RIGA TECHNICAL UNIVERSITY

Information Technology and Management Science

ISSN 2255-9094 (online)
ISSN 2255-9086 (print)
December 2019, vol. 22, pp. 22–31
doi: 10.7250/itms-2019-0004
https://itms-journals.rtu.lv

Application of Genetic Algorithms for Decision-Making in Project Management: A Literature Review

Ieva Ancveire[1], Inese Poļaka[2]
[1,2] *Riga Technical University, Riga, Latvia*

Abstract **– In software development projects, managers still have to face a variety of organisational and technical limitations despite the development of technology and approaches to improve the project management process. Projects, Human Resources and Costs are planned for a specific period of time. However, in the progression of project execution, there is a need to make various decisions and to dynamically adjust the work plan during the project in order to conform to its evolution. Thus, there is a need for a method that employs the latest technology to support the project management decision-making process.**

The aim and the expected result of the article are to identify and collect available information in the scientific literature to answer the following questions: (1) Which challenges of project management have been addressed using genetic algorithms? (2) What are the opportunities and limitations of genetic algorithms in the project management decision-making process? (3) What are the potential solutions to the identified genetic algorithm problems?

Keywords **– Algorithm limitations, genetic algorithm, project management.**

I. Introduction

There are many examples of the application of the genetic algorithm (GA), which also include project management in software development projects. In this article, we will prefer PMBOK, which is a commonly used and applied industry standard in project management. Following good practices [1], project management includes different areas of knowledge[1].

In the project management context, Project Integration Management includes the processes and activities to identify, define, combine, unify, and coordinate the various processes and project management activities within the project management process groups. Management of the project scope is primarily concerned with defining and controlling what is and is not included in the project.

Project Cost Management is primarily concerned with the cost of the resources needed to complete project activities. The main tasks in the Project Quality Management include the processes for incorporating the organisation's quality policy regarding planning, managing, and controlling project and product quality requirements to meet stakeholders' objectives. Project Resource Management helps ensure that the right resources are available to the project manager and project team at the right time and place for successful completion of the project.

Project Communication Management consists of two parts. The first part develops a strategy to ensure that communication is useful for stakeholders. The second part carries out the activities necessary to implement the communication strategy. The objectives of project risk management are to increase the probability and/or impact of definite risk and to decrease the probability and/or impact of adverse risk in order to optimise the chances of project success. Project Procurement Management includes the management and control processes required to develop and administer agreements, such as contracts, purchase orders, memoranda of agreements or internal service level agreements. Project Stakeholder Management identifies the people groups or organisations that could impact or be impacted by the project [1].

Many models of GA solutions vary depending on the project's size, complexity, duration, working cost, and requirements [2]. There is no uniform methodology for tests and evaluation of genetic algorithm application in project management.

Most studies include subjective evaluations based mainly on tasks that they solve. However, the following steps of the algorithm can be identified at a high level:

1. Initial chromosome (solution) population generation [2]–[4].
2. Defining weights for each individual [3], [5].
3. Estimation of the fitness value (goodness). If necessary, scaling of the fitness value [6]–[10].
4. Selection of pairs of individuals for the crossover operation [5], [11]–[14].
5. Identification of mutation in new individual chromosomes. Mutation [15]–[18].
6. Selection of individuals for the next generation [18]–[22].
7. If the stopping criterion of the algorithm is not met, go to Step 2.

All these steps include, evaluate and employ the following parameters: the number of generations, number of

[1] (1) Project Integration Management; (2) Project Scope Management; (3) Project Schedule Management; (4) Project Cost Management; (5) Project Quality Management; (6) Project Resource Management; (7) Project Communication Management; (8) Project Risk Management; (9) Project Procurement Management; (10) Project Stakeholder Management [1].

chromosomes in the population, crossover and mutation probability, types of different genetics operators [2].

The aim of the article is to answer the following questions: (RQ1) Which challenges of project management have been addressed using genetic algorithms? (RQ2) What are the opportunities and limitations of genetic algorithms in the project management decision-making process? (RQ3) What are the potential solutions to the identified genetic algorithm problems?

The rest of the paper is organised as follows. Section 2 details the research design process, describing the process of selection and analysis of scientific publications. The description of articles and answers to research questions RQ1, RQ2, and RQ3 are presented in Sections 3 and 4. Finally, Section 5 presents conclusions related to the literature review and suggests directions for further research.

II. Research Method

In the analysis process, a systematic literature review method has been used to identify articles in the fields of project management, decision-making, and genetic algorithms.

In the literature analysis process, we seek answers to the following research questions: (1) Which challenges of project management have been addressed using genetic algorithms? (2) What are the opportunities and limitations of genetic algorithms in the project management decision-making process? (3) What are the potential solutions to the identified genetic algorithm problems?

Before the search stage of scientific articles, it was necessary to identify the aim of the literature analysis, i.e., to find answers to the research questions, to select relevant scientific articles, to review them and analyse the literature selected from scientific repositories: EBSCO, IEEE Xplore Digital Library, ScienceDirect, SCOPUS, ACM Digital Library. These repositories were chosen because they are popular, extensive and also provided by the university to students and researchers; other repositories were not examined for similar reasons.

The following results were expected at the end of the literature research process: (1) a set of keywords describing the research issue; (2) articles related to the research question for the research issue; (3) articles representing the research issues.

First, to obtain a complete view of genetic algorithm application to the project management decision-making process, different keywords were used in repositories for search.

The following keywords were used to find the answers to the research questions RQ1, RQ2, and RQ3: (1) "genetic algorithm project management"; (2) "genetic algorithm project management challenges"; (3) "genetic algorithm project management problems"; (4) "genetic algorithm project management limitations"; (5) "genetic algorithm project management problems solutions"; (6) "genetic algorithm project management software development".

To limit the number of scientific articles we decided to limit articles only to the field related to our work area.

While articles related to other industries, which are available in scientific repositories, are an interesting source of research for specialists in other industries, the authors selected articles only in the field of computer science. Scientific articles from 2010 onwards were selected and reviewed. We chose this time period to find out the trends of recent years, at the same time including as various articles as possible.

The last step of the study of the selected articles was the analysis of abstracts, article duplicates and full-text articles. However, after a full-text review, 45 most relevant articles were selected to answer research questions RQ1, RQ2, and RQ3.

III. Genetic Algorithms in Software Development Project Management

Different scientific articles were selected from different scientific repositories to provide a comprehensive analysis of the problem area. After a full-text analysis, the selected scientific articles were analysed in more detail, thus reaching the previously identified expected results of the research process.

All articles can be divided into two categories: (1) articles that review the general use of genetic algorithms to solve project management problems and make decisions, (2) articles that discuss the use of genetic algorithms in software development project planning and decision-making.

According to the analysis, 13 out of 45 reviewed articles apply to the general use of genetic algorithms for project management problem-solving and decision-making. These articles have been included in the review because the described GAs could also be used to solve decision-making problems when managing software development projects. It was found out that 32 out of 45 articles described the use of genetic algorithms for project management in software development projects.

All of these articles presented different approaches to find different solutions, such as identifying technical factors in a project and optimising various project issues. This section answers research questions RQ1 and RQ2.

When grouping the articles based on the contribution to a particular project management problem (see Table I), one can conclude that the most common problems addressed in recent literature are the following: (1) Project Schedule Management (reviewed in 33 articles) and (2) Project Resource Management (reviewed in 19 articles).

There are ten types of problems identified in the Knowledge Area of Project Schedule Management. The most common problem identified involves Project, Resource, Team, Activities, Tasks: Planning and Scheduling processes. This problem type is addressed in articles the most (in 15 articles out of 33).

The genetic algorithms presented in the articles can be divided into two groups based on their complexity: (1) standard genetic algorithms that use common definitions and operations, and (2) genetic algorithm modifications and improvements proposed by the authors.

To solve the problem type related to Project, Resource, Team, Activities, Tasks: Planning and Scheduling, standard algorithms are used (2 articles), as well as versions are proposed and improved by authors (13 articles). The number of the improved solutions shows that the tasks are quite specific and ask for a specific adaptation of standard methods. However, even the improved methods have their limitations, and the authors of the present article plan to address these issues in future research.

From the review, it can be inferred that in all cases the representations of solutions (GA chromosome) and encoding strategies are very project-specific, for example, project costs, project duration and project overtime [43]; task start date [11], [35]; a specific mode for each activity and activity lists [7], [23], [24], [33], [38], [41]; human resources [11], [23], [33] and their priorities [12]. The most common types of chromosome encoding in the reviewed studies are binary code [11], [12], [18], [23], [24], [35], [36] and integer number strings [7], [33], [41]. Other types also used in some studies are chromosome encoding types, such as the list of strings [8], graph nodes [38] and encoding method proposed by the authors [43]. In two of the reviewed articles [6], [19], the authors do not suggest a specific chromosome encoding method.

The authors agree that genetic algorithms also have their limitations. The authors identified these limitations and recognised a need to discuss them in future research.

All identified and analysed limitations of genetic algorithms can be divided into two categories: technic-specific limitations and ones defined by project tasks — all identified limitations described in this section relate to relevant problems of the knowledge area.

TABLE I
PROJECT MANAGEMENT KNOWLEDGE AREA AND PROBLEMS IN THE ARTICLE

Knowledge area	Project management problems in the article	Source
Project Cost Management	• Cost estimation management and minimisation problems	[23]–[26]
Project Resource Management	• Resource, Project, Team: levelling and planning problems • Project, Resource, Team, Activities, Tasks: planning and scheduling problems • Resource-constrained project (one and multi) scheduling problems	[2], [5], [21]–[23], [27]–[36]
Project Risk Management	• Project risk identification, scheduling, and management	[13], [17], [37],
Project Schedule Management	• Effort estimation problems • Project scheduling conflicts, identifications and resolutions problems • Project scheduling strategy problems • Resource, Project, Team: levelling and planning problems • Project, Resource, Team, Activities, Tasks: planning and scheduling problems • Resource-constrained project (one and multi) scheduling problems • Critical path problem in software planning and management problems • Business process: optimisation and performance problems • Software bug management problems	[2] - [12], [14], [16], [18]–[25], [27], [29], [31]–[33], [35], [36], [38]–[45]

When solving Project, Resource, Team, Activities or Tasks: Planning and Scheduling problems with genetic algorithm, most of the limitations, which are possible to identify, are task-based genetic algorithm limitations.

Task-specific limitations can be classified into three major groups: (1) project team related; (2) associated to project, tasks, and activities; (3) cost related.

Limitations related to project team are based on a lack of additional and credible information, such as to (1) identify the degree to which a member of the development team has or does not have a specific skill based on project tasks [36]; (2) measure the difference between the personality characteristics, which are required to perform a task, and the personality characteristics for completing tasks for each assigned member of the development team [36]; (3) create and keep an up-to-date list of project team competences [36].

Project, Task and Activity-based limitations include limitations, such as (1) a lack of methods to assign priorities for activities (at the moment, the authors use the random principle) [38]; (2) the need for a project to finish by the deadline [7], [8] and the way to meet the project deadline [8]; (3) specific nature of a model, which is project-dependent and mostly cannot be used in other projects or different areas of knowledge [38]; (4) task/activity execution sequences, defined start and end activities, and their due dates [11], but more importantly the duration of the task, as well as the preservation of task relations [8]; (5) preventing the situation when development team members need performing more than one task at the same time in the project [36]; (6) employees not working overtime (however, there are situations when some people work more than 8 hours per day) [12]; avoiding the over-booking of the resources [8]; (7) no relation between the number of human resources participating in one activity and the effort for communication [12], the classification of human resources based on their skills as well as price for a time unit defined by a certain group [8]; (8) the number of developers (human resources) assigned to a task, which results in resources that are "wasted" on tasks that can be accomplished with fewer development team members, as well as reducing communication between development team members [24], [36].

Cost-based limitations include costs for the projects and project human resources that should be met during the project (appear at the beginning of the tasks, but the contractor gets paid at the end of the project) [8].

However, technical limitations are frequently the same for standard GA and GA proposed and improved by authors. This group of limitations includes:

- Population size [12], [19], [38], [35];
- Maximal limit of iterations [12], [35], [38];
- Application of only one or two selection methods (not all) [7];
- Probability of crossover operator, mutation operator and weight vector limitations [12], [35], [38], [41];
- The algorithm routes or generations limitations [41];
- A limited number of activities/tasks and a specific sequence for performing the tasks [38], [41];
- Fitness function value calculation model [41].

Resource-Constrained Project (one and multi) Scheduling problems (5 out of 33 articles) and Effort Estimation problems (4 out of 33 articles) are the second type of most common project management problems.

As before, the relevant articles are categorised into the categories detailed above.

Overall, 1 standard genetic algorithm and 8 advanced genetic algorithms are used to solve Resource-Constrained Project (one and multi) Scheduling and Effort Estimation problems.

In the articles, it is possible to observe that the GA chromosome encoding strategies are project-specific input data types, for example, Priorities, Delay Time Release Dates [10], Activities [20], Activity lists [32], [5] and Edges [20], Actual Effort [3], Estimated Effort [3], Historical Project Information [3], Effort Drivers (development team effort, development team support, computer operation involvement, end-users or clients) [16].

The most applied types of chromosome encoding in the genetic algorithm are binary code [10], [44], graph nodes [20], [44], and integer number string [5], [32]. Four of the reviewed articles ([3], [16] [31], [42]) do not suggest chromosome encoding methods.

When solving Resource-Constrained Project (one and multi) Scheduling and Effort Estimation problems, a genetic algorithm mostly displays technical limitations. All technical limitations are usually the same for standard GA and GA improved by authors. This group includes the following limitations:

- Choosing and running the appropriate population size [3];
- The algorithm step size limitations [5];
- The algorithm route or generation limitations [10];
- Algorithm improvement capabilities [31], [32];
- Computational time and power (computer performance limitations) [42];
- Independent experiment run limitations [42].

However, task-based limitations can be classified into two categories: Project, Task, Activity-based and GA model-based limitations.

Project, Task, Activity-based limitations include execution sequences of tasks/activities [20], [44]. GA model-based limitations include Fuzzy Analogy algorithm limitations [16]. The definition of labels for the fuzzy sets is generated in order to obtain the expected interpretation of Fuzzy Analogy and allow Fuzzy Analogy to deal with categorical data other than linguistic value.

There are also other project planning and management problems, which are not described in articles frequently:

- Project scheduling conflicts, identification and resolution problems (addressed in 1 out of 33 articles);
- Project scheduling strategy problems (addressed in 1 out of 33 articles);
- Resource, Project Team: levelling and planning problems (addressed in 2 out of 33 articles);
- Critical path problem in software planning and management problems (addressed in 1 out of 33 articles);
- Business process optimisation and performance problems (addressed in 2 out of 33 articles);
- Software bug management problems (addressed in 1 out of 33 articles).

When researching solutions to this problem area, the studies were also divided based on methods: standard and improved GAs. To solve Project Scheduling Conflict Identification and Resolution problems, Project Scheduling Strategy problems, Resource, Project Team levelling and planning problems, as well as Business process optimisation and performance optimisation problems, the authors used GAs specifically modified for these problems (5 articles).

However, the Critical path problem in software planning and management, Business process optimisation and performance optimisation problems and Software Bug management problems were solved using the standard GAs (3 articles).

In the articles, we see that the GA chromosome encoding strategies are project-specific, for example, project activities [2], [9], [14], [39] and tasks [14], [21], [39], process models [39], historical project information, resources [39], work packages and package list [4], as well as project team [4].

The most popular types of chromosome encoding for the genetic algorithm are binary code [2], [9], integer number string [4], nets or graph nodes [2], [14], [39].

Two of the reviewed articles [21], [40] do not define chromosome encoding methods.

Similar to other Project Schedule Management problems, the most common limitations are the task-based genetic algorithm limitations. These limitations can be divided into three categories: (1) Project team-based limitations; (2) Project, Task, Activity-based limitations; (3) Cost-based limitations.

Project team-based limitations include (1) project resources that are limited and cannot be overbooked when assigning tasks [9]; (2) the number of human resources that participate in one activity does not reflect the effort for communication [39]; (3) randomly specified execution speed of each employee [39].

Project, Task, Activity-based limitations include limitations related to meeting the project deadline [9] and project conflicts, for example:

- Managing an activity that does not exist [9];
- Developing an activity that does not exist [9];
- First activity conflicting with the other activity [9];

- Executing an activity at a location, which does not exist at different places [9];
- Activity categorisation limitations [4];
- Software process valid solution limitations [39].

Cost-based limitations described in the article are, for example, restriction for extra project costs [39].

Technical limitations are frequently the same for standard GA and GAs improved by authors. This group includes the following limitations:

- Population size [14], [21], [39], [40];
- Maximum limit of iterations [14], [21], [39], [40];
- Crossover operators, mutation operator probability limitations and weight vector limitations [14], [39], [40];
- Application of only one or two selection methods (not all) [2];
- Limited number of activities/tasks and a specific sequence for performing the tasks [39];
- Computational time and power (computer performance limitations) [21];
- Independent experiment run [21].

The third most analysed knowledge area in the reviewed articles is Project Resource Management, and most articles address the problem of Project, Resource, Team, Activity, Task planning and Scheduling problems (in 10 out of 19 articles).

The articles in this knowledge area are divided into two categories: (1) articles that review standard genetic algorithms; (2) articles that review genetic algorithms proposed and developed by the authors.

During the analysis process, the articles on the knowledge area are divided based on the methods: 2 of the articles apply standard genetic algorithms, and 8 apply genetic algorithms proposed and improved by the authors.

In this case, the GA chromosome encoding strategies are project-specific input data types, for example, a specific mode for each activity [12], [34], [38]; tasks [22], [29], [35], [36]; task priorities [12], [35], [46] and activity list [12], [33], [38]; assigned resource list [33] and human resources [22], [23], [29], [34]; employees' knowledge and competences [29], [36] and work packages [46].

The most applied types of chromosome encoding in the genetic algorithm are binary code [12], [23] [29], [34], [35], [46] and graph nodes [22], [38]. In one case, authors also use their own encoding methods [33]. One of the reviewed articles [36] does not suggest chromosome encoding methods.

In Project, Resource, Team, Activities, Tasks: Planning and Scheduling problems, it is possible to identify task-based genetic algorithm limitations. This type of limitations can be divided into three types: (1) GA model-based limitations; (2) Project team-based limitations; (3) Project, Task and Activity-based limitations.

GA model-based limitations, which are presented in the articles, include underestimating implementation potential for use. Some articles do not discuss real problem solving, but the experiment described is based on realistic data. The results prove that 100 % positive evaluation is not achieved [29].

Project team-based limitations are the following limitations: (1) the activity prioritisation and process assigning to human resource limitations [38]; (2) no relation between the number of the human resources participating in one activity and the effort for communication [12]; (3) some people working more than 8 hours per day [12]; (4) the need to compare and estimate the technical skill levels [36]; (5) a lack of update for a list of project team competences [23], [36].

Project, task, and activity-based limitations include two limitations: (1) the lack of conditions on the sequence of activities, and the necessity for a development team member to perform more than one task at a time throughout the project [36]; and (2) the number of developers (human resources) assigned to a task, which results in resources being "wasted" on tasks that can be accomplished with fewer development team members, as well as reducing communication between development team members [33], [36].

Technical limitations are frequently the same for standard GA and GA proposed and improved by authors. This group includes the following limitations:

- Population size limitations (including activities and tasks as population) [35], [38], [46];
- The maximum number of iterations/generations [35], [38], [46];
- Data set size for model training [22];
- Objective function limitations [34];
- One selection method [35];
- Crossover rate and mutation rate limitations [35], [46].

The articles also include Resource-Constrained Project (one and multi) Scheduling problems (in 6 out of 19 articles) and Resource, Project Team: levelling and planning problems (in 6 out of 19 articles).

Articles devoted to Resource-Constrained Project (one and multi) Scheduling and Resource, as well as Project Team Analysing can be divided into groups based on the methods used: 1 article that discusses the problem of using a standard genetic algorithm, and 10 articles (assuming one of the reviewed articles deals with both method groups) that review genetic algorithms proposed and improved by authors.

In this case, the GA chromosome encoding strategies are project-specific input data, for example, priorities [10]; delay time [10], [28] and release dates [10]; tasks [21], [22], [29], [30]; project-specific rules [15], [27], [28]; employee DB, employees' knowledge and competences [15], [29]; human resources [22], [27], [28], [29]; scheduled activities and activity lists [5], [31], [32].

The most applied types of chromosome encoding approaches in the genetic algorithm are graphs, nets or tree nodes [22], [27], [30], [31] or integer number string [5], [28], [32]. Authors also use chromosome encoding approaches [10], Island-based encoding method [15] and binary code [29]. In one of the reviewed articles [21], the authors do not suggest a chromosome encoding method.

The most common limitations in solutions for Resource-Constrained Project (one and multi) Scheduling problems and Resource, Project Team: levelling and planning problems are Task-based genetic algorithm limitations.

GA model-based limitations include: (1) the algorithm not being tested on real data [29], [31]; (2) underestimated implementation potential for use (real problem solving is not discussed, but the described experiment is based on realistic data; the results show that 100 % positive evaluation is not achieved) [29], [31]; (3) a lack of historical database, which describes the previous algorithm activity [28]; (4) suitability of the algorithm for one specific project and a lack of customisation as a universal solution (there are different assumptions and dependencies in each project, which may not be required in other cases/projects) [28]; (5) a lack of indication of necessary time and resources to solve problems [27]; (6) a lack of evaluation of the implementation potential for use [27].

Project team-based limitations include the need to adapt part of the genetic algorithm model to additional project team optimisation [15] and keep an up-to-date list of project team competences [15].

The observed Project, Task and Activity-based limitations are the following: (1) a lack of automated conversion of project specifications into fuzzy logic criteria (the necessity to use text context in order to automate the redesign of project specification criteria) [15]; (2) few projects where the method would be suitable not only in project start-up phase [15]; (3) incomplete detailing and design specifications, which do not allow for a full review and identification of the required knowledge and skills areas of the project human resources [15].

The technical limitations for the standard and adapted GAs are the following:

- Backward–forward step size limitations [5];
- The maximum number of iterations/generations [10], [21], [27], [30];
- Mutation rate and crossover rate limitations [10], [21], [22], [27];
- The number of population limitations [21], [27], [30];
- Computational time and power (computer performance limitations) [5], [21];
- Limited independent experiment run [21];
- The need for more records in the dataset. Limit on the number of Activities, Tasks, Resources to use the algorithm [21];
- Algorithm improvement opportunities [31], [32].

Project Cost Management problems and Project Risk Management problems are solved less often using genetic algorithms. Within these areas of knowledge, two decision-making problems in project management are identified: Cost Estimation, Management and Minimisation problems (identified in 4 articles) and Project Risk Identification, Scheduling and Management problems (identified in 3 articles).

All four articles describe the application of the genetic algorithms proposed and improved by the authors.

In this case, GA chromosome encoding strategies are project-specific input data types, for example, tasks and activities [23], [24], effort and effort availability [25], [26], human resources [23], and project-specific rules [26].

The most applied type of chromosome encoding is binary code [23], [24], [26]. One of the reviewed articles [25] does not suggest a specific chromosome encoding method.

In Cost Estimation, Management and Minimisation problem-solving, genetic algorithms have task-based limitations. This type of limitations can be divided into three types: (1) GA model-based limitations; (2) Project team-based limitations; (3) Project, Task and Activity-based limitations.

GA model-based limitations are the following: (1) software subject to failures during execution caused by faults remaining in the software [25]; (2) testing of the algorithm not in real projects but rather using datasets taken from multiple projects [26]; (3) dependency of less reliability growth during the testing phase upon the testing efforts spent on testing [25].

Project, Task and Activity-based limitations include: (1) activities in diagrams cannot be divided, and the resource rates are uniform throughout each activity duration [23]; (2) the workload for each activity will not change during the optimisation process [23]; (3) the method has not investigated the relationship between effort estimation and time allocation [25]; (4) during the fault removal process, new faults can be generated, and the fault generation rate is proportional to the rate of fault removal [25]; (5) the automated method for parameter optimisation needs to be improved [26], and projects performed in optimal working sequences and the resource optimisation process do not change the relationship between activities [23]; (6) For a specific project task, there is a substitution relationship between two different types of resources with a loss of work productivity [23]; (7) The work-fit area is defined in working hours, excluding projects that do not fit in this defined area [26].

The cost-based limitation is related to project costs, which are often dependent on the number of resources used for the project (the fewer resources, the better) [24]. There are unlimited resources available for the project, and there are idle costs for standby resources and costs for resource organisations [23].

Technical limitations are the following:

- Population size [25];
- The application of only one or two selection methods (not all) [25], [26];
- The maximum number of iterations/generations [25].

In Project Risk Management, the authors use standard genetic algorithms (overall in 1 article) and genetic algorithms improved and proposed by authors (in 2 articles).

The GA chromosome encoding strategies used in these cases are project-specific input data types, for example, costs [13], [17], tasks/activities [13], [17], and project risks [17], [37].

The most commonly applied type of chromosome encoding approaches is integer number string [13], [17]. Chromosome encoding types are also used, such as graphs, nets or tree nodes [37].

However, in Project Risks: identification, scheduling and management problems, when they are solved using a genetic algorithm, the most common limitations are technical ones.

This group includes the following limitations:

- Population size [13], [17], [37];
- The application of only one or two selection methods (not all) [13];
- Iterations/generation size [13], [37];

- Probability, mutation rate and crossover rate limitations [13], [37];
- Fitness value limitations [13].

Task-based limitations can be divided into two groups: (1) Project, Task and Activity-based limitations, and (2) Cost-based limitations.

Project, Task, Activity-based limitation encountered in the article is related to the lack of parallel task execution, which affects algorithm performance by increasing project time [17].

Cost-based limitation, occurring in the article, is cost determination (the algorithm uses a wide range of costs) [17].

IV. Opportunities, Limitations and Solutions in Project Management Decision-Making Process

By collecting and analysing the available information about the application of the genetic algorithms, the authors of the present article identified the domain-related problems and limitations imposed by limited computation resources and the shortcomings of genetic algorithms.

Although several problems have been identified, the current state of research does not propose specific solutions. The authors believe that many of them can be solved by improving domain knowledge and the further implementation of genetic algorithms and herein propose the directions of future research.

In the previous section, the answers to research questions RQ1 and RQ2 were discussed. In this section, research questions RQ2 and RQ3 are addressed. The review of the obtained results is presented in Table II.

The most frequently identified domain-based limitations of the genetic algorithm application are Cost-based limitations, input values and Project, Tasks or Activity-based limitations.

The domain-based problems can only be approached by compromising between effort and benefits. For example, Project, Task or Activity-based limitations often include wrong sequence of the tasks and activities.

TABLE II
Project Management Knowledge Area and Genetics Algorithm Limitations

Project management problems in the article	Limitations	Type of limitations	Source
• Cost estimation management and minimisation problems	Input values and algorithm performance limitations	Technical limitations	[25], [26]
	GA model-based limitations	Task limitations	[25], [26]
	Project, Task or Activity-based limitations	Task limitations	[23], [25], [26]
	Cost-based limitations	Task limitations	[23]
• Resources, project team: levelling and planning problems • Project, Resource, Team, Activities, Tasks: planning and scheduling problems • Resource-constrained project (one and multi) scheduling problems	Input values and algorithm performance limitations	Technical limitations	[5], [10], [21], [22], [27], [30]–[32], [34], [35], [38], [46]
	GA model-based limitations	Task limitations	[28], [29], [31]
	Project team-based limitations	Task limitations	[12], [15], [23], [36], [38]
	Project, Task or Activity-based limitations	Task limitations	[15], [33], [36]
	Backward–forward step size limitations	Technical limitations	[5]
• Project risk identification, scheduling and management	Input values and algorithm performance limitations	Technical limitations	[13], [17], [37].
	Project, Task or Activity-based limitations	Task limitations	[17]
	Cost-based limitations	Task limitations	[17]
• Effort estimation problems • Project scheduling conflicts, identification and resolution problems • Project scheduling strategy problems • Resource, Project, Team: levelling and planning problems • Project, Resource, Team, Activities, Tasks: planning and scheduling problems • Resource-constrained project (one and multi) scheduling problems • Critical path problem in software planning and management problems • Business process: optimisation and performance problems • Software bug management problems	Team-based limitations	Task limitations	[9], [36], [39]
	Project, Task or Activity-based limitations	Task limitations	[4], [7]–[9], [11], [12], [20], [24], [36], [38], [44]
	Cost-based limitations	Task limitations	[39]
	Input values and algorithm performance limitations	Technical limitations	[2], [3], [5], [7], [10], [12], [14], [19], [21], [31], [32], [35], [38] - [42]
	GA not solving a practical problem	Task limitations	[18], [45]
	GA model-based limitations	Task limitations	[8], [16]

This can be controlled by implementing additional checks of chromosomes against rules: setting dependencies with other tasks and activities for each tack/activity (which have to be finished before the start of the task/activity, and which tasks/activities can only be started when the specific task/activity is finished.

The Cost-based limitations call for additional checks against acceptable overdraft limits, which can be set by a project manager, and budget breakdown preferences, e.g., setting equal limits for all project phases or introducing other preferences based on the cash-flow.

Project team-based limitations ask for a database of the project team (competences/workloads etc.).

The algorithm implementation-related limitations can be solved by either increasing the computational resources or compromising between detailing and speed. While computational resource availability is growing with every year, the resources are not unlimited. The compromise has to be reached between chromosome encoding (the detailing necessary in the solution: how many characteristics of each variable should be included), population size, proportion of individuals in crossover, and number of generations. Any of these, when being significantly increased, consume resources. Therefore, if a detailed chromosome is necessary, population size could be decreased, while generating more populations and making sure that the best solutions are not lost (e.g., using elitism etc.).

V. Conclusion and Discussion

The studies discussed and analysed in this paper use genetic algorithms for decision-making in project management. In the research process, we used two scientific repositories and 6 keyword phrases. Then 45 articles were selected after paper abstract analysis, duplicate audit, full-text article analysis and selecting literature in the period from 2010. Within the framework of the study, 45 scientific articles were investigated within the literature analysis finding answers to the research questions: RQ1, RQ2 and RQ3.

The most frequently identified domain-based limitations of the genetic algorithm application are Cost-based limitations, input values and Project, Tasks or Activity-based limitations. The domain-based problems still remain the main issue and they can only be approached by compromising between effort and benefits.

Although the technology and knowledge advances are paving the way for the return of the genetic algorithm in this field, the required skills to implement such a solution (genetic algorithm with the most efficient problem encoding into chromosomes) is another drawback that is not being lifted. This can be an expensive approach for a tool that helps with decision support. A possible solution could be a commercial tool that can be used by different companies but here we face the biggest challenge: the solutions presented in research articles so far are very specific to the problem they are built to solve. Although there are characteristics that are the same for all software development projects, there can be different preferences and specifics. It is the most important problem with the application of genetic algorithms for project management decision support in the coming decades.

The articles describe various project management artefacts, which are mentioned before when identifying algorithm input data. Planning of activities and the level of details in tasks depend on the main aim of the project management task. Therefore, algorithm application in project management can require a predefined clear aim. A more detailed project activity and task definition would improve the order of task assignment for the management of the people involved in a project [38]. Plans of project activities have to include the order of activity execution, while still conforming to the project deadlines. Analysis according to the current industry practice, when planning activities, requires identification of the minimum (pessimistic) and maximum (optimistic) critical path of the project activities, which would allow evaluating the execution of activities and their impact on other project artefacts: time, costs, human resources and quality. However, it is impossible to lift limitations, such as capacity of an employee to execute several tasks at the same time [36] and the 8-hour working days [12]. When planning activities, it is important not only to focus on the competences of resources involved in the project and their ability to work in different circumstances. In the perfect situation of project execution, the project manager, when building the project team, chooses team members creating his/her own perfect matrix of resources. However, in reality, when planning resources, one can choose only the available human resources. Though, some people will work more than 8 hours per day [12], it cannot be accepted as a widespread norm. Each employee is different, and their work capacities and task completion rates are different. The manager also has to consider other risks that can occur due to human factors, e.g., missing work due to health problems, vacations etc., and another important factor: the microclimate within a project team.

In order to reduce the limitations of the algorithm, the list of competences of a project team has to be constantly updated, evaluating differences between the personality characteristics required to complete the task, the personality traits, which are required to complete the task by each assigned development team member, and skills mismatches for the development team member and a specific project task [23], [36].

When choosing project team members, it is necessary to take into account not only their competences for carrying out tasks and activities, but also their interests. This can be done by including relevant interests of employees and previous project experiences into the project team database. When evaluating workload of each project team member, it is important to assess the ability of the employee to carry out the work in the designated period of time. When the task is limited in time, the best choice would be an employee who could carry out the task not only based on their competence but also on their previous interests and working capacities and agility.

It is also important to create knowledge bases that include historical data, which would allow exchanging and accumulating experience about project activities, risks and another knowledge.

The analysis of the scientific articles shows that the previous studies do not scrutinise the interaction and connections among genetic algorithms, their objects, operations and results, and project management. The studies mostly focus on the application of genetic algorithms, but the authors do not convey how to use them to support decisions in software development projects.

In the future, it is necessary to combine all of the relevant knowledge areas in one application of genetic algorithms, which utilises the maximum of domain knowledge (parallel algorithms that use global evaluation at stopping points, which would evolve solutions in different areas that match and together make a suitable solution).

References

[1] Project Management Institute. *A Guide to the Project Management Body of Knowledge* (PMBOK® Guide)-Sixth Edition. Project Management Institute, 2017. [Online] Available: https://www.pmi.org/pmbok-guide-standards [Accessed: September 26, 2019].

[2] M. Karova and N. Avramova, "A genetic algorithm basic approach for software management project," *Proceedings of the 13th International Conference on Computer Systems and Technologies - CompSysTech '12*, pp. 103–110, 2012. https://doi.org/10.1145/2383276.2383293

[3] S.-J. Huang, N.-H. Chiu, and L.-W. Chen, "Integration of the grey relational analysis with genetic algorithm for software effort estimation," *Eur. J. Oper. Res.*, vol. 188, no. 3, pp. 898–909, Aug. 2008. https://doi.org/10.1016/j.ejor.2007.07.002

[4] Y, Zhou and Y. Chen, "Business process assignment optimization, " *IEEE International Conference on Systems, Man and Cybernetics*, Yasmine, vol. 3 Hammamet, Tunisia, 2002, pp. 6. [Online] Available: https://ieeexplore.ieee.org/abstract/document/1176100 [Accessed: September 26, 2019].

[5] A. Agarwal, S. Colak, and S. Erenguc, "A Neurogenetic approach for the resource-constrained project scheduling problem," *Comput. Oper. Res.*, vol. 38, no. 1, pp. 44–50, Jan. 2011. https://doi.org/10.1016/j.cor.2010.01.007

[6] J. Kroll, S. Friboim, and H. Hemmati, "An empirical study of search-based task scheduling in global software development," in *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track, ICSE-SEIP 2017*, 2017., pp. 183–192. https://doi.org/10.1109/ICSE-SEIP.2017.30

[7] S. Nesmachnow, "Efficient parallel evolutionary algorithms for deadline-constrained in project management scheduling," *Int. J. Innov. Comput. Appl.*, vol. 7, no. 1, pp. 34–49, Mar. 2016. https://doi.org/10.1504/IJICA.2016.075468

[8] M. Karova, I. Penev, G. Todorova, and M. Todorova, "Genetic Algorithm for Managing Project Activities System," in *Proceedings - 2014 International Conference on Mathematics and Computers in Sciences and in Industry, MCSI 2014*, 2014., pp. 267–271. https://doi.org/10.1109/MCSI.2014.46

[9] M. Ramzan, A. Jaffar, A. Iqbal, S. Anwar, A. Rauf, and A. A. Shahid, "Project scheduling conflict identification and resolution using genetic algorithms (GA)," *Telecommun. Syst.*, vol. 51, no. 2–3, pp. 167–175, Nov. 2012. https://doi.org/10.1007/s11235-011-9426-3

[10] J. F. Gonçalves, J. J. M. Mendes, and M. G. C. Resende, "A genetic algorithm for the resource constrained multi-project scheduling problem," *Eur. J. Oper. Res.*, vol. 189, no. 3, pp. 1171–1190, Sep. 2008. https://doi.org/10.1016/j.ejor.2006.06.074

[11] Y. Ge and C. Chang, "Capability-based project scheduling with genetic algorithms," in *CIMCA 2006: International Conference on Computational Intelligence for Modelling, Control and Automation, Jointly*, pp. 161–161, 2006. https://doi.org/10.1109/CIMCA.2006.63

[12] J. Xiao, Q. Wang, M. Li, Q. Yang, L. Xie, and D. Liu, "Value-based multiple software projects scheduling with genetic algorithm," in *Conference on Software Process, ICSP, 2009*, vol. 5543 LNCS, 2009., pp. 50–62. https://doi.org/10.1007/978-3-642-01680-6_7

[13] F. Reyes, N. Cerpa, A. Candia-Véjar, and M. Bardeen, "The optimization of success probability for software projects using genetic algorithms," *J. Syst. Softw.*, vol. 84, no. 5, pp. 775–785, May 2011. https://doi.org/10.1016/j.jss.2010.12.036

[14] M. M. Rosli, N. H. I. Teo, N. S. M. Yusop, and N. S. Mohammad, "The design of a software fault prone application using evolutionary algorithm," in *2011 IEEE Conference on Open Systems*, Langkawi, 2011, pp. 338–343. https://doi.org/10.1109/ICOS.2011.6079246

[15] D. Strnad and N. Guid, "A fuzzy-genetic decision support system for project team formation," *Appl. Soft Comput. J.*, vol. 10, no. 4, 2010, pp. 1178–1187. https://doi.org/10.1016/j.asoc.2009.08.032

[16] F. A. Amazal, A. Idri, and A. Abran, "Software development effort estimation using classical and fuzzy analogy: a cross-validation comparative study," *Int. J. Comput. Intell. Appl.*, vol. 13, no. 03, p. 1450013, Sep. 2014. https://doi.org/10.1142/S1469026814500138

[17] J. Pfeifer, K. Barker, J. E. Ramirez-Marquez, and N. Morshedlou, "Quantifying the risk of project delays with a genetic algorithm," *Int. J. Prod. Econ.*, vol. 170, pp. 34–44, Dec. 2015. https://doi.org/10.1016/j.ijpe.2015.09.007

[18] J. F. Gonçalves, M. G. C. Resende, and J. J. M. Mendes, "A biased random-key genetic algorithm with forward-backward improvement for the resource constrained project scheduling problem," *J. Heuristics*, vol. 17, no. 5, pp. 467–486, Oct. 2011. https://doi.org/10.1007/s10732-010-9142-2

[19] L-Y. Tseng, S-C. Chen, "A hybrid metaheuristic for the resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 175, no. 2, pp. 707–721, Dec. 2006. https://doi.org/10.1016/j.ejor.2005.06.014

[20] C. Poloni, D. Quagliarella, J. Périaux, N. Gauger, K. Giannakoglou, and F. Gargiulo, "A Hybrid Genetic Algorithm For The Resource Constrained Project Scheduling Problem (RCPSP)," in *Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*, 2011, pp. 244–252. [Online] Available: https://www.researchgate.net/publication/266327628_A_Hybrid_Genetic_Algorithm_For_The_Resource_Constrained_Project_Scheduling_Problem_RCPSP [Accessed: September 26, 2019].

[21] A. Tzanetos, C. Kyriklidis, A. Papamichail, A. Dimoulakis, and G. Dounias, "A Nature Inspired metaheuristic for Optimal Leveling of Resources in Project Management," in *Proceedings of the 10th Hellenic Conference on Artificial Intelligence - SETN '18*, 2018, pp. 1–7. https://doi.org/10.1145/3200947.3201014

[22] S. Samath, D. Udalagama, H. Kurukulasooriya, D. Premarathne, and S. Thelijjagoda, "Collabcrew-An intelligent tool for dynamic task allocation within a software development team," in *International Conference on Software, Knowledge Information, Industrial Management and Applications, SKIMA*, Malabe, 2017, pp. 1–9. https://doi.org/10.1109/SKIMA.2017.8294131

[23] M.-R. Yan, "Evolutionary optimization model for managing project changes with minimum cost," in *Sixth International Conference on Natural Computation*, Yantai, 2010, pp. 3978–3982, 2010. https://doi.org/10.1109/ICNC.2010.5584799

[24] M. Yan, "Resource-based optimization model for dynamic project planning and cost management," *Appl. Math. Inf. Sci.*, vol. 11, no. 4, pp. 1091–1096, Jul. 2017. https://doi.org/10.18576/amis/110415

[25] P. K. Kapur, A. G. Aggarwal, K. Kapoor, and G. Kaur, "Optimal testing resource allocation for modular software considering cost, testing effort and reliability using genetic algorithm," *Int. J. Reliab. Qual. Saf. Eng.*, vol. 16, no. 06, pp. 495–508, Dec. 2009. https://doi.org/10.1142/S0218539309003538

[26] D. Milios, I. Stamelos, and C. Chatzibagias, "A genetic algorithm approach to global optimization of software cost estimation by analogy," in *Intelligent Decision Technologies*, vol. 7, no. 1, 2013, pp. 45–58. https://doi.org/10.3233/IDT-120150

[27] C. Kyriklidis and G. Dounias, "Application of Evolutionary Algorithms in Project Management," in *Artificial Intelligence Applications and Innovations*, vol. 436, 2014, pp. 335–343. https://doi.org/10.1007/978-3-662-44654-6_33

[28] M. Di Penta, M. Harman, and G. Antoniol, "The use of search-based optimization techniques to schedule and staff software projects: An approach and an empirical study," *Softw. - Pract. Exp.*, vol. 41, no. 5, pp. 495–519, Apr. 2011. https://doi.org/10.1002/spe.1001

[29] M. Hameed, H. Khalid, U. Qamar, and S. K. Abass, "Optimizing software project management staffing and work-force deployment processes using swarm intelligence," in *Proceedings of Computing Conference*, pp. 78–84, 2017. https://doi.org/10.1109/SAI.2017.8252084

[30] K. Pawiński and K. Sapiecha, "An Efficient Solution of the Resource Constrained Project Scheduling Problem Based on an Adaptation of the Developmental Genetic Programming," in *Recent Advances in Computational Optimization: Results of the Workshop on Computational Optimization WCO 2014*, S. Fidanova, Ed. Cham: Springer International Publishing, 2016, vol. 610, pp. 205–223. https://doi.org/10.1007/978-3-319-21133-6_12
[31] R. Zamani, "An evolutionary implicit enumeration procedure for solving the resource-constrained project scheduling problem," *Int. Trans. Oper. Res*., vol. 24, no. 6, pp. 1525–1547, Nov. 2017. https://doi.org/10.1111/itor.12196
[32] V. Valls, F. Ballestín, and S. Quintanilla, "A hybrid genetic algorithm for the resource-constrained project scheduling problem," *Eur. J. Oper. Res*., vol. 185, no. 2, pp. 495–508, March 2008. https://doi.org/10.1016/j.ejor.2006.12.033
[33] V. Yannibelli and A. Amandi, "A knowledge-based evolutionary assistant to software development project scheduling," *Expert Syst. Appl.*, vol. 38, no. 7, pp. 8403–8413, Jul. 2011. https://doi.org/10.1016/j.eswa.2011.01.035
[34] C. Stylianou and A. S. Andreou, "A multi-objective genetic algorithm for software development team staffing based on personality types," in *IFIP Advances in Information and Communication Technology*, vol. 381, no. PART 1, 2012, pp. 37–47. https://doi.org/10.1007/978-3-642-33409-2_5]
[35] C. Stylianou and A. S. Andreou, "Intelligent software project scheduling and team staffing with genetic algorithms," in *IFIP Advances in Information and Communication Technology*, vol. 364, no. PART 2, 2011, pp. 169–178. https://doi.org/10.1007/978-3-642-23960-1_21
[36] C. Stylianou, S. Gerasimou, and A. S. Andreou, "A novel prototype tool for intelligent software project scheduling and staffing enhanced with personality factors," in *Proceedings - International Conference on Tools with Artificial Intelligence*, vol. 1, 2012, pp. 277–284. [Online] Available: https://ieeexplore.ieee.org/document/6495057 [Accessed: September 26, 2019].
[37] C. Li, P. Li, and G. Lu, "Enterprise projects set risk element transmission chaotic genetic model," *Res. J. Appl. Sci. Eng. Technol*., vol. 4, no. 17, 2012, pp. 3162–3167. [Online] Available: https://www.semanticscholar.org/paper/Enterprise-Projects-Set-Risk-Element-Transmission-Li-Li/6b770e3d32afdcb8ab4f4f4c4edd412416c5c0a8 [Accessed: September 26, 2019].
[38] H. Li and X. Dong, "Multi-mode resource leveling in projects with mode-dependent generalized precedence relations," *Expert Syst. Appl*., vol. 97, pp. 193–204, May 2018. https://doi.org/10.1016/j.eswa.2017.12.030
[39] X. Liu, Y. Yang, J. Chen, Q. Wang, and M. Li, "Achieving on-time delivery: A two-stage probabilistic scheduling strategy for software projects," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5543, 2009, pp. 317–319. https://doi.org/10.1007/978-3-642-01680-6_29
[40] Y. Zhou and Y. Chen, "Project-oriented business process performance optimization," in *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance*, vol. 5, 2004, pp. 4079–4084. [Online] Available: https://ieeexplore.ieee.org/document/1245626 [Accessed: September 26, 2019].
[41] S. J. Sadjadi, R. Pourmoayed, and M. B. Aryanezhad, "A robust critical path in an environment with hybrid uncertainty," *Appl. Soft Comput.*, vol. 12, no. 3, pp. 1087–1100, March 2012. https://doi.org/10.1016/j.asoc.2011.11.015
[42] F. A. Abulalqader and A. W. Ali, "Comparing Different Estimation Methods for Software Effort," in *2018 1st Annual International Conference on Information and Sciences (AiCIS)*, 2018, pp. 13–22. https://doi.org/10.1109/AiCIS.2018.00016
[43] A. García-Nájera and M. del Carmen Gómez-Fuentes, "A Multi-objective Genetic Algorithm for the Software Project Scheduling Problem," in *MICAI 2014: Nature-Inspired Computation and Machine Learning*, 2014, pp. 13–24. https://doi.org/10.1007/978-3-319-13650-9_2
[44] A. L. I. Oliveira, P. L. Braga, R. M. F. Lima, and M. L. Cornélio, "GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation," in *Information and Software Technology*, vol. 52, no. 11, 2010, pp. 1155–1166. https://doi.org/10.1016/j.infsof.2010.05.009
[45] N. Bhasin and N. Gupta, "Critical Path Problem for Scheduling Using Genetic Algorithm," in *Soft Computing: Theories and Applications*, 2018, pp. 15–24. https://doi.org/10.1007/978-981-10-5687-1_2
[46] G. Antoniol, M. Di Penta, and M. Harman, "Search-Based Techniques for Optimizing Software Project Resource Allocation," in *Genetic and Evolutionary Computation – GECCO 2004*, 2004, pp. 1425–1426. https://doi.org/10.1007/978-3-540-24855-2_162

Ieva Ancveire holds a Professional Master's degree in information technology and Qualification of Project Manager in information technology at Riga Technical University, Latvia, in 2016. Since 2016 she has been a PhD student at Riga Technical University. Her main areas of interests are ERP systems, business analysis, system analysis and project management.
E-mail: ieva.ancveire@edu.rtu.lv
ORCID iD: https://orcid.org/0000-0003-0925-4157

Inese Poļaka received her Doctoral degree (Dr. sc. ing.) in information technology from Riga Technical University in 2014. She has since worked at Riga Technical University and holds the position of Assistant Professor at the moment. Her fields of interest are machine learning, data mining, especially in medical applications, evolutionary algorithms, transparent and interpretable supervised and unsupervised learning methods, as well as bioinformatics (main focus on prokaryotic metataxonomics and metagenomics).
E-mail: inese.polaka@rtu.lv
ORCID iD: https://orcid.org/0000-0002-9892-7765