

O. Haitan¹, O. Nazarov²

¹ National University «Yuri Kondratyuk Poltava Polytechnic», Poltava, Ukraine

² Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

HYBRID APPROACH TO SOLVING OF THE AUTOMATED TIMETABLING PROBLEM IN HIGHER EDUCATIONAL INSTITUTION

Abstract. The paper describes a hybrid approach to solving of the automated timetabling problem in higher educational institution based on the ant colony optimization, the genetic algorithm, and the Nelder–Mead method. The ant colony method is the basis of this algorithm, which forms the initial population for the genetic algorithm. The combination of this method with the genetic algorithm and the Nelder–Mead method reduces time of the convergence of an algorithm and eliminates the strong dependence of the results on the initial search parameters, which usually are selected experimentally. The Nelder–Mead method is used to find the parameters of the ant colony optimization method. Use of the genetic algorithm allows for reducing of algorithm running time and increasing of global optimum finding probability. The educational process timetabling in higher school is an important component of the educational process assurance system, since the schedule quality determines the comfort of the educational process participants and its quality and effectiveness. Therefore, the development of methods for computer-aided timetable generation is **an important challenge**. **The subject of study** is adaptive methods of automated university timetabling. **The objective of the work** is development of a hybrid approach to addressing the problem of automated timetabling in university. **The results** are development and research of a hybrid method and software for university timetabling that been implemented this method.

Keywords: Multi-Objective Optimization, Ant Colony Optimization, Genetic Algorithm, Nelder–Mead Method, Timetable, Timeslot.

Introduction

The educational process timetabling in higher school is an important component of the educational process assurance system, since the schedule quality determines the comfort of the educational process participants and its quality and effectiveness.

The task of the university timetabling is characterized by [1]:

- a significant amount of diverse operational information from various structural units of the university, such as departments, deaneries, the educational office, etc.;

- the difficulty in identifying and formalizing the schedule parameters and limitations; quality of taking them into account completely depended upon the experience and qualification of the dispatch;

- a conflict of interests of the educational process main participants: students and teachers, departments - owners of rooms and specialized equipment, that imposes restrictions on the freedom of their use in the schedule;

- the complexity of adaptation of universal timetabling algorithms to the needs of a particular educational institution, especially taking into account the new working conditions of educational institutions.

Despite the fact that manual timetabling takes a lot of time, it is still widespread due to the significant cost or lack of appropriate computer programs.

Therefore, the study of appropriate methods and the development of effective software for automated timetabling is an important challenge.

The relevance of the development of the effective methods of automated university timetabling has been demonstrated by the international conferences on the Practice and Theory of Automated Timetabling (PATAT) [2]. PATAT is held biennially and serves as a forum for an international community of researchers

and practitioners on all aspects of computer-aided timetable generation. PATAT-2020 conference will be held on 25 Aug – 28 Aug 2020, Bruges, Belgium. PATAT has been supporting various competitions and challenges, in particular International Timetabling Competition ITC 2019, investing in the timetabling community for the benefit of the field.

This work is concerned with research of the adaptive hybrid method of university timetabling and the development of a software system for the course timetabling. It is envisaged to test the developed system on the real data of National University «Yuri Kondratyuk Poltava Polytechnic», at the Educational and Scientific Institute of Information Technologies and Mechatronics.

Analysis of the problem and problem statement

The problem of computer schedule generation was being considered in many papers. In [3], a survey on the methods for university course timetabling was provided, and their benefits and disadvantages were analyzed. Description of the basic methods used for solving this problem is also presented in [4].

Both classical methods, including linear integer programming [5], graph coloring method [6], simulation modeling [7] etc., and metaheuristic methods, including simulated annealing [8], evolutionary algorithms [9, 10] etc. are widely used for automatic course timetabling. It was noted [11] that most of the classical methods use an iterative technique of continuous optimization, it allows finding only a local optimum, and a global optimum can be found only by chance.

In order to go beyond the local optimum and find a global optimum, let combine the ant colony method with the genetic algorithm which will be run each time after the next iteration of the ant colony method.

So, **the objective of the work** is development of a hybrid approach to addressing the problem of automated

university timetabling which will ensure the creation of a schedule during the allowed time with minimal violation of the schedule restrictions.

The subject of study is adaptive methods of automated university course timetabling.

The task of university course timetabling can be presented in a formalized form as follows.

Input data:

According to the curriculum there are d courses S_1, S_2, \dots, S_d to teach in the current semester: for each $i, i = \overline{1..d}$ course S_i consists of lec_i lectures, $pract_i$ practical studies and seminars, and lab_i laboratory studies. Each class s is held by the teacher pr_j . There are gr

curricula G_1, G_2, \dots, G_{gr} , which are sets of courses taught by given student group Gr_k . It means that courses in $G_{ik}, k = \overline{1..gr}$ must be scheduled for different periods.

Let ts_max be number of time periods (timeslots), and r_max_k maximum number of classes, that can be scheduled for period t_k (number of rooms available at period $t_k, k = \overline{1..ts_max}$).

It is necessary to schedule classes for all courses within a certain number of rooms and time periods, i.e. find so s_{tr}^{ijk} ($i = \overline{1..d}, j = \overline{1..pr}, k = \overline{1..gr}, t = \overline{1..ts}, r = \overline{1..r_max}$), that

$$\forall i = \overline{1..d} \sum \{s_{tr}^{ijk} | j \in Pr, k \in Gr, t \in Ts, r \in R\} = lec_i + pract_i + lab_i - \text{scheduling of all classes};$$

$$\forall t = \overline{1..ts_max} \sum \{s_{tr}^{ijk} | i \in D, j \in Pr, k \in Gr, t \in Ts, r \in R\} \leq r_max_t - \text{use of limited room quantity};$$

$$\forall t = \overline{1..ts_max} \forall r = \overline{1..r_max} \sum \{s_{tr}^{ijk} | i \in D, j \in Pr, k \in Gr, t \in Ts\} \leq 1 - \text{absence of room overlaps};$$

$$\forall t = \overline{1..ts_max} \forall k = \overline{1..gr} \sum \{s_{tr}^{ijk} | i \in D, i \in G_k, j \in Pr, t \in Ts, r \in R\} \leq 1 - \text{absence of group overlaps};$$

$$\forall t = \overline{1..ts_max} \forall j = \overline{1..pr} \sum \{s_{tr}^{ijk} | i \in D, k \in Gr, t \in Ts, r \in R\} \leq 1 - \text{absence of teacher overlaps}.$$

The schedule restrictions can be structured as follows (Fig. 1).

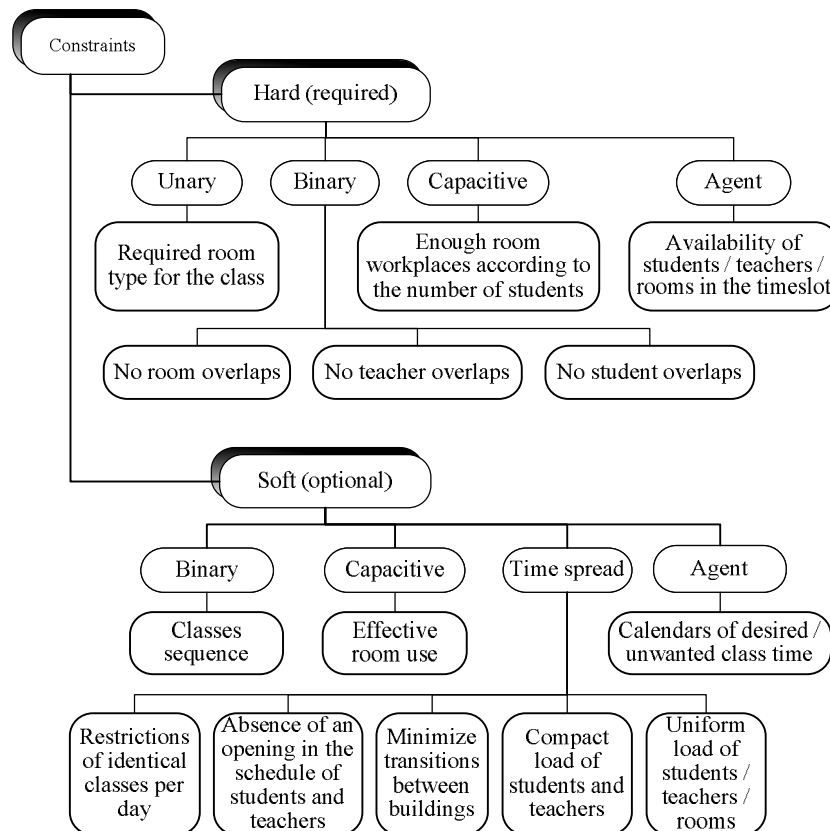


Fig. 1. Restrictions of the generated schedule

Basic material and results

The teaching process unit and therefore the schedule unit is class (lesson).

The class (lesson) is characterized by a triple <Course, group, teacher> (Fig. 2).

Each student can be part of an academic group, subgroup, generalized group (stream), or virtual group.

Lectures on the same course are held by one teacher for all groups of the stream at the same time; therefore, it is necessary to ensure the concept of a generalized group (stream).

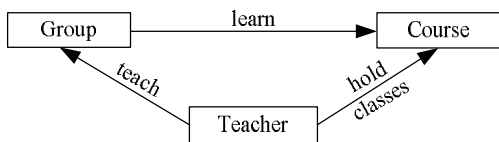


Fig. 2. Triple <Course, group, teacher>

Since students can choose courses to teach (from list presented in optional block of a curriculum), it is necessary to form temporary (virtual) groups from various academic groups to teach a specific discipline. In different subgroups of the same academic group classes can be held at the same time, if they are held by different teachers.

Classes in physical education or disciplines of the humanitarian block, such as philosophy, sociology, religion, etc., can be held simultaneously for all students of one level by various teachers of the department, therefore, it is necessary to introduce the concept of a generalized teacher, which consists of real teachers.

A teacher may not be associated with real teachers if this class is held by an external person, for example, a representative of the business or industry

organizations. In this case we introduce the concept of a virtual teacher.

Timetabling for such generalized groups and teachers requires controlling the employment of all academic groups and all real teachers as the part of these unions.

Therefore, the triple <Course, group, teacher> takes the following form (Fig. 3).

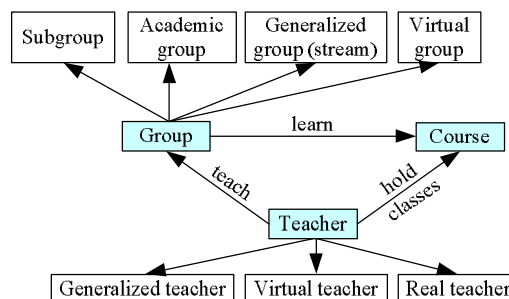


Fig. 3. Relationship among the class entities

The class diagram, which describes the relationship between these entities, is presented in Fig. 4.

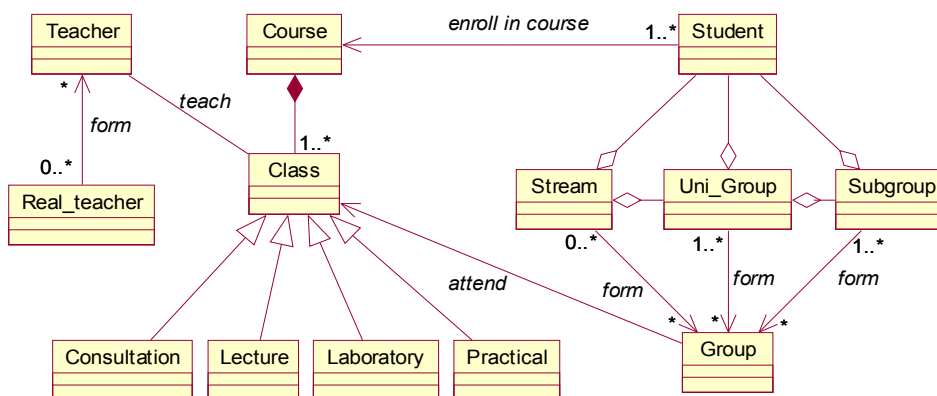


Fig. 4. The class diagram

1. Representation of classes in a schedule

Let use the matrix representation of the schedule in the algorithm. The most optimal schedule presentation is the parallelogram form, where axes are room, day and time, their intersection is class [12]. In this case, only the class number is indicated, by which it is possible to identify uniquely the teacher, group and course.

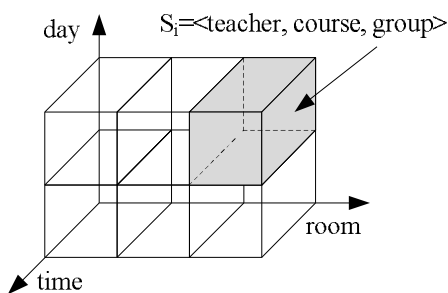


Fig. 5. Representation of the classes in the schedule

Such a presentation will ensure that there are no room overlaps and the possibility of quick data selection

and visualization, for example, using OLAP technologies. Mathematically we describe this parallelogram in the form of a three-dimensional matrix *candidate_list*.

$$candidate_list \{room, day, time\} = \begin{cases} i, & \text{if } S_i \text{ is taken on } day, \text{ time in } room; \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Relationship among the classes in the overall schedule of the educational institution is represented in the form of *union matrix* and *conflict matrix*, which are created according to the curriculum.

The union matrix is a formalization of the streaming. Classes S_i and S_j , for which $M_union_{ij} = 1$, are held simultaneously in the same room.

The union matrix is formed as follows:

$$\forall (s_i, s_j) | s_i \in S, s_j \in S ((s_i^{ducu} = s_j^{ducu}) \wedge (s_i^{kurs} = s_j^{kurs}) \wedge (s_i^{spec} = s_j^{spec}) \wedge (s_i^{lec} \wedge s_j^{lec} \wedge (s_i^{modul} = s_j^{modul})) \rightarrow M_union_{ij} = 1. \quad (2)$$

The conflict matrix is a formalization of the requirement that two classes cannot overlap at a time because of presence of one group / subgroup or teacher in these classes.

Classes S_i and S_j , for which $M_conflict_{ij} = 1$, cannot be taught at the same time.

The conflict matrix is formed as follows:

$$\begin{aligned} & \forall (s_i, s_j) \mid s_i \in S, s_j \in S ((i \neq j) \wedge (M_union_{ij} = 0) \wedge \\ & ((s_i^{pr} = s_j^{pr}) \vee ((s_i^{gr} = s_j^{gr}) \wedge (s_i^{subgr} = s_j^{subgr}))) \vee \\ & ((s_i^{gr} = s_j^{gr}) \wedge (\neg s_i^{subgr} \vee \neg s_j^{subgr}))) \rightarrow M_conflict_{ij} = 1. \end{aligned} \quad (3)$$

2. Used methods

2.1 Ant colony optimization method

The ant colony method is a probabilistic technique for solving the computational problems, which can be reduced to finding the best ways with help of graphs using artificial ants that inherit the behavior of natural ant colony.

This iterative algorithm is based on the idea of successive approximations to the optimal solution.

Each iteration is the launch of an artificial ant that, according to some rule, tries to choose the best route to “food” (optimum function), using the labels of its predecessors.

Each ant performs a chain of steps:

1. Ant moves from node to node with transition probabilities. In order to determine the next edge of the path, the probability of transition to the j -th node is calculated when the agent is in the i -th node:

$$P_{ij} = \frac{(\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}{\sum (\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}, \quad (4)$$

where $\tau_{i,j}^\alpha$ is the amount of pheromones in the node i, j (accumulated statistical information on the quality of choice for position i of requirement j); α is a parameter to control the influence of $\tau_{i,j}^\alpha$; $\eta_{i,j}^\beta$ is node attractiveness (heuristic information about the quality of choice for position i of requirement j); β is a parameter to control the influence of $\eta_{i,j}^\beta$.

2. Recalculation (updating) of the pheromone trace (production and evaporation of the pheromone) is carried out according to the formulas:

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \Delta\tau_{i,j}, \quad (5)$$

where $\tau_{i,j}$ is the amount of pheromone deposited on the node i, j (global pheromone trace); ρ is the pheromone evaporation coefficient; $\Delta\tau_{i,j}$ is the amount of pheromone deposited by the current ant (local pheromone trace).

3. The best solution is the one with the highest value of the local pheromone trace.

Several extensions of the ant system method were developed [13], which include: Elitist Ant System; Rank-

based Ant System (ASrank); Ant Colony System (ACS); Max-Min Ant System (MAX-MIN AS – MMAS).

In this case, we will use the ant colony system (ACS) method, which accumulates benefits and improves other methods.

ACS uses a pseudo-proportional rule to determine the next node to move [13]: with probability q_0 , the agent moves to the point for which the product of the pheromones quantity and heuristic information is maximum; with probability $1 - q_0$ the basic approach that described in the method of ant systems will be applied in determining the next point for the transition.

In this case, a strict elite strategy is used while restoring pheromones on the edges. Only the best agent (globally or locally) adds pheromones after each iteration of the method.

The ant colony method has been developed in detail for the traveling salesman problem (TSP), but the works on course timetabling with this method contain only general theoretical consideration of the possibility of using this method for this task.

The use of the ant colony method for solving optimization problems was considered in [13]. In [14] the timetabling tool based on Best-Worst Ant System and Best-Worst Ant Colony System is considered.

The advantages of the ant colony method are the convergence to the optimal solution, as well as stochasticity, that is, random search, which eliminates the possibility of looping in a local optimum.

At the same time, the uncertainty of the convergence time is noted in [13], while the convergence is guaranteed; a strong dependence of the method results on the initial search parameters, which are selected experimentally.

Therefore, in our opinion, the use of a hybrid approach is an effective approach. The ACS algorithm provides good results in combination with other methods, for example, with local search or genetic algorithms.

2.1.1 Parameters of the algorithm

The initial algorithm parameters, which affect the speed and quality of the solution, are the parameters of influence α , β , and attractiveness of the node $\eta_{i,j}^\beta$. Let consider formation of these parameters.

The parameters of influence α and β are usually selected experimentally. We use the Nelder–Mead method to determine these parameters.

The Nelder–Mead method (also downhill simplex method) is a simple and effective method for optimizing (finding of minimum or maximum) of an objective function in a multidimensional space; it is a way to optimize functions without using gradients. The method is reliable and, as a rule, shows good results, while there is no convergence theory.

The algorithm is generation of the simplex and its further deformation in the direction of minimum using three operations: reflection, expansion, and contract.

In order to form a simplex we divide the ant colony into three groups, for each of which we set individual parameters of the method:

$$\alpha_1 = \text{random}(0.5, 1), \beta_1 = \text{random}(0, 0.5) - \tau_{i,j}^\alpha \text{ is dominant};$$

$$\alpha_2 = 0,5, \beta_2 = 0,5 - \text{uniform influence of } \tau_{i,j}^\alpha \text{ and } \eta_{i,j}^\beta;$$

$$\alpha_3 = \text{random}(0, 0.5), \beta_3 = \text{random}(0.5, 1) - \eta_{i,j}^\beta \text{ is dominant.}$$

If the necessary solution is not reached during *iteration_max* iterations, it is necessary to change the parameters of the ant colony method, calculating the new ones using the downhill simplex method.

In this case, the current value of the parameters $v_1 = (\alpha_1, \beta_1), v_2 = (\alpha_2, \beta_2), v_3 = (\alpha_3, \beta_3)$ is selected as the initial simplex.

Using the Nelder-Mead method requires to calculate the value of the objective function at each point $f_1(\alpha_1, \beta_1), f_2(\alpha_2, \beta_2), f_3(\alpha_3, \beta_3)$, whose equation does not exist, so the maximum local pheromone trace during *i* iterations in each group of ants is taken as this value.

Sorting the points by the values of the objective function at these points, we obtain a double inequality:

$$f(\text{worst}) \leq f(\text{good}) \leq f(\text{best}). \quad (6)$$

Let use the operation of reflection; it is the projection of *worst* point through the center of gravity:

$$\text{new} = \text{mid} + a(\text{mid} - \text{worst}), \quad (7)$$

$$\text{mid} = ((\alpha_{\text{good}} + \alpha_{\text{best}})/2; (\beta_{\text{good}} + \beta_{\text{best}})/2),$$

where $a = 1$.

We get 3 sets of parameters that are selected as the current simplex $v_1 = \text{best}, v_2 = \text{good}, v_3 = \text{new}$.

The algorithm is repeated until the desired schedule is obtained or during *iteration_max* iterations with a new set of parameters.

If the necessary solution is not reached during *iteration_max* iterations, it is necessary to change the parameters of the ant colony method, calculating the new ones similar to the previous paragraph, but instead of the reflection, use the expansion operation according to the formula:

$$\text{new} = \text{mid} + \gamma(\text{worst} - \text{min}), \quad (8)$$

where $\gamma = 2$.

The algorithm is repeated until the desired schedule is obtained or during *iteration_max* iterations with a new set of parameters.

If the necessary solution is not reached during *iteration_max* iterations, it is necessary to change the parameters of the ant colony method, calculating the new ones similar to the previous paragraph, but instead of the expansion, use the contract operation according to the formula:

$$\text{new} = \text{mid} + \beta(\text{worst} - \text{min}), \quad (9)$$

where $\beta = 0,5$.

$\eta_{i,j}^\beta$ is heuristic information about attractiveness of location of class *i* on position *j* in the schedule.

This parameter is calculated heuristically by one of the rules:

1. Under the general rule:

$$\eta_{i,j} = \frac{1}{s_max} \text{ (uniform distribution when there is}$$

no information about the benefits);

2. Under the rule of subjective preferences or fuzzy measure.

With the help of a fuzzy measure, the convenience of placing classes on day *d* in time *p* is defined as a subjective probabilistic measure (Fig. 6).

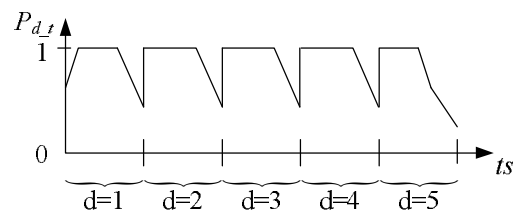


Fig. 6. Subjective probabilistic measure

It is undesirable to hold a lecture on the first lesson on Monday, daily on the fifth or sixth lessons and on the last lessons of the last day of the week, when workability decreases and the least number of students is present. For example, $p_{d,t} = 0$ for $d = 6$ at 5-day workweek, $p_{d,t} = 0,2$ for $p = 6$ (undesirable lecture on the last lesson). The convenience of taking a class in the *r*-th room is determined by the compliance of the room type to the class type, capacitive room indicators, belonging of room to the department, necessary laboratory equipment, software and hardware, as well as the subjective preferences of the teacher. The simplest definition: $\mu_r = 1$ at full conformity of the room, $\mu_r = 0$ at complete discrepancy of the room, $\mu_r = 0,5$ at undesirable taking of class in this room.

According to $\mu_{d,t}$ and μ_r , parameter $\eta_{i,j}$ is defined:

$$\eta_{i,j} = \max(\mu_{d,t} \cdot \mu_r), \quad (10)$$

where $0 \leq \mu_{d,t} \leq 1$ is the convenience of scheduling classes on day *d* in time *t*; $0 \leq \mu_r \leq 1$ – is the convenience of scheduling classes in *r*-th room. Such a presentation allows taking into account the wishes of the educational process participants during course timetabling. If $\beta > \alpha$ at the iteration $i = 1, \dots, N$, an unallocated position with the highest value $\eta_{i,j}$ is selected.

The pheromone trace. In the classical ant colony algorithm, implemented, for example, for the traveling salesman problem, the pheromone is applied to the graph edges, since the edge is a transition path from point to point.

In this case, the main sources of information in the graph are the vertices (timeslot / audience combination), therefore, it is proposed to put the pheromone on the vertices, and not on the graph edges.

2.2 Genetic algorithms

Genetic algorithms are the stochastic heuristic optimization methods, the main idea of which is taken from the evolutionary theory [4]. The main mechanism of evolution is natural selection, whose essence is that more adapted individuals are more likely to survive and reproduce and, accordingly, bring more offspring than less adapted individuals. Moreover, due to the transfer of genetic information, descendants inherit their main qualities from their parents. The procedure of the genetic algorithm:

1. A chromosome structure for storage of solutions is being developed. In this case the schedule as a “chromosome” is considered. The selected structure should take into account all the features and restrictions of the desired solution. An initial chromosomes form population.

2. Each individual in the population is evaluated using the fitness function; the best solutions are copied to the new population unchanged. The fitness function determines the suitability of the chromosome for the formation of the next generation. The principle of elitism allows saving the best decisions and provides increased convergence of the algorithm.

3. The best chromosomes are selected for further reproduction from the set of chromosomes using their fitness function.

4. Selected chromosomes cross-pair with each other to obtain a new population. If the received schedule does not conform to the curriculum, the crossover operation is repeated until the correct schedule is received. In this case, it is better to provide a heuristic mechanism for schedule correcting.

5. The mutation operator is applied to the population.

6. The fitness function is recalculated for the resulting population

7. Go to the step 2.

The genetic algorithm is stable to local minimums, and also provides a relatively quick search.

But application of the evolutionary method for solving the timetabling task causes the problem of

chromosome formation. Therefore, it will be relevant using a combination of the considered approaches – the ant colony algorithm for iterative approximation to the optimum and the initial chromosome formation, and the genetic algorithm for search of the global optimum.

The use of genetic algorithms for timetabling is presented in [4, 9, 11, 15].

3. Hybrid course timetabling method

Let’s consider the main stages of the hybrid method.

Let’s introduce the following variables: s_max is a total number of classes according to the curriculum, d_max is a number of teaching days during the week, p_max is a maximum number of classes per day, $t_max = d_max \times p_max$ is a number of timeslots during the week, r_max is a number of rooms, S is an array of unallocated classes.

Step 1. Initialization of the method.

1.1 Creation of the classes array S with dimension s_max and its sorting by decreasing freedom of location. First of all, classes that have limited opportunities for the time of the classes or require large rooms are located.

1.2 Creation the *Pheromone_matrix* $\{s_i, t_j, r_k\}$ with dimension $s_max \times t_max \times r_max$ and initialization it with the initial pheromone value at the vertices *Pheromone_start* for the implementation of the ant colony method.

1.3 Creation of an of artificial ant colony (agents) $ants[]$. For each ant it is determined: *candidate_list* $\{r_i, d_j, p_k\}$ with dimension $r_max \times d_max \times p_max$, ant path *ant_tour* with dimension s_max , which are initialized with zero values.

1.4 Separation of the ant colony of into three groups for the formation of a simplex by the Nelder-Mead method. For each ant group the initial parameters of the method $\langle \alpha_1, \beta_1 \rangle, \langle \alpha_2, \beta_2 \rangle, \langle \alpha_3, \beta_3 \rangle$ are set.

Step 2. Iterative movement of ants (Fig. 7).

2.1 Setting the initial values of the path, initialization of *candidate_list* and *ant_tour* with zero values.

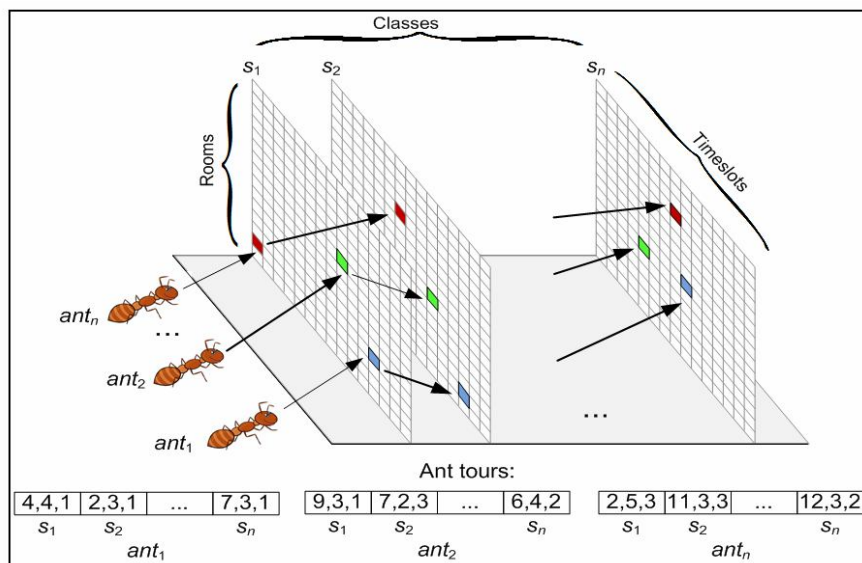


Fig. 7. Schematic representation of the ant path

2.2 Each class i is sequentially scheduled on position j using the transition probabilities. Ant moves from node to node with probability:

$$p_{ij} = \frac{(\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}{\sum (\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}, j \in \Omega, \quad (11)$$

where Ω is the set of classes not scheduled yet.

The rule by which class i is located on position j is defined as follows:

$$j = \begin{cases} \arg \max_{h \in \Omega} (\tau_{i,h}^\alpha)(\eta_{i,h}^\beta), q < q_0; \\ j \text{ is determined randomly} \\ \text{according to the probability} \\ \text{distribution, } q \geq q_0, \end{cases} \quad (12)$$

where $0 \leq q_0 \leq 1$ is the algorithm parameter ($q_0 = 0.5$), and the value q is calculated randomly at each step.

The result of the selection is fixed in the ant path (*ant_tour* matrix) and *candidate_list* matrix.

An agent moves only to nodes that have not been visited yet and for which *candidate_list* $\{r_i, d_j, p_k\} = 0$.

In this case, the *candidate_list* matrix plays the role of a tabu list and ensures the satisfying of hard restrictions “no room overlaps”. Absence of the teacher and group overlaps must be controlled separately.

Step 2.2 is repeated until each agent completes his path.

2.3 Calculation of the fitness (penalty) function F by the formula:

$$F = \sum_j \omega_j F_j, \quad (13)$$

where F is the schedule quality (the number of violations of soft restrictions); F_j is the number of violations of soft restrictions according to the j -th quality criterion; ω_j is the weight coefficient of the j -th quality criterion.

2.4 Application of the local search method and recalculation of the target (penalty) function F .

Local search methods:

Option 1:

– random class $S_0(r_0, d_0, t_0)$ is selected in the schedule;

– all classes $S_1(r_1, d_1, t_1) \dots S_n(r_n, d_n, t_n)$ with the same timeslot (d_0, t_0) are considered;

– classes i and j alternately change places, while the objective function is recalculated;

– a replacement that gives the best value of the objective function is fixed, other classes return to their positions.

Such change will improve the schedule according to the criteria “effective room use” and “uniform room load”. Since the time of the classes does not change, there are no violations of strict restrictions.

Option 2:

– random class $S_i(r_i, d_i, t_i)$ is selected in the schedule;

– class S_i is transferred to all vacant positions in turn, while the fitness function is recalculated;

– a replacement that gives the best value of the ob-

jective function is fixed, other classes return to their positions.

Such change will improve the schedule according to the criteria “compact load of teachers and students” and “uniform load of educational process participants”.

These options of local search can be considered as a genetic algorithm that is applied to one individual: example 1 is selection operation, example 2 is mutation operation.

2.5. The local trace is determined – the amount of pheromone deposited by the i -th ant on each vertex of the path, according to the formula:

$$\Delta \tau_{i,j} = \frac{1}{1 + F_i}, \quad (14)$$

where $\Delta \tau_{i,j}$ is the amount of pheromone deposited by the i -th ant; F_i is penalty for violation of soft restrictions in the schedule created by the i -th ant.

For each group of ants, the largest value of the local pheromone trace is calculated and stored.

2.6. The global trace in the pheromone matrix is calculated by the formula:

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \Delta \tau_{i,j}, \quad (15)$$

if class i is on the position j in the best schedule.

Otherwise,

$$\tau_{i,j} = (1 - \rho)\tau_{i,j}, \quad (16)$$

where ρ is the rate of the pheromone evaporation, uniformly distributed in the range $[0,1]$.

At the first iteration, each vertex has the opportunity to be selected. To gradually remove the vertices relevant to the worst positions, the pheromone evaporation procedure is applied to all the vertices.

If $|ant_tour^{best} - ant_tour^{worst}| < \varepsilon$, the pheromone matrix must be restarted.

Schedules obtained by the ant colony method and improved by local search are transferred to the genetic algorithm as chromosomes.

Step 3. Genetic algorithm

3.1 On the basis of fitness functions, the best chromosomes in each ant group are selected for further reproduction and the creation of a new generation with the best indicators.

3.2 Selected chromosomes cross-pair with each other to obtain a new population, exchanging rooms or timeslots. Genes for crossover (classes) are chosen by the roulette-wheel method. The resulting chromosome is checked for validity – whether it satisfies the hard restrictions. If the received schedule does not conform to the curriculum or the crossover worsened the room use indicators, the crossover operation is repeated until the correct schedule is received.

3.3 Mutation operator is applied to a population. It is a random replacement of a room or timeslot.

3.4 The objective function (fitness function), which fixes the satisfying of soft restrictions, and the local pheromone trace are recalculated for the resulting population.

Steps 2 – 3 are performed until the desired schedule is obtained or during *iteration_max* iterations.

The algorithm stops after achievement of the desired result or after *iteration_max* iterations, as well as if during the *i* iterations the algorithm does not give a significant improvement i.e.

$$F_{global_best} - F_i < \varepsilon .$$

Schematically, the work of the genetic algorithm is presented in fig. 8.

The algorithm flowchart is presented in fig. 9.

Software for university course timetabling was developed based on this hybrid algorithm.

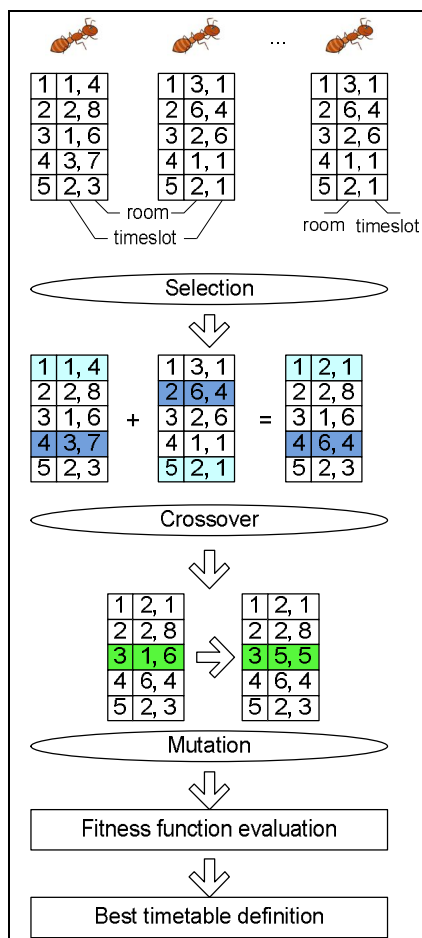


Fig. 8. Genetic algorithm

It provides the possibility of manual and automated course timetabling for higher education institution using the discussed methods and drag-and-drop technologies. The permanent data is stored in the sql-database, and operational semester information is imported from xml files.

The screenshots demonstrating room data updating and manual course timetabling are presented in Fig. 10, 11.

The fragment of the schedule that has generated as a result of aromatic timetabling is presented in Fig. 12.

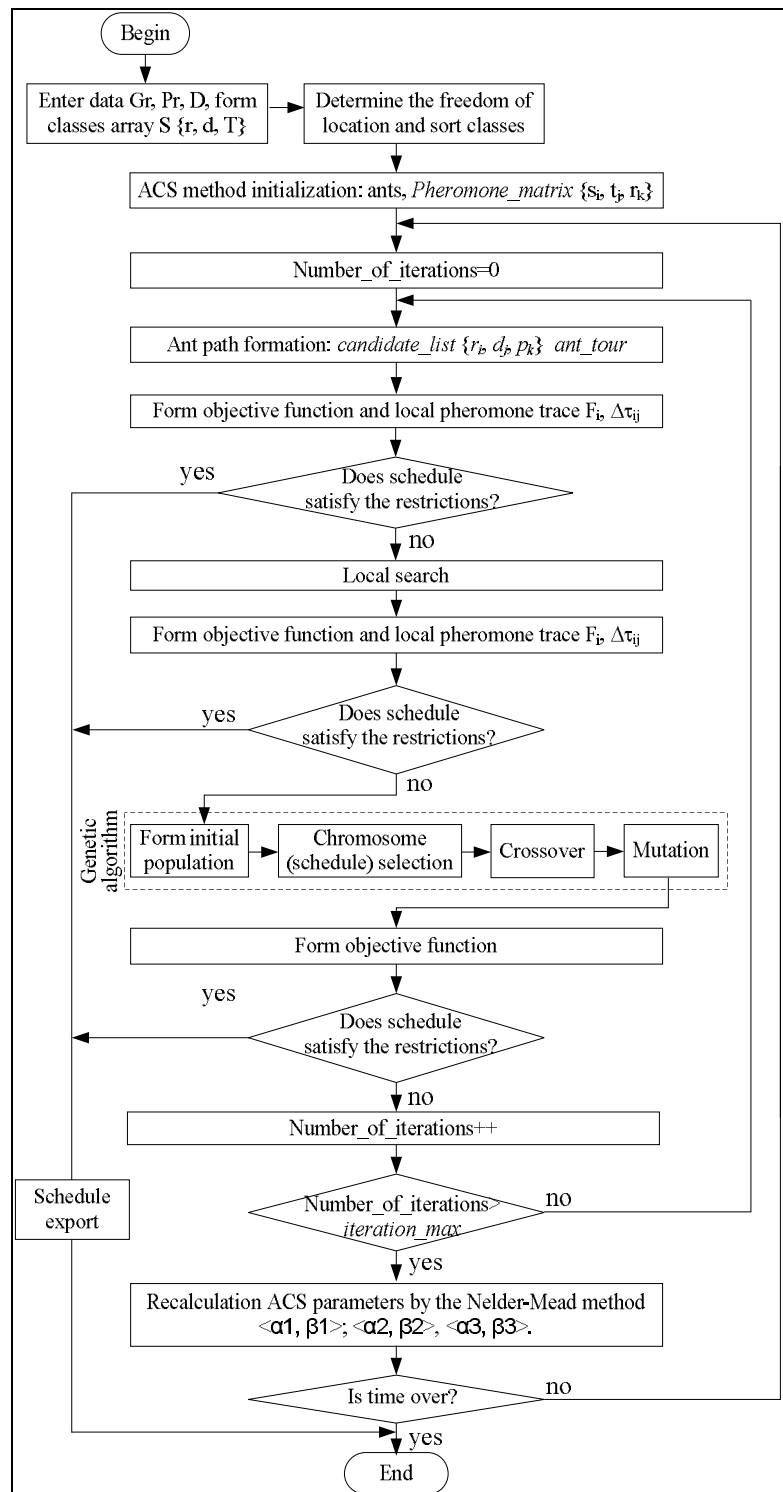


Fig. 9. The algorithm flowchart

The default restriction weights are equal 1, but it is possible to adjust them in manual mode, as well as taking into account individual quality criteria.

The developed system was tested on the following data:

- 1 faculty,
- 56 teachers,
- 54 groups,
- 20 classrooms.

Testing results confirmed the effectiveness of the proposed method.

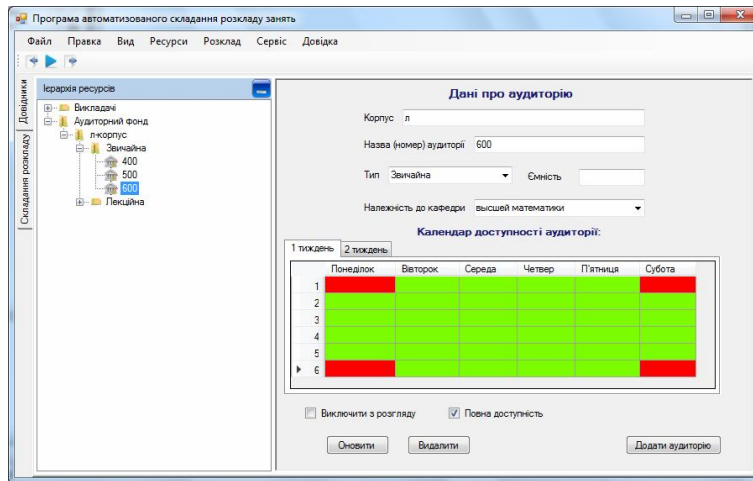


Fig. 10. Room data updating form

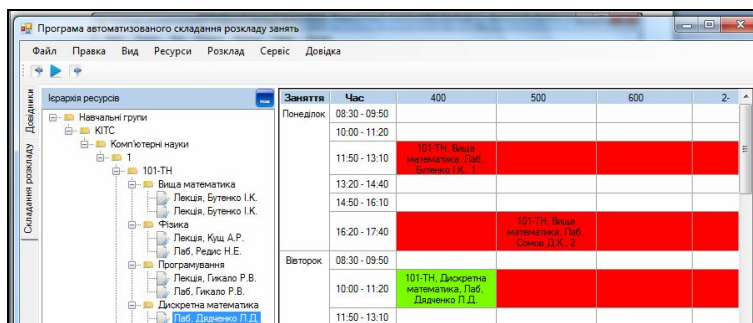


Fig. 11. Manual course timetabling

	A	B	C	D
1	День	Час	101-ТН	102-ТН
2	Понеділок	08:30 - 09:50		Дискретна математика, Лаб., Дядченко П.Д., 303п
3		10:00 - 11:20	Вища математика, Лекція, Бутенко І.К., 301п	Вища математика, Лекція, Бутенко І.К., 301п
4		11:50 - 13:10	Дискретна математика, Лаб., Дядченко П.Д., 303п	
5		13:20 - 14:40		
6		14:50 - 16:10		
7		16:20 - 17:40		
8		Вівторок	08:30 - 09:50	Дискретна математика, Лаб., Дядченко П.Д., 304п
9	10:00 - 11:20		Фізика, Лекція, Куш А.Р., 100п	Фізика, Лекція, Куш А.Р., 100п
10	11:50 - 13:10		Програмування, Лаб., Гикало Р.В., 101п	Вища математика, Лаб., Сомов Д.К.(2), 102к
11	13:20 - 14:40			
12	14:50 - 16:10			
13	16:20 - 17:40			
14	Середа	08:30 - 09:50		Програмування, Лаб., Гикало Р.В., 101п
15		10:00 - 11:20	Вища математика, Лекція, Бутенко І.К., 301п	Вища математика, Лекція, Бутенко І.К., 301п
16		11:50 - 13:10	Фізкультура	Фізкультура
17		13:20 - 14:40		
18		14:50 - 16:10		
19	16:20 - 17:40			
20	Четвер	08:30 - 09:50	Дискретна математика, Лаб., Дядченко П.Д., 304п	Фізика, Лаб., Редис Н.Е., 410ф
21		10:00 - 11:20	Вища математика, Лаб., Бутенко І.К.(1), 101к/Вища математика, Лаб., Сомов Д.К.(2), 102к	Дискретна математика, Лаб., Дядченко П.Д., 303п
22		11:50 - 13:10		
23		13:20 - 14:40		
24		14:50 - 16:10		
25	16:20 - 17:40			
26	П'ятниця	08:30 - 09:50	Фізика, Лаб., Редис Н.Е., 410ф	Дискретна математика, Лаб., Дядченко П.Д., 303п
27		10:00 - 11:20	Програмування, Лекція, Гикало Р.В., 100п	Програмування, Лекція, Гикало Р.В., 100п

Fig. 12. Results of automatic timetabling

Conclusions

As a result of the work, a hybrid approach to solving of the automated timetabling problem in higher educational institution based on the ant colony optimization, the genetic algorithm, and the Nelder–Mead method has been developed. Use of this algorithm allows for reducing of algorithm running time and

increasing of global optimum finding probability. The developed hybrid method is the basis of the software for the university course timetabling. Use of this system at the university will facilitate and increase the efficiency of the dispatchers who are involved in scheduling, and the quality of the educational process by taking into account wish of the educational process participants.

REFERENCES

1. Khasukhadzhiyev A.S., Sibikina I.V. Obobshchennyi algoritm sostavleniya raspisaniya v vuze s uchetom novykh trebovaniy federal'nykh gosudarstvennykh obrazovatel'nykh standartov // Vestnik Astrakhan. gos. tekhn. un-ta. Ser. upravleniye, vychisl. tekhn. inform., 2016, № 3. Pp. 78–86.
2. PATAT Conferences. URL: <https://patatconference.org>.
3. Dvoryankin A.M. Chalyshev V.S. Obzor metodov sostavleniya raspisaniya vuzov // Izv. VolgGTU. Seriya Aktual'nyye problemy upravleniya, vychislitel'noy tekhniki i informatiki v tekhnicheskikh sistemakh: mezhvuz. sb. nauch. st., 2011. Vyp. 11, № 9. Pp. 110–113.
4. Tomashevskiy V.M. Novikov YU.L., Kamins'ka P.A. Skladannya rozkladiv zanyat' u dystantsiynykh systemakh navchannya // Visnyk Natsional'noho tekhnichnoho universytetu Ukrainy "Kyivskiy politekhnichnyy instytut". Ser.: Informatyka, upravlinnya ta obchyslyval'na tekhnika, 2010. Vyp. 52. Pp. 118–130.
5. Lagosha B.A., Petropavlovskaya A.V. Kompleks modeley i metodov optimizatsii raspisaniya zanyatiy v vuze // Ekonomika i matematicheskiye metody, 1993. № 4. Pp. 48–56.
6. Bania Kumar Rubul, Duarah, Pinkey. Exam Time Table Scheduling using Graph Coloring Approach // International Journal of Computer Sciences and Engineering, 2018. № 6. Pp. 84–93.
7. Bezginov A.N. Tregubov S.YU. Obzor sushchestvuyushchikh metodov sostavleniya raspisaniy // Informatsionnyye tekhnologii i programirovaniye. Mezhevuzovskiy sbornik statey, M., 2005. Vypusk 2 (14).
8. Leite Nuno, Melicio Fernando, Rosa Agostinho. A fast simulated annealing algorithm for the examination timetabling problem // Expert Systems with Applications, 2018. Vol. 122.
9. Astakhova I.F., Firas A.M. Sostavleniye raspisaniya uchebnykh zanyatiy na osnove geneticheskogo algoritma // Vestnik VGU, seriya: Sistemyy analiz i informatsionnyye tekhnologii, 2013. № 2. Pp. 93 – 99.
10. Zhukova M.YU., Al'-Gabri V.M. Avtomatizatsiya postroyeniya raspisaniya ekzamenov VUZa s ispol'zovaniyem geneticheskogo algoritma // Inzhenernyy vestnik Dona, 2017. №3.
11. Boyko O.M. Evolyutsiyna tekhnolohiya rozv'yazuvannya zadachi skladannya rozkladiv navchal'nykh zanyat' // Shtuchnyy intelekt, 2006. № 3. Pp. 341 – 348.
12. Mulyava I. YA. Systema formuvannya rozkladu navchal'noho zanyat' z vykorystanniam sub'yektyvnykh perevah // International scientific journal, 2016. № 7. Pp. 22–27.
13. Ustenko S.V., Bibko O.O. Vykorystannya metodu murashynoyi koloniyi dlya rozv'yazannya optymizatsiynykh zadach. Naukovyy visnyk NLTU Ukrainy, vyp. 25, vyp. 3, Berezen', 2015. Pp. 351–359.
14. Thepphakorn T, Pongcharoen P, Hicks C. An ant colony based timetabling tool // International Journal of Production Economics, 2014, № 149(3). Pp. 131–144.
15. Yurchak I.YU., Moskovych T.R. Doslidzhennya henetychnykh alhorytmiv ta zastosuvannya yikh v avtomatyzovaniy systemi rozpodilu navantazhennya dlya vykladachiv i studentiv. URL: <http://com.lp.edu.ua/sntk/doc/ksm2018/moskovytch.pdf>.

Received (Надійшла) 12.03.2020

Accepted for publication (Прийнята до друку) 06.05.2020

Гібридний підхід до розв'язання задачі автоматизованого складання розкладу вищого навчального закладу

О. М. Гайтан, О. С. Назаров

Анотація. У роботі описано гібридний підхід до розв'язання задачі автоматизованого складання розкладу вищого навчального закладу на основі методу мурашиних колоній, генетичного алгоритму та методу деформованого багатогранника. Метод мурашиної колонії є основою даного алгоритму, що формує початкову популяцію для генетичного алгоритму. Комбінація даного метода з генетичним алгоритмом та методом деформованого багатогранника спрямована на усунення таких недоліків даного метода як невизначеність часу збіжності алгоритму та сильна залежність результатів роботи методу від початкових параметрів пошуку, які зазвичай підбираються експериментально. Метод деформованого багатогранника використовується для знаходження параметрів методу мурашиних колоній. Використання генетичного алгоритму дозволяє зменшити час роботи алгоритму та збільшити ймовірність попадання в глобальний оптимум. Складання розкладу навчального процесу у вищому навчальному закладі (ВНЗ) є важливою складовою системи забезпечення навчального процесу, оскільки від якості розкладу залежить комфорт учасників навчального процесу та його якість і ефективність, отже, розроблення методів генерації комп'ютерного розкладу є актуальною темою. Предметом дослідження є адаптивні методи автоматичного складання розкладу ВНЗ. Мета роботи – розроблення гібридного підходу до розв'язання задачі автоматизованого складання розкладу вищого навчального закладу. Результати – розробка та дослідження гібридного методу та програмна реалізація системи складання розкладу навчального процесу в університеті з використанням розглянутих методів.

Ключові слова: багатокритеріальна оптимізація, метод мурашиних колоній, генетичний алгоритм, метод деформованого багатогранника, розклад, таймслот.