EDN: PQZQBH
УДК 519.8

Two Heuristic Algorithms for RCPSP with NPV Criterion

Aleksandr M. Bulavchuk*
Daria V. Semenova†
Siberian Federal University
Krasnoyarsk, Russian Federation

Received 09.06.2023, received in revised form 20.07.2023, accepted 16.08.2023

Abstract. The resource constrained project scheduling problem (RCPSP) with the criterion of maximizing the net present value (NPV) is considered. We propose two heuristic algorithms for RCPSP based on idempotent algebra methods. To assess the quality of the algorithms, a zero-one integer linear programming model was built for the problem under consideration. This model makes it possible to find exact solutions to the problem using the IBM ILOG CPLEX. Experiments show that the proposed heuristic algorithms demonstrate high performance. In a series of experiments, schedules corresponding to exact solutions were obtained, among other things.

Keywords: scheduling problem, investment project, NPV, idempotent mathematics, genetic algorithm, simulated annealing.

Citation: A.M. Bulavchuk, D.V. Semenova, Two Heuristic Algorithms for RCPSP with NPV Criterion, J. Sib. Fed. Univ. Math. Phys., 2023, 16(5), 639–650. EDN: PQZQBH.

Introduction

The scheduling problem is a classical optimization problem with a wide practical application. An important version of the scheduling problem is statements in which project resources are described by cash flows. An overview of such formulations is given in the works [1,2]. One of the differences in the approaches is the choice of discounting method. However, the continuous and compound interest formulas used in different works are easily converted into each other. Another important difference between the approaches is the features of accounting for the components of cash flows. For example, these cash flows can be considered as regular or irregular. In addition, there are differences in the choice of the objective function. The profit traditionally used in this capacity can be adjusted, for example, by the amount of fines for violating the deadlines for completing work.

The paper considers the resource constrained project scheduling problem (RCPSP) in the formulation proposed in [3]. The optimality criterion is the profit reduced to the beginning project implementation. It is assumed that a partial order is given on the set of activities. The only resource of the project is financial. The stocks of funds, the needs for financing activities and the income generated by them are known at each of the integer points in time. Money is a stored resource. In addition, there is the possibility of obtaining additional income from holding free funds at a given risk-free rate. Restrictions on the stock of funds can be removed

*ABulavchuk@sfu-kras.ru https://orcid.org/0000-0001-6990-6925
†DVSemenova@sfu-kras.ru https://orcid.org/0000-0002-8670-2921

by introducing the possibility of lending. This also means additional costs associated with the payment of interest.

The proven NP-hardness of the problem under consideration has led to the popularity of heuristic methods for solving it [3]. This paper is a continuation of our research on the development of effective heuristic algorithms for solving the resource constrained project scheduling problem with the criterion of maximizing the net present value [4]. The proposed algorithms are based on the methods of idempotent algebra. It is shown that, in terms of idempotent algebra, the project schedule can be represented as a solution to a linear equation over an idempotent semiring. A sufficient condition for the admissibility of the schedule from the point of view of the partial order of work and the duration of the project is formulated. This fact is a theoretical justification for the use of vectors from an idempotent semimodule in the development of heuristic algorithms GASPIA and SASPIA.

The first of the proposed algorithms is a modification of the genetic algorithm GASPIA[‡] [5]. The use of vectors from an idempotent semimodule in the GASPIA genetic algorithm as individuals of a population makes it possible to ensure the fulfillment of the conditions of a partial order of work when applying genetic operators. The modified crossover mechanism is based on the division of project work into profitable and unprofitable ones.

The second heuristic algorithm for solving this problem is based on the annealing simulation method. As a rule, the simulated annealing method is used to solve the scheduling problem with the criterion of minimizing the total project time. For example, [6] considers the application of the simulated annealing method to a problem with several types of resources. As variables, the authors consider the priority vectors of project activities. In [7] a hybrid algorithm is proposed that combines the simulated annealing method and the MINSLK method, which is used to find the starting solution. The authors conclude that this approach leads to a significant reduction in the number of steps compared to the standard procedure for generating a random starting solution. Other well-known heuristics can be used to find the starting solution. For example, in the work [8] for this purpose, the method ranking works by their duration (SPT) is used. A feature of our approach to solving RCPSP is the use of vectors from an idempotent semimodule in a new algorithm for simulating annealing SASPIA[§]. This allows you to search among schedules that satisfy the conditions of a partial work order.

The rest of the article is organized as follows. In Section 1, we give the formal definition of the RCPSP. In 2 we give a zero-one statement of the problem. This allows you to find exact solutions for small projects using the IBM LOG CPLEX solver. The Sections 4 and 5 describe the algorithms we developed: the genetic algorithm and the annealing simulation algorithm for the scheduling problem. The Section 6 presents the results of computational experiments and characterizes the quality of the algorithms. Our main contribution is the application of idempotent algebra methods in the implementation of heuristic algorithms for solving the problem of scheduling investment projects.

1. Problem formulation

Consider an investment project consisting of a set of activities $V = \{1, 2, \ldots, N\}$. Partial order relations E are given on the set V. The timing of the activities is known and set as an

[‡]GASPIA — **G**enetic **A**lgorithm for Resource-constrained Project **S**cheduling **P**roblem Using **I**dempotent **A**lgebra Methods

[§]SASPIA — **S**imulated **A**nnealing for Resource-constrained Project **S**cheduling **P**roblem Using **I**dempotent **A**lgebra Methods integer number of time periods (months, years, etc.) — $Q = (q_1, q_2, \ldots, q_N)$. We assume that the budget, costs and income of the project can be measured in monetary units. For each moment of time $t \in \{0, 1, \ldots, T-1\}$, the amount of money K_t at the disposal of the organization is known. These funds, as well as income from previous periods, are the source of financing for the project. In addition, it is possible to reinvest free funds at the rate of r_0. Each of the activities j is characterized by the balance of income and expenses $c_j(\tau)$ at the beginning of the period $\tau \in \{0, 1, \ldots, q_j\}$. To evaluate the results of the project implementation, we will use the net present value indicator (NPV). This indicator represents the value of the profit from all planned activities, discounted to the time the investment was started. Thus, the problem of scheduling investment project that we are considering is to find the optimal start time for each of the activities, at which the restrictions on the sufficiency of funds, the relationship of activities are met and the maximum NPV. Taking into account the introduced notation, the RCPSP with the NPV maximization criterion will take the following form [3, 5].

$$NPV(S) = \sum_{j \in V} \sum_{\tau=0}^{q_j} \frac{c_j(\tau)}{(1+r_0)^{\tau+s_j}} \to \max_S, \tag{1}$$

$$s_i + q_i \leqslant s_j, (i, j) \in E, \tag{2}$$

$$\sum_{t=0}^{t^*} \frac{K_t}{(1+r_0)^t} + \sum_{t=0}^{t^*} \sum_{j \in N_t} \frac{c_j(t-s_j)}{(1+r_0)^t} \geqslant 0, \qquad t^* = 0, 1, \ldots, T-1, \tag{3}$$

where $S = (s_1, s_2, \ldots, s_N)$ is the project schedule consisting of the start times of activities, and N_t is the set of activities performed in the interval $[t; t+1)$. Constraints (2) correspond to conditions of partial order, and constraints (3) formalize the requirements for positive balance of funds, taking into account the possibility of reinvestment at time t^*. An RCPSP problem of the form (1)–(3) is NP-hard in the strong sense [3].

2. Zero-one model

To find the exact solution of the problem under consideration, we proposed a zero-one formulation. The Tab. 1 contains a description of the variables for the zero-one model.

Taking into account the introduced notation, the binary model of the RCPSP problem can be represented in the following form.

$$NPV(X) = \sum_{t=0}^{T-1} \sum_{j=1}^{N} z_{tj} \cdot x_{tj} \to \max_X, \tag{4}$$

$$\sum_{t=0}^{T-1} x_{tj} \cdot t \leqslant T - q_j, \qquad j = 1, \ldots, N, \tag{5}$$

$$\sum_{t=0}^{T-1} (x_{ti} - x_{tj}) \cdot t \leqslant -q_i, \quad (i, j) \in E, \tag{6}$$

$$\sum_{t=0}^{T-1} \sum_{j=1}^{N} \mathbf{G}^*{}_{tj}(t^*) x_{tj} \geqslant -k^*_{t^*}, \qquad t^* = 0, 1, \ldots, T. \tag{7}$$

This model makes it possible to find exact solutions to the problem of non-high dimension using the IBM ILOG CPLEX.

Table 1. Notation

Symbol	**Meaning**
X	a zero-one matrix, where $x_{tj} = 1$ if $s_j = t$ for $j \in \{1,\dots,N\}$, $t \in \{0,1,2,\dots,T-1\}$
$k_t^* = \sum_{\tau=0}^{t} \frac{K(\tau)}{(1+r_0)^\tau}$	cumulative discounted budjet at time $t \in \{0,1,2,\dots,T-1\}$
$z_{tj} = \frac{NPV_j}{(1+r_0)^t}$	NPV of activity j started at time $t \in \{0,1,2,\dots,T-1\}$
G	matrix of discounted cash flows, where $g_{tj} = \frac{c_j(t-s_j)}{(1+r_0)^t}$, if $t \leqslant s_j + q_j$ and $g_{tj} = 0$ otherwise
$\mathbf{G}^*{}_{tj}(t^*) = \sum_{\tau=0}^{t^*} \mathbf{G}_{tj}(\tau)$	cumulative discounted cash flows for periods $t^* = 0,1,\dots,T$, where $\mathbf{G}_{tj} = G_j(s_1,\dots,s_{j-1},t,s_{j+1},\dots,s_N)$

3. A sufficient condition for admissibility of a schedule

In terms of idempotent algebra, the space of admissible project schedules by RCPSP deadlines in the form (1)–(3) is determined by solving a linear equation over an idempotent semiring [5]. The definitions and notation of idempotent algebra used in this paper

Consider an idempotent semiring $\mathbb{R}_{\max,+}$ with operations $\oplus = \max$ and $\odot = +$. This semiring contains the zero element $\mathbb{0} = -\infty$ and the identity element $\mathbb{1} = 0$.

The partial order relations of project activities can be represented as a weighted directed graph with an adjacency matrix $A = (a_{ij})_{i,j=1}^N$, where

$$a_{ij} = \begin{cases} q_j & \text{if } (i,j) \in E, \\ 0 & \text{if } (i,j) \notin E \text{ and } i = j, \\ -\infty & \text{if } (i,j) \notin E \text{ and } i \neq j. \end{cases}$$

In the semiring $\mathbb{R}_{\max,+}$, the elements of the matrix A are described by the relations

$$a_{ij} = \begin{cases} q_j & \text{if } (i,j) \in E, \\ \mathbb{1} & \text{if } (i,j) \notin E \text{ and } i = j, \\ \mathbb{0} & \text{if } (i,j) \notin E \text{ and } i \neq j. \end{cases}$$

Let us formulate a sufficient condition for the admissibility of the project schedule in terms of the idempotent algebra [5].

Theorem 3.1. *Let A be the partial work order matrix of the investment project in normal form, $Q = (q_1, q_2, \dots, q_N)$ be the vector of work durations, and T be the target completion date of the project. Then the schedule $S = A^+\upsilon$ will be admissible from the point of view of the partial order and duration of the project, if for all $j = 1,\dots,N$*

$$\upsilon_j^{\min} \leqslant \upsilon_j \leqslant \upsilon_j^{\max}, \tag{8}$$

where $\upsilon^{\min} = (0, \dots, 0)$, $\upsilon^{\max} = (b^{-}A^{+})^{-}$ *and* b *— a string of length* N *and* b_j *is equal to* $T - q_j$.

The proof of the theorem is based on the fact that the partial order matrix of the investment project is decomposable, and its normal form has a lower triangular form. For such a matrix, the linear equation $A \cdot S = S$ has a nontrivial solution represented as $S = A^{+}\upsilon$, where υ is an arbitrary vector from $\mathbb{R}^N$ [5].

The sufficient condition for the admissibility of the project schedule in terms of timing (Theorem 3.1.) is fundamental in the two heuristic algorithms proposed below.

4. Description of the GASPIA algorithm

The genetic algorithm is a heuristic optimization method based on modeling the evolution of a population. Each individual in the population will be characterized by genotype and phenotype. A genotype is a set of independent characteristics of an individual that interests us in the context of the problem. A phenotype is a characteristic of an individual, defined as a function of its genotype. Let the individuals of the population represent the admissible schedules of activities for the project. Then the vectors $S = (s_1, s_2, \dots, s_N)$ could be considered as a genotype, the components of which determine the start time of the corresponding work. However, such a variant of the genotype makes crosses difficult, since the components of the parental genotypes cannot be arbitrarily mixed without violating the partial order conditions. In this regard, as the genotype of individuals, we will consider vectors υ, whose coordinates satisfy the relations (8).

Such vectors allow you to obtain schedules that meet the requirements of the order of work and the deadline for completing the project. In the process of forming the initial population, it is necessary to generate the required number of vectors that satisfy the relations (8). The schedule S corresponding to the vector υ will be determined using the formula $S = A^{+}\upsilon$.

This can be achieved, for example, by including a constraint violation penalty in the fitness function. Then the phenotype will be the NPV of the project adjusted for the penalty:

$$F\big(S(\upsilon)\big) = \sum_{j=1}^{N} \sum_{\tau=0}^{q_j} \frac{c_j(\tau)}{(1+r_0)^{s_j+\tau}} + \alpha_L \cdot \min\left\{0, L\big(S(\upsilon)\big)\right\}, \tag{9}$$

where $\alpha_L > 0$ is a configurable parameter, and $L\big(S(\upsilon)\big)$ is the cumulative deficit of funds needed to complete the work for this schedule. The use of a penalty makes it possible to separate schedules that are inadmissible according to the budget from those that are admissible during the algorithm execution.

If the budget constraints are not too strict, then this approach will make it possible to exclude budget-inadmissible schedules at the stage of selecting individuals with the highest value of the fitness function. With a small number of schedules allowed by the budget, the application of the genetic algorithm will be difficult due to the lack of genetic diversity.

Crossover. Crossover is one of the basic operations of the genetic algorithm. The mechanism of crossing should ensure the growth of the fitness function in the population. The version of the algorithm considered in this paper differs from that described in [5] by the crossing procedure.

The project activities can be divided into profitable and unprofitable by the value of NPV_j. Let's calculate the auxiliary vector η with coordinates:

$$\eta_j = \begin{cases} 1 - \alpha \cdot \left(1 - \dfrac{NPV_j}{\max\{NPV_1, NPV_2, \dots, NPV_N\}}\right) & \text{if } NPV_j \geqslant 0, \\ \alpha \cdot \left(1 - \dfrac{NPV_j}{\min\{NPV_1, NPV_2, \dots, NPV_N\}}\right) & \text{if } NPV_j < 0, \end{cases} \tag{10}$$

where α is a custom parameter.

For a pair of parents υ^k, υ^l and a random number $p \in [0,1]$, the genotype of the offspring will be given by the vector $\upsilon^{k\times l}$ with coordinates

$$\upsilon_j^{k\times l} = \begin{cases} \min\left\{\upsilon_j^k, \upsilon_j^l\right\} & \text{if } \eta_j \geqslant p, \\ \max\left\{\upsilon_j^k, \upsilon_j^l\right\} & \text{if } \eta_j < p. \end{cases} \tag{11}$$

This modification of crossover makes it possible to take into account not only the profitability of activities, but the differences within the groups of profitable and unprofitable activities.

Mutation. The mutation operator is intended to introduce diversity into the population in order to avoid premature convergence of the algorithm. In the algorithm settings, we set the proportion of individuals in the population whose genotype will undergo mutation. During the application of the mutation operator, we will replace one randomly selected element of the genotype so that the relations (8) are satisfied. This procedure also does not violate the partial order conditions. Schedules that do not meet budget constraints will be screened out at the stage of selecting the best individuals. The mutation algorithm can be described as follows.

1. Choose a vector υ in the population.
2. Generate a random number $rnd\_\upsilon \in [0,1]$.
3. If $rnd\_\upsilon < p\_mutation$, where $p\_mutation$ is the given mutation probability, choose a random element of the vector υ_j.
4. Replace the element υ_j with the random number $\upsilon_j^{\min} \leqslant rnd\_\upsilon_j \leqslant \upsilon_j^{\max}$.
5. Repeat for all individuals in the population.

Selection. After the application of genetic operators, the number of individuals in the population is guaranteed to increase. In addition, the characteristics of some individuals that have undergone fixed mutations change. To determine the most viable individuals, we sort the population in descending order of the fitness function (9). A given number of the best individuals enter the next generation population. Since the fitness function includes a penalty for violating the budget constraint, the next generation will predominantly include individuals for which the constraint is not violated. We will consider the given number of generations as the criterion for stopping the algorithm.

5. Description of the SASPIA

Simulated annealing is an ordered random search method. The method was created on the basis of a substance crystallization model and has found application in solving various optimization problems. Application of the method to a specific problem involves the description of the following components [9]:

– variables;

– objective function;

– cooling schedule;

– acceptance probability;

– generation rule for new configurations.

Let us characterize these components in the context of the model (1)–(3).

As variables in the developed annealing simulation algorithm, vectors υ from the idempotent semimodule $\mathbb{X}^N$ are used. As an objective function, consider the NPV of the project, adjusted by the amount of the penalty for violating budget constraints:

$$F\big(S(\upsilon)\big) = \sum_{j=1}^{N} \sum_{\tau=0}^{q_j} \frac{c_j(\tau)}{(1+r_0)^{s_j+\tau}} - M \cdot \delta(S), \tag{12}$$

where $M > 0$ is a sufficiently large number, and $\delta(S)$ equals 1 if at least one of the constraints (3) is not satisfied. Otherwise $\delta(S) = 0$. This function assumes a more significant penalty than function (9).

In practice, various cooling schedules are used, which differ in the rate of temperature decrease [9]. Consider the following scheme:

$$\mathcal{T}(k) = \frac{\mathcal{T}_0}{k}, \tag{13}$$

where $\mathcal{T}_0$ is the initial temperature value and k is the step number.

As a function of the acceptance probability we will use the following:

$$h(\Delta F, \mathcal{T}) = \exp(\Delta F/\mathcal{T}).$$

For positive values of ΔF, the value of $h(\Delta F, \mathcal{T})$ will be greater than 1. This guarantees a transition to a solution with a large value of the function $F(S(\upsilon))$. In order to get a new solution υ', let's change the randomly chosen coordinate of the vector υ. This transformation makes it possible to obtain schedules $S(\upsilon)$ satisfying the constraints (2).

6. Computational experiments

To evaluate the quality of the proposed heuristic algorithms, a program was written in Python. All experiments were carried out on a personal computer with a 2.3 GHz CPU and 8 Gb RAM with Windows 10. When constructing the network diagram of the project, examples from the PSLIB [10] test database were used. The maximum size of projects is 120 activities. A discount rate of 10 % per annum was used in the calculations. Payment flows for projects were generated in such a way that among the activities there were both profitable and unprofitable ones.

In the GASPIA experiments, a population of $n_p = 100$ individuals was modeled. The termination condition of the algorithm was the number of generations $n_g = 500$. Experiments with the annealing simulation algorithm were carried out for cooling schedule (13). The termination condition was the $n_k = 10000$ steps. The temperature was updated even if there was no transition to a new solution at the current step. The exact solution for a project of 120 activities was found using the IBM ILOG CPLEX solver.

The Tab. 2 shows the results of 100 experiments on finding a solution using the proposed algorithms for the examples $j120$ [10]. Exact solutions are marked with asterisks.

Table 2. Results of experiments for the GASPIA and tha SASPIA algorithms

Algorithm	**Set**	**Maximum** NPV	**Average** NPV	**Deviation from maximum, %**
GASPIA	$j120\_1$	13.791*	11.347	17.720
	$j120\_2$	40.711*	39.369	3.298
	$j120\_3$	18.646	16.050	14.027
	$j120\_4$	9.249	8.509	12.029
	$j120\_5$	36.770	34.342	8.155
	$j120\_6$	22.329*	20.859	6.585
	$j120\_7$	16.722	15.245	8.899
	$j120\_8$	32.952	29.750	10.445
	$j120\_9$	26.839	26.220	8.559
	$j120\_10$	40.191	39.428	7.819
SASPIA	$j120\_1$	13.791*	10.852	21.310
	$j120\_2$	40.711*	39.559	2.832
	$j120\_3$	18.669*	18.097	3.063
	$j120\_4$	9.625	8.687	10.196
	$j120\_5$	37.391*	35.798	4.261
	$j120\_6$	22.329*	21.624	3.158
	$j120\_7$	16.725	15.293	8.610
	$j120\_8$	33.220*	31.284	5.830
	$j120\_9$	28.674*	27.488	4.137
	$j120\_10$	41.409	39.626	7.356

Fig. 1 shows a graph of the maximum value of NPV depending on the generation number for project $j120\_2$. The algorithm demonstrates a stable growth of the value of the objective function. Fig. 2 shows NPV graph for SASPIA steps. This figure also shows an increase in the NPV value when moving to the next steps of the algorithm.

Fig. 3 shows box plots for set $j120\_2$. Both algorithms provide an exact solution (dashed line), but the scatter of NPV values for algorithms is significantly different. The experimental results for the SASPIA have a higher median value than for the GASPIA algorithm. The same can be said about the minimum NPV values. We also note a significantly larger number of outliers for results of applying the GASPIA algorithm.

Thus, based on the results of the experiments, we can conclude that the proposed algorithms allow finding optimal or close to optimal solutions to the problem. The considered versions of the algorithms demonstrate similar quality, however, the smallest deviation from the maximum is provided by the annealing simulation algorithm with the cooling schedule $\mathcal{T}(k) = \dfrac{100}{k}$.

7. Discussion

The proposed algorithms were also tested on projects of other dimensions. For small projects, high accuracy solutions can be obtained in a small number of iterations. Comparison of the results for projects of different sizes allows us to conclude that it is necessary to determine the

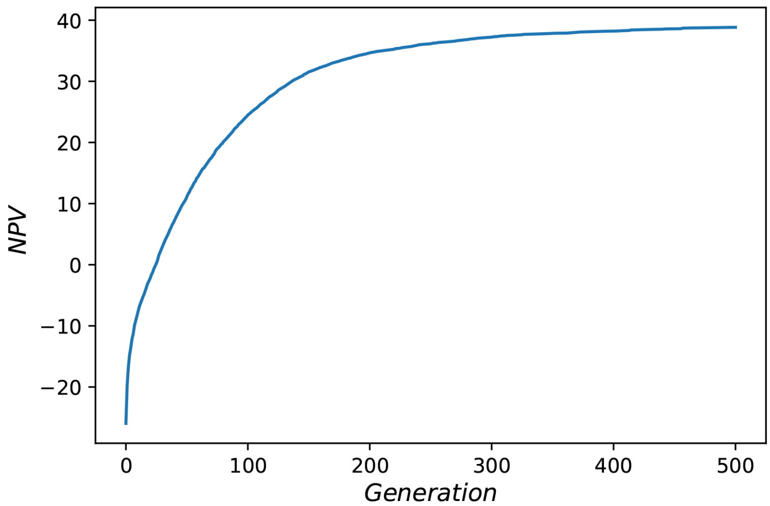

Fig. 1. Dynamics of NPV depending on the generation for GASPIA (set $j120\_2$)

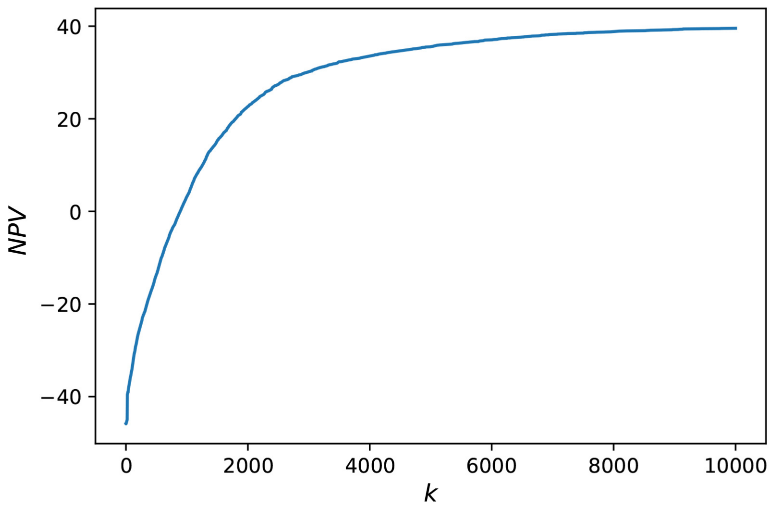

Fig. 2. The value of NPV at each step of the algorithm for SASPIA (set $j120\_2$)

termination condition. The number of steps or generations, as well as the size of the population, should be larger for the larger the project size. In addition, it is required to select other parameters of the algorithms. For example, the probability of mutation or crossover, cooling schedule, etc.

Another important feature of the application of the considered heuristic algorithms is the dependence of their efficiency on the hardness of budget constraints. Fig. 4 shows how the share of schedules that are inadmissible according to the budget behaves in the process of population evolution in the GASPIA algorithm. However, if a significant proportion of schedules in the population turns out to be unacceptable in terms of the budget, then applying genetic operators to them does not give the desired effect.

The paper considers a deterministic formulation of the problem, however, in practice, many

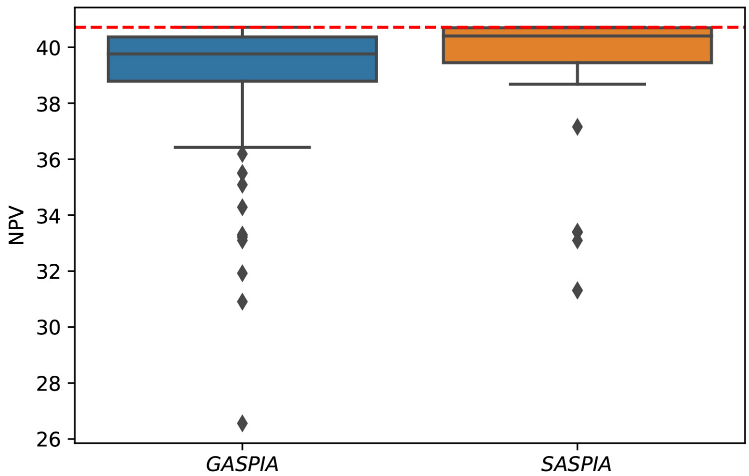

Fig. 3. Box plots for experimental results using the GASPIA and SASPIA algorithm for set $j120\_2$

project parameters may be uncertain. This is especially true for components of cash flows. There are stochastic and fuzzy RCPSP models [11, 12]. In the paper [13] it was shown that the GASPIA algorithm can also be successfully used for problems in fuzzy formulation. The algorithm can be adapted for various fuzzy number ranking functions [14]. Note that only cases of independent activities were considered. In the future, it is planned to study the formulations in which the components of the cash flows for different activities are interconnected. In stochastic formulations, one can single out such a direction of analysis as determining the level of risk of NPV deviation from the planned values. The proposed algorithms can be easily modified to solve the risk minimization problem.

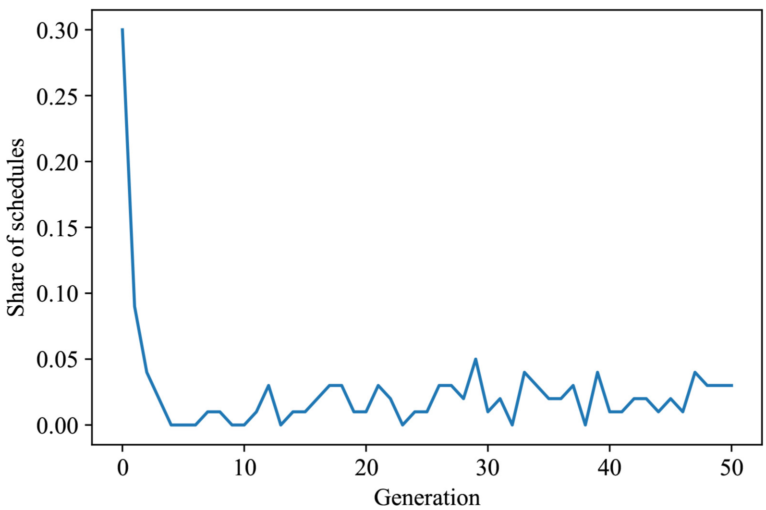

Fig. 4. The share of schedules that are inadmissible according to the budget

This work is supported by the Krasnoyarsk Mathematical Center and financed by the Ministry of Science and Higher Education of the Russian Federation (Agreement no. 075-02-2023-936).

References

[1] S.Hartmann, D.Briskorn, An updated survey of variants and extensions of the resource-constrained project scheduling problem, *European Journal of Operational Research*, **1**(2022), 1–14. DOI: 10.1016/j.ejor.2021.05.004

[2] F.Habibi, F.Barzinpour, S.Sadjadi, Resource-constrained project scheduling problem: review of past and recent developments, *Journal of Project Management*, **3**(2018), 55–88. DOI: 10.5267/j.jpm.2018.1.005

[3] A.Kazakovtseva, V.Servakh, The complexity of the project scheduling problem with credits, *J. Appl. Industr. Math.*, **9**(2015), no. 4. 489–496. DOI: 10.1134/S1990478915040055

[4] A.Bulavchuk, D.Semenova, Genetic algorithm based on idempotent algebra methods for RCPSP, 2021 IEEE 15th International Conference on Application of Information and Communication Technologies (AICT), 2021, 1–4.

[5] A.Bulavchuk, D.Semenova, Application of idempotent algebra methods in genetic algorithm for solving the scheduling problem, *Prikl. Diskr. Mat.*, **58**(2022), 112–124 (in Russian). DOI: 10.17223/20710410/58/11

[6] J.-H.Cho, Y.-D.Kim, A Simulated annealing algorithm for resource constrained project scheduling problems, *The Journal of the Operational Research Society*, **48**(1997), 736–744. DOI: 10.1057/palgrave.jors.2600416

[7] N.-H.Pan, T.-C.Hsiang, W.-T.Chen Using hybrid simulated annealing algorithm in resource-constrained project scheduling problem, *The International Journal of Organizational Innovation*, **12**(2020), 25–46.

[8] K.Bouleimen, H.Lecocq, A new effcient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version, *European Journal of Operational Research*, **149**(2003), 268–281. DOI: 10.1016/S0377-2217(02)00761-0

[9] A.Lopatin, Simulated annealing, *Stochastic optimization in computer science*, **1**(2005), 133–149 (in Russian).

[10] Project scheduling problem library. Available online: https://www.om-db.wi.tum.de/psplib/ (accessed on 25 January 2023).

[11] H.Shavandi, A.Najafi, A.Moslehirad, Fuzzy project scheduling with discounted cash flows, *Economic Computation and Economic Cybernetics Studies and Research*, **46**(2012), 219–232.

[12] B.Ashtiani, R.Leus, M.Aryanezhad, New competitive results for the stochastic resource-constrained project scheduling problem: Exploring the benefits of pre-processing, *Journal of Scheduling*, **14**(2011), 157–171. DOI 10.1007/s10951-009-0143-7

[13] A.Bulavchuk, D.Semenova, Features of risk analysis in solving RCPSP using the GASPIA algorithm, 2022 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON), 2022, 970–973
DOI: 10.1109/SIBIRCON56155.2022.10017031

[14] A.El-Kholy, M.El-Shikh, S.Abd-Elhay, Which fuzzy ranking method is best for maximizing fuzzy net present value?, *Arabian Journal for Science and Engineering*, **42**(2017), 4079–4098 DOI: 10.1007/s13369-017-2573-4

[15] N.Krivulin, Methods of idempotent algebra in problems of complex systems modeling and analysis, Publisher: St. Petersburg University Press, St. Petersburg, Russia, 2009 (in Russian).

О двух эвристических алгоритмах для задачи календарного планирования инвестиционных проектов с ограниченными ресурсами и NPV-критерием

Александр М. Булавчук
Дарья В. Семенова
Сибирский федеральный университет
Красноярск, Российская Федерация

Аннотация. Рассматривается задача календарного планирования инвестиционных проектов с ограниченными ресурсами (RCPSP) и критерием максимизации чистой приведенной стоимости (NPV). Мы предлагаем два эвристических алгоритма для RCPSP, основанных на методах идемпотентной алгебры. Для оценки качества работы алгоритмов была построена бинарная модель целочисленного линейного программирования для рассматриваемой задачи. Эта модель позволяет находить точные решения задачи с помощью программного комплекса IBM ILOG CPLEX. Эксперименты показывают, что разработанные нами эвристические алгоритмы демонстрируют высокую результативность. В серии экспериментов были получены в том числе расписания, соответствующие точным решениям.

Ключевые слова: задача календарного планирования, инвестиционный проект, NPV, идемпотентная математика, генетический алгоритм, имитация отжига.