A New Approach Based on Genetic Algorithm for Prioritizing Quality Scenarios in Enterprise Architecture Evaluation

Majid Karimi, Sayed Mehran Sharafi, Mohammad Naderi Dehkordi
Department of Computer Engineering
Najafabad Branch, Islamic Azad University
Najafabad, Iran
mid.karimi@gmail.com, me_sharafi@yahoo.com, naderi@iaun.ac.ir

***Abstract*— Enterprise Architecture is an approach for understanding, engineering, and managing all enterprise elements and their relationships. In order to better explain the concepts defined in the quality attributes in enterprise architecture and their relationships, the quality scenarios are used. Because of the breadth and variety of enterprise architecture quality scenarios, the cost of implementation scenarios is high. Therefore, prioritization and selection of optimal scenarios, in terms of quality attributes satisfaction, before implementation, is very important. Due to the diversity of stakeholders, large number of scenarios and possible selections, prioritization scenarios involves searching a large state space and considering all of the possible selections which is not precise. Genetic Algorithm is the intelligent algorithm that solves the problems based on metaheuristic search. This paper presents an innovative method for prioritizing quality scenarios, based on the knowledge and experience of stakeholders using genetic algorithm. The validity of proposed algorithm is evaluated in two case studies and it's precision is compared with similar methods. The results of evaluation show the correctness and performance of this algorithm to prioritize large number of quality scenarios with higher precision and lower computational complexity in comparing to other methods.**

Keyword: Genetic Algorithm, Quality Scenario, Enterprise Architecture Evaluation

I. INTRODUCTION

Today's organizations are complex and integrated systems including processes, organizational units, individuals, information and supporting technologies as well as dependencies and communications between different elements. To achieve and sustain organizational performance, knowledge, engineering and management of these social, technical and infrastructure aspects are vital. This need led to the creation of enterprise architecture. Based on the system approach to organizations, it can be concluded that the attributes and problems in the design of complex systems are also related to complex organizations. Some of these system attributes also called quality attributes are selected as utility to any organization, and any decision can lead to move their superiority over other organizations. Examples of these attributes include expandability, flexibility, maintainability, Interoperability and sustainability [1]. Since the function of enterprise architecture is identification and management of organizational elements and their relationships with a wide range of models and data, quality attributes of organizations are presented in models and information of architecture and can be analyzed based on them In order to better explain the concepts defined in the quality attributes in enterprise architecture and their relationships, the quality scenarios are used [2].

Due to the breadth of quality scenarios of enterprise architecture, the cost of finance, human resources and time for implementing scenarios is high; So evaluating enterprise architecture scenarios in terms of meeting the quality and optimized scenarios selection, before implementation is very important [3].

In this paper, the generalized ATAM [4] method of analyzing enterprise architecture is used. This method uses quality scenarios to evaluate architecture in order to identify decisions and tradeoffs and then determine whether they are compatible in architectural structure with quality attributes or not. One of the main steps in this method is the prioritization of quality scenarios.

In order to prioritizing Enterprise Architecture quality scenarios, there are a few things to be considered:

1) An EA is composed of (or realized by) four "sub-architectures" (business architecture, information or data architecture, application architecture and technology or infrastructure architecture) [5].
2) The elements of an EA include stakeholders [5].

3) the diversity of stakeholders and the number of scenarios to consider in an EA can become intractable, risking spotty coverage of quality attribute requirements or leading to a very long process to achieve adequate breadth [5].
4) Each quality scenario is made of a set of criteria and sub criteria [3].

Therefore the problem of prioritizing quality scenarios is a decision-making problem in which multiple criteria have impact [3]. Due to the large number of scenarios, stakeholders and their different selection criteria and their interactions, finding an optimal order of scenarios includes a search in a large that is in accurate and mistake-prone, and its computational complexity is placed in the category of NP-Complete problems. To solve these problems, metaheuristic algorithms are used. One of the most famous of metaheuristic algorithms is Genetic Algorithm. Its success is due to avoiding systematic search of the whole problem space and reducing its computational complexity [6]. In this paper an attempt is made to improve prioritizing quality scenarios by using genetic algorithms and opinions of stakeholders on different criteria of scenario selections.

The validity of proposed algorithm is evaluated in two case studies and the accuracy of its priority is compared with similar approaches. Assessment results indicate the accuracy and performance of this algorithm in prioritizing the quality scenarios with greater accuracy and less computational complexity than other methods.

The structure of this paper is as follows: In Section 2 related works are reviewed and discussed. In section 3 the proposed fitness function in genetic algorithm is introduced. In section 4, the computational complexity of the algorithm is calculated and two case studies in the domains of enterprise architecture and software architecture, evaluate its application. The conclusions and future work are stated in section 5.

II. RELATED WORKS

The software architecture evaluation methods tried to predict the software quality attributes that will be produced based on an architecture. Among the evaluation methods , ATAM techniques with tools, strategies, and very good examples of practical application range and it is more appropriate for evaluating [7]. To evaluate the enterprise architecture , it is necessary to use methods based on ATAM [5]. Proposed method to Prioritize in this research is based on prioritizing quality scenarios step in the ATAM method. Many research projects discussed the different aspects of organizational enterprise architecture evaluation. Some of these research projects evaluate the organization's enterprise architecture. For instance, provided methods by a group of researchers at KTH university in Sweden([8, 9, 10, 11, 12, 13]) and AHP-fuzzy approach [3] can be mentioned. The big problem that most of these methods have, is neglecting all of the prioritizing criteria and inaccuracy as well as high computational complexity to prioritize large number of scenarios that could affect the accuracy and performance.

III. THE PROPOSED GENETIC ALGORITHM

This section first describes the requirements for implementing the proposed algorithm, then defines the algorithm and its functions' details and their operators in order to prioritize quality scenarios.

A. Criteria to Prioritize Quality Scenarios

Criteria to prioritize quality scenarios that should be considered in enterprise architecture assessment can be defined in 6 separate criteria. These criteria are:

1) Priority in every quality scenario based on the stockholders comment: In every organization there are people as stakeholders and architecture decision makers that based on their comments, each quality scenario has a priority. These priorities are emerged from their position and vision of the organization [3].
2) The difficulty level of acquiring or implementing of a scenario, based on architect's comment: each scenario, depending on how hard or easy to access it is, has a priority based on architect's comment. Scenarios with high difficulty have a higher importance for evaluation [2].
3) The importance of each criterion or sub criterion of quality attributes related to the scenario based on the stockholders comment: each related quality attribute has criteria or sub-criteria, which have different degrees of importance and priority [3].
4) A measure of the impact of each scenario on the criteria or sub criteria of the quality attributes, based on the organization's stakeholders comment: a measure of the amount of each scenario from concrete scenarios relevant to a quality attribute has effect on any criteria or sub criteria, that reflects its compliance with the relevant quality attribute [3].
5) The effect of scenario on other quality attributes, based on the organization's stakeholders comment: each quality scenario can affect other considered organization's quality attributes. A tradeoff should be considered in scenarios selections [3]. For better results, scenarios should have the least negative impact on other quality attributes after their priority based on stakeholders comment.

6) Priority of stakeholders participating in the prioritization process based on architect comment: there are several people in the organization that their comments are used to prioritize scenarios. These people are totally called stakeholders [3]. They can have different priorities for prioritizing based on architect's comment.

These criteria are used as fitness function's entry of genetic algorithm that will be introduced as follows. Since these criteria are qualitative, quantitative values of these parameters are used to express.

B. Implementation of the Proposed Genetic Algorithm

After the preparations for the implementation of the algorithm, this section will describe and examine how to implement it. To work with the genetic algorithm it's needed to implement chromosome structure or set of possible solutions, the initial population, fitness function, selection function and crossover and mutation operators [6]. This paper doesn't describe these issues and it is assumed that the reader is familiar with basic concepts of genetic algorithms. The design of proposed genetic algorithm consists of the several steps that will be described below.

1) The Structure of Chromosome

Chromosome length is fixed and equal to the number of quality scenarios. Each scenario is supposed as a gene on a chromosome that their order of placement shows their priority during the running of algorithm. The goal is to find the optimal sequence of scenarios.

As described in previous section, the values of the criteria for prioritization of scenarios are qualitative, and quantitative equivalents are used to express them. The numerical values as the fitness function entry, must be saved for any scenario where the structure of cell arrays are used. In Figure 1, the structure of cell array of 6 elements for each scenario is shown, as well as the component identifier QID relevant quality attribute.

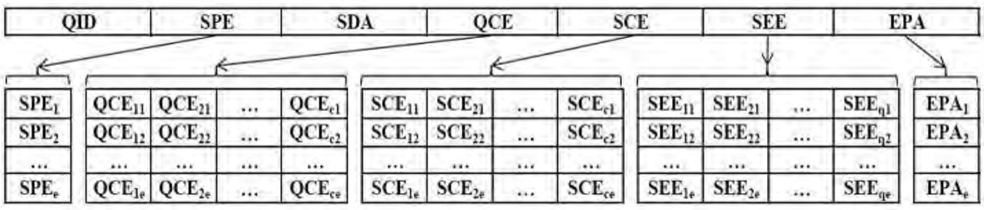


Figure 1. The structure of scenario

In this structure, the constants c, e, and q respectively characterize the number of stakeholders , number of sub criteria and total number of quality attributes. SPE is an array of e elements, representing the priority of scenarios based on stakeholders comment. SDA is the numerical equivalent of difficulty to implement scenarios. QCE is a matrix with $e \times c$ dimensions that represent stakeholders comments about the importance of each quality attribute's sub criteria. SCE is the matrix with the same dimensions of QCE and characterize the impact of scenario response to quality attributes' sub criteria. SEE is the matrix of size $e \times q$, which represents the impact of this scenario on other quality attributes. The last element is EPA that is an array with e elements that determines the priority of stakeholders.

2) Creating the Initial Population

In the first stage of the genetic algorithm, the initial population is generated. The initial population consists of an initial generation of chromosomes that are randomly produced. A random population of N chromosomes, where N, is the population size or the number of members of the first generation that is created. The size of initial population was fixed to the end of the algorithm's execution time. This should be the maximum possible value provided that it isn't an impediment to algorithm's executing speed.

3) Fitness Function

This function is a criterion for diagnosing the superiority or eligibility of an answer over other answers. Knowing that each chromosome represents an answer, by assigning a numeric value, suitability or fitness of each chromosome against other chromosomes of its own generation must be determined [6]. In the quality scenarios prioritization problem, each answer shows a sequence of scenarios which is scenarios prioritization. The best answer is of the chromosome in which the order of the scenarios includes compliance the best standards and priorities are as follows:

1) descending order of scenarios priority criteria
2) descending order of difficulty of achieving criteria
3) descending order of importance of quality attribute sub criteria
4) descending order of response impact to quality attribute sub criteria
5) ascending order of effect on other quality attribute criteria

6) Scenarios placement in each chromosome with no duplicate

Fitness function with the input similar to Figure 1 and considering the optimal response criteria, shows a numerical value that indicates the merit of this responses to the other response. Fitness function F(X) for chromosome X can be expressed by (1).

$$F(X) = \sum_{i=1}^{n} \left(\left((W_1 \times FP_i) + (W_2 \times FD_i) + (W_3 \times FC_i) + (W_4 \times FE_i) \right)^{1/i} \right) \quad (1)$$

The input parameter of fitness function is an array named X with the length of n that the array elements show the ID of scenarios in it. n, is the total number of scenarios. The output of this function indicates fitness of the input chromosome and the higher numbers shows the more merit of a chromosome.

The proposed fitness function consists of several sub-functions. Each of these sub-functions has the responsibility to calculate the quantities of each prioritization criteria. W_k represents the importance or weight of corresponding sub-function. The reason for using this ratio is that each of the sub-functions has an importance in relation to the total amount of fitness function. Total sum of W_ks will be equal to 1. The amount of weight depends on the type of scenario or organization that the architect will determine the coefficients or weights. Sub-functions of the fitness function are described below. In these sub-functions priority criteria of each stakeholder (EPA_j) is used in the calculation of each of the other criteria.

The responsibility of sub-function FP is calculating priority of each scenario based on comments of stakeholders. To obtain this measure, the sum of the priorities of stakeholders comments in relation to stakeholder's priority to the organization about scenario priorities, is calculated. This sub-function is expressed in (2).

$$FP = \sum_{j=1}^{e} \left(SPE_j \times EPA_j \right) \quad (2)$$

Sub-function FD is equal to the degree of difficulty achieving or implementing scenario that is a fixed number and based on the architect's comment, so FD = SDA.

Sub-function FC shown in (3) is responsible to calculate the impact of response criteria values of this scenario to criteria or sub-criteria of quality attribute.

$$FC = \sum_{j=1}^{e} \left(QC_j \times SC_j \right) \quad (3)$$

This sub-function includes sub-function QC (4) to calculate the priority of quality attribute's criteria or sub-criteria based on the stakeholders' comment and sub-function SC (6) for calculation of the impact of scenario response measure to criteria of quality attribute.

$$QC_j = \frac{QCE_j}{\sum_{f=1}^{c} QCE_f} \quad (4)$$

The value of QCE used in (4) is calculated in (5).

$$QCE_j = \sum_{k=1}^{e} \left(QCE_{k,j} \times EPA_k \right) \quad (5)$$

Responsibility of sub-function SC is calculating average value of stakeholders comment about the effectiveness of the impact of scenario response measure to criteria of quality attribute.

$$SC_i = \sum_{k=1}^{e} \left(SCE_{k,j} \times EPA_k \right) \quad (6)$$

Responsibility of sub-function FE is calculating the effect of this scenario on other quality attributes which are located after this scenario in the order of the chromosomes. How to calculate the FE sub-function is expressed in (7).

$$FE = \sum_{j=1}^{q} \left(\sum_{k=1}^{e} \left(SEE_{k,j} \times EPA_k \right) \right) \quad (7)$$

To calculate the fitness function, a power of (1 / i) is used. The reason for this, is that the value calculated for each scenario is powered to reverse of its rate on the chromosome so that for producing new chromosomes, those scenarios which have higher priority and numeric value will be at the beginning of the chromosome.

4) Selection Function

Selection function selects those chromosomes as the parent which have more elegance than other chromosomes in the same generation. In this algorithm, the roulette wheel selection method [6] is used. In this method, all possible values of selection probability that are fractions of division of the fitness function to the total value, stacked together to generate a random number in the range 0 to 1, which indicates its position in the chromosome of the roulette wheel. Obviously, the probability of selecting worthy chromosomes is more than others.

5) Genetic Algorithm Operators

After Selecting candidate parents, for creating a new generation, two operators are used to crossover and mutate that are described below.

a) Crossover Operator

This operator selects a pair of parents, crossovers their genes to produce the new generation of chromosomes that have good features of the previous generation [6]. In this algorithm the uniform crossover techniques that produce children with uniform random selection of genes from each parents are used.

b) Mutation Operator

This function changes the genes of a chromosome to create unexpected results [6]. In this algorithm, the swapping mutation technique is used that first selects several genes from chromosome, then swaps their values.

6) Proposed Genetic Algorithm Procedure

Proposed genetic algorithm procedure is shown in Figure 2. At first, the initial population that consists of N scenario sets, randomly generated and the maximum number of generations, the crossover probability (P_c), mutation probability (P_m) and Probability of convergence (P_{cn}) are determined.

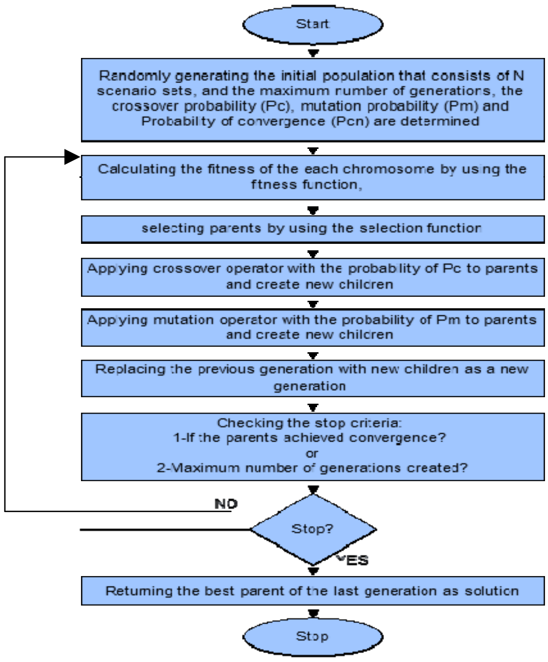


Figure 2: Procedure of the proposed algorithm

Then by using the fitness function, the fitness of the first population is calculated. Furthermore, parents are selected using the selection function, crossover and mutation operators with the probability of P_c and P_m applied to parents. Created children as a new generation replace the previous generation. If the parents achieved convergence or maximum number of generations, the algorithm stops and the best parent of the last generation is returned as solution, otherwise algorithm resumes with a new generation and applying the fitness function's step.

IV. Evaluation of the Proposed Genetic Algorithm

In order to evaluate the proposed genetic algorithm, the computational complexity of the proposed algorithm is compared with similar methods and algorithms. Then by doing a case study in the field of enterprise architecture, its accuracy is evaluated. At the end of this section to demonstrate the capability and flexibility of the proposed algorithm in the field of software architecture, a case study of this area is given.

A. Evaluating Computational Complexity of the Proposed Algorithm

The problem of finding an optimal scenario is a multi-criteria decision and involves searching a very large state space that the computational complexity of those problems are NP-Complete. Common prioritization methods with pairwise comparisons of the prioritization criteria and search in the state space, attempt to find an optimal order of scenarios. Doing comparisons for the large number of scenarios will be order of O(n!) that is a very bad running time. Genetic algorithms can be used to reach an acceptable solution, with a running time significantly reduced in comparison with these methods. It is important to note that the genetic algorithm does not guarantee to achieve the best results but in a much shorter time than previous methods. It offers an acceptable answer that could be the best answer, or close to it [6]. The reason of running genetic algorithms with lower computational complexity than the other methods is avoiding the systematic search of the solution in the problem state space. With a suitable choice of functions and operators, genetic algorithms can be used to reach an optimal computational complexity with respect to the performance of genetic algorithm , certainly less than the computational complexity of previous methods and can be around order of $O(n^2)$. On the other hand, it can be used for parallel execution capabilities of genetic algorithms. The execution time can be reduced to a considerable extent. In Figure 3 the comparison of the computational complexity of the proposed method and other methods for prioritization is shown.

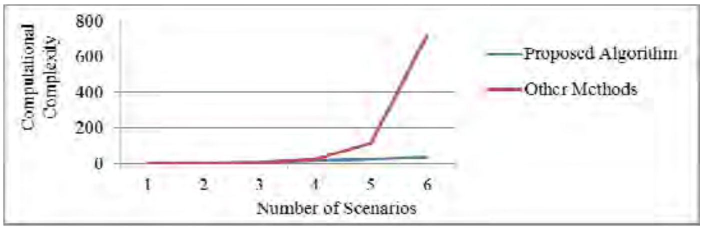


Figure 3: The comparison of the proposed genetic algorithm complexity with conventional methods

B. A Case Study in the Field of Enterprise Architecture

In this part a case study that was introduced in [3] is used. This case study relates to the assessment Enterprise Architecture PMO Iran. Initially desired quality attributes which can affect the choice of quality scenarios must be clearly identified. Maintainability and Interoperability are specified as quality attributes for this organization. In this case study the proposed structure in [2] is used to identify the quality attributes and their criteria. For simplicity, only the first level criteria of the quality attributes are used. Maintainability's criteria are shown in Figure 4 and Interoperability's criteria are shown in Figure 5.

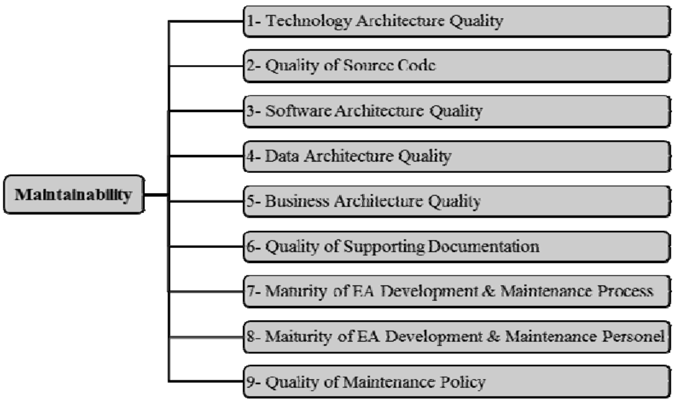


Figure 4: The first level criteria of maintainability

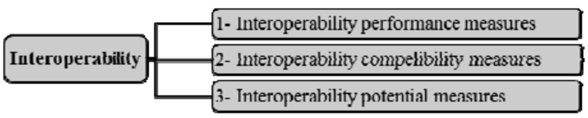


Figure 5: The first level criteria of Interoperability

In this case study the comments of five stakeholders are used their priorities are not mentioned, the equal priority will be considered. Stakeholders will be asked to prioritize the criteria of quality attributes with the values in the range of 0 to 1 so that the sum of the given priority equals to 1. The results are shown in Table 1 and Table 2.

Table 1: The priority of interoperability criteria

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	0.25	0.40	0.33	0.30	0.40
2	0.35	0.35	0.33	0.40	0.25
3	0.40	0.25	0.33	0.30	0.35
Sum	**1**	**1**	**1**	**1**	**1**

Table 2: The priority of maintainability criteria

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	0.11	0.12	0.13	0.07	0.06
2	0.11	0.12	0.13	0.07	0.06
3	0.11	0.12	0.13	0.07	0.06
4	0.11	0.12	0.13	0.07	0.06
5	0.11	0.12	0.13	0.07	0.06
6	0.24	0.22	0.21	0.53	0.14
7	0.07	0.06	0.05	0.04	0.14
8	0.07	0.06	0.05	0.04	0.26
9	0.07	0.06	0.04	0.04	0.04
Sum	**1**	**1**	**1**	**1**	**1**

Considering the two quality attributes and organization’s option, for each quality attribute two scenarios and a total of four concrete scenarios have been proposed. The details of these four scenarios are presented in [3]. Priority of scenarios based on stakeholders’ comments are shown in Table 3.

Table 3: Priority of scenarios based on stakeholders’ comments

Scenarios	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	0.37	0.25	0.37	0.37	0.42
2	0.37	0.25	0.37	0.37	0.42
3	0.13	0.25	0.13	0.13	0.08
4	0.13	0.25	0.13	0.13	0.08
Sum	**1**	**1**	**1**	**1**	**1**

Difficulty to implement or access scenarios based on architect’s comment is not stated therefore the value of 0.25 is considered for each scenario.

The measure of the impact of each scenario on the criteria or sub criteria of the quality attributes, based on the organization's stakeholders comment is initialized in the range of -2 to 2, that their definitions are given in Table 4.

Table 4: defined values for the impacts on criteria

Very positive effect	Positive effect	No effect	Negative effect	Very negative effect
2	1	0	-1	-2

Comments of five stakeholders about the impact of each scenario on the criteria or sub criteria of the quality attributes, relevant scenarios 1 to 4, are shown in Table 5, Table 6, Table 7 and Table 8.

Table 5: Impact of scenario #1 on maintainability

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	1	1	0	1	1
2	0	0	0	0	0
3	0	0	0	0	0
4	2	1	2	2	1
5	1	1	1	1	1
6	2	1	1	2	2
7	1	1	1	0	0
8	2	2	1	2	2
9	1	0	1	0	1

Table 6: Impact of scenario #2 on maintainability

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	0	1	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	2	2	2	1	1
5	0	0	0	0	0
6	-2	-2	-1	-1	-2
7	-2	-2	-2	-2	-2
8	-1	-1	-1	-1	-1
9	-1	-2	-2	-1	-2

Table 7: Impact of scenario #3 on interoperability

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	-1	-1	0	0	-1
2	-1	-2	-2	-2	-1
3	-2	-2	-1	-1	-1

Table 8: Impact of scenario #4 on interoperability

Criteria	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
1	1	1	1	1	1
2	2	2	1	2	2
3	2	2	2	2	2

The stakeholders will be asked to express the impact of each scenario on other quality attributes based on the numbers shown in Table 4. The results of which have been shown in Table 9, Table 10, Table 11, Table 12.

Table 9: The impact of scenario #1 on other quality attributes

Quality Attribute	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
Maintainability	0	0	0	0	0
Interoperability	1	2	1	1	2

Table 10: The impact of scenario #2 on other quality attributes

Quality Attribute	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
Maintainability	0	0	0	0	0
Interoperability	1	-1	0	0	1

Table 11: The impact of scenario #3 on other quality attributes

Quality Attribute	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
Maintainability	-1	-2	-2	-1	-2
Interoperability	0	0	0	0	0

Table 12: The impact of scenario #4 on other quality attributes

Quality Attribute	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5
Maintainability	2	2	1	1	2
Interoperability	0	0	0	0	0

After collecting information about scenarios prioritization criteria and using them as input data for the proposed genetic algorithm, Scenario with number 4 had higher priority and Scenario with number 1 , 2 and 3, respectively, were held to assess priorities for the next.

The result of running of this algorithm in MATLAB software is shown in Figure 6. As specified, after maximum number of generations in the algorithm, the convergence and satisfactory answer has been prioritized based on the results obtained in [3] to verify that, with the difference that performance and computational complexity of proposed algorithm are better than this method.

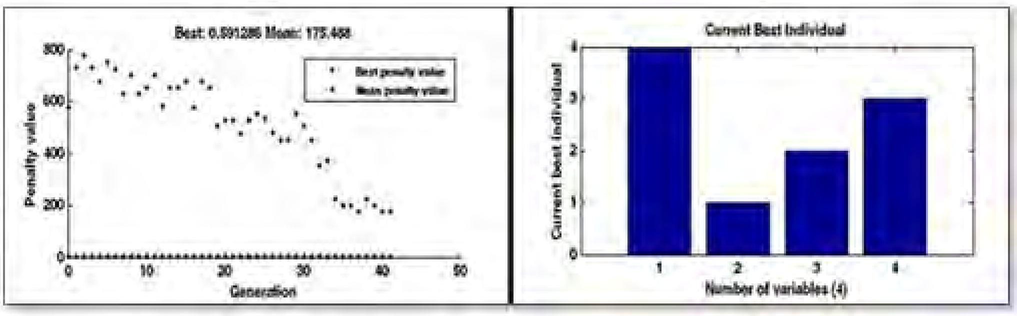


Figure 6: Result of proposed genetic algorithm in the field of Enterprise Architecture Case Study

C. A Case Study in the Field of Software Architecture

In order to demonstrate the usability and flexibility of the proposed algorithm in the field of software architecture a case study of Chapter 11 of the reference [4], is used.

In this case study, the quality scenario prioritization that is a step of ATAM method, based on a utility tree with only two criteria, importance of the scenario and the difficulty to implementation, will be done. This approach has a lower accuracy of priority because other criteria involved in the evaluation are not considered and is also a manual method and error prone.

The aim of this research is to overcome the lack of precision in prioritization and reduce computational complexity; therefore in the following to prioritize quality scenarios in case study, proposed genetic algorithm is used.

In this case study 29 quality scenarios are identified. Quality Scenarios introduced in Table 13, along with quality attributes and sub criteria associated with them, their importance according to stakeholders' comment and the difficulty of acquiring are introduced. For using qualitative information extracted from this case study in the proposed algorithm, they should be converted into equivalent quantities. If the structure given in Figure 1 considered as the proposed scenario prioritization criteria along with proposed scenario structure, therefore at first, quality scenarios priority must be specified based on stakeholder's comments.

In this case study, stakeholders' comments are expressed with one element, so only one number in the range of [0,1] in ascending order from low to high level of importance will be associated.

The second criterion for prioritization of scenarios is achieving difficulty that is expressed with the terms of low, medium and high hardness. This criterion should be turned into on equivalent quantity as scenarios importance in ascending order from 0 to 1.

The third criterion is the level of the importance of quality attributes' sub-criteria. Since this criterion is not mentioned directly, therefore the priority level of quality attribute which is mentioned in the second step of the first phase of the evaluation is used and prioritization of quality attributes' sub-criteria is considered with the same priority of quality attributes. In Table 14, each of the priority of quality attributes' criteria and priority of quality attributes without sub-criteria for consideration is shown.

The corresponding quantitative values for applying these criteria have been used in genetic algorithms as the previous criteria; so criteria are evaluated at the normal level. Ascending order of the values 0 and 1 are used to express priority from low to high.

The fourth measure is the impact of each scenario on the criteria or sub criteria of the quality attributes, that used value of 1 for the sub-criteria with which scenarios are associated and the value of 0 for the other sub-criteria of same quality attribute are used.

The fifth criteria is scenario effect on other quality attributes that same as the third criteria is not mentioned directly but with the explanation in the reference about interactions and disturbing effects between quality attributes, quantitative equivalent values can be used to express the amount of effects. Equivalent values are shown in Table 4.

The last criterion for prioritizing is the priority of stakeholders and participants in the evaluation process that was not used due to the fact that in this case study the average value of stakeholders' comment are used and comments are not expressed individually which shows the flexibility of the proposed algorithm.

After collecting information on criteria to prioritize scenarios in this case study, these quantitative values were given as input to proposed genetic algorithm with cell array structure.

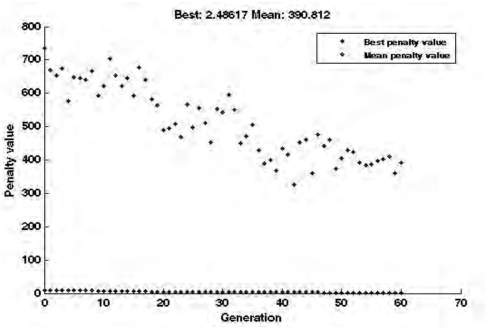
Table 13: Quality attributes and their sub criteria, their importance according to stakeholders comment and the difficulty of acquiring

Scenario	Quality Attribute	Quality Attribute Sub-Criteria	Scenario Importance	Difficulty to Achieve
1	Performance	Transaction response time	H	M
2	Performance	Transaction response time	L	M
3	Performance	Throughput	M	M
4	Usability	Proficiency training	M	L
5	Usability	Proficiency training	H	L
6	Usability	Normal operations	M	M
7	Configurability	-------	H	L
8	Maintainability	-------	H	M
9	Maintainability	-------	M	L
10	Maintainability	-------	H	M
11	Extensibility	Adding new products	M	M
12	Security	Confidentiality	H	M
13	Security	Integrity	H	M
14	Availability	-------	H	L
15	Availability	-------	L	L
16	Scalability	Growing the system	L	H
17	Scalability	Growing the system	L	M
18	Scalability	Growing the system	L	M
19	Scalability	Growing the system	M	L
20	Modularity	Functional subsets	M	L
21	Modularity	Flexibility to replace COTS products	H	M
22	Modularity	Flexibility to replace COTS products	H	M
23	Modularity	Flexibility to replace COTS products	H	M
24	Modularity	Flexibility to replace COTS products	H	M
25	Modularity	Flexibility to replace COTS products	H	M
26	Modularity	Flexibility to replace COTS products	H	M
27	Modularity	Flexibility to replace COTS products	H	M
28	Modularity	Flexibility to replace COTS products	H	M
29	Interoperability	-------	M	M

Table 14: Priority of quality attributes' criteria and priority of quality attributes without sub-criteria

Sub-Criteria of Quality Attributes	Priority
Transaction response time	H
Throughput	H
Proficiency training	H
Normal operations	H
Configurability	L
Maintainability	H
Adding new products	L
Confidentiality	M
Integrity	M
Availability	M
Growing the system	M
Functional subsets	M
Flexibility to replace COTS products	M
Interoperability	L

Figure 7 is an example of the running proposed algorithm based on the average and best fitness function value and prioritized scenarios; that is shown to achieve an optimal convergence and the maximum number of generations.



Final point:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
25	18	4	12	3	4	29	19	8	16	9	4	9	9	16	19	11	22	8	24	6	27	10	26	1	14	21	17	23

Figure 7: Result of proposed genetic algorithm in the field of Software Architecture Case Study

It will not necessarily produce the same results on every run. Since the proposed algorithm is to help the architect to make decision, so can be run several times and observe the results and make appropriate decisions about optimal priorities of quality scenarios. As shown in the output of the algorithm, the high priority of scenarios based on stakeholders' comments are not necessarily in the beginning of the prioritized list. The reason is considering the proposed algorithm to another prioritization criteria, including difficulty to implementing, sub-criteria of quality attributes, effects on other quality attributes, etc. For example scenario number 25 that is on the top priority list, has medium priority based on stakeholder's comment but because of the positive impact on other quality attributes and also high effect on sub-criteria of quality attributes, is considered with the first priority in the final output.

The results demonstrate the high accuracy of the proposed algorithm compared to other prioritization methods that have benefited from fewer criteria or methods that used intuitive or experimental prioritization.

V. CONCLUSION AND FUTURE WORK

In this paper, a qualitative method of prioritizing scenarios, considering all the parameters that influence the quality of enterprise architecture based on genetic algorithm was presented. The proposed algorithm helps evaluating teams, architects, and organization decision makers to have optimize priority of the quality scenarios based on the organization's quality attributes. As the results shows the proposed algorithm is more accurate and has less computational complexity compared to other methods and the accuracy of its output is shown in case studies.

For future work, proposed genetic algorithm can be used in other phases of architecture evaluation such as selecting the proper architecture plan.

REFERENCES

[1] Nightingale D.J., "Principles of Enterprise Systems," in Second International Symposium on Engineering Systems MIT, 2009.

[2] S. A. F. Razavi M., "Characterization of Enterprise Architecture Quality Attributes," in Proceedings of Advances in Quality of Service Management Workshop (AquSerM09), EDOC 2009, Auckland, New Zealand, 2009.

[3] S. A. F. Razavi, "A Fuzzy AHP Based Approach Towards Enterprise Architecture Evaluation," in Europea Conference on information Management and Evaluation (ECIME 2009), Gothenburg, Sweden, 2009.

[4] L. C. P. K. R. Bass, Software Architecture in Practice, 3nd Ed, Addison-Wesley Professional, 2012.

[5] J. a. G. M. J. Klein, "A Workshop on Analysis and Evaluation of Enterprise Architectures," Software Engineering Institute, p. 429, 2010.

[6] R. L. H. a. S. E. Haupt, Practical Genetic Algorithms, Second Edition, Wiley-Interscience, 2004.

[7] R. T. A. ,. Z. K. Khayami, "Evaluating Quality Characteristics of Enterprise Architecture," Knowledge and Information Systems, 2009.

[8] J. E. S. T. U. J. A. Johnson P., "Tool for Enterprise Architecture Analysis," in 11th IEEE Enterprise Distributed Object Computing Conference, 2007.

[9] L. R. N. P. S. M. Johnson P., "Enterprise Architecture Analysis with Extended Influence Diagrams," Information Systems Frontiers, 2007.

[10] J. P., "Extended Influence Diagrams for Enterprise Architecture Analysis," in 10th IEEE EDOC Conference, 2006.

[11] E. M. N. P. Gammelgård M., "Architecture Scenario Analysis– Estimating the Credibility of the Results," in 17th International Symposium of the Council on Systems Engineering, 2007.

[12] L. A. J. P. Simonsson M., "Scenario-based Evaluation of Enterprise - a Top-Down Approach for Chief Information Officer Decision Making," in 7th International Conference on Enterprise Information Systems ICEIS, 2005.

[13] J. P. N. L. Närman P., "Enterprise Architecture: A Framework Supporting System Quality Analysis," in 11th IEEE EDOC Conference, 2007.