BABES-BOLYAI University of Cluj-Napoca
Faculty of Mathematics and Computer Science
Proceedings of the National Symposium ZAC2018
(Zilele Academice Clujene, 2018), p. 0-0

Introspection: Metrics-based Assessment for Component-based Systems. Impact and Future Directions.

by
A.Vescan and C. Șerban [1]

Abstract. *An introspection of our previous research approaches is conducted with the aim to identify the way they are linked together, their dependencies and connections. This paper also identifies various viewpoints of the investigated problems.*

Another aspect refers to the impact of our research, reflected by the citations of our work. Analyzing those approaches and their relations with our proposals, we have endeavored to formulate forward/future research objectives

Keywords: components, metrics, assessment

1. Introduction

Introspection helps us in many ways. Introspection would lead you to knowing yourself in a better way. You can find a mistakes you have made in life and can help you to learn from your mistakes. It would draw your attention towards the things which can be improved in ourselves.

The definition of *introspection* provided by Cambridge dictionary [1] is ”examination of and attention to your own ideas, thoughts, and feelings” and by Merriam dictionary [2] is ”a reflective looking inward: an examination of one’s own thoughts and feelings”.

Following the above definitions and benefits, a research introspection is important to be considered. Thus, this study has been taken as a research introspection and research impact of the authors. The goal of the paper is to analyze authors previous research proposals, the way they are linked together, their dependencies and connections, as well as to study its impact on the literature.

[1] UBB Cluj-Napoca, *camelia/avescan@cs.ubbcluj.ro*

Another key aspect that will result after the analysis of our research and its impact on the literature refers to future directions: enunciation of new themes and research objectives in the future. This is absolutely necessary, without such an analysis the efforts have no continuity, everything reducing to a leap research.

The authors have studied in parallel, Component Based Programming and Software Metrics. Having discussions regarding their research, questions raised in their mind: How could be measured if the system being built meets the requirements? How can we assess its quality? How can we quantify the reusability of a component? As a response in this direction, software metrics came as an immediate solution therefore this was the first intersection point of the authors' research work.

A lot of problems is encountered when a metric-based approach is used and our research work tried to identify solutions for some of them. For instance, the lack of standard terminology and formalism in the definitions of metrics hampers their applicability. To overcome this, metrics for Component Based System and Object Oriented Design were defined in a formal manner using mathematical notations from the theory of sets and algebraic relations. Another important problem regarding a metric-based approach is the interpretation of the obtained measurements results. In this respect, two conceptual frameworks for both Object Oriented Design and Component Based Systems were proposed.

The rest of the paper is organized as follows. Section 2 briefly discusses the main investigated topics from this paper, i.e. Component Selection Problem and Metrics-based Software Assessment. The detailed description of the proposed introspection is presented in Section 3. Four research questions regarding our previous research proposal were formulated as research method, the authors trying to argue the responses in a well documented manner. Section 3.2 described the impact of our research work, whereas Section 4 concludes the paper.

2. Investigated Research topics

As we have mentioned earlier, we started our research in two subdomains of Software Engineering, Component Based Systems and Metrics for Object Oriented Design. We have approached these two roads for a while until we have reached an intersection where the two roads have been united. We have noticed that in order to build a component-based system, the assessment of the components in isolation and into the assembly is a necessity. Additionally, when we select a component to integrate into an assembly is important to have a pre-classification of those components, classification based on some defined measurements.

Have all the above mentioned into account, we could define a taxonomy for our research investigated problems as follows:

- Component Selection Problem – Problem
    - ∗ Single criterion
    - ∗ Multi criteria
  - Genetic algorithm
    - ∗ Single-objective (weights)
    - ∗ Multi-objective (Pareto front)
- Metrics-based Software Assessment
  - Component Systems Assessment
    - ∗ Component Assessment in Isolation
    - ∗ Component Assembly Assessment
    - ∗ Component Classification based on metrics (fuzzy analysis)
    - ∗ Conceptual framework for Component Based System Assessment
  - Object Oriented Design Assessment
    - ∗ Design flaws detection based on metrics
    - ∗ Design entities classification (fuzzy analysis)
    - ∗ Formal Definition of Metrics
    - ∗ Conceptual framework for OOD Assessment

This taxonomy of our research investigated problems can be seen as a map that describes the roads that we have paved in our research, their intersections and unifications, but most importantly, drawing this map through this introspection brought us to light new roads/pathways which we could explore to continue our research.

3. Introspection of research

This section presents the questions that we aim to answer by the introspection and the results revealed.

3.1. **Application methods.** The steps of the introspection and impact research analysis are documented below.

Our introspection aims at answering the following stated questions:

Q1: How can our (current) research on software component selection and metrics-based software assessment be classified?

Q2: What is the current state of software component selection and metricsbased software assessment research with respect to this classification?

Q3: What is the contribution of the authors’ s research work in the field of software component selection and metrics-based software assessment? This question leads to another sub-question: What are the citation of the authors’ s previous work?

Q4: What can be learned from the current research results that will lead to topics for further investigation?

3.2. **Introspection Results.** This section presents the results of our research instrospection.

Next, we provides for each stated question the revealed outcome, emphasizing the key elements characterizing our approaches and perspectives of the investigated problems.

3.2.1. *Research classification - response for Q1.* This section answers/responds to the first question: Q1: How can our (current) research on software component selection and metrics-based software assessment be classified?

Our research activities focused on notions regarding component selection and composition and metrics-based software assessment. We have considered joining the research, thus applying the findings of object-oriented assessment based on metrics to the component selection and composition.

This section presents the classification of our research work based on various criteria. The authors of the current paper have published multiple research papers in various national and international conferences and journals. Papers that this research considers are presented in Table 2.

We have considered for the analysis our papers published between 20062017. Four major research directions intertwine with these articles to a certain extent. Therefore, we aim to define classification of these papers, considering as classification criteria the following four research directions:

(1) Various mixt approaches

(2) Genetic algorithms
(3) Fuzzy clustering
(4) Metric-based assessment

Figure 1 contains a pie chart to illustrate numerical proportion for each identified method: various genetic algorithms and methods that use metricsbased assessment were most used in our research. A more detailed and complex analysis is following in the next sections, emphasizing various concepts used in our approaches and various metrics and perspectives used.

Figure 1. Classification of our published articles based on used method

3.2.2. *Current State Research - response for Q2.* This section answers to the second question: Q2: What is the current state of software component selection and metrics-based software assessment research with respect to this classification?

Various proposed approaches

In order to answer to the second question we need first to analyze the previous published approaches. Considering the classification discovered in the previous section, we first investigate the chronological publications for each used method. As seen in Figure 2, most of the publications concentrate around Genetic algorithms and Metrics-based assessment, around 2008-2010 and then 2015-2017. As a remark, it is worth mentioning that those dates are related to our PhD program and to our interleaved parental leaves from 2009 to 2014.

Figure 2 contains the published articles in chronological order group by used method in the approach.

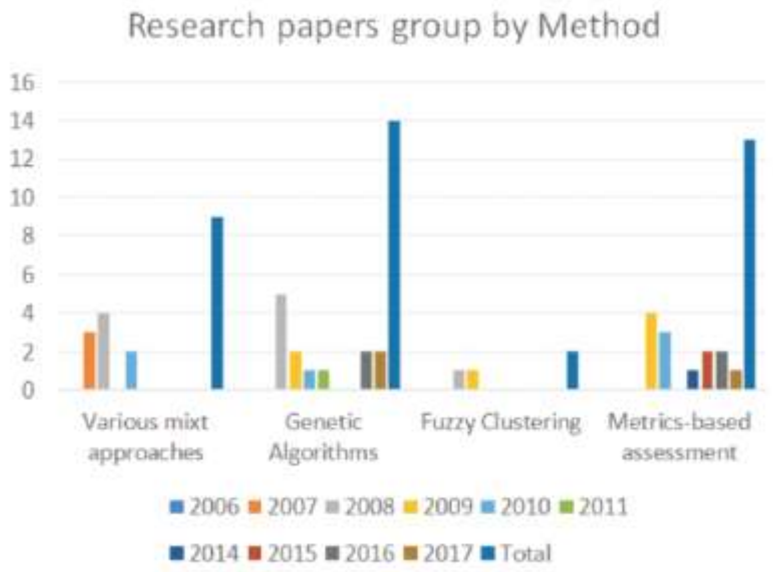

Figure 2. Published articles in chronological order group by used method in the approach

In order to study the relations between the four basic approaches we will next construct a concept matrix, mapping for each article the main considered concepts. But first, how we have identified the main concepts that characterize our approaches? We have scanned the papers and identified, other than the domain related concepts (as provided/required interfaces or software quality attribute, ect), several key elements that define particularly each approach:

- For Genetic algorithms-based approaches
  - Chromosome representation
    - ∗ Based on requirements
    - ∗ Based on components
- For Metrics-based assessment approaches
  - Metrics design for
    - ∗ A component
    - ∗ The assembly/composition of components
  - Fuzzy clustering
    - ∗ Same partition
    - ∗ Changed partition
- For both approaches (Genetic algorithms and Metrics-based assessment)
  - provided/required interfaces

- dependencies between requirements
- formal model/meta-model for object-oriented/component-based systems
- multilevel structure
- dynamic composition

Table 1 contains the identified concepts used in the papers being analyzed. The concepts are drawn from both Component Selection Problem perspective and Metrics-based software assessment and are presented in Table 2 for each studied paper.

Concept Identifier	Concept Name
C1	Provided/required Interfaces
C2	Dependencies between requirements
C3	Chromosome representation-on requirements
C4	Chromosome representation-on components
C5	Metrics-for a component
C6	Metrics-for assembly of components
C7	Fuzzy-same partition
C8	Fuzzy-changed partition
C9	Software quality
C10	Formal model for OOD/CBD
C11	Multilevel structure
C12	Dynamic components/dynamic requirements

TABLE 1. Concepts Identifiers

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
[8]				✓								
[21]		✓			✓							
[20]	✓					✓						
[27]		✓										
[3]	✓											
[36]		✓	✓	✓								
[34]		✓	✓									
[33]				✓								
[29]		✓	✓									
[28]	✓						✓					
[24]		✓										
[38]		✓										
[31]			✓									
[37]	✓	✓										
[25]							✓	✓				
[15]									✓			
[14]									✓			
[30]		✓	✓									
[32]			✓									
[23]										✓		
[22]		✓								✓		
[16]										✓		
[35]	✓			✓							✓	
[44]				✓								✓
[6]	✓	✓				✓	✓					
[13]									✓			
[26]									✓			
[12]									✓			
[41]	✓			✓							✓	✓
[40]	✓			✓							✓	✓
[42]					✓	✓					✓	
[39]	✓			✓							✓	✓
[43]				✓	✓	✓					✓	✓

TABLE 2. Concept Matrix

From a in depth analysis we can see that the concepts used in various approaches are intertwined, for example dependencies between requirements, metrics for a component and metrics for the assembly of the components. Moreover some of the concepts are first used in some approaches and then a combination of several ones are proposed in later proposals: multilevel and dynamic concepts are first introduced in two different papers and later combine in a single approach.

Considering as *criteria* the *type of dissemination* used, i.e. conferences or journals, we have discovered that most of our publications (that we considered in this study) are published in various national and international conferences (21 papers out of 33 considered), few are published directly in journals (6 articles), and the rest are disseminated at conferences but published in journals.

Considering as *criteria* the *used validation*, several of our approaches were validated using comparison with baseline heuristic algorithms ([34], [29], [31], etc), several were validated using comparison with random search ([41], [39], etc), some approaches used internal comparison (by defining metrics for comparison, [33]), and few considered statistical analysis, i.e. Wilcoxon signed ranks test ([41], [40]).

Considering as *criteria* what *approach was used to validation*, we have noticed that some approaches used examples ([21], [27], etc), some other approaches used Experiments and various comparison analysis ([36], [31], [30], etc), other approaches used academic Case studies.

In what follows we will focus on analyzing the two major perspectives: Genetic algorithms-based approaches and Metrics-based assessment approaches. **Approaches using Genetic algorithms**

One of the first proposed approaches related to component composition is the proposal from [8] by introducing a new computational intelligence-based method for component composition analysis. Starting from a automaton-based model and using integration properties we develop a genetic algorithm to analyze the component composition process. Given a set of components to be integrated into a system we evolve the solution (the component-based system(s)) using genetic algorithms. Some properties for the correct composition of components and component call dependencies between components were encoded into the evaluation of a chromosome. Next, on paper [27] we introduced temporal composition restraints to assist the assembly process when selecting the next feasible component to integrate into the final system.

The paper [3] defines formerly the Component Selection Problem (CSP): choosing the minimum number of components from a set of components such that their composition satisfies a set of objectives. The paper outline an algorithm for component selection: the component with the maximum number of provided operations that are needed by the final system is considered at each step of the algorithm.

Dependencies in the CSP were first introduced in paper [28] and were further integrated in all approaches that followed.

Evolutionary approaches are next used for the CSP: various representations were used (requirements-based chromosome representations and componentsbased chromosome representation) and two approaches for the optimisation problems were used (weighted sum method and Pareto principle). The published papers considered various combinations of requirements/componentsbased representations and optimization problems ([36], [34], [29], [33]). Extensions and in depth analysis of the genetic algorithm using weighted sum method and requirements-based representation was provided in paper [30].

Paper [31] formulated the problem as multiobjective, involving 2 objectives: the number of used components and the cost of the involved components. We use the Pareto dominance principle to deal with the multiobjective optimization problem. The approach used is an evolutionary computation technique (a steadystate evolutionary algorithm).

The first paper that fusioned the genetic algorithm with the metrics was [32], the approach considering various experiments in which different metrics for the involved components were considered.

Paper [35] approached the multilevel CSP. All the previous approaches did not considered the multi-level structure of a component-based system. They all constructed the final solution as a one level system, but components are themselves compositions of components. This give rise to the idea of composition levels, where a component on level i may be decomposed (using more components) at level i + 1 or compositions at level i + 1 serves as component at level i. We have proposed a multiobjective evolutionary approach, 4 objectives being involved. For each level (a compound component) of the final system we have applied the algorithm and the solution corresponding to that level was obtained. For a level we have compared the best fitness values using two solution and because they are overlapping we could conclude that the same best solution may be obtained starting from the same components repository

Our next approach [[44] was to investigate the problem in an dynamic or changing environment for which two practical scenarios are envisaged: the repository containing the components varies over time and the system requirements change over time. The algorithm employed uses a combination of evolutionary algorithms and repair mechanism. The idea behind this was to avoid restarting the algorithm from randomly generated solutions and to make use of the ones found at the previous step.

The new concepts related to the CSP that were introduced in [35] and [44] were next merged into a new approach discussed in papers [41], [40] and [39]. The approach dealt with the component selection problem with a multilevel system view in an dynamic environment. We approached the problem as multiobjective using the Pareto dominance principle. The model aims to minimize the cost of the final solution while satisfying new requirements (or having available new components) keeping the complexity of the system as minimum as possible (in terms of used components).

Our latest approach [43] considered for the CSP a combinations of multilevel with functional/nonfunctional requirements and architecture evaluation based on metrics. As advantages over the existing literature we mention: the automatic computation of component interactions and of the constituent components for each module, multiple solutions obtained in a single run, and the new architecture evaluation step based on metrics values that is not present in other approaches.The decomposition offers valuable insights about internal structure of the system which led us to identify metrics to assess coupling and cohesion of the architecture design. The internal structure influences the external quality; thus, a highly cohesive module exhibits high reusability and loosely coupled systems enable easy maintenance. In this context, when selecting the best solution out of a set of available ones, we aim to maximize the cohesion of modules and to minimize the coupling between them, obtaining thus the best reusable and maintainable solution.

To conclude, our proposals for the Component Selection Problem started and considered the key elements that define the building block, i.e. the component: provided and required interfaces, dependencies between components, metrics to assess either the component in isolation or the component in composition. Other specific characteristics that was used in our proposals was the multilevel structure of the final system and the dynamic changing of both components from the repository or the requirements of the system.

Approaches using Metrics-based assessment

Our starting point on metrics based component systems assessment was highlighted in the paper [21]. We have studied metrics for components from the perspective of component assessment. Two new metrics for the assessment of coupling between components were proposed. The influence of metrics values on software components quality attributes such as reusability, modularity, understandability, testability was also investigated in this paper.

In order to select a component to integrate it in an assembly, aspects such as provided/required functionalities are essentials. Metrics that quantify these aspects were presented in paper [20] addressing some quality attributes being of interest for the assembly evaluation. Two new metrics regarding the component coupling grade were also proposed. The paper discussed the influence of metrics values on quality attributes.

The problem of selecting the best candidate from a set of available components was discussed in paper [24]. Fuzzy clustering analysis was used to classify the components into different groups based on the values of metrics. Four metrics were considered for the analysis.

The paper [19] used metrics and fuzzy clustering analysis in order to detect object-oriented design flaws. A suite a papers [16], [23], [14][17] [18] followed after [19] with the goal to define a quantitative evaluation model for design assessment. A suite of the most used metrics for object-oriented design were formally defined. Papers [17] [18] present an experimental evaluation of our proposed methodology for object-oriented design (OOD) assessment, comparing the proposed approach with related approaches based on detection strategies.

Having into account the analysis proposed in [24], the paper [25] proposed a new algorithm for constructing a software system by assembling components. The process of selecting a component from a given set takes into account some quality attributes. Metrics are defined in order to quantify the considered attributes. Using these metrics values, a fuzzy clustering approach groups similar components in order to select the best candidate.

Another important perspective that we approached in our previous work regarding component based systems assessment was to define a conceptual framework for component-based system metrics definition [22]. The paper [23] continued the work from [22] defining *A Formal Model for Component-Based System Assessment*. The idea to use metrics for the assessment of CBS and to define a formal approach in this context, came up from some ours previous papers [20], [24] [19] , [15], [14], [16] some of them approached the problem of metrics based assessment for object-oriented design. Trying to identify similarities between OOD and CBS, we have adapted some metrics from OOD to CBD, but at the same time we identified software components and the CBS specificities for building a new model for CBS assessment. Starting with the papers [26], [12], [13] we aimed to extends out research idea regarding metrics based assessment. In this respect, an ontology was proposed in paper [13]. This paper gave us an insight into applying the approach for an CBS also.

Table 3 emphasizes, taking into account both perspectives of our approaches for software systems assessment - object oriented design and component based systems - the context used. As it can be seen, this context is built upon some metrics used in the assessment, different criteria, aspects and concepts that appear in metrics definitions or are related with the proposed assessment, internal and external quality attributes correlated with the selected metrics. All these elements defined the contextual framework of our proposed assessment approaches based on Metrics.

Context	Papers	Metrics used	Aspects used in metrics definition or related with the assessment	Internal quality attributes	External quality attributes
CBS	[21]	CBC DDT BDT	Assembly Dependencies Tree.	Coupling	Adaptability
	[20]	Cost	Provided/Required Service	Cohesion	Functionality
	[24]	PSU RSU	A meta-model for component based systems		Portability
	[22], [23]	IDC IIDC	Design Assessment Objectives		Customizability
	[42]	OIDC	Fuzzy partition of software components		Understandability
	[43]	CPSU CRSU			Interfaces Complexity
	[41]	AIDC			Reusability
	[40]	Func			Usability
OOD	[19]	CBO	A meta-model for object-oriented design systems.	Coupling , Cohesion, Complexity	Maintainability,
	[14]	WMC	Design flaws: God Class, Shotgun Surgery	Proper data abstraction	Reliability
	[15]	ATFD	Design principle and heuristics: high coupling, low cohesion, manageable complexity, proper, proper data abstraction		Readability
	[16]	LCOM	Fuzzy partition of design entities		Understandability
	[17] [18]	TCC	Ontology		Testability
	[12] [39] [13]				

TABLE 3. Metrics-based assessment approaches - used metrics and various aspects of them, internal and external quality attributes and metrics related to them

3.2.3. *Impact of the Research (response for Q3).* This section answers to the third question: Q3: What is the contribution of the authors s research work in the field of software component selection and metrics-based software assessment? This question leads to another sub-question: What are the citation of the authors s previous work?

To study the impact of the proposed approaches we first investigate the citations (Conference classified according to http://portal.core.edu.au and Journals according to Impact Factor IF/AIS). Thus, we have considered only A*, A, B and C classifications for conferences and A*, A, B, C for journals, due to the impact of those conferences and journals to the research advancement (most of our other papers and citations being ranks as D). Figure 3 contains some of our papers and the citations for them: most of our work from 2008 have citations in 2013 and more recently in 2016 and 2018. Most of papers that cite our work regarding CSP are in Journals of category A (5A) and B (4B) and some are in conferences (1 of rank A and 4 of rank C).

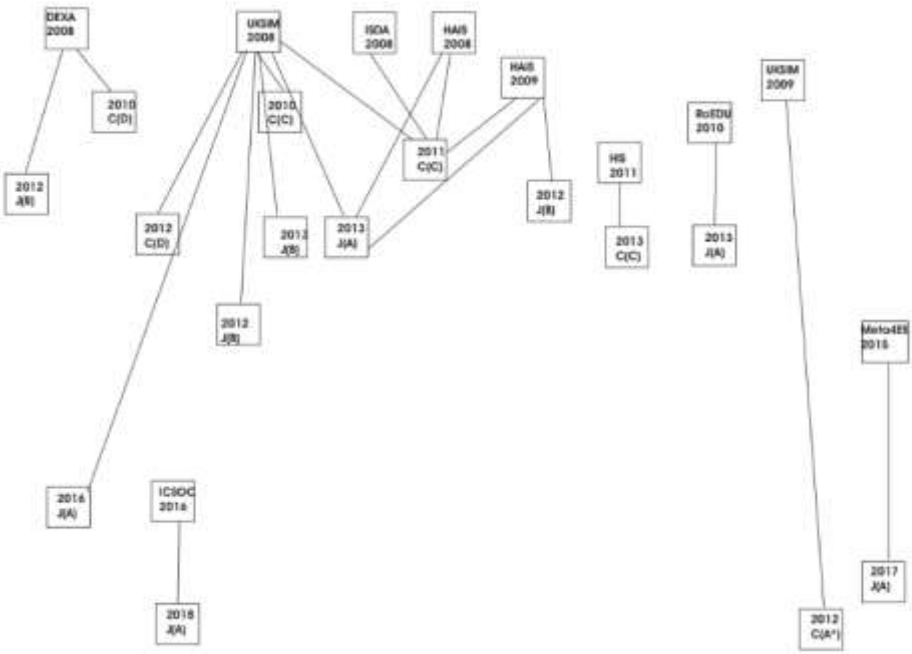

Figure 3. Our Papers and their Citations.

From Figure 3 we can observe that the approach discussed in UKSIM 2008 was most cited: Pareto-dominance based approach for the CSP. Also, it worth mention that the approach from UKSIM 2009 was recently cited in an prestigious international conference of rank A*. Another aspects that is important is that our latest approach published in a rank A conference in 2016 is cited in 2018 in an rank A journal, thus emphasizing the fact that our work gets immediate attention.

Regarding the metrics-based assessment we have identified several citations. Paper [22] was cited in 2013 in a rank A journal. Paper [25] obtained two important citations, one of them in an prestigious international conference of rank A*(2013) and the other one in a rank B journal (2012). These important references of our [25] paper came from Software Engineering domain. The paper [13] is also cited by a prestigious rank A journal in 2017.

An important aspect regarding the impact of our proposed approaches refers to measure to what extent some of our concepts were used by other researchers. Investigating the papers that cited our work we have discovered that:

- The approach in paper [11] also used dependencies between requirements and the weights for various characteristics of components.

  They proposed another strategy, i.e. to shortlist the candidate components.
- The proposal from paper [7] also used cost and other non-functional attributes to assess the architecture of the software. They also use similar notion for our dynamic changing, i.e. evolution of software architecture.
- In paper [5], the authors used of nonlinear regression analysis to extract the dependencies of the system, and various scenarios were considered when selecting the components.
- Paper [4] investigates what metrics are used for the CBSS level from specification. They make a distinction between Component evaluation versus CBSS evaluation.

An in depth analysis of our proposed research and of the research citing us will be conducted in a future paper.

3.2.4. *Recommendations for future research directions (response for Q4).* This section answers to the fourth question: Q4: What can be learned from the current research results that will lead to topics for further investigation?

Having scanned and analyzed our previous work presented in the papers discussed above, we drew out new ideas that we aim to approach in near future. Some of them are discussed in what follows.

The problem investigated by our research has its origin in the paper [10] and our first approach considered two kind of compositions (parallel and serial) in order to discover, based on interdependencies, how to link the components [9] This first rustic, based approach was later investigated and using the notation, the Component Selection Problem was defined. The problem was addressed using various methods; various representations and perspectives were considered when solved by the Genetic algorithms approach. Different representations ([36], [34], [29], [33]) for the chromosome were considered, emphasizing the benefits of one representation over the other. Also, regarding the optimization problem two approaches were used: weighted sum method and Pareto principle. The multilevel structure of the component-based system was considered and studied in our approaches [35] . A more realistic view was also considered when investigating the problem in a dynamic or changing environment [44] for which two practical scenarios were stipulated: changing the components in the repository or changing the requirements of the final system over time. A combination of multilevel and dynamic changing was proposed in [41], [40] and [39]. Our newest approach [43] considered for the CSP a combinations of multilevel with functional/nonfunctional requirements and architecture evaluation based on metrics. *Analyzing the previous proposals and studying the advantages and drawbacks of each of them, the distinct elements of each approach and the particularities of them, we will bring forth new approaches that will combine the elements that have been made improvements for the defined problem: using metrics values when assessing a component from the set of candidate components in the genetic algorithm approach, applying various strategies for selecting next population to find better results at either every level or at the final obtained system.*

Starting from the papers [20], [CimSim2010] we discussed the importance of studying the influence of metrics values on quality attributes of software components. To add more clarity and rigorosity regarding the study of metrics values influence on quality attributes a novel quality evaluation methodology based on ontologies was defined in paper [13]. The major contribution of this paper is conceptualising the elements of the ISO25010 software quality model and their relations to object oriented metrics into an ontology, and a proposal for software quality assessment based on this ontology. This work was carried on with the paper [12] studying the influence of four important objectoriented design metrics on maintainability quality attributes. The metrics considered for the analysis were related to four important characteristics of an good object oriented design: high coupling, low cohesion, manageable complexity and proper data abstraction. The model is using ISO25010 standard for software quality evaluation, and includes all eight characteristics with their corresponding subcharacteristics. In our study, we focused on maintainability, whose subcharacteristics are: modularity, reusability, analysability, modifiability and testability. *Having analysing our contributions on the papers mentioned above we can draw the conclusion that the proposed methodology for the assessment of object oriented design systems needs to be extended for component based systems as well. There is a lack of formalisation regarding the assessment of a component based system, as far as we know, and we believe that a conceptualisation based on ontologies would need to exist.*

4. Conclusion and Future Work

An introspection of our previous research approaches was conducted with the aim to identify connections between various perspectives and to formulate forward/future research objectives. So far, we channeled our efforts in the assessment of both object oriented design and of component based systems, trying to define new approaches based on metrics in order to quantify different aspects that defined the assessment objectives. In what follows, we also aim to focus our research in predicting the quality attributes of a software components or of an object oriented design, prediction based on some machine learning methods.

References

[1] https://dictionary.cambridge.org/dictionary/english/introspection.

[2] https://www.merriam-webster.com/dictionary/introspection.

[3] H.F. Pop A. Vescan. Automatic configuration for the component selection problem. In *roceedings of the 5th international conference on Soft computing as transdisciplinary science and technology (Workshop on Autonomous and Autonomic Software-based Systems (ASBS))*, pages 479–483, 2018.

[4] Majdi Abdellatief, Abu Bakar Md Sultan, Abdul Azim Abdul Ghani, and Marzanah A. Jabar. A mapping study to investigate component-based software system metrics. *Journal of Systems and Software*, 86(3):587 – 603, 2013.
[5] A. Agarwal, G. L. Hamza-Lup, and T. M. Khoshgoftaar. A system-level modeling methodology for performance-driven component selection in multicore architectures. *IEEE Systems Journal*, 6(2):317–328, June 2012.
[6] A.Vescan and C. erban. Metrics based component systems assessment. a survey. In *Proceedings of the National Symposium ZAC2014*, pages 13–10, 2014.
[7] Vittorio Cortellessa, Raffaela Mirandola, and Pasqualina Potena. Managing the evolution of a software architecture at minimal cost under performance and reliability constraints. *Science of Computer Programming*, 98:439 – 463, 2015.
[8] A. Fanea and L. Diosan. Computational intelligence-based model for component composition analysis. In *Proceeding of the International Conference on Computers, Communications and Control, ISSN: 1841-9836 (indexed ISI-SCI-E)*, pages 474–479, 2006.
[9] A. Fanea and S. Motogna. A formal model for component composition. In *Proceedings of the Cluj Academic Days National Symposium*, pages 160–167, 2004.
[10] Michael Roy Fox, David C. Brogan, and Jr. Paul F. Reynolds. Approximating component selection. In *WSC '04: Proceedings of the 36th conference on Winter simulation*, pages 429–434. Winter Simulation Conference, 2004.
[11] Muhammad Ali Khan and Sajjad Mahmood. A graph based requirements clustering approach for component selection. *Advances in Engineering Software*, 54:1 – 16, 2012.
[12] S. Motogna, A. Vescan, C. Serban, and P. Tirban. An approach to assess maintainability change. In *2016 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pages 1–6, 2016.
[13] Simona Motogna, Ioana Ciuciu, Camelia Serban, and Andreea Vescan. Improving software quality using an ontology-based approach. In *On the Move to Meaningful Internet Systems: OTM 2015 Workshops*, pages 456–465. Springer International Publishing, 2015.
[14] C. Serban. A Formal Approach for OOD Metrics Definition. *First International Conference on Modelling and Development of Intelligent Systems, Sibiu, Romania*, pages 262–269, 2009.
[15] C. Serban. High Coupling Detection Using Fuzzy Clustering Analysis. *Special Issue of Studia Universitatis Babes-Bolyai Informatica: Proceeding of The International Conference on Knowledge Engineering: Principles and Techniques*, pages 223 – 226, 2009.
[16] C. Serban. A conceptual framework for object-oriented design assessment. In *UKSim 4th European Modelling Symposium on Mathematical Modelling and Computer Simulation, Pisa, 17 - 19 November*, pages 90–95, 2010.
[17] C. Serban. God Class Design Flaw Detection In Object Oriented Design. A Case–Study. *Studia Universitas Babes-Bolyai, Seria Informatica*, LVI(4):33–38, 2011.
[18] C. Serban. Shutgun Surgery Design Flaw Detection In Object Oriented Design. A Case– Study. *Studia Universitas Babes-Bolyai, Seria Informatica*, LVIII(4):65–74, 2013.
[19] C. Serban and H.F. Pop. Software Quality Assessment Using a Fuzzy Clustering Approach. *Studia Universitas Babes-Bolyai, Seria Informatica*, LIII(2):27–38, 2008.
[20] C. Serban and A. Vescan. Metrics-based selection of a component assembly. *Special Issue of Studia Universitatis Babes-Bolyai Informatica: Proceeding of The International Conference on Knowledge Engineering: Principles and Techniques, ISSN: 1224-869X*, pages 324–331, 2007.
[21] C. Serban and A. Vescan. Metrics for component-based system development. *Creative Mathematics and Informatics, ISSN: 1843-441X*, 16:143–150, 2007.
[22] C. Serban, A. Vescan, and H. F. Pop. A conceptual framework for component-based system metrics definition. In *9th RoEduNet International Conference, June 2010, Sibiu, Romania*, pages 73–78, 2010.

[23] C. Serban, A. Vescan, and H. F. Pop. A Formal Model for Component-Based System Assessment. In *Second International Conference on Computational Intelligence, Modelling and Simulation, Bali, 28 - 30 September 2010*, pages 261–266, 2010.
[24] C. Serban and H. F. Vescan, A.and Pop. Metrics for component-based system development. *Selection based on Fuzzy Clustering Analysis, ISSN: 1843-441X*, -:505–510, 2008.
[25] Vescan A. Serban C. and Pop H.F. A new component selection algorithm based on metrics and fuzzy clustering analysis. In *Proceedings of the 4rd International Workshop on Hybrid Artificial Intelligence Systems*, pages 621–628, 2009.
[26] CAMELIA SERBAN SIMONA MOTOGNA and ANDREEA VESCAN. Metrics-based refactoring strategy and impact on software quality. *Studia Universitatis Babes-Bolyai, series Informatica*, LX,:83–90, 2/2015.
[27] A. Vescan. Components ordered assembly construction based on temporal restraint. In *Proceeding of the 3rd Doctoral Workshop on Mathematical and Engineering Methods in Computer Science, ISBN: 978-80-7355-077-6*, pages 249–256, 2007.
[28] A. Vescan. Dependencies in the component selection problem. In *Proceedings of the 6th ICAM - International Conference on Applied Mathematics*, page (accepted), 2008.
[29] A. Vescan. An evolutionary multiobjective approach for the component selection problem. In *Proceedings of the First IEEE International Conference on the Applications of Digital Information and Web Technologies, ISBN: 978-1-444-264-9 (indexed IEEE)*, pages 252–257, 2008.
[30] A. Vescan. Optimal component selection using a multiobjective evolutionary algorithm. *Neural Network world, ISSN: 1210-0552 (indexed ISI-SCI-E)*, page (accepted), 2008.
[31] A. Vescan. Pareto dominance - based approach for the component selection problem. In *Proceedings of the 2nd UKSim European Symposium on Computer Modeling and Simulation, ISBN: 978-0-7695-3325-4 (indexed IEEE)*, pages 58–63, 2008.
[32] A. Vescan. A metrics-based evolutionary approach for the component selection problem. In *Proceedings of 11th International Conference on Computer Modelling and Simulation (UKSim 2009)*, pages 83–88, 2009.
[33] A. Vescan and C. Grosan. A hybrid evolutionary multiobjective approach for component selection problem. In *Proceedings of the 3rd International Workshop on Hybrid Artificial Intelligence Systems, ISBN: 978-3-540-87655-7 (ISI Proceeding)*, pages 164–171, 2008.
[34] A. Vescan and C. Grosan. Two evolutionary multiobjective approach for component selection problem. In *Proceedings of the Fourth International Workshop on Evolutionary Multiobjective Optimization - Design and Applications (indexed IEEE)*, page (accepted), 2008.
[35] A. VESCAN and C. GROSAN. Evolutionary multiobjective approach for multilevel component composition. *Studia Universitatis Babes-Bolyai, series Informatica*, LV,:18– 32, 4/2010.
[36] A. Vescan, C. Grosan, and H. F. Pop. Evolutionary algorithms for the component selection problem. In *Proceedings of the 2nd International Workshop on Evolutionary Techniques in Data Processing, ISBN: 1529-4188 (indexed IEEE)*, pages 509–513, 2008.
[37] A. Vescan and H. F. Pop. The component selection problem as a constraint optimization problem. In *Proceedings of the Work In Progress Session of the 3rd IFIP TC2 Central and East European Conference on Software Engineering Techniques*, page (accepted), 2008.
[38] A. Vescan and H.F. Pop. Constraint optimization-based component selection problem. *Studia Universitatis Babes-Bolyai, series Informatica*, LIII:3–14, 2/2008.
[39] ANDREEA VESCAN. Third case study for the dynamic multilevel component selection. *Studia Universitatis Babes-Bolyai, series Informatica*, LXII,:15–31, 1/2017.
[40] Andreea Vescan. Case study method and research design for the dynamic multilevel component selection problem. In *Service-Oriented Computing – ICSOC 2015 Workshops*, pages 130–141, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[41] Andreea Vescan. An evolutionary multiobjective approach for the dynamic multilevel component selection problem. In *Service-Oriented Computing – ICSOC 2015 Workshops*, pages 193–204, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[42] Andreea Vescan and Camelia S¸erban. A fuzzy-based approach for the multilevel component selection problem. In *Hybrid Artificial Intelligent Systems*, pages 463–474, Cham, 2016. Springer International Publishing.

[43] Andreea Vescan and Camelia S¸erban. Multilevel component selection optimization toward an optimal architecture. *Soft Computing*, 21(15):4481–4495, Aug 2017.

[44] C.; Shengxiang Yang Vescan, A.; Grosan. A hybrid evolutionary multiobjective approach for the dynamic component selection problem. In *11th International Conference on Hybrid Intelligent Systems (HIS), Malacca, Malaysia, 5 8 December 2011, ISBN: 978-1-4577-2151-9*, pages 714–721, 2011.

Department of Computer Science, Faculty of Mathematics and Computer Science, Babe¸s-Bolyai University, Cluj-Napoca, Romania

E-mail address: camelia@cs.ubbcluj.ro, avescan@cs.ubbcluj.ro