

Selection and prioritization of software requirements using the Verbal Decision Analysis paradigm

Paulo Alberto Melo Barbosa
University of Fortaleza - UNIFOR
Fortaleza – Brazil
albertobmap@hotmail.com

Plácido Rogério Pinheiro
University of Fortaleza - UNIFOR
Fortaleza – Brazil
placido@unifor.br

Francisca Raquel de Vasconcelos Silveira
University of Fortaleza - UNIFOR
Fortaleza – Brazil
raquel.vsilveira@hotmail.com

Marum Simão Filho
7 de Setembro College
Fortaleza – Brazil
marum@fa7.edu.br

Abstract— Selecting and prioritizing the most stable software requirements within a set of requirements and engaging them in releases that satisfy the most customers is a difficult task for the decision maker. Many methods have been employed to solve this type of problem. But we do not find many solutions that use Verbal Decision Analysis. Therefore, in this paper we aim to select and prioritize software requirements using Verbal Decision Analysis techniques as a tool, exploring the ZAPROS III-i method and comparing the results with those obtained by the NSGA-II, SPEA2 and Moeckel metaheuristics and also with a random algorithm.

Keywords- *Software Release Planning, Multi-objective optimization, Verbal Decision Analysis, ZAPROS III-i*

I. INTRODUCTION

Requirements are critical in the software development process. They provide the basis for estimating costs and effort, as well as allowing the development of estimates and test specifications [1]. During the software release planning process we can find several constraints, such as: project budget and precedence between requirements [2].

It is essential that the task of selecting and prioritizing requirements to take effect in the most efficient way possible. In the literature, we can find several methods to solve this problem. A common method is the use of metaheuristics that are algorithms used to solve optimization problems. Becceneri [3] say that metaheuristics is algorithmic tool, which can be applied to different optimization problems, with relatively small modifications, in order to make them adaptable to a specific problem.

Deciding which requirement to prioritize for implementation is typically a decision problem. We mean that a high degree of subjectivity is present in the decision-making process. This is a suitable scenario for Verbal Decision Analysis (VDA), which consists of an approach based on multi-criteria problem solving through its qualitative analysis [4], that is, VDA methods take into account criterion subjectivity.

Therefore, structured context in this work was the Software Releases Planning, which indicates a methodology that uses Verbal Decision Analysis to be used by software managers as a means to obtain an effective planning taking into account the selection of more priority requirements, having as criterion Stability of Requirements.

Our goal is to obtain a solution that contains an order of requirements to be implemented considering constraints (technical precedence between requirements and resources available to the project) and objectives (maximizing customer satisfaction by selecting the most important requirements for key customers and maximize stability among requirements, initially implementing those with the highest degree of stability). The results of the approach proposed here are compared with Barbosa's [5], which is an extension of Brasil's [16] work that used a similar methodology to solve problems with uncertainty of the number of requirements. In section 2 we will see works related to the planning of software releases and the approach proposed by [5]. In section 3 we will see the methods adopted to solve the problem proposed in this paper. In section 4 the results achieved and discussions and in section 5 the conclusions and proposals of future work.

II. RELATED WORK

The work published by Bagnall [6] deals with the determination of the requirements that must be executed for the next release of the software. The author predicts that customers have different levels of importance to the company and points out the requirements that have prerequisites. The algorithms applied in this strategy show the obtaining of quick solutions to small problems.

Therefore Greer et al. [7] say that defining which release to deliver the requirement is a decision that depends on several variables that relate in complex ways. The different perspectives of the stakeholders and the planning of releases, including effort restraint, are discussed.

Brasil [16] presents a structured approach in multi-objective optimization using metaheuristics for the problem of selection and prioritization of software requirements, considering the

inherent characteristics of real project and an uncertain number of requirements. Among these characteristics, she considered: i) stability of the requirement; ii) costs to implement the requirement; iii) technical precedence between requirements; iv) importance of stakeholders to the company and v) its preferences regarding requirements. The objective of this work was to compare the efficacy of metaheuristics in problem solving through performance measures and to compare the performance of metaheuristics with the result of the implementation and execution of a solution generated from a random algorithm.

The work differential [5, 16] was used as the selection criterion. Unstable requirements to be implemented late to avoid diverting project resources. The work of [5, 16] compared as solutions generated from the execution of the metaheuristics NSGA-II [8], Mocal [1] and a random algorithm. The results showed that metaheuristics were the best methods to reach higher quality solutions.

The task of ordering requirements can be complex and challenging. Many proposals for metaheuristic solutions are found in the research work. However, methods structured in VDA are little known in the field. In the opinion of [5, 6, 7, 16], the methodology of multicriteria support to the decision has several methods that can be applied in the most diverse problems. Therefore, the very choice of a multicriteria decision support method alone is already a multicriteria problem [9].

This work proposes to select and prioritize software requirements in the order in which they will be implemented using a VDA method known as ZAPROS III-*i* [10]. The results will be compared to those obtained when using quantitative methods (metaheuristics).

For the problem considered in this work, the application of the ZAPROS III-*i* method came from the acceptance of the method by the decision maker, which meant that the issues that were being presented to the decision maker made sense to him, and he was confident to answer them. In addition to this point, the need to evaluate the acceptance of the data, its properties used by the method, and whether the result supported in the decision-making process was exalted.

For the development of this work, we divided it into three stages: i) the generation of instances that represent a set of requirements to be implemented, ii) the classification of these requirements using ZAPROS III-*I*, iii) the comparison of the results obtained relating these with results obtained through the application of metaheuristic NSGA-II used in [5], SPEA2 [11], Mocal used in [1] and the solution of random algorithm and iv) Finally, we will evaluate the results. The figure 1 illustrates this flow.

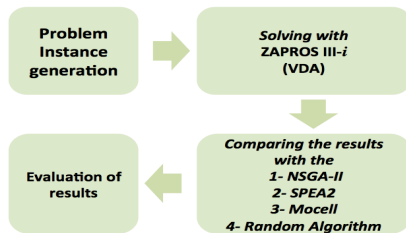


Figure 1. Considered Workflow

III. METHODS

The choice of a multicriteria method among those available, applied to a given context, should be adequate for the characteristics of the problem in question. An important point will be an evaluation of the problem, of the decision objects and the available information. The choice of method should be the result of an evaluation of the chosen criteria, the type and precision of the data, the form of the decision maker's thinking and his knowledge of the problem [12]. It is also emphasized that the direct consequence of possibility of choice between several methods that results can be discordant and even contradictory..

A. Instance Generation

Initially an application was developed to generate simulations of problems inherent in the software development process. In the work of [18] the results of the ZAPROS III-*i* method were compared with those obtained by the NSGA-II metaheuristic and good results were obtained with simulations involving 10 requirements and 5 clients. In this work we will increase the difficulty, especially the number of requirements that has been increased to 20, to analyze the behavior of the ZAPROS III-*i* method in this new situation involving a larger set of variables. Each simulation of the problem here contains:

- i) Number of requirements to be implemented, which in this work were fixed in 20 requirements
- ii) Number of customers interested in the project, which in this work was set at 7
- iii) Cost Total of the project that in the experiments was considered between 70% or 80% of the total value needed to implement all the requirements
- iv) Importance of each client to the company (whose value 01 represents the least important customer and 10 Importance),
- v) Cost of each requirement (where 10 is the lowest cost and 20 the highest),
- vi) Stability level of requirements, ranging from 1 to 10, where 1 means little stable and 10 is very stable
- vii) Technical precedence matrix among the requirements, signaling that a particular requirement should only be implemented after another requirement indicated in that matrix
- viii) Importance of the requirements for the customer that scores from 1 (minimum) to 10 (maximum) a customer's preference for a given requirement. The variations of these situations are presented in a simplified way in Table 1, which shows the four main characteristics adopted by this work

TABLE I. CHARACTERISTICS OF PROBLEM SIMULATIONS

File name	Number of requirements	Number of Clients	Percentage of technical precedence for the requirement	Budget available for the project
I.20.7.10.70	20	7	10%	70%
I.20.7.10.80	20	7	10%	80%
I.20.7.20.70	20	7	20%	70%
I.20.7.20.80	20	7	20%	80%

B. Classification using ZAPROS III-i

When a decision can generate a considerable impact, such as management decisions, and must take into account some factors, the use of methodologies to support the decision making process is suggested, because choosing the inappropriate alternative can lead to waste of resources, time, and money, affecting the company [13].

The ZAPROS III method [14] is an evolution of the ZAPROS-LM one, with the application of the same procedure to elicit the preferences, but with modifications that make it more efficient and more accurate with respect to inconsistencies. Another difference between these methods is that the ZAPROS III is based on the elicitation of preferences around values that represent the distances between the evaluations of two criteria, called Quality Variations (QV), instead of comparing criteria estimates, as in its older version..

ZAPROS III-i method applies i) the Formal Index of Quality (FIQ) [14], which was used with the purpose of reducing the number of pairs of alternatives to be compared, ii) the ideas of comparison between alternatives through ordering the values of their quality vectors in ascending order [17] and iii) the comparison considering all possible alternatives for the problem, which can be used for solving complex decision making process.

Therefore, ZAPROS III-i presents a valuable alternative to solve requirements selection problems, since the opinion of the decision-maker is taken into account in this process.

C. Application of the methodology

To rank order the factors that project managers should consider when allocating tasks in distributed software development projects [17], we applied a methodology consisting of four main steps, which are explained next: 1) Identification of the Alternatives; 2) Definition of the Criteria and the Criteria Values; 3) Characterization of the Alternatives; and 4) The ZAPROS III-i Method Application.

1. The alternatives considered in this work were the 20 software requirements to be implemented. Each requirement has its own characteristics and will be known ahead.

TABLE II. CHARACTERISTICS OF PROBLEM SIMULATIONS

Criteria	Criteria values
A Cost	A1 Requirement has low cost A2 Cost of requirement is reasonable A3 Cost of requirement is high
B Stability	B1 The requirement will hardly change B2 The requirement may change B3 The requirement will change
C Stakeholders	C1 The stakeholder is very important C2 The stakeholder has partial and isolated importance C3 The stakeholder is of little importance
D Customer requirement value	D1 The requirement is of great value to the customer D2 The requirement is of low importance to the customer

2. For the purpose of comparison, the criteria adopted in this study were the same as those adopted by [5] and [18]. As [3, 5] used quantitative methods and this work adopted the qualitative methodology, the data were converted to the

qualitative methodology, where, for example, the Cost criterion represented on a scale of 10 to 20 was discretized into three values, whose values between 10 and 13.3 are represented by criterion A1, values between 13.4 and 16.6 represented by criterion A2 and values between 16.7 and 20 represented by criterion A3. This same methodology was adopted for the other criteria. Thus, the criteria were ranked from the most preferable (A1B1C1D1) to the least preferred (A3B3C3D2), according to table 2.

3. The characterization of alternatives was done according to the values contained in the requirements of the work problems of [5, 18]. Thus, considering the definition of criterion presented in the previous item and taking into account the criterion Cost, a requirement that presents Cost 15 in [3, 5] was classified as Cost A2 considering table 2. In the particular case, as the criteria 'Stakeholders' And 'Customer requirement value' have more than one customer by punctuating the same requirement, the arithmetic mean between the scores indicated by those clients for the classification represented in table 2 was considered.
4. After defining and characterizing the alternatives, we moved on to the stage of ordering. At this stage, we applied the ZAPROS III-i method to put in order the influencing factors, such that it is possible to establish a ranking of them, how make in [13].

In order to facilitate the decision-making process and perform it consistently, we used the ARANAÚ tool, presented in [10, 19, 20] which was implemented in Java platform, was first developed in [21] to support ZAPROS III method. In work of [13], was used the updated version to ZAPROS III-i method. The use of ZAPROS III-i method in the ARANAÚ tool requires four steps, as follows: 1. Criteria and criteria values definitions, 2. Preferences elicitation, 3. Alternatives definition, and 4. Results generation. The process runs as follows. First, we introduced the criteria presented in table 2 into the ARANAÚ tool. Next, the decision-maker decides the preferences. The interface for elicitation of preferences presents questionings that can be easily answered to obtain the scale of preferences. The questions provided require a comparison considering the two reference situations [4]. Once the scale of preferences is structured, the next step is to define the problem's alternatives. The quantitative values of [5] were used and converted to qualitative values.

D. Overlap of results

For each simulation of the files of table 1 of the problem two solutions were extracted: a set of solutions of the NSGA-II, SPEA2, Moeckl, Random Algorithm execution and a solution of the ZAPROS III-i. NSGA-II and SPEA2 are similar and proposes a set of multi-criteria solutions coming from the Pareto front and the project manager selects the solution closest to his preferences. Moeckl is a multiobjective algorithm based on the Genetic Algorithm model. The ZAPROS III-i methodology contains a single solution that was generated from the preferences indicated by the project manager. The random algorithm uses no technique and was used to depict an execution without any search strategy.

IV. RESULTS

At the end of each execution, the ARANAÚ [21] tool provided a ranking with the requirements ordered according to their execution priority. In table 3, we can see the result of this execution for file I.20.7.10.80.

TABLE III. RANKING OF ARANAÚ TOOL FOR THE FILE I.20.7.10.80

Ranking	1	2	3	4	5	6	7	8	9	10
Requirements	8	10	19	20	7	12	15	18	5	14
Ranking	11	12	13	14	15	16	17	18	19	20
Requirements	2	17	3	4	11	16	1	13	6	9

To facilitate comprehension and analysis, the results obtained were plotted and compared graphically. In Figure 2, 3, 4 and 5 we present the results of these executions with the NSGA-II, SPEA2, Mocell, Random Algorithm and the ZAPROS III-*i*, where each graph represents a simulation of the problem with its solutions.

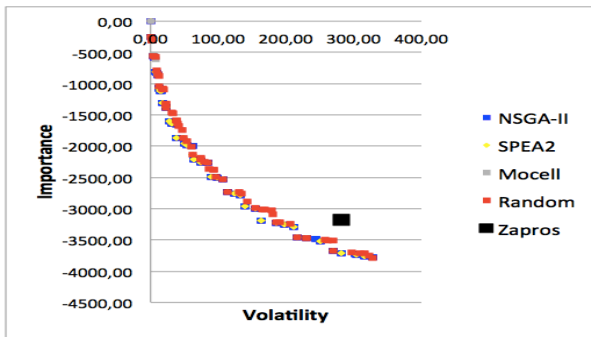


Figure 2. Result of file I.20.7.10.70

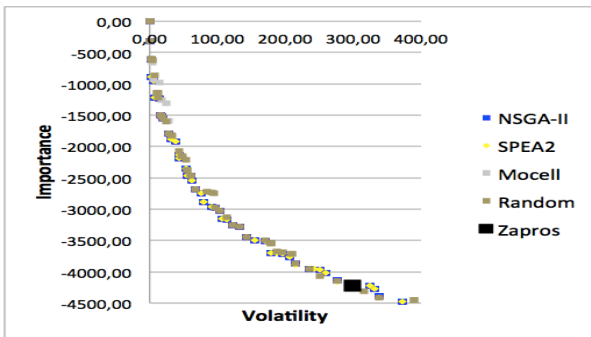


Figure 3. Result of file I.20.7.10.80

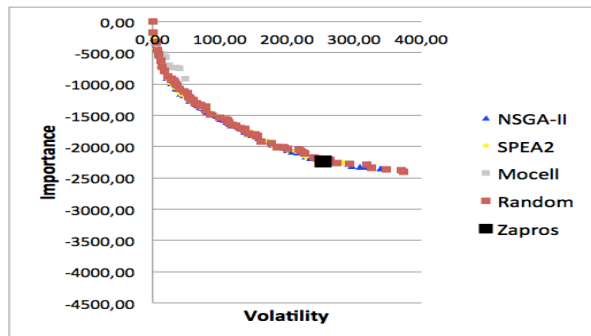


Figure 4. Result of file I.20.7.20.70

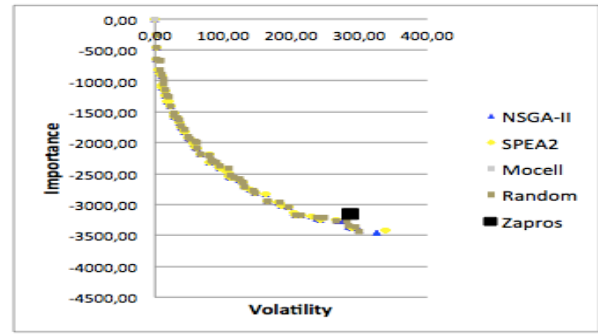


Figure 5. Result of file I.20.7.20.80

We can see five fronts of solutions. The Pareto fronts show a set points where each of them represent a possible ordering solution generated by each method. According to the legend of the graphs above, the colored dots correspond to the results obtained by the NSGA-II, SPEA2, Mocell, and Random. The black dots represent the solutions generated by ZAPROS III-*i*. Figures 2, 3, 4 and 5 represent solutions to problems with 20 requirements and 7 clients interested in these requirements. Figures 2 and 3 represent problems with 10% technical precedence between the requirements and figures 4 and 5, 20%. Accordingly, Figures 2 and 4 represent 70% of the budget available for the execution of the project, while Figures 3 and 5 represent 80%.

The stability criterion represented on the graph by the antonym (Volatility) was introduced in this work to assist the decision maker in choosing those more stable requirements to be implemented first. However, the solutions showed that the decision maker, in the tests applied, chose to value the importance that the important customer to the company gave to a given requirement. For this reason we see the points of ZAPROS III-*i* below the chart (valuing the client) and more right prioritizing more volatile requirements. This fact occurred in the four situations shown in Figure 2, 3, 4 and 5.

In Figure 2 and 3 we can see that the results obtained by NSGA-II and SPEA2 are superimposed. This fact occurs because of the similarity of the resolutive form chosen by these methods to solve the same problem.

Independent of the different situations, the solution generated by the ZAPROS III-*i* qualitative methodology was close to the solutions generated by the NSGA-II quantitative methods. This is a plausible fact if we consider the difference between the quantitative methods, adopted in NSGA-II, SPEA2, Mocell, and the qualitative methods, existing in ZAPROS III-*i*. This result shows that it is possible for qualitative methods to explore areas other than those already studied. The fact that a ZAPROS III-*i* solution, a decision maker's opinion, is close to solutions with complex search strategies for better solutions, shows that qualitative methods have a potential to solve this type of problem as well as metaheuristics They've been doing it for a while. In this case, the computational cost of the quantitative methods is replaced by the form presented to the decision maker, through the ARANAÚ tool that, as said, deals with the chosen alternatives and defines an order using the ZAPROS III-*i* method.

The solutions present in front of pareto can be used by a decision maker to facilitate the choice of the solution that is most appropriate for the proposed problem. The solutions found by the NSGA-II algorithm illustrate this Pareto front in Figures 3 and 4. As seen, the decision maker has a set of solutions to choose the one closest to his reality. The results obtained with the ZAPROS III-i qualitative method show very promising, since its solution was very close to the Pareto front, generated by metaheuristics that are quantitative algorithms. As in this case this solution was generated from information provided by the decision maker in a qualitative way, we conclude that this solution is what he really expected, since it is structured in real information provided by himself. In figure 3 and 4 the solution generated was equal to one of the solutions found by NSGA-II.

V. CONCLUSION AND FUTURE WORKS

The main contribution of this work is to apply a qualitative methodology structured in ZAPROS III-i Verbal Decision Analysis method to order software requirements and compare the solution generated with quantitative methodologies already known for doing this sort of ordering. The ARANAÚ tool provided support for this work, allowing good performance during testing and execution. As mentioned earlier, there are very few jobs that attempt to sort requirements using VDA methods. This work aimed to show that, although the results are not better than those obtained by the metaheuristics, the results obtained by ZAPROS III-i were very close to the front of the generated metaheuristic method.

In relation to the work developed by Barbosa [5], Brasil [16], and others to which these were extended, what was proposed in this work is a new methodology to prioritize software requirements using VDA giving a new alternative to the decision maker, which until then had only automated methods (metaheuristics).

As future work, we can increase the number of requirements to be sorted within the possibilities of VDA. This is a challenge because it is known that the problems solved by qualitative methods are limited in size due to the complexity of the methods themselves. To compare the results obtained by ZAPROS III-i with other metaheuristics that have populations of larger and more complex solutions. Increase the number of criteria to cover other types of problems related to ordering requirements. We can explore real-world problems by adapting the solutions according to the reality experienced by software development companies. We can apply other characteristics of the requirements, such as risk, estimated time for implementation and degree of complexity of the requirement.

ACKNOWLEDGMENT

The author Plácido Pinheiro is thankful to the National Counsel of Technological and Scientific Development (CNPq) for the support received for this project.

REFERENCES

- [1] DURILLO, J. J., NEBRO, A. J., LUNA, F., DORRONSORO, B., ALBA, E.: *jMetal: a Java Framework for Developing Multi-Objective Optimization Metaheuristics*. TECH-REPORT ITI-2006-10, Departamento de Lenguajes y Ciencias de la Computación, Campus de Teatinos, Universidad of Málaga, Malaga (2006)
- [2] RUHE, G., SALIU, M. O.: *The Art and Science of Software Release Planning*. IEEE Software, 47-53p (2005)
- [3] BECCENERI, J. C.: *Metaheurísticas e otimização*. r.t. lac, inpe. Computers and Operations Research (2007)
- [4] LARICHEV, O. I., Moshkovich, H. M.: *Verbal Decision Analysis for Unstructured Problems*, Boston: Kluwer Academic Publishers (1997)
- [5] BARBOSA, P. A. M.: *An Optimal-Based Approach to the Priorization of Software Re-quirements, Considering the Stability of the Requirement*. Master Thesis – Academic Mas-ter in Computer Science, Ceará State University (2013)
- [6] BAGNALL, A.J., RAYWARD-SMITH V.J., e WHITTLE I.M.:“The next release problem.”*Inform. and Soft. Techn.*,vol. 43, 883-890p (2001)
- [7] GREER, D., RUHE, G.: *Software release planning: an evolutionary and iterative approach*, *Inform. and Software Technology*, 243-253p (2004)
- [8] DEB, K., AGRAWAL, S., PRATAB, A., MEYARIVAN, T.: *A fast and elitist multiobjective geneticalgorithm:NSGA-II*.v.6,n.2,182-197p (2002)
- [9] OZERNOY, V. M., *Choosing the Best multiple criteria decision making method*. *INFOR*, v. 30, n. 2, p. 159-171 (1992)
- [10] TAMANINI, I., Pinheiro, P. R.: *Reducing Incomparability in Multicriteria Decision Analysis: An Extension of The ZAPROS Methods*, *Pesquisa Operacional* (Print), v. 31, n. 2, p. 251-270, DOI: 10.1590/S0101- 74382011000200004 (2011)
- [11] ZITZLER, E., LAUMANN, M., THIELE, L.: *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*. (2001)
- [12] BOUYSSOU, D., Marchant, T., Pirlot, M., Perny, P., Tsoukiás, A., Vincke, P.: *Evaluation and decision models: a critical perspective*. Boton: Kluwer Academic (2000)
- [13] FILHO, M. S., PINHEIRO, P. R., ALBUQUERQUE, A. B.: *Applying Verbal Decision Analysis to Task Allocation in Distributed Development of Software*. SEKE. DOI refer-ence number: 10.18293/SEKE2016-181 (2016)
- [14] LARICHEV, O.: *Ranking Multicriteria Alternatives: The Method ZAPROS III*. *European Journal of Operational Research*, v. 131, n. 3, p. 550-558 (2001)
- [15] TAMANINI, I., PINHEIRO, P. R., MACHADO, T. C. S., ALBUQUERQUE, A. B.: *Hy-brid Approaches of Verbal Decision Analysis in the Selection of Project Management Ap-proaches*. *Procedia Computer Science*, v. 55, p. 1183-1192, DOI <http://dx.doi.org/10.1016/j.procs.2015.07.093> (2015)
- [16] BRASIL, M. M. A., SILVA, T. G. N., FREITAS, F. G., SOUZA, J. T., CORTÉS, M. I.A *Multiobjective Optimization Approach to the Software Release Planning with Undefined Number of Releases and Interdependent Requirements*. *Lecture Notes in Business Inform. Proces*, v. 102, p. 300-314, 2012. DOI 10.1007/978-3-642-29958-2_20 (2012)
- [17] TAMANINI, I., PINHEIRO, P. R., MACHADO, T. C. S.: *Project management aided by verbal decision analysis approaches: a case study for the selection of the best SCRUM practices*. *Intern. Trans. in Oper. Res.*, v. 22, p. 287-312, DOI 10.1111/itor.12078 (2015)
- [18] BARBOSA, P. A. M., PINHEIRO, P. R., SILVEIRA, F. R. V., FILHO, M. S.: *Applying Verbal Analysis of Decision to prioritize software requirement considering the stability of the requirement*. 6th Computer Science On-line Conference 2017(CSOC) *Advances in Intelligent Systems and Computing*, Vol. 575. ISBN: 978-3-319-57141-6. DOI 10.1007/978-3-319-57141-6_45 (2017).
- [19] TAMANINI, I., PINHEIRO, P. R.: *Challenging the Incomparability Problem: An Approach Methodology Based on ZAPROS*. *Communications in Computer and Information Science* (Print), v. 14, p. 338-347, DOI 10.1007/978-3-540-87477-5_37 (2008)
- [20] TAMANINI, I., Machado, T. C. S., Mendes, M. S., Carvalho, A. L., Furtado, M. E. S., Pinheiro, P. R.: *A model for mobile television applications based on verbal decision analysis*, *Advances in Computer Innovations in Information Sciences and Engineering*1,1,399–404 (2008)
- [21] TAMANINI, I., MACHADO, T. C. S., PINHEIRO, P. R.: *Verbal Decision Analysis Applied on the Choice of Educational Tools Prototypes: A Study Case Aiming at Making Computer Engineering Education Broadly Accessible*. *International Journal of Engineering Education*, v. 30, p. 585-595 (2014)