A Multi-attribute Decision Making Approach for Resource Allocation in Software Projects

A. Ejnioui¹, C. E. Otero¹, and L. D. Otero²
¹Information Technology, University of South Florida Lakeland, Lakeland, Florida, USA
²Engineering Systems, Florida Institute of Technology, Melbourne, Florida, USA

Abstract—For most software companies, the production of reliable software within the planned time schedule is of paramount importance. Many times, inadequate resource allocation can lead to high costs and low quality in software products. Hence, it is critical to put in place well-planned processes for personnel assignment that take into consideration the skill sets of the personnel with the objective of reducing costs and training time as well as increasing product quality. To this end, this paper presents a novel methodology that considers multiple project-specific skills for assigning human resources to software projects. This methodology takes into account the existing capabilities of personnel to determine the best fit based on the required skills for the task. Because personnel selection is essentially an imprecise task, this methodology uses fuzzy sets to represent the problem of personnel assignment as a fuzzy multi-attribute decision problem. This problem is solved by ranking personnel candidates based on the expected value operator of fuzzy numbers. A sample case study is used to show the methodology’s capabilities.

Keywords: Expected Value Operator, Fuzzy Sets, Software Engineering, Resource Allocation, Multi-Attribute Decision Problem.

1 Introduction

Task assignment decisions in software development environments are critical because they influence the performance of workers and quality of products [1]. As documented in [2], the U.S. Department of Defense (DOD) spent nearly 8 billion dollars in 2004 to rework software. This large financial figure serves as evidence that quality-related issues continue to be a major struggle for software companies. Furthermore, “evidence reveals that the failure of software development projects is often a result of inadequate human resource project planning” [3]. In [4], Linberg stated that only about 16.2% of software projects are on time and within budget. A major contributor to this problem is the inefficient allocation of resources that may result in schedule overruns, decreased customer satisfaction, decreased employee morale, reduced product quality, and negative market reputation. The inevitable consequence is a decrease in potential profit for companies.

Despite all the research and advances in the field, software development is still very challenging due to its unpredictable nature. The fast pace at which new technologies and techniques are being developed today to improve the design and development of products increases the demand for specialized individual skills in the workforce. Most of the time, candidates with the required skills to work on specific tasks are not available, and decision makers are forced to assign resources to tasks based on subjective measures [1]. Therefore, further studies of the processes and techniques for personnel management are necessary to provide better solutions in terms of quality, cost, and schedule. Often, assigning resources is not certain and can be very fuzzy.

This paper proposes a fuzzy multi-attribute decision making approach for allocating human resources to task assignment in software engineering. The approach uses a fuzzy function ranking to provide a unified metric representative of the suitability between the complete set of skills available from candidates and skills required for tasks [5]. As such, decision makers can quantitatively assign resources to tasks even when the most desirable skills are not available from the existing workforce. The approach is extensible to consider a wide variety of project specific capabilities, such as years of experience, level of perceived expertise on a particular language, operating system, domain knowledge, etc. Moreover, managers can use this methodology as a tool to increase the efficiency of resource allocation.

This paper is divided into seven sections. Section 2 describes the literature related to resource allocation in software projects. Section 3 briefly describes the proposed methodology for personnel assignment. Section 4 presents a detailed coverage of ranking of fuzzy variables using the expected value operator. Section 5 introduces concepts related to how credibility spaces are used to compute the expected values of fuzzy numbers. Section 6 presents results of the approach on a small case study. Finally, section 7 concludes this paper.

2 Related Work

In recent literature regarding the assignment of software developers to tasks, Acuña et al. stated that software managers typically make assignments based on “their experience, heuristic knowledge, subjective perception, and
instinct” [1]. In [6], Duggan et al. developed a multi-objective optimization model for software task allocation based on genetic algorithms. The competencies of developers were modeled using a categorical variable with five levels. Each competency level was associated with an expected productivity per day, and an expected number of defects per unit of productivity. Other studies have developed procedures for allocating personnel to software tasks based on the assessment of behavioral competencies [1, 7].

In [3], the authors proposed selecting resources using the CRD method and the Taguchi’s parameter design approach. The CRD was used because it focused on resource scheduling rather than activity scheduling to represent human-resource workflow and tasks’ precedence. The Taguchi’s parameter design was used to obtain a scheme that would optimize the selection of engineers for tasks under dynamic and stochastic conditions. The Taguchi’s parameter design approach is based on the concept of target value [8]; that is, any deviations from the target value will result in additional costs. Deviations are attributed to controllable and uncontrollable factors. The aim is to achieve the optimal levels of the controllable factors while minimizing the variation caused by the nuisance (uncontrollable) factors [9].

In [3], the skill levels of resources were estimated as an average number of software lines of code (SLOC) per day. The authors commented on the importance of including in their model stochastic factors affecting the selection of resources. Specifically, emphasis was placed on the stochastic behavior of tasks complexities, since they are very difficult to measure or even estimate, causing most of the variability of calculated project completion times.

Other methodologies used to evaluate staffing alternatives include the Analytical Hierarchy Process (AHP) and linear programming (LP). In [10], Ho-Leung used AHP to tackle the problem of human resource substitution, considering several organizational, client, and application attributes. Even though the proposed model did not consider the relationship between known and required skills, the author commented on the importance of developing faster methods for human resource substitution. In [11], the authors proposed an LP assignment model to match resources to tasks when optimum skill sets are not available. Their model takes into account existing capabilities of candidates, required levels of expertise, and priorities of required skills for the task.

While the approaches described above use parameters that are quantified as crisp values, most often qualitative criteria cannot be precisely defined particularly when performing capability assessment in skill-based environments. Because these criteria tend to be imprecise, they have been modeled using fuzzy sets. One of the earliest attempts in this direction used fuzzy expected values to represent the capability of an employee [12]. Others addressed this problem by modeling competency levels using fuzzy variables [13-15]. In addition, fuzzy relationships have been integrated in expert systems to match employees with specific tasks [16, 17].

From the reviewed literature, it is evident that there is much room for improved personnel assignment methodologies in software projects. The literature shows that the most common measure of the ability of a software developer is an estimated value of SLOC per day. Furthermore, this estimated SLOC value is usually a function of developers’ experience level. To the best of our knowledge, a readily available methodology that considers complete set of capabilities of candidates, levels of skills required, and priorities of required skills for tasks is nonexistent in the software development literature.

3 Proposed Approach

To properly make resource allocation decisions in software engineering projects, decision-makers must follow a decision-making process that takes into consideration the fundamental efficiency metrics present in specific projects. The creation of such processes is achieved as follows. First, experienced project leaders must identify the particular skill-set required for a particular project. Then, from the pool of available candidates, each candidate is assessed using a small set of skill levels based on previous performance, educational background, or combination of both. In the proposed approach, we suggest five skill levels: None, Novice, Proficient, Advanced, and Expert. After each candidate is evaluated in all required skills, these skill levels are used as input to the proposed fuzzy multi-attribute decision problem to compute the expected value of a weighted fuzzy function.

![Figure 1. Proposed approach.](image)

Finally, the candidates with the highest expected value of this function get selected for the software project.
4 Preliminaries

This section introduces basic concepts in credibility theory that are critical in developing the approach proposed in this paper. Most of these concepts have been published in [5, 18, 19].

4.1 Fuzzy Variables and Credibility

Let $\Theta$ be a nonempty set and $P(\Theta)$ its power set. Each element in $P(\Theta)$ is called an event. The credibility of an event $A$, denoted by $\text{Cr}(A)$, is a number that represents the credibility that $A$ will occur.

Definition 1 Let $\Theta$ be a nonempty set, $P(\Theta)$ the power set of $\Theta$, and $\text{Cr}$ a credibility measure. Then the triplet $(\Theta, P(\Theta), \text{Cr})$ is called a credibility space.

Definition 2 A fuzzy variable is defined as a (measurable) function from a credibility space $(\Theta, P(\Theta), \text{Cr})$ to the set of real numbers.

Theorem 1 (Credibility Inversion) Let $\xi$ be a fuzzy variable with membership function $\mu$. Then for any set $B$ of real numbers, we have

$$\text{Cr}(\xi \in B) = \frac{1}{2} \left( \sup_{x \in B} \mu(x) + 1 - \sup_{x \in B^c} \mu(x) \right)$$

where $B^c$ is the complement set of $B$. The proof of this theorem can be found in [18, 19].

4.2 Expected Value

Although there are many ways to define an expected value operator, we choose to focus on the most general definition of this operator [5, 18-21]. This definition is applicable to both continuous and discrete fuzzy variables.

Definition 3 Let $\xi$ be a fuzzy variable and $\theta$ a real number. Then the expected value of $\xi$ is defined as

$$E[\xi] = \int_0^{+\infty} \text{Cr}(\xi \geq \theta) d\theta - \int_{-\infty}^0 \text{Cr}(\xi \leq \theta) d\theta$$

provided that at least one of the two integrals is finite.

Definition 4 Let $A = (l, m, r)$ be a triangular fuzzy number characterized by its grade membership function as:

$$\mu_A(\chi) = \begin{cases} 
1 - \frac{(m-x)}{l}, & \chi = m, \\
1 - \frac{(x-m)}{r}, & m \leq \chi \leq m + r, \\
0, & \text{otherwise}
\end{cases}$$

The values $l$, $m$, and $r$ are respectively the left, middle and right spreads of $A$.

Theorem 2 Based on Definition 3, the expected value of a triangular fuzzy number $A = (l, m, r)$ can be calculated as follows:

$$E[A] = \int_0^{+\infty} \text{Cr}(\xi \geq \theta) d\theta - \int_{-\infty}^0 \text{Cr}(\xi \leq \theta) d\theta = m + \frac{r - l}{4}$$

The proof of this theorem can be found in [22].

Theorem 3 Let $\xi$ and $\eta$ be independent fuzzy variables with finite expected values. Then for any numbers $a$ and $b$, we have

$$E[a\xi + b\eta] = aE[\xi] + bE[\eta]$$

This property is called the linearity of expected value operator of fuzzy variables. The proof of this theorem can be found in [18].

4.3 Ranking of Fuzzy Variables

Contrary to the set of real numbers, fuzzy variables do not have a natural order in a fuzzy world. As such, several approaches were devised to rank fuzzy variables [23]. One approach is based on the expected value operator of a fuzzy variable.

Definition 4 (Expected Value Criterion) Let $\xi$ and $\eta$ be fuzzy variables with finite expected values. We say $\xi \geq \eta$ if and only if $E[\xi] \geq E[\eta]$ where $E$ is the expected value operator of a fuzzy variable.

This definition can be readily applied to triangular fuzzy numbers.

5 Expected Value Method

This section formulates the problem of resource allocation in software projects as a fuzzy multi-attribute decision problem. Then, it proposes a solution to this problem based on the expected value of fuzzy numbers in the problem formulation.

5.1 Problem Formulation

In making decision to allocate resources in software projects, we assume a set of candidates $C = \{C_1, C_2, \ldots, C_m\}$ and a set of skills required to complete a project $S = \{S_1, S_2, \ldots, S_n\}$. We assume that the evaluation of each candidate with regard to each skill has been completed by a project manager resulting in a fuzzy decision matrix $A = [\xi_{ij}]_{m \times n}$ where each fuzzy number $\xi_{ij}$ represents the skill level of candidate $C_i$ in skill $S_j$. In fact, each skill can be viewed as a fuzzy variable characterized by its membership functions based on a set of linguistic concepts defining the level of expertise in each skill. We also assume a set of weights $W = \{w_1, w_2, \ldots, w_m\}$ that represents the weights of the skills in $S$. This formulation is known as the fuzzy multi-attribute decision problem (FMADM) where the skill set $S$ represents the attributes in the decision matrix $A$. 
5.2 Matrix Normalization

In most FMADM problems expressed in matrix form, normalization is necessary in order to transform the matrix and weight vector numbers to comparable values. In our case, normalization is based on the expected value operator [5]. For each fuzzy number \( \xi_{ij} \) in \( A \), transform this number as follows:

If \( \xi_{ij} \) is a benefit:

\[
\eta_{ij} = \frac{\xi_{ij}}{\sqrt{\sum_{i=1}^{m} (E[\xi_{ij}])^2}} \quad (6)
\]

for \( i = 1, 2, \ldots, n \) and \( j = 1, 2, \ldots, m \).

If \( \xi_{ij} \) is a cost:

\[
\eta_{ij} = \frac{p_j - \xi_{ij}}{\sqrt{\sum_{i=1}^{m} (E[\xi_{ij}])^2}} \quad (7)
\]

for \( i = 1, 2, \ldots, n \), \( j = 1, 2, \ldots, m \), and \( p_j = \max_{i=1} \sum_{m} \sup \{x_{ij} | \mu_{ij}(x_{ij}) > 0 \} \). Note that both expressions use in their denominators the expected values of the fuzzy variables of each column or attribute. Also, note that the \( \mu \) are the membership functions of the fuzzy variables representing the attributes. The obtained normalized matrix is \( B = [\eta_{ij}]_{m \times n} \). In addition to the decision matrix, the set \( W \) can be normalized as follows:

\[
\omega_j = \frac{w_j}{\sum_{i=1}^{m} E[w_i]} \quad (8)
\]

for \( j = 1, 2, \ldots, n \). The final normalized weight vector is \( \omega = [\omega_1, \omega_2, \ldots, \omega_n] \).

5.3 Expected Value Approach

Given a normalized matrix of fuzzy numbers and a normalized weight vector, a simple additive weighting approach can be used to compute the following \( m \) fuzzy variables as follows [5]:

\[
f_i = \sum_{j=1}^{n} \omega_j \eta_{ij} \quad (9)
\]

for \( i = 1, 2, \ldots, m \). Each fuzzy variable \( f_i \) can be viewed as the real-value function associated with each candidate. A utility value function \( E[f_i] \), \( i = 1, 2, \ldots, m \), based on the expected value operator can be devised to rank the \( m \) fuzzy variables. Assuming that the fuzzy variables have triangular membership functions, this utility \( E[f_i] \) can be computed using equation (4). In this case, the real-valued function \( f \) can be computed as follows if we assume \( \eta_{ij} = (a_{ij}, b_{ij}, c_{ij}) \) and \( \omega_j = (d_j, e_j, g_j) \):

\[
f_i = \left( \sum_{j=1}^{n} a_{ij} d_j, \sum_{j=1}^{n} b_{ij} e_j, \sum_{j=1}^{n} c_{ij} g_j \right) \quad (10)
\]

for \( i = 1, 2, \ldots, m \). In this case, the utility function of \( f_i \) can be computed as:

\[
E[f_i] = \frac{1}{4} \left( \sum_{j=1}^{n} a_{ij} d_j + 2 \sum_{j=1}^{n} b_{ij} e_j + \sum_{j=1}^{n} c_{ij} g_j \right) \quad (11)
\]

for \( i = 1, 2, \ldots, m \).

6 Case Study

This section presents results of a resource allocation case study using the proposed approach. The case study assumes a scenario where 10 candidates are available. The identified required skill set involves knowledge of the C# language, Windows platform, web programming, and knowledge of the satellite communications domain. In addition, cost is identified as a decision making unit. Skills are modeled using the fuzzy sets shown in Figure 2 while weights are modeled using the fuzzy sets shown in Figure 3. The first set shows five skill levels: None, Novice, Proficient, Advanced, and Expert. On the other hand, the second set shows five levels of importance: Not Important, Somewhat Important, Moderately Important, Important, and Very Important. Costs are modeled in five levels of severity similar to the levels used in the weights, but their membership functions are identical to the ones shown in Figure 2. Using synthetic data, the skill assessment matrix is presented in tabular form as shown in Table 1. The weights are shown in the bottom of Table 1.

![Figure 1. Fuzzy set of skill levels.](image1)

![Figure 2. Fuzzy sets of weights.](image2)
The columns l, m, and r represent the left, middle and right values of each fuzzy number representing a skill level in the table. Using equations (6)-(8), the assessment matrix is normalized as shown in Table 2. Normalization is based on the expected values of the triangular numbers in the matrix as equation (4) shows. This table is used to compute the ranking of the candidates based on equation (11). As seen from this particular scenario, candidates 8, 7, and 3 are the top three candidates that fit the required skills. These candidates display rankings of 52.83%, 50.74%, and 44.87% in terms of skills for the project at hand. In fact, candidate 8 displays expert skills in C# and Windows, as well as proficient skills in satellite communication.

7 Conclusion

The research presented in this paper develops a systematic approach for planning resource allocation in software projects based on multiple criteria. Specifically, it presents a methodology based on the expected value operator for ranking fuzzy numbers. This concept is used to rank fuzzy numbers representing skill levels for each candidate. Through a small case study, the approach is proven successful in providing a way for analyzing resource assignment for application-specific projects.

There are several important contributions from this research. First, the approach is simple and readily available for implementation using a simple spreadsheet. This can promote usage in practical scenarios, where highly complex methodologies for resource allocations are impractical due to schedule and budget constraints. Another important contribution from the approach presented in this research is the ability to consider numerous decision-making factors in the decision-making process. For example, beside the skills presented in the case study, the approach can be easily extended to incorporate project-specific factors, such as expected availability, level of clearances, etc. Finally, the results provided by this approach can be used by program managers to tailor scheduling goals to make them more realistic. Depending on the overall skill rankings of all candidates in a team, program managers can determine if either more resources need to be allocated for the task, or
scheduling requirements need to be relaxed to complete the project.

8 References


