Hindawi Publishing Corporation
Advances in Software Engineering
Volume 2012, Article ID 765635, 10 pages
doi:10.1155/2012/765635

Research Article

Applying a Goal Programming Model to Support the Selection of Artifacts in a Testing Process

Andreia Rodrigues da Silva, Fernando Rodrigues de Almeida Júnior, and Placido Rogerio Pinheiro

Graduate Program in Applied Informatics, University of Fortaleza (UNIFOR), Avenue Washington Soares, 1321, 60811-341 Fortaleza, CE, Brazil

Correspondence should be addressed to Placido Rogerio Pinheiro, placidrp@uol.com.br

Received 11 August 2012; Revised 20 October 2012; Accepted 12 November 2012

Academic Editor: Zhenyu Zhang

This paper proposes the definition of a goal programming model for the selection of artifacts to be developed during a testing process, so that the set of selected artifacts is more viable to the reality of micro and small enterprises. This model was based on the IEEE Standard 829, which establishes a set of artifacts that must be generated throughout the test activities. Several factors can influence the definition of this set of artifacts. Therefore, in order to consider such factors, we developed a multicriteria model that helps in determining the priority of artifacts according to the reality of micro and small enterprises.

1. Introduction

According to the Ministry of Science and Technology of Brazil, about 60% of software development enterprises in this country are classified as micro and small enterprises [1, 2]. In order to remain on the market, these companies need to invest significantly in improving the quality of their products because of the inherent complexity of the software development activity, which depends mainly on the interpretation skills of those involved. For that reason, this activity is susceptible to various issues, including the possibility of developing software other than what is expected by the user.

In this context, the test activity is fundamental in supporting the quality assurance of products. However, it is important to note that according to the estimates obtained in recent years, 50% of development costs are allocated to software testing [3] and, in the scenario of micro and small enterprises (MSEs), where resources availability is limited, software testing activities are reduced or, in many cases, eliminated [4], because of the lack of skilled professionals in the area, the variety of techniques existing, and the difficulty of implementing a testing process. These companies do not have the necessary capital to hire such professionals, besides not having the know-how of testing techniques and having much difficulty to deploy a testing process practical enough.

A variety of micro and small enterprises still do not have a formal testing process and even have the ability to implement a process that meets the needs and ensure the correct execution of activities. Generally testing activities, when included in the development process of software these companies are carried out by developers or system analysts. These professionals do not have knowledge about the techniques and testing criteria and, therefore, cannot benefit from the application of the techniques most appropriate to the context of the organization and the characteristics of the software being developed.

About these limitations, it is important to define an approach that is not large or unviable, to allow the utilization of a testing process in these companies. As you can see, the testing process is very expensive and time consuming, spending valuable resources in such activity [5].

Small businesses have unique and distinct characteristics: they develop software generally smaller and less complex; do not have many financial resources; avoid expensive tools, sophisticated and complex procedures; processes and methods are unique [6].

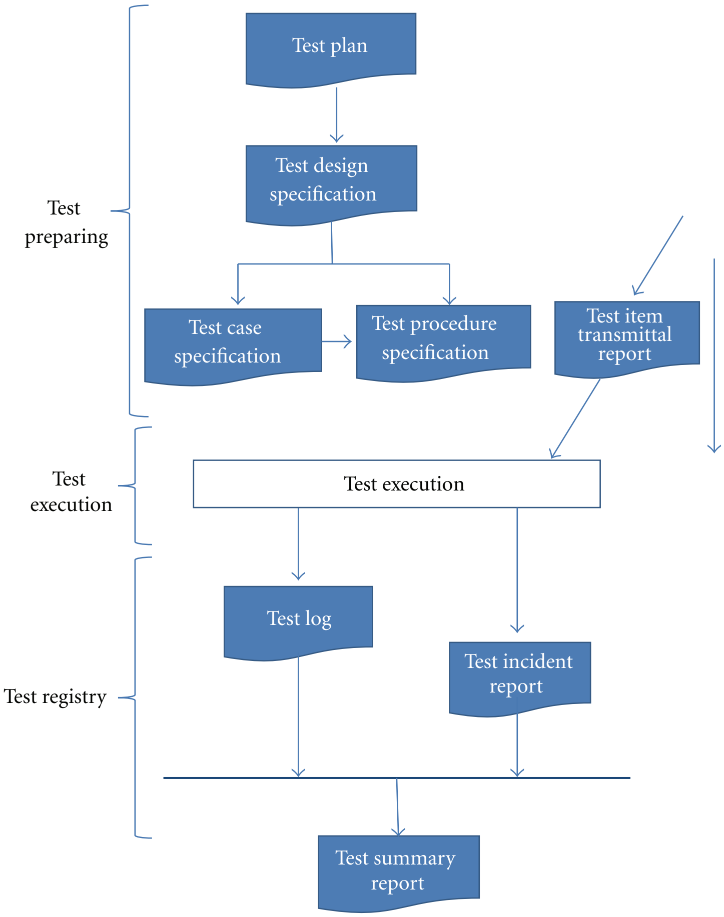


Figure 1: Relationship between testing documents.

Thus, from the definition of constraints and objectives expected for a testing process and considering the favorable and unfavorable conditions of the enterprise, it is possible to use operations research techniques such as linear programming, game theory, queuing theory, dynamic programming, risk analysis, goal programming, among others. These techniques and methods provide the decision maker with the possibility of increasing the degree of rationality of the decision, assisting in defining the actions to be taken, since they allow considering various relevant aspects of the decision making process.

The remainder of this paper is organized as follows. Section 2 explains the importance of software testing activity for enterprises in general and how the main standards and maturity models for software address such activities. Section 3 explains the basis of the multicriteria model and how this model can be used for the decision support process. Then, a multicriteria model is defined to support the definition of a software testing methodology in Section 4. Next, Section 5 describes the goal programming model, the proposed model and the results obtained with application from this model. Finally, a conclusion of this work is taken in Section 6.

2. The Importance of Using a Process

Software testing has been covered in the most acknowledged process maturity models and standards, such as CMMI [7], MPS BR [8], ISO/IEC 12207 [9], and ISO/IEC 15504 [10]. All of these maturity models and standards provide a guide to support the definition of a software testing process so that the testing activities are performed in an organized, disciplined manner and based on a set of well-defined activities. However, the definition of that process and the artifacts generated are the responsibility of the company that implements the standard or model and should be in accordance with their needs and the particular characteristics of each project.

Likewise, the evaluation and improvement models of software testing processes, such as Testing Maturity Model [11] and Test Process Improvement [12], were created in order to provide support to companies wishing to improve the testing process. Nevertheless, it is not the purpose of these models to assist the company in the definition of the testing process. However, the ISO/IEC 829 [13] Standard is slightly different from other standards and models because it establishes a set of documents to be generated over the testing activities. Altogether there are eight documents included from planning and specification activities until testing reports, as shown in Figure 1.

3. Related Works

According to [14], although the IEEE 829 can be used to test software products of any size or complexity, for small or low-complexity projects, some proposed documents can be grouped to reduce the management and production cost of documents. Moreover, the content of the documents can also be shortened.

GRASP metaheuristic is applied to the regression test case selection problem, defining an application related to prioritization of tests in [15]. In another work [16], an application for test redundancy detection is developed in a useful way, as a collaborative process between testers and a proposed redundancy detection engine to guide the tester to use valuable coverage information.

4. Fundamentals of Multicriteria

The objective of multicriteria is to reduce the subjectivity in the decision making process, using multiple criteria and application of mathematical calculations. However, it is important to emphasize that the subjectivity factor will always be present in the decision making process since the items to be mathematically evaluated are a result of human opinions. Its focus is, therefore, to support the decision maker in the information analysis and seek the best strategy among existing alternatives.

The adoption of a multicriteria decision support approach in this work is due to the fact that this approach, that is in constant growth, provides the group involved in the decision making process with subsidies required to obtain a solution that best fits the needs of the group [17]. Also, the approach focuses on issues that include qualitative and/or quantitative aspects, based on the principle that the experience and knowledge of people are at least as valuable as the data used for decision making [18]. Another factor that encouraged the adoption of the multicriteria approach in our study was the fact that this method has already been applied in the software engineering field [19–21].

5. A Multicriteria Model to Support the Definition of a Software Testing Methodology for Micro and Small Enterprises

A multicriteria model, developed by Rodrigues et al. [20–23], based on objective and subjective criteria was applied in order to support the prioritization of the artifacts available in the IEEE 829-1998 Standard, selecting the most relevant items for micro and small enterprises.

The model consists of a series of generic steps that were distributed among the three main phases of the decision supporting process: (1) structure, (2) evaluation, and (3) recommendation. However, in the context of our work, we used only the structure phase of the model because other phases were replaced by the goal programming model, resulting in a hybrid model. The steps of the structure phase are detailed in Table 1.

5.1. Structure. In the step "identify criteria," all the involved actors participated in a conference session to choose the criteria to be analyzed for prioritization of the documents of the IEEE 829-1998 Standard. The criteria selected for the evaluation are shown in Table 2.

In the step "identify actors and their weights," two groups of actors were selected to participate in this decision making process: the first group, consisting of two experts in software testing, and the second group composed of two professionals working in micro and small enterprises.

Then the actors answered a survey composed of three sections covering the characterization of the respondent, the prioritization of the criteria used in the model and the analysis from the point of view of the actor regarding the criteria and documents analyzed. The documents, in turn, represent the alternative solutions in decision making process.

To calculate the weight for a specific actor, a value for each of the answer options was established for each item of the survey. Also an adjustment factor was established for each group of actors, testing experts and professionals working in micro and small enterprises in relation to each of the criteria considered in the model. This is needed to establish a differentiated weight for each group of actors by criterion evaluated in the model. Thus, the weight of each actor consists of the sum of the scores referring to the actor's response to each item of the survey, multiplied by the adjustment factor of the criterion in question, considering the group to which the actor belongs to. The formula applied for obtaining the actor's weight is given below:

$$\mathrm{PA}(a,c) = \sum_{i \in A} R_i * F_c, \tag{1}$$

where $\mathrm{PA}(a,c)$ is the actor's weight a for the criterion c, R_i corresponds to the actor's answer for the item i of the survey, F_c corresponds to the multiplication factor of the criterion in question, and A corresponds to the number of items of the survey.

It is worth noting that the actor's weight for the criterion does not vary according to the document being assessed.

In the step "assign prioritization to the criteria," each actor ranked the selected criteria according to their degree of relevance, not by specific document, but for the company's testing process, as shown below:

$$\begin{gathered} \text{Priority}\,(a,c_i) \in \{1,\ldots,i,\ldots,n\} \Big| \,\forall c_i\,, c_j \Big|\, i \neq j \\ \Longrightarrow \text{Priority}\,(a,c_i) \neq \text{Priority}\,\left(a,c_j\right), \end{gathered} \tag{2}$$

where Priority (a,c_i) is the criterion priority c for the actor a, n is the number of criteria, and $i, j > 0$.

Next, for each criterion, a partial evaluation of the score was made. This partial evaluation is derived by multiplying the values obtained from the viewpoint of the actor for each document, the actor's weight for each criterion and

Table 1: Model's steps.

Phase	Step	Description
Structure	(7) Identify criteria	Identifies the criteria to be used to evaluate the artifacts and defines the priority of these criteria.
	(8) Identify actors and their weights	Identifies the actors who will expose their points of view, also considering their roles in the decision making process.
	(9) Assign prioritization to the criteria	Each actor should assign a weight to the criterion, considering the full test process and not only a specific artifact.
	(11) Execute a partial evaluation	During this step, the values are normalized by placing them on the same base (base 1), in order to perform a equality partial evaluation. This evaluation is derived by multiplying the values obtained from the point of view of the actor for each document by the actor's weight and the criteria priority.
	(12) Calculate the general scores of the criteria	It is the calculation of the median obtained using values resulting from the previous step.

Table 2: Selected criteria.

Criterion	Description
Difficulty level	Is the level of difficulty to obtain the information necessary to fulfill the artifact low?
Information relevance	Is the information contained in the artifact relevant to the reality of the organization?
Reuse possibility	Is there a high possibility for the information contained in the artifact to be reused?
Effort	Is the effort required to complete the artifact low?
Knowledge	Is the level of knowledge in software testing required to fulfill the artifact appropriately quite low?
Cost	Is the cost for fulfilling the artifact low?

the priority assigned by the actor to the criterion. This multiplication represents the score (E) of each actor for each criterion of each document and will serve for further classification and prioritization of documents, as shown below:

$$E_x(a,c) = [\mathrm{PV}_x(a,c)]_1 * [\mathrm{PA}(a,c)]_1 * [\mathrm{Priority}\,(a,c)]_1. \tag{3}$$

It is noteworthy that the values applied for the criteria priority and actor's weight were equalized in order to ensure an equitable evaluation, as shown below

$$[\mathrm{Priority}\,(a,c)]_1 \in \left\{\frac{1}{n}, \frac{2}{n}, \ldots, \frac{n}{n}\right\}, \tag{4}$$

where n represents the number of criteria

$$[\mathrm{PA}\,(a,c)]_1 = \frac{\sum_{i\in A} Rx_i * F_c}{m}, \tag{5}$$

where m is the number of actors.

After the results obtained in the partial evaluation for each actor, in the step "calculate the general scores of the criteria" was carried out, the calculation of the arithmetic mean to obtain the final score of each document for each criterion, as shown below:

$$\mathrm{Average}\,(x,c) = \frac{\sum_{i=1}^{m} E_x}{m}, \tag{6}$$

where E_x is the corresponds to the partial score of the document x_i, determined by the actor a referring to the criterion c, m is the number of actors.

The average value found was used as a basis for the prioritization of the documents during the application of the goal programming model.

6. Goal Programming

In real problems, the occurrence of multiple objectives to be considered at the time of resolution is frequent and mutually conflicting goals are not uncommon.

The goal programming is an extension of linear programming that has emerged as an option to the basic mathematical programming models, allowing to solve, simultaneously, a system of complex goals, rather than a single, simple goal [24].

This is a method that requires an iterative solution procedure by which the decision maker investigates a variety of solutions. However, a peculiarity of goal programming is the impossibility of representing all the goals in a single-objective function (often conflicting or unrelated). Therefore, goal programming considers all the objectives and seeks a solution that least deviates this set and at the same time meets the restrictions of the problem.

The goal programming and the approaches for multicriteria decision support have in common the fact of presenting a series of feasible solutions to the problem so that the decision makers can find/choose the one that best meets their needs and expectations. This similarity between the approaches motivated the creation of the hybrid model for the selection of documents to be produced during the testing activities.

The main idea of goal programming problems is to describe a function for each goal and also set a goal value for each of these. These functions represent the constraints of the problem, and then formulate an objective function that minimizes the sum of the deviations of all goals. The following presents a goal programming model simplified, first presented by Charnes and Cooper [25]:

$$\text{Mininmize} \quad \sum_m (y_m^+ + y_m^-) \tag{7}$$

$$\begin{aligned}&\text{Subject to} \quad f_m(X) - v_m = y_m^+ - y_m^-;\\ &\text{for each goal } m\end{aligned} \tag{8}$$

$$\begin{aligned}&y_m^+,\ y_m^-,\ x_n \geq 0;\\ &\forall m, i = 1, \ldots, n,\end{aligned} \tag{9}$$

where $X = \{x_1, \ldots, x_n\}$ corresponds to the vector of decision making variables that at the end of the process of resolution of the model will provide a configuration that leads to better results, considering all the goals and constraints applied to the model; $f_m(X)$ is a function $f : \Re^n \rightarrow \Re$ that defines the objective of goal m; v_m is the numerical value to be achieved by the goal m; y_m^+ and y_m^- are the positive and negative deviation variables, respectively, that calculate the variation of the goal m. The variable y_j^- represents the absolute value that each goal is below the originally desired and y_j^+ represents the absolute value that each goal is above the originally desired. In order that a negative deviation does not cancel a positive deviation, the model should measure it in absolute terms, and, therefore, the objective function is represented by minimizing the sum of the deviations of each goal.

Note that in the model suggested by Charnes and Cooper [25], we assumed that all objectives and all goals have the same weight.

6.1. Proposed Goal Programming Model. The purpose of our work is to decide which documents, among those suggested by the ISO/IEC 829 Standard, presented in Section 2 of this paper, and are more relevant to the reality of micro and small enterprises, in order to make the amount of documents produced during the testing activity more viable to the context of the MSE. These variables are represented by the binary variables u_x, which represent

$$u_x = \begin{cases} 1, & \text{if the document } x \text{ is selected,} \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Next, we define the goals for each objective. The goals were defined based on the criteria considered relevant for the selection of artifacts. Each goal relates directly to one of the six selected criteria during the application of the multicriteria model, as shown in Table 2. The definition of the goals can be seen below:

(i) Goal 1—select artifacts whose level of difficulty to obtain the information needed to its completion is low;

(ii) Goal 2—select the artifacts whose information is of most relevance to the reality of the enterprise;

(iii) Goal 3—select the artifacts in which the possibility of reusing the information contained therein is high;

(iv) Goal 4—select artifacts whose effort required to complete them is low;

(v) Goal 5—Select artifacts whose level of knowledge in software testing required to complete them properly is pretty low;

(vi) Goal 6—Select the artifacts whose cost to complete is considered low.

As you can see, the goals defined do not have values directly associated. Therefore, the values obtained using the multicriteria model (step "calculate the general scores of the criteria"), depicted in the previous section, were used as the value of each one of the goal defined herein. Thus, two possible values v_m were established for each goal m associated with a criterion c, and that will be applied to the constraints of inequality (8), according to the section Goal Programming:

(I) $v_m = \sum_{x \epsilon D} E_{x,c}$: summation of the scores of documents to the criterion c;

(II) $v_m = \sum E_{x,c}/|D|$: average of the scores of the documents to the criterion c,

where D represents the set of all documents.

After assignment of values of each goal, the constraint (8) presented in the simplified model of Charnes and Cooper [25], was redefined using the decision variables and the goal values assigned, as shown below:

$$\sum_{x \in D} E_{x,c} * U_x \geq v_m, \tag{11}$$

where x is the document and c is the criterion related to goal m.

Then, the objective function (7) was also redefined by considering that all the positive deviations can be discarded since it is a maximization problem. For a maximization problem it does no matter how much a positive deviation is exceeded, it matters only how to minimize the maximum negative deviations, which represents how much is missed for achieving the goal. Another difference between the objective function redefined and the function originally presented in the simplified model of Charnes and Cooper [25] is the inclusion of weights for each of the goals, which makes the model closer to reality, considering that each goal will have a differentiated factor of relevance according to the particularities of each enterprise. Therefore, we have the following objective function:

$$\text{Minimize} \sum_m w_m y_m^-. \tag{12}$$

Finally, to avoid the obvious solution, which in the context of this work is the use of all the documents in order to achieve all the objectives, it was necessary to add constraint (13), in order to obtain the best subset of n documents,

$$\sum_{x \in D} U_x \leq q, \tag{13}$$

where q is a parameter of the model created with the purpose of limiting the amount of documents used. Thus, if $q = 1$, then the model will define the best document considering all the goals and constraints. If $q = 2$, then the model will define the two best documents, given that the first document was already known after the first run of the model. Therefore, the successive execution with the value of q incremented step by step will result in a formulation of the top ranking documents.

The final model consists of the objective function (12) plus the constraints (9), (11), and (13). The results obtained for the goal values (I) and (II) are presented in the next section:

$$\begin{aligned} &\text{Minimize} && \sum_{m} w_m y_m^-, \\ &\text{Subject to} && \sum_{x \epsilon D} E_{x,c} * U_x \geq v_m, \\ &&& \sum_{x \epsilon D} U_x \leq q, \\ & y_m^+, y_m^-, x_i \geq 0; && \forall m, i = 1, \ldots, n. \end{aligned} \tag{14}$$

7. Application of Hybrid Model for the Selection of Test Artifacts

The proposed testing methodology for micro and small software development companies is based on the ISO 829, with regard to the standard documentation to be developed during the planning and execution of the tests. We chose to use this standard as a basis for this work, given that, in addition to presenting a set of documents that can be adapted for specific organizations or projects, this standard provides a set of information relevant to the testing of software.

The adaptation of ISO 829 with the purpose of enabling its use in micro and small enterprises was based on Crespo (2003). According to Crespo (2003), although the IEEE 829 can be used for testing of software products of any size or complexity, for the projects of small or low complexity, some proposed that documents can be grouped so as to decrease the manageability and production cost of documents. Furthermore, document content may also be abbreviated.

That way, the proposed methodology was defined with the support of experts in software testing and professionals engaged in micro and small enterprises through the hybrid approach involving multiple criteria and goal programming technique, as presented in the following section.

7.1. Application of Multicriteria Model in the Prioritization of Documents Made Available by ISO 829.

Initially the multicriteria model was applied to the documentation provided by the IEEE 829-1998, aiming at obtaining a goal value to each of the documents and, thus, supporting the selection of the items most relevant to the needs of micro and small enterprises through the application of goal programming model. The structuring phase of the multicriteria model consists of the steps of identifying criteria, identification of the actors and their weights, and realization of partial evaluation.

During the stage of identification of criteria, all actors participated in a section for the choice of the criteria to be considered for prioritization of documents of IEEE 829-1998. Two groups of actors were selected to be part of this decision making process. The first group consists of 2 (two) experts in software testing, and the second group consists of 2 (two) professionals working in micro and small enterprises.

These actors have responded to a questionnaire consisting of 3 (three) sections, covering the characterization of the interviewee, the prioritization of criteria used in the model, and the analysis of the actor's point of view against the criteria and analysed documents. The documents, in turn, represent the decision making process solution alternatives.

The calculation of the actor's weight was obtained through the sum of the score corresponding to each of the items of the questionnaire. For example, in the case of the item of the questionnaire that considers the degree of the actors, each one of the options: bachelor, specialization, masters, and Ph.D., corresponds to a specific score, ranging between 0.25; 0.50; 0.75; 1.0, respectively. The weight of each actor is presented in Table 3.

Still regarding the weight of the actor, each criterion was established with the support of experts and through conferences, a factor that is differentiated for each group (experts and practitioners of micro and small enterprises), which multiplied the weight of the actor obtained previously, corresponds to the actor's weight to the criterion in question. Table 4 presents the factor defined for each criterion and a group of actors.

To facilitate the understanding, the calculation of the final weight of one of the actors participating in the process in relation to the relevance of information is presented below:

$$\text{PA}(a, c) = \sum_{i \in A} Rx_i * F_c, \tag{15}$$

so $\text{PA}(\text{Professional MPE 2, Relevance of information}) = (1 + 1 + 1 + 0,75) * 0,66 = 2,475$.

Then each actor ranked the selected criteria as your perception of their priority for the process as a whole, as can be seen in Table 5.

Continuing the process, for each criterion, we performed a partial evaluation of the score. This partial evaluation is the result of the multiplication of the values obtained from the point of view of the actor for each document, the actor's weight, and priority of the criterion. It is worth mentioning that the values applied to priority and weight of the actor, were equalized, aiming to ensure equal assessment. Thus, whereas the priority value presented in Table 5, for the professional actor MPE 2 regarding the criterion "relevance of information" is 4, for the calculation of partial evaluation, we have

$$[\text{Priority}(a, c)]_1 \in \left\{ \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n}{n} \right\}, \tag{16}$$

Table 3: Actor's weight.

Criteria	Expert 1	Expert 2	Professional MPE 1	Professional MPE 2
Difficulty level	0,99	0,7425	3,25	3,75
Information relevance	2,49	1,8675	2,145	2,475
Reuse possibility	1,5	1,125	1,0725	1,2375
Effort	1,98	1,485	1,625	1,875
Knowledge	3	2,25	0,52	0,6
Cost	0,48	0,36	2,6975	3,1125

Table 4: Adjustment factor per criterion and group of actors.

Group of professionals	Difficulty level	Information relevance	Reuse	Effort	Knowledge	Cost
Professionals in micro and small enterprises	1	0,66	0,33	0,5	0,16	0,83
Expert in software testing	0,33	0,83	0,5	0,66	1	0,16

where n represents the amount of criteria, so: Priority (Professional MPE 2, Relevance of information) = 4/6, Priority (Professional MPE 2, Relevance of information) = 0,66.

Similarly, the actor's weight was also standardised. Therefore, standardisation of professional actor weight MPE 2, presented in Table 3, for the relevance of the information, is shown below:

$$[\mathrm{PA}(a,c)]_1 = \frac{\sum_{i\in A} Rx_i * F_c}{4}, \tag{17}$$

so PA(Professional MPE 2, Relevance of information) = 2,475/4, PA (Professional MPE 2, Relevance of information) = 0,61.

Finally, the calculation of partial evaluation representing the score obtained for the test plan document of professional actor MPE 2, with respect to the criterion "relevance of information," is presented below:

$$\begin{aligned} &E_x(a,c) = [\mathrm{PV}_x(a,c)]_1 * [\mathrm{PA}(a,c)]_1 * [\mathrm{Priority}\,(a,c)]_1, \\ &E_{\text{test plan}}(\text{Professional MPE 2, Relevance of information}) \\ &\quad = [1 \times 0,61 \times 0,66], \\ &E_{\text{test plan}}(\text{Professional MPE 2, Relevance of information}) \\ &\quad = 0,40. \end{aligned} \tag{18}$$

After the result was obtained in partial evaluation for each actor, the average calculation was performed to obtain the final score of each document for each criterion. Thus, for the test plan document, criterion "relevance information," considering the scores of all the actors, we have

$$\begin{aligned} &E_{\text{test plan}}(\text{Expert 1, Relevance of information}) \\ &\quad = [1 * 0,62 * 0,5] = 0,31, \\ &E_{\text{test plan}}(\text{Expert 2, Relevance of information}) \\ &\quad = [1 \times 0,46 * 0,5] = 0,23, \\ &E_{\text{test plan}}(\text{Professional MPE 1, Relevance of information}) \\ &\quad = [1 * 0,53 * 1] = 0,53, \\ &E_{\text{test plan}}(\text{Professional MPE 2, Relevance of information}) \\ &\quad = [1 * 0,61 * 0,66] = 0,40, \\ &\mathrm{ME}(\text{test plan, Relevance of information}) = 0,3675. \end{aligned} \tag{19}$$

Table 6 presents the result of the average of each document and criteria. This result, as shown in Section 5.1, the value of goal of each criterion was evaluated in the model obtained based on:

(I) $v_m = \sum_{x\epsilon D} E_{x,c}$: summation of the scores of documents to the criterion c;

(II) $v_m = \sum E_{x,c}/|D|$: average of the scores of the documents to the criterion c.

The value corresponding to the sum of the scores of documents for each criterion, as well as the average of the scores of the calculated documents was obtained based on the presented values.

Finally, the goal programming model structured based on the results obtained through the application of multicriteria model is presented below. The average of the priorities assigned to each criteria by the actors involved in the decision process was used as the weight of the criteria in the objective function of goal programming model. Likewise, as shown in

Table 5: Priorities criteria by actor.

Criteria	Expert 1	Expert 2	Professional MPE 1	Professional MPE 2
Difficulty level	5	2	4	1
Information relevance	3	3	6	4
Reuse possibility	1	4	1	2
Effort	6	5	3	5
Knowledge	2	1	2	3
Cost	4	6	5	6

Table 6: Overall score of the criteria.

	Calculation of scores of the criteria					
Document	Difficulty level	Information relevance	Reuse possibility	Effort	Knowledge	Cost
Test plan	0,000	1,493	0,335	0,495	0,000	0,080
Test design specification	0,062	1,182	0,103	0,804	0,250	0,090
Test case specification	0,206	1,493	0,166	0,495	0,250	0,080
Test procedure specification	0,542	1,493	0,166	0,203	0,250	0,642
Test log	0,966	0,311	0,000	0,513	0,462	0,732
Test incident report	0,966	1,493	0,000	1,089	0,368	1,420
Test summary report	0,218	1,493	0,000	1,195	0,344	0,948

Section 5.1, the model was carried out twice with different amounts of target. The first run used as target value for each criterion, the overall mean score of each document for each criterion, as can be seen below in the constraint definition R_Crit_0

$$\begin{aligned}\text{Minimize}\quad & \text{obj}: 0.5\text{Yn\_Dif} + 0.667\text{Yn\_Rel} + 0.333\text{Yn\_Reu} + 0.792\text{Yn\_Esf} + 0.333\ \text{Yn\_Con} + 0.875\text{Yn\_Cst} + \text{id173}\\ \text{Subject to}\quad & \text{R\_Crit\_0}: \text{Yn\_difficulty} + 0.015\text{U\_EPjT} + 0.052\text{U\_ECT} + 0.135\text{U\_EPT} + 0.242\text{U\_HT}\\ & \qquad + 0.242\text{U\_RIT} + 0.055\text{U\_RRT} - \text{Yp\_difficulty} = 0.1058571\\ & \text{R\_numDocs}: \text{E\_EPjT} + \text{E\_ECT} + \text{E\_EPcT} + \text{E\_HT} + \text{E\_RIT} + \text{E\_RRT} + \text{E\_PT} = 6.\end{aligned} \tag{20}$$

In the second run, the value of the summation of the scores of each general document for each criteria was used as the target value for the constraint R_Crit_0. In this case, the value 0.1058571 was replaced by 0.8468571. The following are the results of applying the hybrid model to support the decision of the test artifacts selection.

In the second run, the value of the summation of the scores of each general document for each criteria was used as the target value for the constraint R_Crit_0.

7.2. Computational Results. The proposed model was solved using the software IBM ILOG OPL 6.3, assigned to the academic community. Two scenarios were performed using the sum (I) and the average (II) as the goal values. The results are shown in Figure 2.

We can note that the prioritization of the documents was carried out satisfactorily, considering the two goal values options established. However, the application of the model using the average as a goal value obtained a result closer to what had already been obtained by applying a multicriteria model and evaluation of the actors involved in decision making process presented by Rodrigues [26]. This result is possibly more feasible because it includes documents produced during the main stages of the testing activity. On the other hand, the result that considered as a goal value of the summation could be more interesting for an enterprise in which the documents produced for the stage of execution and evaluation of the tests are of higher priority than those documents produced for the planning stage of tests.

8. Conclusion and Future Works

This paper proposed a hybrid model for prioritization of documents to be produced during the software testing activity in an enterprise. The purpose of applying the model to obtain the prioritization of the documents available on IEEE 829 [13] Standard is to establish a set of documents that may be considered, possibly closer to the reality of micro and small enterprises when they are not able to produce all documents suggested by the IEEE 829.

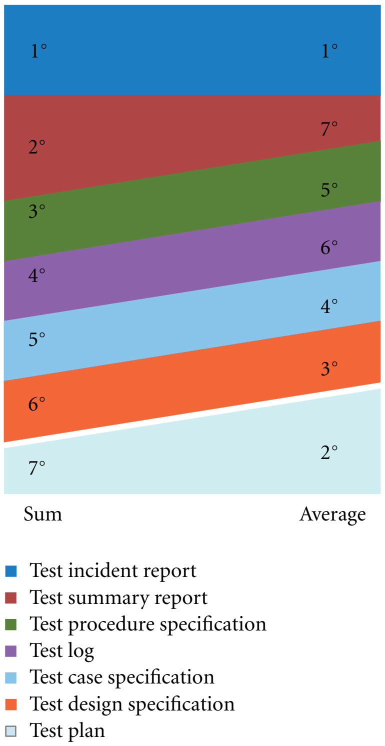


Figure 2: Comparison of results using as goals the sum (I) and average (II).

The application of the proposed model met the goal of prioritization while at the same time allowed to replace the way enterprises define, often using an evaluation without criteria formally established, the documents that are part of the scope of their processes. Thus, the application of a decision support model attempts to decrease the subjectivity, with which decisions are made, but it is important to note, not necessarily the result of decision making will be exactly equal to the result obtained with application of the prioritization model. However, future works, it is essential that the differences among the results obtained with the application of the model and the judgments expressed by the participants involved in the decision making process are evaluated, with the purpose of adjusting the model as close as possible to the reality.

Although the proposed approach aims the prioritization of documents to form a testing process, we believe that it can be applied to different prioritization problems, such as prioritization of use cases to perform peer review.

Acknowledgments

The first and second authors would like to thank FUNCAP—Foundation for Support in Scientific and Technological Development State of Ceará—that has been financially sponsored this work, and the third author would like to thank CNPq—National Counsel of Technological and Scientific Development, via Grant no. 305844/2011-3. The authors also acknowledge IBM for making the IBM ILOG CPLEX Optimization Studio available to the academic community.

References

[1] MCT. Ministério da Ciência e Tecnologia. Qualidade e Produtividade no Setor de Software Brasileiro: Result-ados da pesquisa, 2005, http://www.mct.gov.br/index.php/content/view/3253/Pesquisa_2005.html.

[2] SEPIN (Secretaria de Política de Informática), MCT (Ministério da Ciência e Tecnologia), Qualidade e Produtividade no Setor de Software—Resultados da Pesquisa, 2005, http://www.mct.gov.br/index.php/content/view/3253/Pesquisa_2005.html.

[3] S. Wagner and T. Seifert, "Software quality economics for defect-detection techniques using failure prediction," *ACM Software Engineering Notesn*, vol. 30, no. 4, pp. 1–6, 2005.

[4] A. Rodrigues, A. Bessa, and P. R. Pinheiro, "Barriers to implement test process in small-sized companies," *Communications in Computer and Information Science*, vol. 112, no. 2, pp. 233–242, 2010.

[5] P. Laplante, F. Belli, J. Gao et al., "Software test automation," *Advances in Software Engineering*, vol. 2010, Article ID 163746, 2 pages, 2010.

[6] L. E. S. Sartori, *Melhoria do processo de teste para pequenas empresas [M.S. Dissertação]*, Departamento de Ciência da Computação, Centro Universitário Eurípides de Marília, Fundação de Ensino Eurípides Soares da Rocha, 2005.

[7] "SEI (Software Engineering Institute). CMMI for Development (CMMI-DEV), Version 1.3," Tech. Rep. CMU/SEI-2010-TR-033, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, USA, 2010.

[8] SOFTEX (Associação para Promoção da Excelência do Software Brasileiro—SOFTEX), MPS.BR—Guia Geral, 2011, http://www.softex.br.

[9] R. Singh, "ISO/IEC draft international standard 12207, software life-cycle processes," *IFIP Transactions A*, no. 55, pp. 111–113, 1994.

[10] ISO/IEC 15504, *Information Technology—Software Process Assessment, Parts 19*, International Organization for Standardization and the International Electrotechnical Commission, Geneva, Switzerland, 2003.

[11] I. Burnstein, T. Suwanassart, and R. Carlson, "Developing a testing maturity model for software test process evaluation and improvement," in *Proceedings of the 1996 IEEE International Test Conference*, pp. 581–589, Washington, DC, USA, October 1996.

[12] T. Koomen and P. O. L. M, *Test Process Improvement: A Practical Step-By-Step Guide to Structured Testing*, Addison-Wesley, 1999.

[13] IEEE Standard 829-1998: Standard for Software Test Documentation, IEEE Press.

[14] A. N. Crespo, O. J. Silva, C. A. Borges, C. F. Salviano, M. Argollo, and M. Jino, "Uma metodologia para teste de Software no Contexto da Melhoria de Processo," in *III Simpósio Brasileiro de Qualidade de Software (SBQS '04)*, Brasília, Brazil, 2004.

[15] C. L. B. Maia, R. A. F. Carmo, F. G. Freitas, G. A. L. Campos, and J. T. Souza, "Automated test case prioritization with reactive GRASP," *Advances in Software Engineering*, vol. 2010, Article ID 428521, 18 pages, 2010.

[16] N. Koochakzadeh and V. Garousi, "A tester-assisted methodology for test redundancy detection," *Advances in Software Engineering*, vol. 2010, Article ID 932686, 13 pages, 2010.

[17] C. A. Bana e Costa, J. M. De Corte, and J. C. Vansnick, "MACBETH," *International Journal of Information Technology and Decision Making*, vol. 11, no. 2, pp. 359–387, 2012.

[18] Â. M. A. Schmidt, *Processo de apoio à tomada de decisão—abordagens: AHP e MACBETH [M.S. Dissertação]*, UFSC, Florianópolis, Brazil, 1995, http://www.eps.ufsc.br/disserta/engait95.html.

[19] F. Gonçalves, G. S. Márcia et al., "Multicriteria model for selection of automated system tests," in *Proceedings of the International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS '06)*, Viena, Áustria, 2006.

[20] A. Rodrigues, P. R. Pinheiro, M. M. Rodrigues, A. B. Albuquerque, and F. M. Gonçalves, "Towards the selection of testable use cases and a real experience," *Communications in Computer and Information Science*, vol. 49, pp. 513–521, 2009.

[21] A. Rodrigues, P. P. Rogerio, M. M. Rodrigues, C. A. Albuquerque, and F. M. Gonçalves, "Applying a multicriteria model for selection of test use cases: a use of experience," *International Journal Social and Humanistic Computing*, vol. 1, pp. 246–260, 2010.

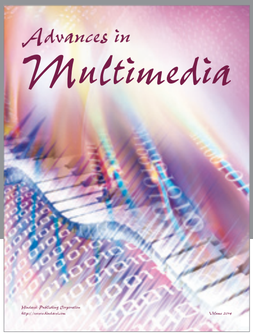
[22] A. Rodrigues, C. A. Albuquerque, P. P. Rogerio, A. Bessa, A. Diego, and M. Thiago, "Uma abordagem de teste de software para micro e pequenas empresas," in *Proceedings of the 4th International Conference on Research and Practical Issues of Enterprise Information Systems*, 2010.

[23] A. Rodrigues, P. R. Pinheiro, and A. Albuquerque, "The definiton of a testing process to small-sized companies: the Brazilian scenario," in *7th International Conference on the Quality of Information and Communications Technology, QUATIC 2010*, pp. 298–303, Porto, Portugal, October 2010.

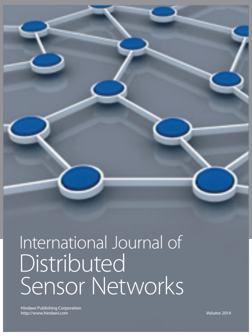
[24] A. C. Hax and D. Candea, *Production and Inventory Management*, Prentice-Hall, 1984.

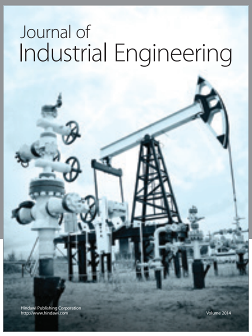
[25] A. Charnes and W. W. Cooper, "Goal programming and multiple objectives optimizations," *European Journal of Operational Research*, vol. 1, pp. 39–54, 1977.

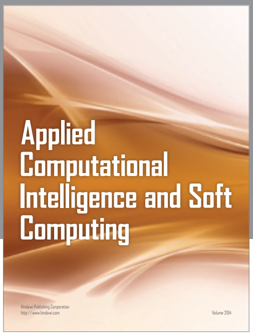
[26] A. Rodrigues, *Uma metodologia de testes em software para micro e pequenas empresas estruturada em multicritério [M.S. dissertation]*, Graduate Program in Applied Informatics, University of Fortaleza, 2011.

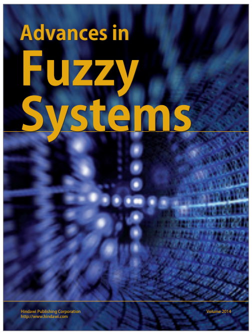

Advances in
Multimedia

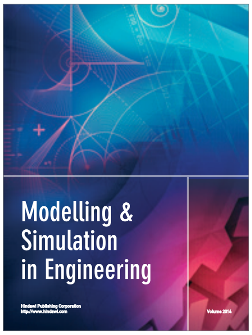
The Scientific
World Journal

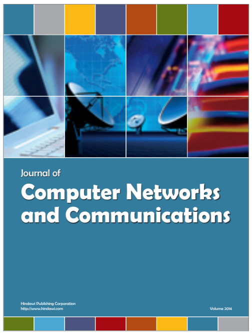

International Journal of
Distributed
Sensor Networks

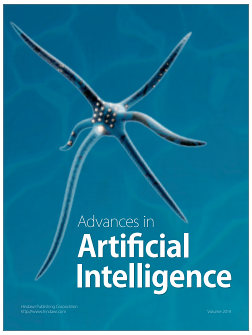

Journal of
Industrial Engineering

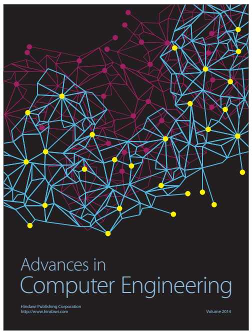

Applied
Computational
Intelligence and Soft
Computing


Advances in
Fuzzy
Systems

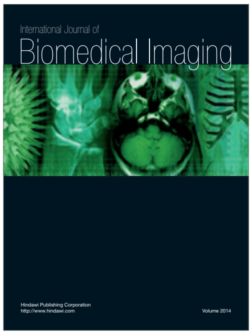

Modelling &
Simulation
in Engineering

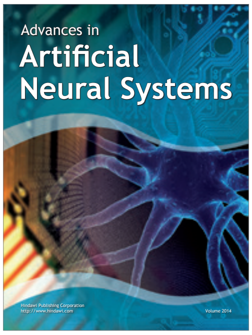

Journal of
Computer Networks
and Communications

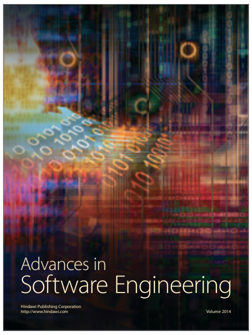

Advances in
Artificial
Intelligence


Advances in
Computer Engineering

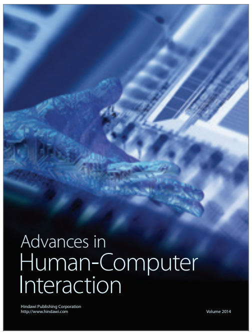
International Journal of
Computer Games
Technology

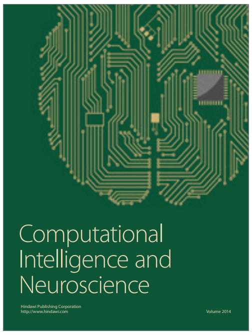

International Journal of
Biomedical Imaging


Advances in
Artificial
Neural Systems


Advances in
Software Engineering

Journal of
Robotics


Advances in
Human-Computer
Interaction


Computational
Intelligence and
Neuroscience

International Journal of
Reconfigurable
Computing

Journal of
Electrical and Computer
Engineering