Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

OPTIMIZATION OF THE SOLUTION OF THE PROBLEM OF SCHEDULING THEORY BASED ON THE EVOLUTIONARY-GENETIC ALGORITHM

A. I. Diveev[1], O. V. Bobr[2,*]

[1]Peoples' Friendship University of Russia (RUDN University), Department of mechanics and mechatronic, Federal Research Centre "Computer Science and Control" of Russia Academy of Sciences, Moscow, Russia

[2]Peoples' Friendship University of Russia (RUDN University), Department of mechanics and mechatronics, Moscow, Russia

Published online: 24 November 2017

ABSTRACT

This article describes the genetic algorithm used to solve the problem related to the scheduling theory. A large number of different methods is described in the scientific literature. The main issue that faced the problem in question is that it is necessary to search the optimal solution in a large search space for the set of feasible solutions in a reasonable time, where the search for valid solutions is a challenging combinatorial task. Typically, for such problems, it is difficult to find any classic method of solution, which would be characterized by acceptable time spent. One of the main reasons is a large number of different constraints. Despite the great achievements and results in this area, the statement of schedule problem is quite abstract and in many cases cannot be used to solve practical problems. Specific details of tasks are considered in the context of each task. In practice, the solution of any tasks related to the planning, based on the use of algorithms to find solutions in a large space.

Author Correspondence, e-mail: bobr_ov@rudn.university

doi: http://dx.doi.org/10.4314/jfas.v9i7s.33

In such cases, the use of genetic algorithms is one of the most common solutions. The purpose of our research was a genetic algorithm with small variations of basic solution finds the optimal solution. Presented genetic algorithm allows to reduce the search space and to propose a variant for the correct schedule. The article presents a mathematical model and describes the algorithm and results of computational experiments, the quality criteria of schedule are described.

Keywords: genetic algorithm; small variations; basic solution; optimal schedule

1. INTRODUCTION

Many scientists in the world have been involved in the higher education class scheduling. With the development of automated means, their introduction into higher education becomes a necessity. The process of scheduling is based on analysis of years of experience and requires a considerable work. Effective resources allocation is a complex and one of the primary tasks facing the universities. At the same time, in the modern fast-changing external and internal environment, effective control is impossible without information support.

With the increased demands for the quality of education, students' work scheduling and rational usage of lecture halls, the actuality of a schedule-making task is also increasing.

Many scientists (V.A. Atroshchenko [1], A.N. Bezginov [2], M. Junginger [3], A. Shraerf [4]have been studying the theory of the schedule, but the whole problem has not been solved yet, both in theory and in practice.

Russian scientists A.A. Lazarev, E.R. Gafarov [5], A.N. Bezginov, S.Yu. Tregubov [6] and foreign scientists E. Burke and C. Petrovic [7] have proved in their works that the principles, the set of models and methods and the approach to optimizing the schedule in a higher educational institution, can claim to be a scientific basis of the schedule theory. A scheduling task refers to an integer programming task. With the number and possible values of variables increasing, the complexity of the task is also increased. Such tasks belong to the NP – complex tasks class. These tasks are characterized by a large number of different-content data, which are difficult to formalize.

2. SOLUTION METHODS

Let us consider the main trends in the development of methods of solving the problem of scheduling.

Polynomial heuristic algorithms were the first developments in this field. Estimation of errors in a resulted solution was found for many of them. These algorithms are called approximate algorithms [8]. The work [9] describes approximate algorithms guaranteeing a relative error and an absolute error in work [8]. Some NP-difficult tasks allow the existence of an approximation scheme that permits to find approximate algorithms with a relative error not exceeding a preset value of $\varepsilon > 0$ in a time-frame, which polynomially depends on $1/\varepsilon$ and on the size of the input information. The development of such schemes was carried out by M.Ya. Kovalev [9], S.V. Sevastyanov and J. Woeginger [10] and many other scientists. It is important to set a limit value ε for tasks that do not have an approximation scheme.

At present, metaheuristic algorithms have been widely distributed and are finding a solution close to optimal at an acceptable time. The drawback of these algorithms is the lack of evaluation of the quality of the solutions received, as it is not possible to determine how much the worst-case solution is different from the optimum.

Much attention is paid to precise methods in solving NP-complex tasks. The most widespread tasks are the screening methods. The application of the branch-and-bound method is described in [11]. If minimized, the lower bounds of the objective function are calculated and the combinatorial properties of the task are used.

Dynamic programming method [12] is also widely used.

Scheduling tasks can often be formulated in the terms of linear integral programming. This method is described in work [13].

One of the successful methods in the scheduling theory is the constraint programming [14].

Using several methods, we can find optimal solutions for the very complex tasks of scheduling theory. These are so-called hybrid algorithms.

The application of the genetic algorithm is considered in work [15].

Genetic algorithm allows to optimize the average performance of the requirements of teachers (according to their wishes, schedule, regularity of classes, etc.). There are works that describe a genetic algorithm with parallelization of the computational algorithm.

The genetic algorithm is the purpose of my research. Previously used methods are directional, but still random search. Their implementation is a resource-intensive process. After the implementation of the algorithm, it is necessary to check a large number of restrictions often not leading to the desired result. A genetic algorithm based on small variations of basic solution, allows you to search the optimal solution in a large search space for the set of feasible solutions in a reasonable time.

3. PROBLEM STATEMENT

Let us consider the optimal schedule task as a multi-criteria task of integer optimization with formalized typical limits.

Formally, the classes schedule can be represented as a matrix

$$X = \left[\mathbf{x}^{i,j}\right],\ i = \overline{0, N-1},\ j = \overline{0, M-1}, \qquad (1)$$

where N is the number of classes per week, M is the number of groups, $\mathbf{x}^{i,j}$ is an integer vector of three components:

$$\mathbf{x}^{i,j} = \left[x_1^{i,j}, x_2^{i,j}, x_3^{i,j}\right]^T \qquad (2)$$

$x_1^{i,j}$ - subject number, $x_1^{i,j} \in X_1 = \{0,1,\ldots,n_1\}$, n_1 - number of subjects; $x_2^{i,j}$ - classroom number, $x_2^{i,j} \in X_2 = \{0,1,\ldots,n_2\}$, n_2 - number of classrooms; $x_3^{i,j}$ - teacher's number, $x_3^{i,j} \in X_3 = \{0,1,\ldots,n_3\}$, n_3 - number of teachers.

Classes schedule for a certain set of students, classrooms, subjects and teachers should be designed. The schedule should meet all the requirements and limitations. The whole range of limitations can be conventionally divided into mandatory and desirable ones. The mandatory ones are the curriculum similarity, fixing lessons for audiences, accounting teachers specializations, non-contradiction to the impossibility of simultaneous lessons conduction, minimizing the total hourly breaks for training groups and the teachers. The desired requirements include, for example, the consideration of teachers' wishes for a schedule.

The requirements to the schedule are described as follows:

$$\sum_{i=1}^{N} \theta\left(x_1^{i,k}\right) = W(i,k), \tag{3}$$

$$\theta\left(x_1^{i,k}\right) = \begin{cases} 1, & \text{if } x_1^{i,k} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $W(i,k)$ is the total number of hours per week for all hours i for the group k. If $x_1^{i,j} = 0$, then group j in the hours i doesn't have any lessons. This expression should be checked when $W(i,k) > 0$ and $x_1^{i,j} > 0$. This requirement means that the number of lessons does not exceed the maximum allowed $W(i,k)$.

$$\sum_{j=1}^{M-1} \sum_{k=j+1}^{M} \theta\left(\left(x_1^{i,j} - x_1^{i,k}\right) + \left(x_2^{i,j} - x_2^{i,k}\right)\right) = 0,\ i = \overline{1,N} \tag{5}$$

This requirement ensures that there are no overlaps for the "training discipline-audience" connection. A training group has a certain classroom for a certain subject.

The set of audiences X_2 is divided into subsets according to the type of their specialization

$$X_2 = \bigcup_{k=0}^{L} \beta\left(x_2^{i,j}, k\right) \tag{6}$$

where $\beta\left(x_2^{i,j}, k\right)$ is a subset of audiences $x_2^{i,j}$ with the specialization k.

Let $\alpha\left(x_1^{i,j}, k\right)$ is the specialization k for the subject $x_1^{i,j}$, then the requirement for the correspondence of the subject to an audience type has the following description

$$\sum_{j=1}^{M} \left(\alpha\left(x_1^{i,j}, k\right) - \beta\left(x_2^{i,j}, k\right)\right) = 0,\ i = \overline{1,N} \tag{7}$$

This requirement ensures that there are no overlaps for the "training discipline-audience" connection: a special classroom will be provided for lectures on a certain subject.

$$\sum_{j=1}^{M-1} \sum_{k=j+1}^{M} \theta\left(\left(x_2^{i,j} - x_2^{i,k}\right) + \left(x_3^{i,j} - x_3^{i,k}\right)\right) = 0,\ i = \overline{1,N} \tag{8}$$

This requirement ensures that there are no overlaps for the "audience - teacher" connection: each classroom is occupied by one teacher for conducting one lesson or there is no lesson in this classroom at all.

Let $\gamma\left(x_3^{i,j}, x_1^{i,j}, k\right)$ be the specialization k of the teacher $x_3^{i,j}$, the subject is $x_1^{i,j}$, then the requirement of the coincidence of the specialization of the audience and the teacher has the following description

$$\sum_{j=1}^{M}\left(\beta\left(x_2^{i,j}, k\right) - \gamma\left(x_3^{i,j}, x_1^{i,j}, k\right)\right) = 0,\ i = \overline{1, N} \qquad (9)$$

This requirement ensures that there are no overlaps for the "audience – teacher" connection: the teacher will be provided with a special classroom suitable for teaching the subject.

$$\sum_{j=1}^{M-1}\sum_{k=j+1}^{M}\theta\left(\left(x_3^{i,j} - x_3^{i,k}\right) + \left(x_2^{i,j} - x_2^{i,k}\right)\right) = 0,\ i = \overline{1, N} \qquad (10)$$

This requirement ensures that there are no overlaps for the "teacher - audience" connection: during the training, the teacher is in the classroom.

$$\sum_{j=1}^{M}\left(\alpha\left(x_1^{i,j}, k\right) - \gamma\left(x_3^{i,j}, x_1^{i,j}, k\right)\right) = 0,\ i = \overline{1, N} \qquad (11)$$

This requirement ensures that there are no overlaps for the "subject - teacher" connection: the subject will be delivered to a training group by a qualified teacher.

Let W be the number of training days. Then N/W is the number of classes a day. The number of each first lesson in the day W is determined from the ratio $r = (\mathrm{w} - 1)\frac{N}{W} + 1$. Last lesson number in the day is calculated by formula $\mathrm{w}\frac{N}{W}$. The day w of the lesson i is determined from the ratio $p = \left[\frac{i-1}{N/W}\right] + 1$. If $x_1^{i,j} = 0$, then there is not lesson i for group j.

For optimization criteria, let's take: the total number of hourly breaks for the training groups and the total number of hourly breaks for teachers:

$$F_1 = \sum_{j=1}^{M}\sum_{\mathrm{w}=1}^{W}\sum_{k=1}^{N/W}\left(q_{k+1}^{j,\mathrm{w}} - q_k^{j,\mathrm{w}} - 1\right) \to min \qquad (12)$$

$$F_2 = \sum_{t=1}^{T} \sum_{\mathrm{w}=1}^{W} \sum_{k=1}^{N/W} \left(q_{k+1}^{t,\mathrm{w}} - q_k^{t,\mathrm{w}} - 1 \right) \to min \qquad (13)$$

where T is the number of teachers, $q^{j,\mathrm{w}}$ is the ordered set of indices of non-empty elements $x_1^{i,j}$: $q^{j,\mathrm{w}} = (k)_{x_1^{r+k,j} \neq 0}$, $k = \overline{1, N/W}$, $\mathrm{w} = \overline{1,W}$, $j = \overline{1,M}$; $q^{t,\mathrm{w}}$ is the ordered set of indices of non-empty elements $x_3^{i,j}$: $q^{t,\mathrm{w}} = (k)_{x_3^{r+k,j} \neq 0}$.

Accounting the criteria F_1 and F_2 the task of an optimal scheduling can be referred to the multi-criteria integer programming class of tasks. To find a solution, we use a genetic algorithm built on the basic solution principle. This approach allows the operation of the genetic algorithm to be applied not to coded solutions (the chromosomes), but to sets of small variations of the basic solution.

Accounting the criteria F_1 and F_2 the task of formulating the optimum solution is referred to the multi-criteria integer programming class of tasks.

Based on the designed mathematical model of the schedule, an evolutionary search of the optimum schedule is implemented using a genetic algorithm built on the basic solution principle [16]. This approach will allow to apply the operations of the genetic algorithm to sets of small variations of the basic solution rather than to coded solutions [17].

3.1 Method of research

Set a small variations vector for a basic solution:

$$\mathrm{w} = [w_1, w_2, w_3, w_4]^T , \qquad (14)$$

where w_1 is the variance code.

Determine the variation as follows:

$w_1 = 0$: simultaneous exchange of any two lines at the lesson w_2 and w_3 in the tables X_1, X_2, X_3.

$w_1 = 1$: exchange of auditoriums during lesson w_2 for groups w_3 and w_4.

$w_1 = 2$: simultaneous exchange of subjects (auditoriums, teachers) for group w_2 at lessons w_3 and w_4.

$w_1 = 3$: exchange of auditoriums for group w_2 during lesson w_3 and w_4.

Let's define the chromosome as a matrix of vectors of the basic solution variations:

$$\mathbf{W} = \left[\mathbf{w}_1^T, \mathbf{w}_2^T, ..., \mathbf{w}_k^T\right]^T, \quad (15)$$

where k is the given length of the chromosome.

Let's define a population $\mathbf{P}$ of chromosomes $\mathbf{W}_i$:

$$\mathbf{P} = [\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_H], \quad (16)$$

where H is the given population size.

Let's apply the Pareto algorithm for ranking the solutions depending on their quality. The fitness function is determined by rank

$$F(\mathbf{W}_i) = \frac{1}{r}, \quad (17)$$

where r is the rank of chromosome $\mathbf{W}_i$, $\mathrm{F}(\mathbf{W}_i)$ is changed in the interval $(0,1]$.

The implementation of the genetic algorithm consists of a sequence of steps:

1) The basic parameters of the genetic algorithm are set. A fixed basic schedule is set. The initial set of chromosomes is generated.

2) Calculate the value of the functionals for the initial set of chromosomes .

3) Based on the values of functionals, rank the chromosomes of the population according to the optimality criterion.

4) Set the number of the current generation .

5) Set the number of the current crossing pair within a single generation .

6) Generate two different random chromosomes in the range .

7) Perform a single-point crossing operation over chromosomes and receive two new descendants .

8) Include the descendants into the population .

9) Apply the variation procedure to and calculate the values of the functionals for all chromosomes of the extended population .

10) Exclude two descendants with the greatest value of functionals from population .

11) . If , then go to step 4, else go to step 11.

12) . If then go to step 3, else go to step 12.

13) End of the genetic algorithm.

3.2 Computational experiment

The proposed algorithm was implemented in the Delphi programming language for an arbitrary basic schedule at the university. The schedule was presented in three tables: X_1 - schedule for subjects, X_2 - schedule for auditoriums, X_3 - schedule for teachers. All elements of the tables have been numbered. The hours of training are located horizontally, the numbers of study groups are located vertically. At the intersection is the number of the discipline (the first table X_1), the number of classrooms (the second table X_2), the number of teachers (in third table X_3). All tables were tested. If the number of the disciplines in the table cell matches the number of the corresponding cell of the table and the violations in the schedule can be avoided. Otherwise, the schedule will be incorrect and will need to conduct another iteration. The genetic algorithm described above was used for optimization. The following algorithm parameters were chosen: number of descendants - 1024, number of generations - 512, number of crossing pairs - 256, chromosome length - 12. The algorithm's run time for one iteration is 30 seconds. Pareto set with two criteria is drawn on the diagram which will be shown on the screen. These criteria are: hourly intervals in the schedule of students and teachers. You could optionally choose any optimal schedule. Should be noted that a schedule had minimum improvements after the fourth iteration.

Table 1. Results of the search for optimal solutions

	1	2	3	4	5	6	7	8	9
F_1	16	6	4	2	2	0	0	0	0
F_2	12	8	6	2	3	3	3	3	3

4. CONCLUSION

This article describes the genetic algorithm with small variations of basic solution found optimal solution in a reasonable time. The found solution is correct, i.e., all requirements and limitations. Achieved the main goal of the algorithm: the number of time intervals in the timetable of students and teachers decreased.

The proposed algorithm can be used for schedule classes in a certain higher educational institution. The benefits include the ability to generate acceptable versions of schedules from the first iteration. Due to introduction of the evaluation function it is possible to choose a desired solution from a variety of suggested solutions in the algorithm. For example, you can choose solutions by reducing the number of intermediary breaks for students or teachers. Using the genetic algorithm in a scheduling system can significantly improve the quality of the schedule.

5. REFERENCES

[1] Atroschenko V.A. (2010). To the question of the design of object-oriented database for high school scheduling. Proceedings of the international conference, St. Petersburg, Russia: 141-144.

Available online at: http://id-yug.com/images/id-yug/Book_id-yug/70-f.pdf

[2] Bezginov A.N. (2011). Set of scheduling algorithms for the university. P.1: The system for schedules quality evaluation based on fuzzy sets; the optimal schedule search algorithm. Bulletin of the Baltic Federal University named after I. Kant, 5: 127-135.

Available online at: https://cyberleninka.ru/article/v/kompleks-algoritmov-postroeniya-raspisaniya-vuza-chast-1-sistema-otsenki-kachestva-raspisaniya-na-osnove-nechetkih-mnozhestv

[3] Junginger W. (1986). Timetabling in Germany - a survey. Interfaces, 16(4): 66-74.

Available online at: http://pubsonline.informs.org/doi/abs/10.1287/inte.16.4.66

[4] Shraerf A. (1999). A survey of automated timetabling. Journal Artificial Intelligence Review archive, 13(2): 87-127

Available online at: https://link.springer.com/article/10.1023/A:1006576209967

[5] Lazarev A.A. (2011). Scheduling theory. Tasks and algorithms. Moscow State University.

Available online at: http://www.twirpx.com/file/984711/

[6] Bezginov A.N., Tregubov S.Yu. (2011). Multi-criteria approach to the assessment schedule based on fuzzy. Control Sciences, 2: 52-59.

Available online at:

https://cyberleninka.ru/article/v/mnogokriterialnyy-podhod-k-otsenke-raspisaniya-zanyatiy-na-osnove-nechyotkoy-logiki

[7] Burke E.K., Petrovic S. (2004). Educational Timetabling. Handbook of Scheduling: Algorithms, Models, and Performance Analysis. Chapman & Hall. 1-45.
Available online at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.65.6156&rep=rep1&type=pdf

[8] Baker K.R., Schrage L. (1978). Finding am optimal sequence by dynamic programming: an extension to precedence-related tasks. Journal Operations Research, 26: 111-120.
Available online at: https://www.statindex.org/articles/29834

[9] PhD thesis Kovalev M.Ya. (1986). Interval and approximate algorithms for solving discrete extremal problem. Belarusian State University, Minsk.
Available online at: http://dep.nlb.by/jspui/handle/nlb/38416

[10] Sevastianov S.V., Woeginger G.J. (1998). Makespan minimization in open shops: A polynomial time approximation scheme. Journal Mathematical Programming, 82: 191-198.
Available online at: https://www.researchgate.net/publication/220589033_Makespan_minimization_in_open_shops_A_polynomial_time_approximation_scheme

[11] Korbut A.A., Sigal I.H., Finkelstein Yu.Yu. (1988). Hybrid methods in discrete programming. Izvestiya USSR Academy of Science, Technical Cybernetics, 1: 65-77.

[12] Bellman R. (2012). Dynamic Programming. The foreign literature publishing house. Moscow. Available online at: http://www.twirpx.com/file/1674300/

[13] Bockmayr A., Pisaruk N. (2003). Detecting infeasibility and generating cuts for MIP using CP. Proceedings of the Fifth International Workshop CPAIOR'03, Montreal: 16-30.
Available online at: https://hal.inria.fr/inria-00107699/document

[14] Carlier J., Pinson E. (1990). A practical use of Jackson’s preemptive schedule for solving the job-shop problem. Journal Annals of Operations Research, 26: 269-287.
Available online at: https://www.researchgate.net/publication/234786317_A_practical_use_of_Jackson"s_preemptive_schedule_for_solving_the_Job-Shop_problem

[15] Berkuta O.N., Lazarev A.A. (1995). Algorithms of 0{n3) complexity for solving the total

delay problem. Journal Research on applied mathematics, 22: 67-78.

[16] Diveev A.I. (2015). Small Variations of Basic Solution Method for Non-numerical Optimization. Proceedings of 16th IFAC Workshop on Control Applications of Optimization, Garmisch-Partenkirchen, 28-33. Available online at: https://www.researchgate.net/publication/285549305_Small_Variations_of_Basic_Solution_Method_for_Non-numerical_Optimization

[17] Diveev A.I., Shmalko E.Yu. (2016). The solution of the problem of two-dimensional packing using the method of variational genetic. Cloud of Science, 3(3): 381-396. Available online at: https://cyberleninka.ru/article/v/reshenie-zadachi-dvumernoy-upakovki-metodom-variatsionnogo-geneticheskogo-algoritma

How to cite this article:
Diveev A I, Bobr O V. Optimization of the solution of the problem of scheduling theory based on the evolutionary-genetic algorithm. J. Fundam. Appl. Sci., 2017, *9(7S), 353-364.*