Bogumiła Krzeszowska

EVOLUTIONARY ALGORITHM WITH DIRECT CHROMOSOME REPRESENTATION IN MULTI-CRITERIA PROJECT SCHEDULING

Abstract

In recent years project scheduling problems became popular because of their broad real-life applications. In practical situations it is often necessary to use multi-criteria models for the evaluation of feasible schedules.

Constraints and objectives in project scheduling are determined by three main issues: time, resource and capital; but few papers consider all of them. In research on project scheduling the most popular is the problem with one objective. There are only few papers that consider the multi-objective project scheduling problem.

This paper considers the multi-criteria project scheduling problem. There are three types of criteria used to optimize a project schedule: resource allocation, time allocation and cost allocation. An evolutionary algorithm with direct chromosome representation is used to solve this problem. In this representation a chromosome is a sequence of completion times of each activity.

The purpose of this paper is to demonstrate how evolutionary algorithms can be used in multi-criteria project scheduling. The paper begins with an overview of previous literature and problem statement; after that there is direct chromosome representation description and at the end final results.

Keywords

Project scheduling, multi-criteria analysis, evolutionary algorithms, multi-criteria scheduling.

Introduction

A project is a unique set of co-ordinated activities, with definite starting and finish times, undertaken by an individual or organization to meet specific objectives within defined schedule, cost and performance parameters [1].

A project is also defined as a temporary (with defined beginning and end) endeavor undertaken to create a unique product or service [2]. The planning, monitoring and control of all aspects of a project to achieve the project objectives on time and to specified cost, quality and performance is called project management [1]. Each project has three main components: activities, resources and precedence relationships [4].

Activities are tasks to do. They build a project. An activity has an expected duration, an expected cost, and resource requirements.

Resources are required to carry out the project tasks. They can be people, equipment, facilities, funding, or anything else capable to perform an activity required for the completion of a project. Resources may be renewable or non--renewable. Renewable resources are available in each period without being depleted. Non-renewable resources are depleted as they are used.

Precedence relationships define the order in which activities should be performed. This order is specified. A precedence relationship is always assigned to activities based on the dependencies of each activity. There are four recognized precedence relationships: Finish-to-Start, Finish-to-Finish, Start-to--Start, Start-to-Finish.

Scheduling concerns the allocation of limited resources to tasks over time. It is a decision-making process that has a goal – the optimization of one or more objectives [9].

The project schedule lists times planned for performing activities. Any schematic display of the logical relationships of project activities can be presented as a network. There are two types of networks for project scheduling problems: AON (Activity On Node) and AOA (Activity On Arc). In an AON network activities are represented by nodes, and they are linked by the precedence relationship to illustrate the sequence in which activities should be performed. In an AOA network activities are presented by arrows. The tail of the arrow represents the start and the head represents the finish of the activity. Activities are connected at nodes to illustrate the sequence in which activities should be performed [10].

A project scheduling problem includes many types of constraints. Type of constraints and optimization criteria are determined by three main components: time, resources and capital. When we take them into consideration we can build various schedule optimizing models. The most popular are problems with one objective. We can build models without constraining time, cost or resources; in those models we can optimize project completion time or cash flows. We can also consider problems with one type of constraint, where we can optimize resource allocation or costs with constrained time, or optimize project completion time or costs with constrained resources allocation or else optimize project NPV with limited costs. Many authors consider problems with two constraint types. It is possible to build and solve a model with three types of constraints but so far there are no studies on it [4]. It is obvious that apart from those constraints the model can have other constraints, e.g. related to a precedence relationship.

Early efforts in project scheduling focused on minimizing the overall project duration (makespan). Scheduling problems have been studied extensively for many years to determine exact solutions by using methods from the field of operation research.

Due to the necessity of using multi-criteria models for evaluation of feasible schedules in practical situations, several methods have been proposed for multiple-criteria project scheduling. So far there are only a few papers that discuss multiple-criteria project scheduling problem.

Vina and de Sousa [11] solve a multiple-criteria project scheduling problem with three objectives. The first objective aims at minimizing project completion time, the second one, at minimizing the mean weighted lateness of activities, the third one at minimizing the sum of the violation of resource availability. They use also some constraints: to ensure that each activity is processed exactly once, that resource consumption of renewable and nonrenewable resources does not exceed the available quantities and that precedence conditions are fulfilled. They presented two heuristics in their paper: Pareto simulated annealing (PSA) and multiobjective taboo search (MOTS).

Lova, Maroto and Tormos [7] presented a multicriteria heuristic algorithm for multiproject scheduling with two phases. It starts from a feasible multiproject schedule and it improves lexicographicly two criteria: one of time type and one of no time type. In the first phase it obtains a good schedule for the multiproject with time criterion. In the second phase the multiple project schedule is improved with a no time criterion. In their paper the authors use the following time criteria: minimizing mean project delay, minimizing multiproject duration and no time criteria: minimizing project splitting, minimizing in-process inventory, maximizing resource leveling and minimizing idle resources.

Leu and Yang [6] proposed model with two optimization directions: minimizing project completion time and minimizing costs. To solve this problem they used GA- based multicriteria method.

Hapke, Jaszkiewicz and Słowiński [3] presented an interactive search for multi-criteria project scheduling. Their approach consists of two stages. In the first stage, a large representative sample of approximately non-dominated schedules is generated by the PSA method. In the second stage, an interactive search method is used. They use three criteria in their paper: minimizing project completion time, minimizing total project cost and minimizing the average deviation from the average resource usage.

The problem presented in this paper is a multi-criteria project scheduling problem in which the following three types of criteria are used to optimize the project schedule: project completion time, resource smoothness (presented as regularity of resource usage) and cost smoothness. To solve this problem an evolutionary algorithm with direct chromosome representation is used. In this representation a chromosome is a sequence of completion times of each activity.

This paper is organized as follows. Section 1 describes the problem of multiple-criteria project scheduling problem. The problem is stated. Section 2 introduces the evolutionary algorithms for project scheduling problems. The scheme of the algorithm and the description of the direct chromosome representation are presented. In Section 3, the results of the application of evolutionary algorithms for the scheduling problem is presented. Conclusions and ideas for future work are in find Section.

1. Problem statement

A multiple-criteria project scheduling problem is presented in this paper. The goal of this problem is to schedule project tasks so that they meet the constraints and optimize the schedule with respect to time, resource usage and costs generated. This problem can be formulated as follows.

We assume that:
- there is a project to schedule,
- project contains activities,
- project is presented on an AON network,
- each activity is described by a triple: duration, costs and resources,
- precedence relationships are of the Finish-to-Start type,
- costs are generated when an activity starts,
- there is one type of resources,
- resources are needed throughout the duration of an activity,
- we do not allow idle times.

The following notation is used:

$k = 1,...,T$ – set of periods,

$i = 1,...,I$ – set of activities,

S_i – the start time of activity i,

F_i – the finish time of activity i,

d_i – duration of activity i,

r_i – amount of resources required by activity i (in this case we assume that we have one type of resources),

c_i – cost generated by activity i.

The criteria functions can be presented as the following objectives:

1. $F_i \to \min$

2. $\max\limits_{k=1,...,T} \sum\limits_{i=1}^{I} r_{ik} \to \min$ for $i = 1,...,I$, $k = 1,...,T$

3. $\max\limits_{k=1,...,T} \sum\limits_{i=1}^{I} c_{ik} \to \min$ for $i = 1,...,I$, $k = 1,...,T$

The constraints are presented as follows:

4. $F_{i+1} \le F_i - d_i$

5. $r_{ik} \ge 0$ for $i = 1,...,I$, $k = 1,...,T$

   $c_{ik} \ge 0$ for $i = 1,...,I$, $k = 1,...,T$

   $F_i \ge 0$ for $i = 1,...,I$, $k = 1,...,T$.

The (1) goal is to minimize the total time it takes to process all tasks; to minimize the finish time of task i. In the criterion (2) we are minimizing the maximum resource usage in each time. This objective takes care of smoothness in resource allocation. The criterion (3) is minimizing the maximum cost level in each period. It takes care of smoothness in capital allocation. Criteria 2 and 3 are most often connected. Cost is resource usage expressed in money. Often in project management we consider separately the resource usage directly connected with project and the cost, which is understood in a wider sense. Additionally, there is also a formula expressing the precedence relations (4), and constraints about nonnegative variables (5).

2. Evolutionary algorithm for project scheduling problem

An evolutionary algorithm with direct chromosome representation is used in this paper. Below is a description of approach used.

Algorithm steps

In this paper we use a classical scheme of evolutionary approach, but with change in selection. The non-dominated solution goes automatically to the next generation. The scheme of the algorithm is presented below:

1. $t \to 0$
2. Set population $P(t)$
3. Evaluate $P(t)$
4. If condition of finish is fulfilled then end
5. $t \to t+1$
6. Chose $P(t)$ from $P(t-1)$
7. Change $P(t)$ using crossover and mutation
8. Evaluate $P(t)$ and go to 4.

The algorithm starts with a population generated randomly. In the next step the individuals are evaluated. If the condition of finish is fulfilled then we can finish the algorithm. After evaluating, the individuals are selected to breed a new generation. At first to the next generation the non-dominated solutions are selected. This population is changing by crossover and mutation operations, then the individuals are evaluated again.

Chromosome representation

An evolutionary algorithm with direct chromosome representation is used to solve this problem. In this representation a chromosome is a sequence of completion times of each activity [5].

For the problem presented in this paper, in which we have eleven activities, a chromosome looks as follows:

$$(F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9, F_{10}, F_{11}).$$

This is an example of the chromosome presented as a sequence of completion times of each activity.

(2, 5, 6, 4, 5, 11, 13, 8, 7, 18, 22).

This chromosome represents the earliest completion times for each activity. It is also acceptable because it satisfies all the constraints.

Based on this chromosome we can build a schedule which represents start and finish times for each activity and resources required by all activities (Figure 2). Resources are needed throughout the duration of an activity.

In the same way we can present a schedule for activities and generated costs. In this approach the costs are generated at the moment when an activity starts.

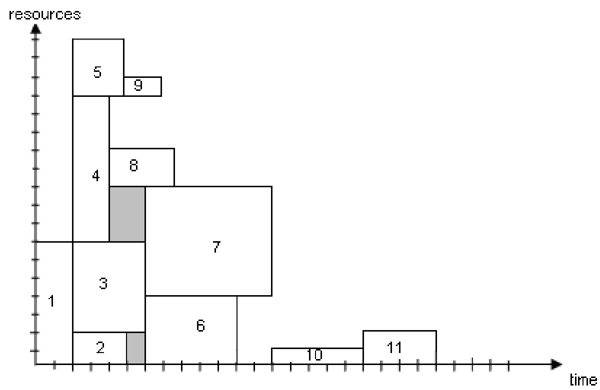


Figure 1. Example of Project schedule

Evaluation (fitness)

The fitness function has three components.

The first component of the fitness function measures the quality of the chromosome related to project completion time. The second component of the fitness function measures the quality of the chromosome related to the smoothness in resource allocation. The third component of the fitness function measures the quality of the chromosome related to the smoothness in cost generating. The costs are generated when an activity starts.

The schedule must satisfy the precedence constraints, so the finish times for activities should satisfy the following inequalities:

$F_2 \geq F_1 + 3$,

$F_3 \geq F_1 + 4$,

$F_4 \geq F_1 + 2$,

$F_5 \geq F_1 + 3$,

$F_6 \geq F_2 + 5$,

$F_7 \geq F_3 + 7$,

$F_8 \geq F_4 + 4$,

$F_9 \geq F_5 + 2$,

$F_{10} \geq \max(F_6, F_7, F_8, F_9) + 5$,

$F_{11} \geq F_{10} + 4$.

There is a penalty for a chromosome that does not meet those inequalities.

Crossover operation

In this approach the classical crossover operation is used. A single crossover point on both parents organism strings is selected. All data beyond that point in either organism string are swapped between the two offspring organisms.

Mutation operation

Mutation is a genetic operator that alters one or more gene values in a chromosome from its initial state. In this paper only one gene value is changed. The new value is generated as a random variable from all the possible finish times.

To compute this problem, the author of this paper wrote a program in the programming language C.

3. Example and results

In this section an example has been solved. The goal was to schedule tasks of activity and find non-dominated solutions. First, the example is presented, then one iteration of the evolutionary algorithm with direct chromosome representation is shown and at the end of this section, the results are discussed.

3.1. Example

Example 1.

A schedule for the project presented in Table 1 should be created. In this project we have eleven activities. For each task its predecessor, duration, costs generated and resources required are presented.

Table 1

Example 1 tasks

Activity	Predecessor	Activity duration	Amount of resources required by an activity	Costs generated by an activity
1	-	2	7	2
2	1	3	2	3
3	1	4	5	3
4	1	2	8	5
5	1	3	3	6
6	2	5	4	3
7	3	7	6	3
8	4	4	2	4
9	5	2	1	4
10	6, 7, 8, 9	5	2	2
11	10	4	2	1

The problem is presented as an AON network (Figure 2) in which activities are on nodes and relationships between them are on arrows. Each activity is determined by three parameters: duration, resources needed for this activity and costs generated by this activity.

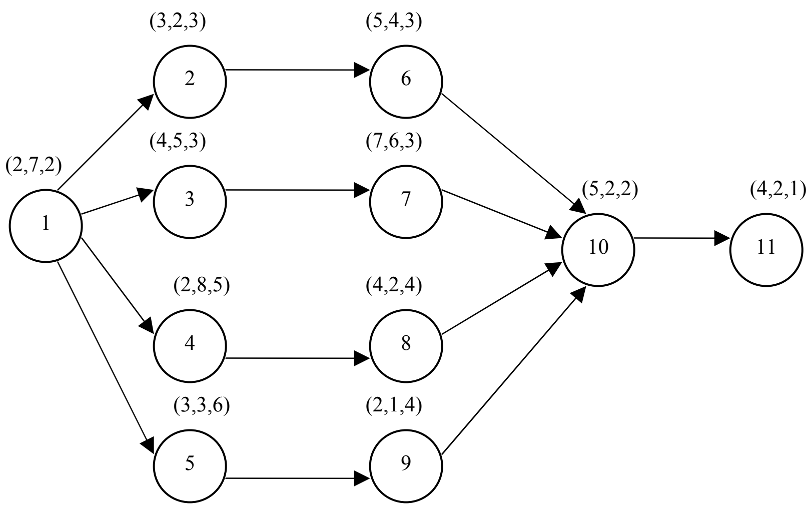


Figure 2. Example 1 network

3.2. Computations

The following EA parameters have been set:

– number of generations: 200,
– crossover rate: 0,9,
– mutation rate: 0,05,
– population size: 20.

Below we show one iteration of evolutionary algorithm with direct chromosome representation.

3. $t \to 0$
4. Set population $P(t)$

The computation starts with a randomly generated population of 20 individuals:

Individual 1 = (2, 5, 6, 4, 5, 11, 13, 8, 7, 18, 22),
Individual 2 = (2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12),
Individual 3 = (2, 5, 6, 4, 3, 7, 4, 2, 8, 12, 30),
Individual 4 = (7, 3, 2, 2, 5, 11, 13, 8, 7, 18, 22),
Individual 5 = (12, 11, 14, 16, 18, 20, 8, 4, 22, 35),
Individual 6 = (5, 6, 8, 5, 20, 16, 18, 17, 17, 22, 26),
Individual 7 = (21, 25, 16, 23, 27, 13, 30, 32, 37, 41, 21),
Individual 8 = (2, 5, 9, 11, 14, 19, 26, 30, 32, 37, 41),
Individual 9 = (2, 5, 6, 15, 13, 10,13,19, 15, 24, 28),
Individual 10 = (16, 23, 27, 13, 32, 36, 4, 7, 12, 15, 16, 14),
Individual 11 = (2, 5, 9, 11, 9, 16, 18, 20, 18, 25, 29),
Individual 12 = (5, 11, 17, 21, 9, 16, 18, 18, 22, 27, 31),
Individual 13 = (21, 12, 11, 9, 8, 7, 16, 18, 10, 21),
Individual 14 = (20,21,19, 22, 18, 17, 16, 37, 19, 11, 23),
Individual 15 = (2, 5, 6, 8, 5, 12, 15, 12, 8, 20, 24),
Individual 16 = (2, 4, 40, 12, 14, 19, 3, 9, 13, 27, 42),
Individual 17 = (2, 5, 6, 20, 16, 18, 13, 24, 18, 29, 33),
Individual 18 = (1, 3, 7, 8, 10, 12, 13, 19, 21, 23, 27),
Individual 19 = (2, 2, 5, 4, 3, 27, 9, 9, 11, 16, 37),
Individual 20 = (2, 6, 8, 4, 7, 12, 15, 16, 14, 21, 25)

3. Evaluate $P(t)$

In this population 6 individuals meet the constraints (individuals: 1, 8, 9, 15, 17 and 20), from those two are non-dominated (individual 1 and individual 8).

4. If the condition of finish is fulfilled then end.

There are 200 generations, so the condition of finish isn`t fulfilled, so we move to the next step.

5. $t \to t+1$

6. Chose $P(t)$ from $P(t-1)$

To the next generation we chose non-dominated individuals first: 1 and 8. Then we fill the population with 18 individuals from the population $P(t-1)$ chosen using proportional selection.

7. Change $P(t)$ using crossover and mutation

Crossover is made with rate 0,9:

Two individuals have been randomized to do crossover. From individuals 11 and 12 after crossover in point 7 we can have two offsprings:

Individual 11` = (2, 5, 9, 11, 9, 16, 18, 18, 22, 27, 31)

Individual 12` = (5, 11, 17, 21, 9, 16, 18, 20, 18, 25, 29)

Individual 11` became a chromosome that meets the constraints.

The mutation rate is 0,05. One mutation point is selected.

Individual 19 after mutation:

Individual 19 = (2, 2, 5, 4, 3, 27, 9, 9, 11, 16, 37),

Individual 19` = (2, **6**, 5, 4, 3, 27, 9, 9, 11, 16, 37).

3.3. Results

After 200 generations nine non-dominated and satisfying the constraints solutions have been found. The solutions are presented in Table 1. For each solution the triple: (time, resource, cost) is presented.

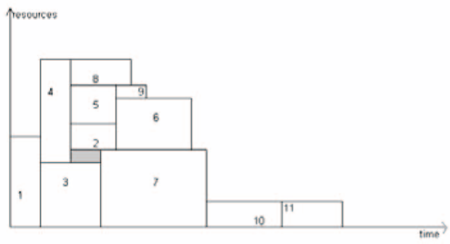
Table 2

Solutions

Non-dominated solution no.	Individual	Level of		
		time	resources	costs
1	(2, 7, 6, 4, 7, 12, 13, 8, 9, 18, 22)	22	13	13
2	(2, 5, 6, 8, 5, 13, 15, 12, 7, 20, 24)	24	12	12
3	(2, 6, 6, 4, 7, 12, 13, 10, 15, 21, 25)	25	10	9
4	(2, 5, 6, 4, 7, 10, 13, 11, 12, 18, 22)	22	15	12
5	(2, 5, 9, 11, 15, 16, 23, 22, 18, 28, 33)	33	8	6
6	(2, 5, 9, 7, 10, 12, 16, 14, 12, 21, 25)	25	13	8
7	(2, 5, 6, 8, 11, 20, 15, 15, 13, 25, 29)	29	9	9
8	(2, 6, 6, 8, 8, 13, 15, 17, 11, 22, 26)	26	11	6
9	(2, 5, 12, 8, 6, 14, 19, 19, 18, 24, 28)	28	10	6

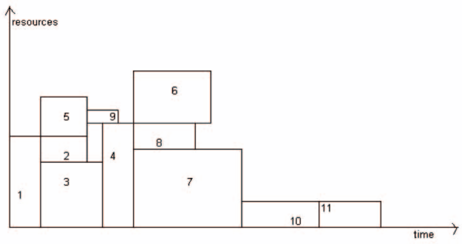
The schedules for solutions are presented below.

Solution 1.



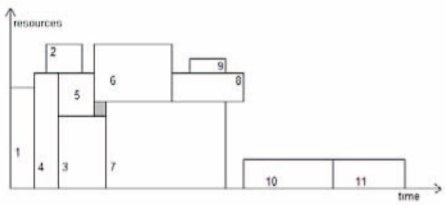
(time, resource, cost) = (22, 13, 13)
Individual = (2, 7, 6, 4, 7, 12, 13, 8, 9, 18, 22)

Solution 2.



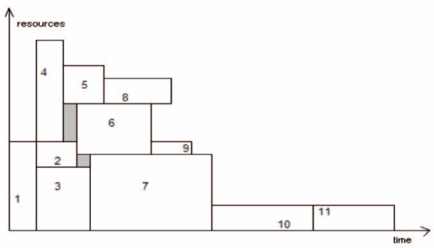
(time, resource, cost) = (24, 12, 12)
Individual = (2, 5, 6, 8, 5, 13, 15, 12, 7, 20, 24)

Solution 3.



(time, resource, cost) = (25, 10, 9)
Individual = (2, 6, 6, 4, 7, 12, 13, 10, 15, 21, 25)

Solution 4.

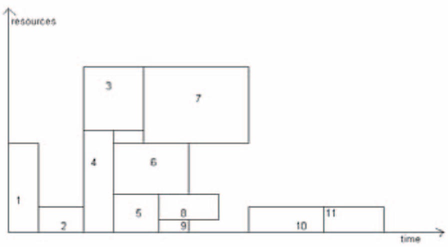


(time, resource, cost) = (22, 15, 12)
Individual = (2, 5, 6, 4, 7, 10, 13, 11, 12, 18, 22)

Solution 5.

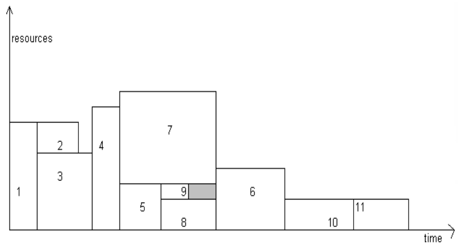
(time, resource, cost) = (33, 8, 6)
Individual = (2, 5, 9, 11, 15, 16, 23, 22, 18, 28, 33)

Solution 6.



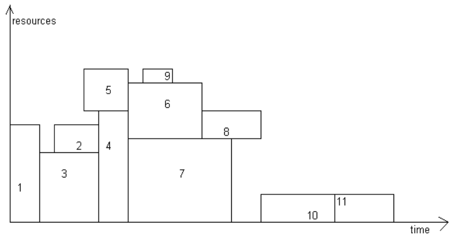
(time, resource, cost) = (25, 13, 8)
Individual = (2, 5, 9, 7, 10, 12, 16, 14, 12, 21, 25)

Solution 7.



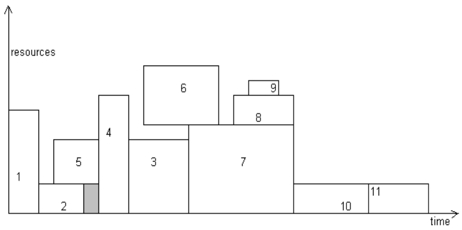
(time, resource, cost) = (29, 9, 9)
Individual = (2, 5, 6, 8, 11, 20, 15, 15, 13, 25, 29)

Solution 8.



(time, resource, cost) = (26, 11, 6)
Individual = (2, 6, 6, 8, 8, 13, 15, 17, 11, 22, 26)

Solution 9.



(time, resource, cost) = (28, 10, 6)
Individual = (2, 5, 12, 8, 6, 14, 19, 19, 18, 24, 28)

The fourth solution is represented by the chromosome: (2, 5, 6, 4, 7, 10, 13, 11, 12, 18, 22). This solution is the best solution because of the time criterion. The shortest time to finish the whole project is 22. It is impossible to finish this project faster without making other criteria worse. The resources in this solution are 15. This is the highest amount of resources needed in one period. The highest amount of costs generated in this project in one period is 12. The same finish time is in the first solution. In this solution the level of costs generated is lower, but the resource level is higher. In the second solution the cost and resource levels are lower, but reduction of these parameters leads to longer project completion time.

The fifth solution is represented by the chromosome: (2, 5, 9, 11, 15, 16, 23, 22, 18, 28, 33). This solution is the best solution because of resource and cost criteria. The resources in this solution are 8 and costs are 6, but the project is finished after 33. The lowest cost level occurs also in the eighth solution, the finish time is better – it is 26 – but the resource level is higher it is 11. A similar situation occurs in the ninth solution, where the finish time is 28 and resource level is 10. The seventh solution also has low cost and resource levels – 9 each, and late finish time 29.

The third solution is represented by the chromosome: (2, 6, 6, 4, 7, 12, 13, 10, 15, 21, 25). This is the most interesting solution because it satisfies all criteria in some way. The project completion time is 25, resource level is 10 and cost level is 9. The same completion time occurs in sixth solution. In this case the cost level is lower but it leads to higher resource level. As we can observe in other solutions, an improvement with respect to one criterion leads to worse levels of other parameters. This situation can be observed in all our solutions.

Conclusion and future works

An evolutionary algorithm with direct chromosome representation for solving a multiple-objective project scheduling problem has been described. It is based on the classical evolutionary algorithm with change in selection where a non-dominated solution automatically goes to the next generation.

This method can be adapted to include new objectives related to needs. It can be also adapted to new chromosome representations and a new algorithm scheme. This approach needs a better solution for indication non-dominated solutions.

It can be useful to try other chromosome presentation to solve a multiple--objective project scheduling problem, eg. permutation without repetition (or with repetition) chromosome representation, priority rule representation, disjunctive graph based representation or random key representation.

We should consider also how to decide which non-dominated individual should be chosen and implemented. It would be useful to use other multicriteria methods, e.g. one of the Electre group method. The first Electre method was proposed in 1966. Since then many adapting techniques have been proposed: to choose the best option (Electre I and Electre IS), to sort solutions (Electre TRI) and to order decision options (Electre II, Electre III, Electre IV) [8].

For further work the elitist evolutionary algorithms can be very useful. This approach uses an archive containing non-dominated solutions previously found (it uses external non-dominated set). At each generation, non-dominated individuals are copied to the external non-dominated set. For each individual in this set, a strength value is computed. The fitness is computed according to the strengths of external non-dominated solutions that dominate it [2].

References

[1] British Standards. *Guide to Project Management*. Nr BS6079-1:2002. BSI, 2002.

[2] Coello Coello C.A., Romero C.E.M.: *Evolutionary Algorithms and Multiple Objective Optimization*. In: M. Ehrgott, X. Gandibleux: *Multiple Criteria optimization of the Art Annotated Bibliographic Surveys*. Kluwer Academic Publishers, 2002, pp. 277-331.

[3] Hapke M., Jaszkiewiecz A., Słowiński R.: *Interactive Analysis of Multiple-Criteria Project Scheduling Problems*. "European Journal of Operational Research" 1998, 107, pp. 315-324.

[4] Kostrubiec A.: *Project Scheduling – Models Review*. In: *Project Management Engineering*. Ed. L. Zawadzka. Gdańsk 2003, pp. 33-52.

[5] Kostrubiec A.: *Project Scheduling Problem Presented in Genetic Algorithm*. In: *Quantitative and Qualitative Aspects of Management*. Ed. L. Zawadzka. Gdańsk 2001, pp. 52-61.

[6] Leu S.S, Yang C.H.: *Ga-based Multicriteria Optimal Model for Construction Scheduling*. "Journal of Construction Engineering and Management" 1999, 125(6), pp. 420-427.

[7] Lova A., Maroto C., Tormos P.: *A Multicriteria Heuristic Method to Improve Resource Allocation in Multiproject Scheduling*. "European Journal of Operational Research" 2000, 127: pp. 408-424.

[8] Nowak M.: *ELECTRE Method In Deterministic and Stochastic Decision Problems.* "Decisions" 2004, 2.

[9] Pinedo M.: *Scheduling – Theory. Algorithms and Systems*. Prentice Hall Englewood Cliffs 1995.

[10] Project Management Institute. *A Guide to the Project Management Body of Knowledge*. Third Edition 2004.

[11] Viana A., de Sousa J.P.: *Using Metaheuristic in Multiobjective Resource Constrained Project Scheduling*. "European Journal of Operational Research" 2000, 120, pp. 359-374.