ISSN: 0976-3104
ISSUE: Computer and Allied Sciences

ARTICLE

MULTI-CRITERIA SELECTION OF TESTING METHODS FOR SEPARATE SOFTWARE MODULES

Vladimir V. Lomakin*, Natalya P. Putivtseva, Tatyana V. Zaitseva, Olga P. Pusnaya

Institute of Engineering Technology and Natural Science, Belgorod State National Research University, Belgorod, 85 Pobedy st., 308015, Belgorod, RUSSIA

ABSTRACT

The paper presents the results of the testing methods selection for individual software modules using the ELECTRE method and the simplified analytic hierarchy process. The selection criteria were formulated, and the main approaches to testing separate software modules were considered as alternatives. It is planned to use the results obtained within the framework of the project to create high-tech production "Development of a methodology and tools for creating applications, supporting the life cycle of information technology and decision-making software for the effective implementation of administrative and management processes within the established authorities."

KEY WORDS
Software testing, multi-criteria selection, ELECTRE method, simplified analytic hierarchy process

Received: 3 Oct 2019
Accepted: 22 Nov 2019
Published: 4 Jan 2020

***Corresponding Author**
Email:
lomakin@bsu.edu.ru

INTRODUCTION

Software testing is a checking the correspondence between the actual and expected behaviour of the program, carried out using a finite set of tests selected in a certain way [1, 2].

You should test some small modules that form program before to start testing this program as a whole. The reasons for this approach are: 1) it becomes possible to control the combinatorics for testing; 2) the process of detecting the place of error and correcting the program text is facilitated; 3) concurrency is allowed, which makes testing several modules at the same time possible.

The testing of modules is mainly focused on the “white box” principle which is explained by the fact that the subsequent stages of testing are focused on detecting errors of various types not necessarily related to the program logic, but arising, for example, due to a program mismatching with the user's requirements. When testing separate modules of a software product, the best option for a number of criteria is to select a modified sandwich method.

There are six main approaches that can be used to merge modules into larger units [3].

1) Ascending testing. The program is assembled and tested from the bottom to up. Only the modules of the lowest level are tested in isolation, i.e. autonomously. Then, those modules are tested that directly call the already tested ones, but no longer autonomously, and together with the already tested lower-level modules. The process is repeated until a top point would be reached. This completes both unit testing and program pairing testing.

2) Top-down testing. The program is assembled and tested from top to bottom. Only the head module is tested in isolation. Then, the modules directly called by the head module are connected to it, one by one, and the resulting combination is tested. The process is repeated until all modules are assembled and tested. In the case when the module under test calls a module from a lower level, which does not exist at the moment, “faceplace” modules are programmed to simulate the functions of the missing modules.

3) Modified top-down testing. Using top-down testing, it is often impossible to test errors or exceptions, as well as defensive checks. The modified top-down testing method requires that each module undergo stand-alone testing before connecting to the program.

4) The "big jump" method. In accordance with this method, each module is tested autonomously. At the end of testing the modules, they are integrated into the system all at once. When using this approach, it is necessary to take into account that the modules are not integrated until the very last moment, serious errors in the interfaces may remain undetected for a long time, and “faceplates” and drivers are necessary for each module.

5) The sandwich method [4]. In this method, they simultaneously start the ascending and top-down testing, assembling the program both from below and from above and meeting somewhere in the middle. The method saves the beginning of system integration at an early stage. Since the top-point of the program comes into operation early, early on we get a working program framework. Since the lower levels of the program are created by the ascending method, the problems of impossibility are removed and some conditions are tested in the depths of the program.

IIOAB JOURNAL ISSN: 0976-3104

6) The modified sandwich method. When testing by the sandwich method, a problem arises that it is impossible to thoroughly test individual modules. In the modified sandwich method, the lower levels are tested strictly "from the bottom up" and the modules of the upper levels are first tested in isolation, and then assembled in a downward manner.

MATERIALS AND METHODS

We will solve the problem of choosing one of the software testing approaches using two methods: the ELECTRE method [5] and the simplified analytic hierarchy process [6].

Consider the solution of the problem by the ELECTRE method, which consists in the following. An integer w, characterizing the importance of the criterion, is assigned to each of the N criteria. It is assumed that the criteria weights are given by the decision maker. The index of agreement with the hypothesis of superiority of alternative x over alternative y is determined as follows:

$$C(x, y) = \sum_{i \in I^+ \cup I^-} \omega_i \Big/ \sum_{i=1}^{N} \omega_i \ ,$$

Where ω i is the weight of the i-th alternative.

The disagreement index d(x,y) with the hypothesis that alternative x is superior to alternative y is defined as follows:

$$d(x, y) = \max_{i \in I^-} \frac{F_i(x) - F_i(y)}{L_i}$$

Where Fi(x), Fi(y) are the estimates of the alternatives x and y according to the i-th criterion,
Li is the length of the i-th criterion scale.

The ELECTRE method sets a binary relationship of superiority between levels of agreement and disagreement. If $C(x, y) \geq \alpha$ & $d(x, y) \leq \beta$, where α and β are given levels of agreement and disagreement, then alternative x is declared to be superior to alternative y. A kernel of non-dominated elements is distinguished for a set of alternatives at the given agreement and disagreement levels. With decreasing α and increasing β, a smaller kernel (contained in the previous one) is identified in the given kernel, etc. An analyst offers to a decision maker a series of possible solutions to the problem in the form of various kernels. In the end, we can get one better alternative.

When solving the problem, the developed ELECTRE program was used. When the program starts, a window appears in which the user enters the number of alternatives and criteria, as well as their values. After entering all the parameters, the user clicks the "Remember" button and goes to the next tab, on which he/she needs to enter estimates of alternatives according to the criteria. After entering, the user must click the "Execute" button, and the program will calculate the values of the parameters L and ω. After that, the user switches to another tab. He/she calculates the agreement / disagreement indices there. Indexes are calculated by the program automatically if to click the "Calculate" button. After that, the tab "Matrix of values of a kernel" is switched on. Here on the left there are two fields where the user enters the coefficients α and β for calculating the kernel of solutions. Below the input fields there are two tables "Alpha" and "Beta", where the user is shown sorted lists of calculated agreement / disagreement indexes. After entering the coefficients, the user must click the "Find a kernel" button and the program will calculate the kernel of solutions using the binary superiority formula between the levels of contest (agreement) and disagreement. The calculations will be displayed on the left in the table, and the kernel of solutions, in which the best solutions will be displayed at the bottom for the given coefficients α and β. The user can change the values of the coefficients in order to achieve one best solution.

IMPLEMENTATION

After analyzing the software testing methods, we can identify the main criteria that are considered when choosing a method. When selecting the criteria, the features of the IC ISKU (developed as part of the joint activities of an enterprise and the educational institution of the Platform) were taken into account [7, 8]. The following criteria were identified: Cr 1 - Assembly; Cr 2 - Time until the appearance of a working version of the program; Cr 3 - Need for drivers; Cr 4 - Need for "faceplates"; Cr 5 - Concurrency at the beginning of work; Cr 6 - Ability to test individual paths; Cr 7 - Ability to plan and control the sequence. For convenience, their comparative analysis is presented in [Table 1].

IIOAB JOURNAL
ISSN: 0976-3104

Table 1: Comparative analysis of software testing methods for selected criteria

Criterion	Ascending Method (A1)	Top-Down Method (A2)	Modified Top-Down Method (A3)	Big Jump Method (A4)	Sandwich Method (A5)	Modified Sandwich Method (A6)
Cr 1	Early	Early	Early	Late	Early	Early
Cr 2	Late	Early	Early	Late	Early	Early
Cr 3	Yes	No	Yes	Yes	Partially	Yes
Cr 4	No	Yes	Yes	Yes	Partially	Partially
Cr 5	Medium level	Weak	Medium level	High	Medium level	High
Cr 6	Easily	Difficult	Easily	Difficult	Medium	Easily
Cr 7	Easily	Difficult	Difficult	Easily	Difficult	Difficult

When solving the problem by the ELECTRE method, initial values were entered (7 criteria and 6 alternatives), the input results are presented in [Fig. 2].

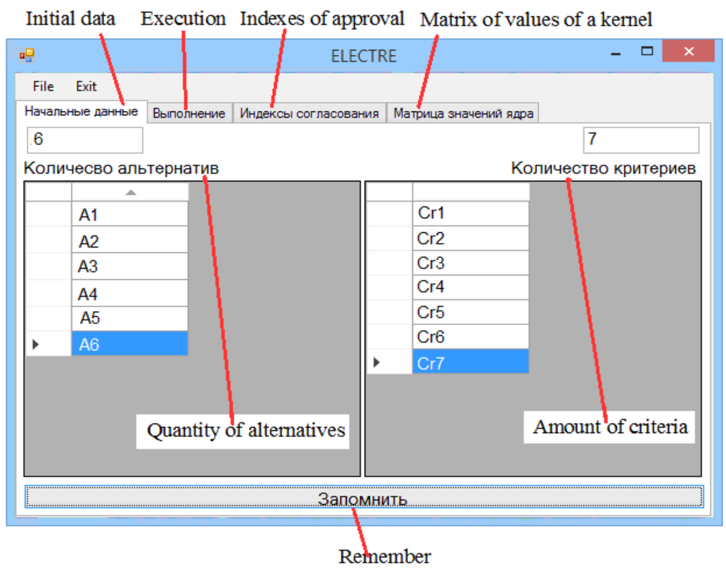

Fig. 2: Filling in the initial values.

Next, estimates of alternatives by criteria are entered [Fig. 3].

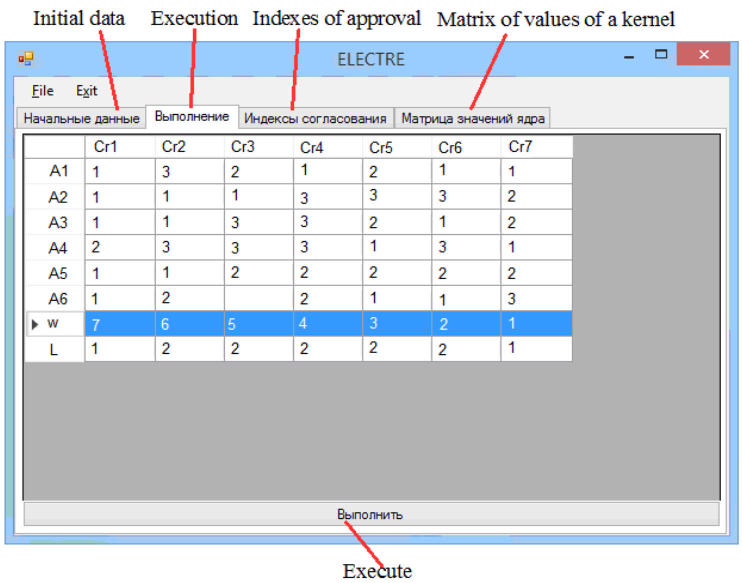

Fig. 3: Filling in the values for comparison of alternatives by criteria.

IIOAB JOURNAL
ISSN: 0976-3104

The following figure shows the calculated indices of contest and disagreement [Fig. 4].

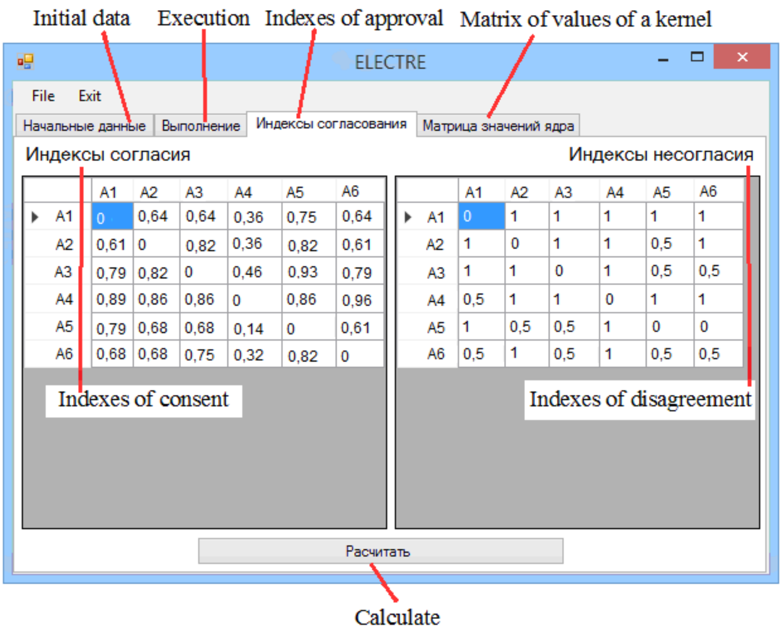

Fig. 4: Calculation of agreement and disagreement indexes.

Next, the values of the coefficients α and β are set [Fig. 5] and the alternatives included in the kernel of the solution are determined [Fig. 6].

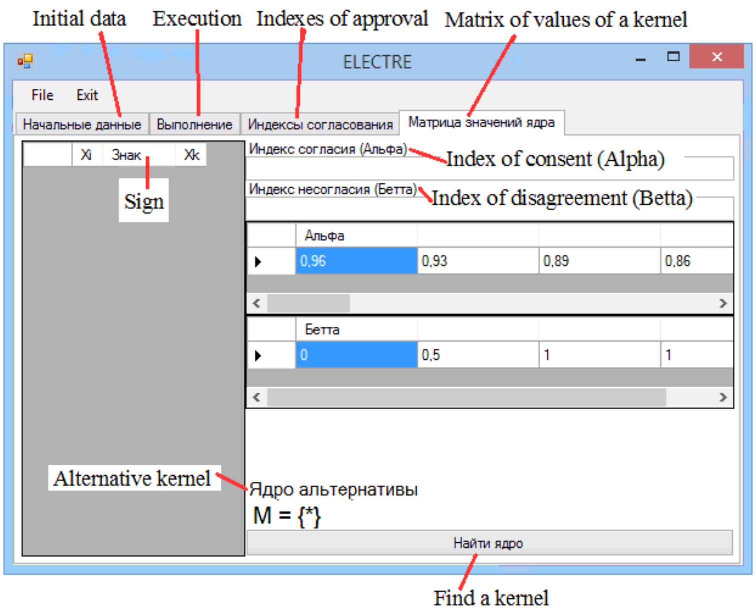

Fig. 5: Selection of coefficient values for finding the solution kernel.

Since, as a result of the experiment, two variants of the sandwich method (classical and modified) were proposed as the best alternatives, which only partially confirms the hypothesis, a second experiment was conducted using the simplified analytic hierarchy process to determine the best alternative according to the selected criteria.

The use of a simplified analytic hierarchy process was considered in [6]. When solving the problem with the simplified analytic hierarchy process, the "Simplified method" module of the program "System for calculating automation of the hierarchy analysis method" was used [9].

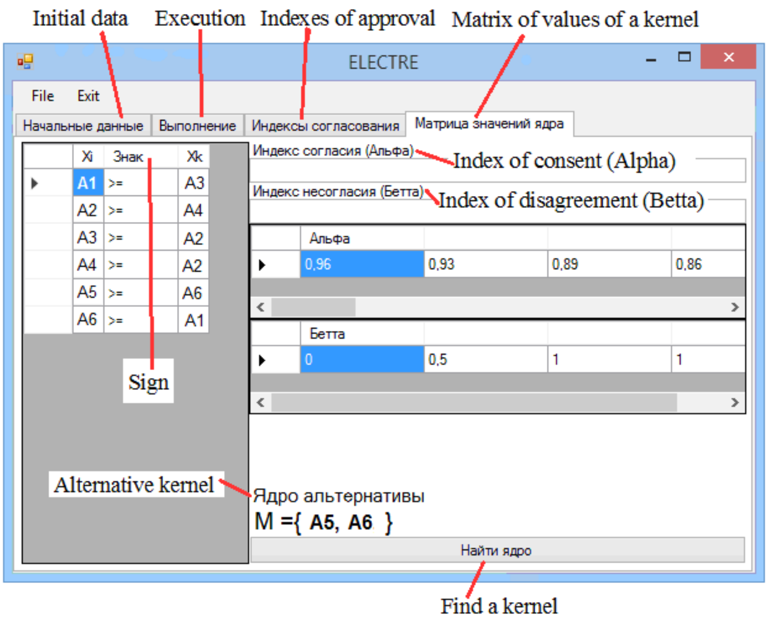

Fig. 6: Definition of alternatives included in the kernel solution.

In the first step, fill in the first row of the pairwise criteria comparison matrix, the remaining rows are filled automatically. Next, the pairwise alternative comparison matrices for each criterion are filled. Fig. 7 shows the result of filling in the pairwise alternative comparison matrix according to the first criterion, while the user filled in only the first row of the matrix, and the remaining values in the rows were calculated.

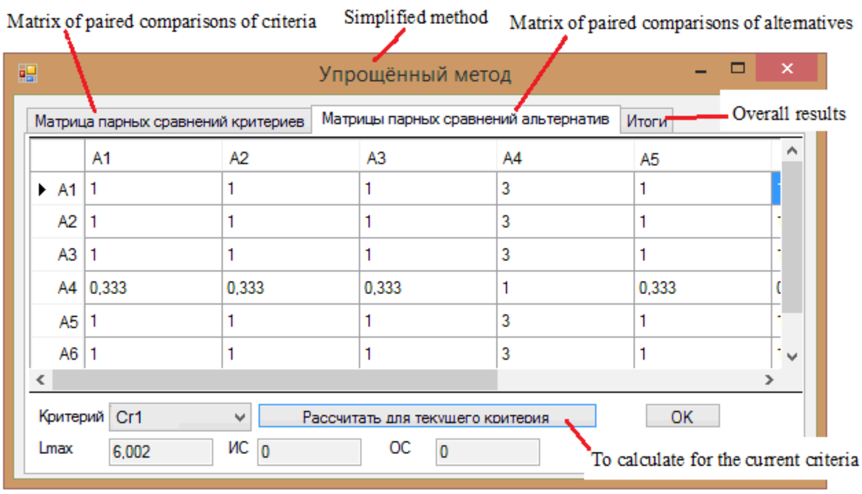

Fig. 7: Filling in the pairwise alternative comparison matrix by the criterion "ease of calculation".

After filling in all the matrices, the total weights of the alternatives were calculated, which is shown in [Fig. 8], and all summary data are presented in [Table 2].

Table 2: Summary table

Alternatives	Criteria							Weights of alternatives
	Cr1	Cr2	Cr3	Cr4	Cr5	Cr6	Cr7	
A1	0.19	0.07	0.1	0.36	0.14	0.25	0.3	0.1878
A2	0.19	0.21	0.4	0.09	0.05	0.06	0.1	0.1527
A3	0.19	0.21	0.1	0.09	0.14	0.25	0.1	0,1605
A4	0.06	0.07	0.1	0.09	0.27	0.06	0.3	0.1374
A5	0.19	0.21	0.2	0.18	0.14	0.12	0.1	0.1554
A6	0.19	0.21	0.1	0.18	0.27	0.25	0.1	0.192

IIOAB JOURNAL
ISSN: 0976-3104

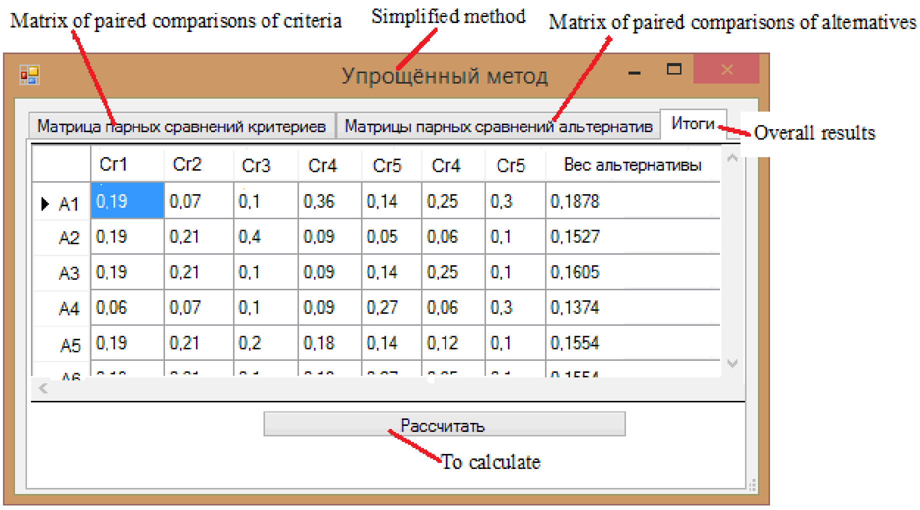

Fig. 8: Calculation of the total weights of the alternatives.

CONCLUSION

According to the ELECTRE method, the following results were obtained: the alternatives are ranked in descending order of importance: A5, A6, A1, A3, A2, A4; The kernel of solutions includes two of the most preferred alternatives: A5 - Sandwich Method and A6 - Modified Sandwich Method; since the ELECTRE method did not give an unambiguous result, and it is also impossible to establish the degree of priority 5 of the alternative over 6, the hypothesis is considered confirmed, since both sandwich methods are recognized as the best alternatives.

According to the simplified analytic hierarchy process, the following results were obtained: the alternatives are ranked in descending order of importance: A6, A1, A3, A5, A2 and A4; the best alternatives are A6 - Modified Sandwich Method and A1 - ascending testing; since the alternative A6 is the most preferable, the hypothesis is considered confirmed.

Thus, comparing the results of two experiments, we can conclude that when testing separate modules of a software product, the best option is to choose a modified sandwich method.

CONFLICT OF INTEREST

There is no conflict of interest.

ACKNOWLEDGEMENTS

Completed as part of implementing a comprehensive project on creation of high-tech production “Development of a methodology and tools for creating applications, supporting the life cycle of information technology and decision-making software for the effective implementation of administrative and management processes within the established authorities”, 2017-218-09-187; Decree of the Government of the Russian Federation dated April 9, 2010 No. 218.

FINANCIAL DISCLOSURE

None.

REFERENCES

[1] IEEE Guide to Software Engineering Body of Knowledge, SWEBOK, [2004].

[2] Software testing - basic concepts and definitions [Electronic resource]. - Access mode: http://www.protesting.ru/testing/ (accessed September 13, 2019)

[3] Soloviev SV, Tsoy RI, Grinkrug LS. [2011] Application software development technology, Academy of Natural Sciences. RU

[4] Sandwich Testing [Electronic resource]. [2019] Access mode: https://www.geeksforgeeks.org/sandwich-testing-software-testing/

[5] Larichev OI. [2000] Theory and decision-making methods, as well as the Chronicle of events in the Magic Countries: A Textbook, 229.

[6] Nogin VD. [2004] A simplified version of the simplified analytic hierarchy process on the basis of nonlinear convolution of criteria, Journal of Computational Mathematics and Mathematical Physics, 44(7):1261–1270, Comput Math Math Phys, 44(7):1194–1202.

[7] Putivtseva NP. [2017] Solving the problem of choosing Russian corporate information systems using the analytic hierarchy process. Bulletin of the Voronezh State University, Series: System Analysis and Information Technologies, 4:85 - 91.

[8] Zaitseva TV. [2018] To the question on the composition of an integrated high-level development tools complex and the corporate-level information system functioning environment. Collection of articles of the XV International Scientific and Practical Conference, Moscow: “Scientific and Publishing Centre, 81-82.

[9] Calculation automation system for the hierarchy analysis method: Certificate on registration of databases and computer programs No. 2018615850 dated 05/17/2018.