84	1.36
R14	1.5	2.6	2.9	1.5	2.6	11.1	2.22	1.49
R15	2.6	2.6	2.6	2.6	2.6	13	2.60	1.61
R16	2.6	1.5	0.9	2.1	0.9	8	1.60	1.26
R17	2.6	2.1	1.5	2.9	0.9	10	5.00	1.41
R18	0.9	1.5	2.1	2.6	1.5	8.6	1.72	1.31
R19	2.1	2.1	2.1	2.1	2.6	11	2.20	1.48
R20	1.5	0.9	1.5	1.5	0.9	6.3	1.26	1.12

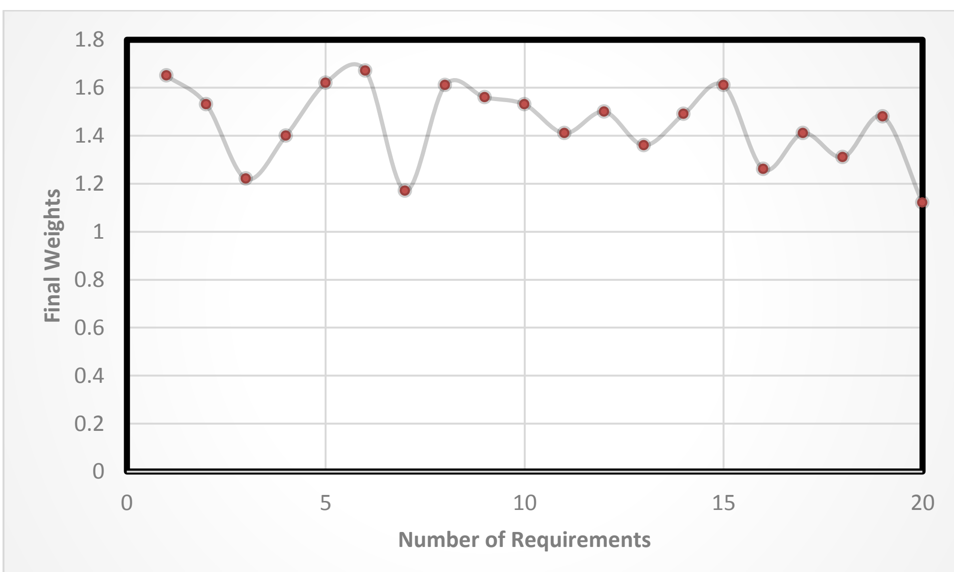


Figure 1 Proposed Approach: Final Weights

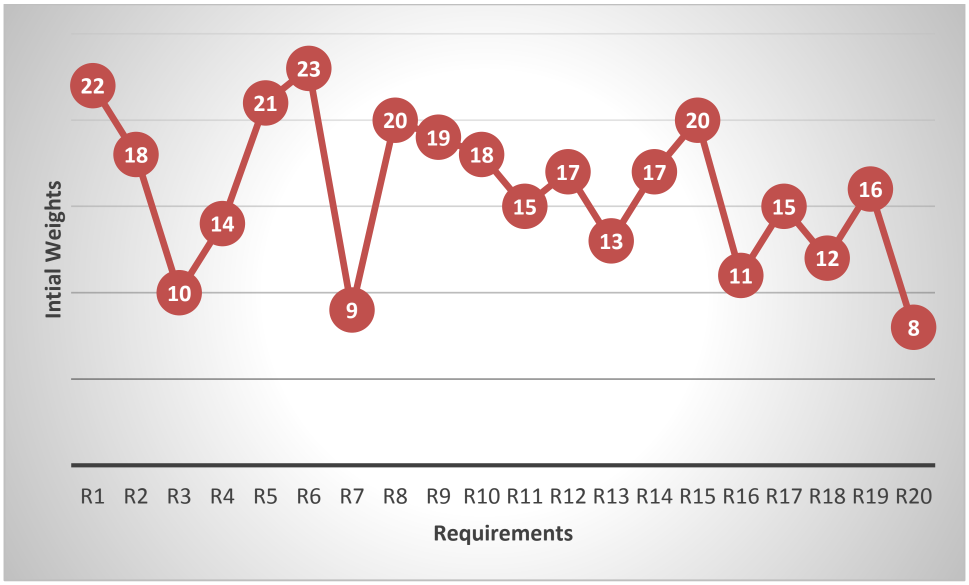


Figure 2 Preference Weights: Initial Weights

Acknowledgement

The authors appreciate the efforts of the Ministry of Science, Technology and Innovation Malaysia (MOSTI) under Vot 4S062 and Universiti Teknologi Malaysia (UTM) for supporting this research.

References

[1] Perini, A. Susi, A. and Avesani, P. 2013. A Machine Learning Approach to Software Requirements Prioritization. *IEEE Transactions on Software Engineering*. 4(39): 445-460.

[2] Ahl, V. 2005. An Experimental Comparison of Five Prioritization Methods-investigating Ease of Use, Accuracy and Scalability, Master's Thesis, School of Engineering, Blekinge Institute of Technology, Sweden. [Online]. From: http://www.bth.se/fou/cuppsats.nsf/all/86a759a57c33591 1c1257088005e42bc?OpenDocument. [Accessed on Feb. 20, 2015].

[3] Thakurta, R. 2012. A Framework for Prioritization of Quality Requirements for Inclusion in a Software Project. Software *Quality Journal*. 1(21): 573-597.

[4] Karlsson, L. Thelin, T. Regnell, B. Berander, P. and Wohlin, C. 2007. Pair-wise Comparisons versus Planning Game Partitioning-experiments on Requirements Prioritization Techniques. *Empirical Software Engineering*. 12(1): 3-33.

[5] Berander, P. K.A. Khan, and L. Lehtola. 2006. Towards a Research Framework on Requirements Prioritization. *6th International Conference on Software Engineering Research and Practice in Sweden. Umeå. October 18-19. 2006*. 39-48.

[6] Karlsson, J. and Ryan, K. 1999. A Cost-Value Approach for Prioritizing Requirements. *IEEE Software*. 14: 67-74.

[7] Kobayashi, M. and M. Maekawa. 2001. Need-based Requirements Change Management. *8th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems*. Washington, DC, USA. April 17 2001. 171-178.

[8] Kassel, N. W. and B. A. Malloy. 2003. An Approach to Automate Requirements Elicitation and Specification. *7th IASTED International Conference on Software Engineering and Applications*. Marina Del Rey, CA, USA. December 1 2003. 544-549.

[9] Tonella, P. Susi, A. and Palma, F. 2012. Interactive Requirements Prioritization using a Genetic Algorithm, *Information and Software Technology*. 55: 173-187.

[10] Karlsson, J. Wohlin, C. and Regnell, B. 1998. An Evaluation of Methods for Prioritizing Software Requirements. *Information and Software Technology*. 39(14): 939-947.

[11] Babar, M., M. Ramzan, and S. Ghayyur. 2011. Challenges and Future Trends in Software Requirements Prioritization. *International conference on Computer Networks and Information Technology (ICCNIT)*. Abbottabad. July 11-13 2011. 319-324.

[12] Zadeh, L. A. 1965. Fuzzy Sets. *Information and Control*. 8(3): 338-353.

[13] Zadeh, L. A. 1999. Some Reflections on the Anniversary of Fuzzy Sets and Systems. *Fuzzy Sets and Systems.* 1(100): 1-3.

[14] Bellman, R. E. and Zadeh, L. A. (1970). Decision Making in a Fuzzy Environment. *Management Science*. 17(4): 141-164.

[15] Dubois, D. and Prade, H. 1980. *Fuzzy Sets and Systems.* Academic Press, New York.

[16] Xu, Z. S. and Chen, J. 2007. An Interactive Method for Fuzzy Multiple Attribute Group Decision Making, *Information Sciences*. 1(177): 248-263.

[17] Cheng, J. Z., Chen, P. T. and Yu, H. C. D. 2005. Establishing a MAN Access Strategy for Future Broadband Service: A Fuzzy MCDM Analysis of SONET/SDH and Gigabit Ethernet. *Technovation*. 25(5): 557-567.

[18] Li, D. F. 2005. An Approach to Fuzzy Multi-attribute Decision Making under Uncertainty. *Information Sciences*. 1(169): 97-112.

[19] Wang, Y. J. 2008. Applying FMCDM to Evaluate Financial Performance of Domestic Airlines in Taiwan. *Expert Systems with Applications*. 1(34): 1837-1845.

[20] Chou, C. C. 2007. A Fuzzy MCDM Method for Solving Marine Transshipment Container Port Selection Problems, *Applied Mathematics and Computation*. 1(186): 435-444.

[21] Narukawa, Y. and Torra, V. 2007. Fuzzy Measures and Integrals in Evaluation of Strategies. *Information Sciences*. 1(177): 4686-4695.

[22] Chou, T. Y. and Chou, S. T. Tzeng, G. H. 2006. Evaluating IT/IS Investments: A Fuzzy Multi-criteria Decision Model Approach. *European Journal of Operational Research*. 1(173): 1026-1046.

[23] Ding, J. F. and Liang, G. S. 2005. Using Fuzzy MCDM to Select Partners of Strategic Alliances for Liner Shipping. *Information Sciences*. 1(173): 197-225.

[24] Lin, F. Ying, H. MacArthur, R. D. Cohn, J. A. Barth-Jones, D. and Crane, L. R. 2007. Decision Making in Fuzzy Discrete Event Systems. *Information Sciences*. 1(177): 3749-3763.

[25] Royes, G. F. and Bastos, R. C. 2006. Uncertainty Analysis in Political Forecasting. *Decision Support Systems*. 42(1): 25-35.

[26] Chang, T. H. and Wang, T. C. 2009. Using the Fuzzy Multi-criteria Decision Making Approach for Measuring the Possibility of Successful Knowledge Management. *Information Sciences*. 179(4): 355-370.

[27] Chang, S. L. and Wang, R. C. and Wang, S. Y. 2006. Applying Fuzzy Linguistic Quantifier to Select Supply Chain Partners at Different Phases of Product Life Cycle. *International Journal of Production Economics.* 1(100): 348-359.

[28] Chen, Y. W. and Tzeng, G. H. 2001. Using Fuzzy Integral for Evaluating Subjectively Perceived Travel Costs in a Traffic Assignment Model. *European Journal of Operational Research*. 130(3): 653-664.

[29] Greer, D. and Ruhe, G. 2004. Software Release Planning: An Evolutionary and Iterative Approach. *Information and Software Technology*. 46(4): 243-253.