Article

Evaluation of the Forms of Education of High School Students Using a Hybrid Model Based on Various Optimization Methods and a Neural Network

Elena Petrovna Dogadina [1,*], Michael Viktorovich Smirnov [1], Aleksey Viktorovich Osipov [1] and Stanislav Vadimovich Suvorov [2]

[1] Department of Data Analysis and Machine Learning, Federal State Budgetary Institution of Higher Education, Financial University under the Government of the Russian Federation, 105187 Moscow, Russia; MVSmirnov@fa.ru (M.V.S.); AVOsipov@fa.ru (A.V.O.)

[2] Federal State Autonomous Educational Institution of Higher Education, Moscow Polytechnic University, 107023 Moscow, Russia; ssw1168@mail.ru

* Correspondence: delena09@yandex.ru; Tel.: +7-9157918524

Abstract: This article deals with the multicriteria programming model to optimize the time of completing home assignments by school students in both in-class and online forms of teaching. To develop a solution, we defined 12 criteria influencing the school exercises' effectiveness. In this amount, five criteria describe exercises themselves and seven others the conditions at which the exercises are completed. We used these criteria to design a neural network, which output influences target function and the search for optimal values with three optimization techniques: backtracking search optimization algorithm (BSA), particle swarm optimization algorithm (PSO), and genetic algorithm (GA). We propose to represent the findings for the optimal time to complete homework as a Pareto set.

Keywords: neural networks; genetic algorithm; backtracking search optimization algorithm; particle swarm optimization; queuing theory; modeling; education

1. Introduction

The period of a child's education at school coincides with a number of important stages in human development. Each stage is characterized by its own characteristics and difficulties that require attention from the older generation. This process is superimposed on a very serious teaching load, which is regulated by SanPiN [1]. The load increases as the student grows up and reaches 34 academic hours per week in high school. In addition to the teaching load, schools are encouraged to provide additional education for schoolchildren. It turns out that the child spends most of his active time at school, and his parents see him only in the evening. In this situation, a significant part of parenting functions has been delegated to school teachers and teachers of additional education.

The COVID-19 pandemic, with forced self-isolation, has produced an experiment to change the way of life of each family. Pupils moved to distance learning, and the functions that had been delegated to school teachers by the parents returned. The result has been both a general decline in school performance and a widening gap between grades and schools. This means that most of the students in the allotted time cannot perform the exercises that their predecessors successfully solved. In order to identify the criteria that affect the quality of school learning and by which the difference between classroom learning and online learning can be established, the authors have studied a number of publications [2–11].

Wang, Fan, and Xu [2] compared characteristics of math exercises in a thorough examination of the contents of School Mathematics Textbooks in ten countries. The researchers examined the quantity, type, openness, and difficulty level of tasks, furthermore their similarities and distinguishing features. According to the openness of conclusion, all the

exercises were divided into "closed" and "open-ended" ones. The type of exercise is another feature that divides the exercises into groups based on the type of questions. The authors of this research described six types of questions such as multiple choice questions or solution questions. An approach based on factor analysis was applied to determine the exercise difficulty.

M. Mohseny et al. [3] studied the influence of the cyber-environment on students' mental health. As a part of an international study, the researchers used a comprehensive questionnaire to obtain many vital characteristics of school students. The researchers examined important domains such as home and school environment, including safety, interaction with teachers, school community, and homework.

The authors of the articles [4,5] worked to highlight the parameters that influence the learning process. C. Masci et al. [4] examined different parameters influencing the teaching process in the framework of the school value-added concept. The researchers studied the effect of a particular class or school in Italy on the achievements of students in reading and mathematics. To do that, the researchers offered the novel statistical method and explored differences and similarities of class effects. As the distance from home to school matters for student's time, M. Febriana et al. [5] studied the distribution of schools and students' home locations in Makassar City, Indonesia. The authors noticed gaps in education in different regions of the country and mentioned the regulatory requirements concerning the zoning of schools. They analyzed spatial data with the use of the k-Means algorithm with the view to better zoning.

Articles [6,7] are devoted to optimizing the learning process. Marcenaro-Gutierrez et al. [6] offered an approach to multiobjective optimization of the teaching process. They considered such characteristics as mean math score, mean reading score, and the levels of achievement of particular success in math and reading. The researchers regressed the said characteristics against students' satisfaction measured in scores and used the regression models to solve the problem with multiobjective programming. Shehab et al. [7] concentrated their efforts on examining the American K-12 educational system learning environment, especially safety and learning facilities. Developing the solution to optimize the quality of learning, the researchers considered such characteristics as the number of served students, the number of English language learner students, and the number of students from low-income families.

A. Shukhman et al. [8] explored the ways to support talented school students with machine learning algorithms that improve traditional learning management systems to introduce individual learning paths. The authors of this article offer an approach to the problem of a quantitative assessment of student's competencies. The researchers proposed formulas to calculate competency levels and ratings.

Based on the above publications [2–11], we established a criteria vector that affects completing homework.

2. Materials and Methods

2.1. Neural Network to Measure Homework Performance

To define the effectiveness of the homework performance, we applied a Multilayer Perceptron (MLP) neural network, realized with the help of Neural Excel VBA Extension Pack Software [12], and trained by the Resilient Propagation method. This method has some advantages over other methods that solve similar problems; mainly, it is easy to implement and has a high convergence rate with low gradient computation error requirements. The algorithm uses the so-called «learning by epochs» when the correction of weights occurs after presenting the network of all data from the training sample. To determine the effectiveness of homework, we chose between the two most appropriate options:

- Multilayer Perceptron (MLP) neural network trained by the Resilient Propagation method;
- Random Forest Algorithm.

The first method has some advantages over other methods that solve similar problems; it is simple to implement and couples a high convergence rate with low computational

requirements. The algorithm uses the so-called "learning by epochs" when the weights are corrected after all the training samples are presented to the network. The disadvantages of this method include the possibility of overtraining (in this case, the accuracy on the training set will significantly exceed the accuracy on the test set) and the need to choose a network for a specific task.

The second method is unpretentious to the data used; it can work with both categorical and numerical data and in the cases of missing or unscaled values. The main disadvantage of this method is its demand for computational resources.

The Multilayer Perceptron (MLP) neural network is implemented using the Neural Excel VBA Extension Pack software [12] and trained using the Resilient Propagation method.

Compared to [13], we added five more neurons to the input layer of the neural network, corresponding to five new parameters; therefore, the total number of parameters now is 12:

1. Number of topics the assignment deals with;
2. Number of types of activities students use while doing the assignment;
3. Number of questions in the assignment;
4. Length of the assignment;
5. Complexity in formulating the assignment;
6. Age of a student;
7. Sex of a student;
8. Distance from home to school;
9. Average math score;
10. Average reading score;
11. Learning mode;
12. Family income.

- The number of topics covered by the assignment—this value is determined by lexemes specific to a particular topic. For example, the "electric current" lexeme means that the assignment has a topic related to electricity.
- The length of the assignment is the number of words required to formulate the assignment.
- The complexity of the assignment formulation is an expert value determined in scores in the range from 1 (the assignment is formulated clear and unambiguously) to 3 (there are redundant data and ambiguous formulations).
- The age of the school student is in years.
- Sex is a binary value.
- The distance from home to school is indicated in kilometers.
- The average math score is the average score in mathematics for the previous schooling period (transferred to a five-point system). Depending on the school, this is a quarter or trimester grade.
- The average score in reading is calculated for lower grades or in the national language for students of senior grades.
- Learning mode—binary value (distant—0 or in-class—1).
- Family income is a value determined in points from 1 (the family receives subsidies from the state) to 3 (the family can afford expenditures higher than average).

The target value will be a number that ranges from 0 to 1, where 0 is an entirely useless assignment with no impact on the perception of the material and 1 is an ideal to be pursued (i.e., an assignment that allows the school student to learn the material thoroughly).

The first five criteria assess the exercises, while the remaining seven are the conditions under which the pupils do their homework.

To build the model, we used a neural network consisting of three layers of neurons (Figure 1):
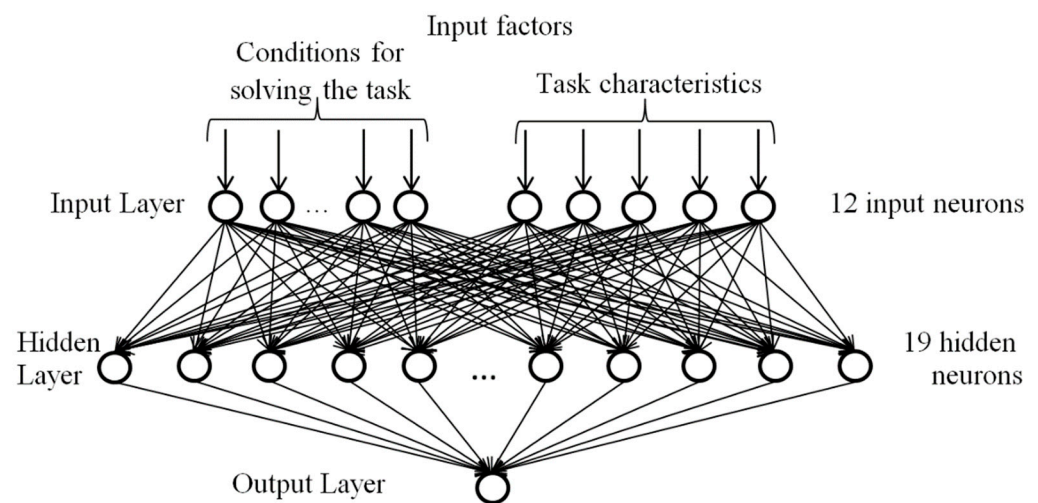
Figure 1. MLP structure.

Input layer—12 neurons;
Hidden layer—19 neurons;
Output layer—1 neuron.

To train the neural network, we prepared a training set consisting of 150 assignments in various subjects and 53 schoolchildren. Approximately half of the schoolchildren solved these assignments before the COVID-19 pandemic and half during the pandemic; for each schoolchild, the values of the effectiveness of each assignment were derived by the method of expert evaluation. The experts took into account both the assignment itself and the conditions influencing its solution. For testing, we also created a test sample of 50 assignments. The model predicted the effectiveness of the assignments with 79% accuracy on the training set and 72% accuracy on the test set.

The Random Forest Algorithm was implemented in Python using the Scikit Learn package and the same training set. When comparing the results, it turned out that the Random Forest Algorithm showed an efficiency of 69% on the test set, which is 3% lower than in the case of a neural network.

The authors of the article compared the results obtained by the neural network when processing data before the pandemic and during it. It was found that the average efficiency of solving the problem by each student during a pandemic decreased by 15%. In addition, there is a significant increase in the scatter of this parameter.

2.2. Optimization Algorithms for Solving Multicriteria Problems

Many real decision-making problems have to deal with several conflicting criteria that need to be optimized simultaneously. The traditional optimization approach, in which a goal is optimized according to a given set of constraints, is not applicable. In such cases, the problem of multicriteria optimization is formulated, which consists of simultaneously optimizing several target functions that mathematically model the criteria, taking into account several constraints defining a valid set of solutions. The set of all optimal solutions is determined by the optimal Pareto set [13,14].

This article assumes the possibility of solving the problem of multicriteria optimization of doing homework by students with the maximum possible efficiency of assignments and minimum labor input for their completion.

To achieve these goals, we considered a multicriteria optimization problem with limitations. We developed a model with a hybrid approach involving the queuing theory, genetic algorithm (GA), and neural network in our solution. The multiobjective optimization problem remains relevant in recent years, especially considering the number of works in the production, engineering, education, and other research fields. Moreover, many such publications provide information on the use of GA, including the analyzed ones [14–16]

where the researchers apply GA to solve scheduling and similar problems. Amjad et al. [14] made a detailed review of recent achievements in the application of GA to solve the flexible job-shop scheduling problem (JSSP). Viana, Junior, and Contreras [15] proposed a new GA with improved crossover and mutation operators for JSSP. Rarità et al. [16] developed a supply chains model through partial and ordinary differential equations. The authors of this article proposed to use GA to control the outflow of the supply chain.

Modern researchers often apply hybrid approaches involving evolutionary (EA), particle swarm optimization (PSO), backtracking search optimization (BSA) algorithms, and neural networks to solve multicriteria optimization problems [17–22]. Y. Hu et al. [17] presented a hybrid solution combining GA, PSO, and backpropagation neural networks (BPNN) to forecast electric load. M. Sedak and B. Rosic [18] used a hybrid approach by introducing differential evolution algorithm mutation operators into the PSO velocity update equation. L. Wang et al. [19] developed advanced BSA improving processes of selection and mutation and applied the algorithm in the field of supply chain management, considering joint replenishment problems.

As generalized criteria, we used the additive optimality criterion in our work:

$$F(\xi, K(X)) = \sum_{j=1}^{n} \xi_j K_j(X) \tag{1}$$

where $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$—weighting coefficients of the relative importance of criteria (vector of convolution parameters).

The weighting coefficients are set in accordance with the conditions:

$$0 \le \xi_j \le 1, j = 1 \ldots n$$
$$\sum_{j=1}^{n} \xi_j = 1, j = 1 \ldots n$$

The task of finding the optimal time for completing homework by students, taking into account the optimality criteria, is implemented using optimization methods, the apparatus of the queuing theory, and a neural network that determines the effectiveness of the assignments proposed to the student.

$$K = f(p(t), X) \to min(max) \tag{2}$$

$$K = (K_1, K_2, \ldots, K_r),$$

$$\psi_i(p(t), X) \le 0, \quad i = \overline{1, l} \tag{3}$$

$$X = (\lambda, \mu, M, Q) \in \Omega_{dop} \tag{4}$$

$$\frac{dp(t)}{dt} = f(X(t), p(t)) \tag{5}$$

$$p_a(0) = p_{a0}, a \in N \tag{6}$$

$$\sum_{a \in N} p_a(t) = 1 \tag{7}$$

where K—vector-function of the selected criteria of optimality of production processes, X—vector of optimized parameters of the system, which depend on the probability density of transitions of the system, λ—vector of intensities of input flows of applications, μ—vector of intensities of their service, M—the number of service devices, Q—the length of the system queue, $p(t)$—vector-function of probabilities of the states of the system in the considered time interval $t \in \{0, T\}$, determined by the model of the form (5–7), discussed in detail in the work [13,23]. The system of constraints (3) and the expression (4) define the scope of acceptable solutions to the problem.

Many methods have been developed to increase the efficiency and speed of implementation of optimal search procedures; however, almost all of them have limitations associated with the mathematical model of the system under study.

The paper reviews the behavior of the objective function and the search for optimal values by three optimization methods: backtracking search optimization (BSA), particle swarm optimization (PSO), and genetic algorithm (GA). The choice of the three presented optimization methods was made according to these criteria: 1—ease of use; 2—the minimum time spent on implementation; 3—the accuracy of the result; 4—the possibility of rapid modernization of the method; 5—search speed for the optimization method; 6—the minimum cost of computer resources (lower minimum total cost).

Each algorithm has its pros and cons; some converge faster at the first point found, others try to find several minima and smoothly approach each of them. To understand the results of the algorithms, it is necessary to briefly describe the processes of their functioning in relation to the problems of multicriteria optimization.

Let us describe the optimization algorithm of the method with a return in case of a failed step, considering the multicriteria problem under the study [23]:

Step 1. Set the starting point $X^{(0)}$, the initial step length $\lambda^{(0)}$, set the iterations counter $r = 0$, set R—the finite number of failed attempts and ε—condition for the end of the search.

Step 2. Set the initial value of failed attempts counter $k = 1$, and constant parameters (step reduction coefficients) β and γ for further modification of the step length.

Step 3. Obtain $\zeta^{(r)}$ and by formula $X^{(r+1)} = X^{(r)} + \lambda^{(r)} * \zeta^{(r)}$ find a trial point $X^{(r+1)}$. If $X^{(r+1)} \in \Omega_{\partial on}$, then go to Step 4, else calculate $\lambda^{(r)} = \lambda^{(r)} * \beta$ and go to Step 3.

Step 4. Calculate the value of the function $F(X^{(r+1)})$ by formula (1) at the point $X^{(r+1)}$.

Step 5. If $F(X^{(r+1)}) > F(X^{(r)})$, then we put $\lambda^{(r+1)} = \lambda^{(r)} * \gamma$, $r = r + 1$ and go to Step 3, else—to Step 6.

Step 6. Set $k = k + 1$. If $k < R$, then go to Step 3, else go to Step 7.

Step 7. Choose optimal solution $X^* = X^{(r+1)}$.

The algorithm presented above considers the multicriteria nature of the optimization problem, which is solved by applying formula (1). The optimized parameters themselves can be represented in the form of both a scalar and a vector. In addition, the values of the required criterion functions are calculated based on the developed generalized mathematical model of the process of doing homework by students, taking into account the time-variable optimization parameters [13].

Let us describe the particle swarm optimization (PSO) algorithm used in this article:

Step 1. Initialize particles swarm. Each of the particles of the swarm can be described by its coordinates $x_i = \{x_{i1}, x_{i2}, \ldots, x_{id}\}$ and velocity $v_i = \{v_{i1}, v_{i2}, \ldots, v_{id}\}$, where i—particle number, d—search space dimension. At this step, we set the coefficient of inertia, acceleration constant, maximum iterations, and minimum allowable error to complete the algorithm.

Step 2. Estimate the initial fitness of each particle.

Step 3. The initial fitness value is used as the current local optimal solution for each particle, and the position corresponding to each fitness value is used as the optimal local solution for each particle.

Step 4. The best initial fitness value is taken as the optimal global solution, and the position corresponding to the best fitness value is taken as the global optimal value position.

Step 5. Update the current flight speed of each particle using the formula.

$$v_{k+1} = \alpha * v_k + \beta * r_1 * (p_k - x_k) + \gamma * r_2 * (g_k - x_k) \tag{8}$$

where p_k and g_k—the best solution coordinates, found by the particle itself and the swarm accordingly, r_1 and r_2—random numbers in the interval of [0,1]. Coefficients α, β and γ determine the degree of influence of each of the three components on the particle velocity.

Step 6. Limit the flight speed of each particle so that it does not exceed the specified maximum flight speed.

Step 7. Update the current position of each particle according to the formula.

$$x_{k+1} = x_k + v_{k+1} \tag{9}$$

Step 8. Compare whether the current fitness of each particle is better than the historical local optimum. If so, the current fitness of the particle is used as the optimal local solution for the particle, and its corresponding position is used as the location of the optimal local solution of each particle.

Step 9. Find the optimal global solution in the current group and use the position corresponding to the current global optimal solution as the optimal global solution for the particle swarm.

Step 10. Repeat Steps 5–9 until the minimum error value, or maximum iteration number is reached.

Step 11. Form Pareto-optimal solutions.

The algorithm presented above is a classical particle swarm method adapted for solving multicriteria problems with the representation of the optimized parameter in both discrete and vector forms. The effectiveness and reliability of PSO largely depend on maintaining the right balance between the stages of exploration of the search space and the localization of the extremum. To regulate the ratio between these stages, such free algorithm parameters as α, β and γ are used.

Let us formulate the genetic algorithm [13,23], comprising the following steps:

1. Generating initial population. Filling the population with individuals in which the array elements (bits) are filled randomly within the boundaries defined by the user.
2. Determining algorithm parameters. The parameters are size of the population N_pop, the number of generations N_pok, the probability of crossover $P_{skresch}$, and the probability of mutation P_{myt}, which determine for each population the number of pairs of crossing chromosomes and the number of mutating chromosomes.
3. Generating initial population. The initial population can be randomly generated.
4. Choosing a parental couple. The selection of the parent pair is carried out using the roulette method, that is, the proportional selection method. Chromosomes are displayed as a segment of lines or roulette sectors in such a way that their size is proportional to the value of the objective function. Next, we randomly generate numbers in the range from 0 to 1, and those individuals in whose segments the random numbers fall are selected as parents. In this case, the chromosome numbers of the parents must be different.
5. Crossover. For a crossover, we pick a random point and choose chromosomes. After that, we use the single-point crossover.
6. Mutation. The number of mutations is determined, and chromosomes for mutation are selected. A single-point mutation is carried out.
7. Checking the condition for completing the evolution process. If the condition for the termination of the algorithm is not met, then go to Step 4; otherwise, go to Step 8. As a condition for the termination of the process, there can be a specified number of generations or a defined number of identical individuals.
8. Formation of a Pareto-optimal solution.

In the process of implementing all three algorithms, the participation of a non-stationary queuing system is necessary, the generalized form of which is presented below [13]:

$$\frac{dp_a(t)}{dt} = - \sum_{\substack{a,b \in N' \\ b \neq a}} d_{ab}(t) \cdot p_a(t) + \sum_{\substack{a,c \in N' \\ c \neq a}} d_{ca}(t) \cdot p_c(t) \tag{10}$$

where N'—a plurality of pairs of indexes of the states $N' = \{(a,b) \in N^2 | d_{ab} = (S_a, S_b)\}$ $N = \{0,1,2,\dots\}$.

The use of variables *a*, *b*, *c* in the mathematical model suggests that the system is in states $S(m_a, z_a, q_a)$, $S(m_b, z_b, q_b)$ and $S(m_c, z_c, q_c)$ respectively.

The initial state is expressed as

$$p_a(0) = p_{a0} > 0, \quad a \in N \tag{11}$$

It is also necessary to observe the condition of normalization

$$\sum_{a \in N} p_a(t) = 1 \tag{12}$$

The density $d_{ab}(a \neq b)$ of the transition from the S_a state to the S_b state is defined as follows

$$d_{ab} = \begin{cases} \mu_a(t), b = a - 1, t \in [0, T]; \\ \lambda_a(t), b = a + 1, t \in [0, T]; \\ 0, a \neq b, a = 0, 1, 2 \dots. \end{cases}$$

3. Results

To analyze the efficiency of the proposed algorithms, we carried out a number of computational experiments. In the experiments, the process of obtaining the optimal time for completing homework by students is considered, taking into account the minimum labor costs and the maximum efficiency of this process. The performance criterion is influenced by the neural network, which considers a number of external factors. One of the parameters of the neural network is the form of training: in-class and distance learning. Therefore, this article focuses not only on the comparative analysis of optimization methods for the studied objective function, but also on obtaining the optimal time for completing homework, taking into account the form of training.

In this optimization problem, we represent students of six classes doing their homework as service channels; applications are tasks in subjects; the input stream is formed by the teachers' requirements for students to complete their homework in accordance with SanPiN and Federal State Educational Standard (for example, the full employment of the student during the time allotted for homework).

It is necessary to introduce the following designations: *M*—the number of students (the model takes into account the possibilities of performing both individual tasks and project tasks for a group of students); *Q*—the length of the queue of items with tasks; $\lambda(t)$—the intensity of receipt of a set of tasks in subjects, distributed according to the exponential distribution law; $\mu(t)$—the intensity of the student's homework, distributed according to the exponential distribution law; *T*—the considered operating time of the system, limited to a 2.5 h working day (for students of six grades).

The mathematical model of doing homework by students has the form (8) with the number of system states *s* depending on the number of students and the capacity of the queue with assignments.

The parameter to be optimized is the average time for completing homework by one student, which is determined through the intensity and is equal to the reciprocal of the average time for making $\bar{t}_{lesson}$ homework $\mu = \frac{1}{\bar{t}_{lesson}}$.

We will use the following characteristics as optimality criteria:

1.　The average relative time for completing homework, taking into account the difficulty of a particular subject, is as follows:

$$K_1 = \frac{1}{M \times T} \times \bar{t}_{lesson} \rightarrow min \tag{13}$$

where $\bar{t}_{lesson}$—average time to complete homework (min.) for one lesson.

The time limit for completing homework (min.) for one lesson is as follows:

$$t_{lesson} = 60 \times D \frac{B_i}{\sum_i B_i P_i} \tag{14}$$

where

D—the time limit for completing homework during the week, according to hygiene requirements (hours);

B_i—the difficulty of a separate i subject;

P_i—the number of hours in the curriculum for the i subject.

2. Average relative efficiency of homework in terms of material assimilation

$$K_2 = \frac{\overline{Ef} \times \overline{t}_{lesson}}{Ef_{max}} \to max,$$
$$\overline{Ef} = a_N * kol, \tag{15}$$

where $\overline{t}_{lesson}$ is the average time for completing homework (min) for one lesson; $\overline{Ef}$—the average efficiency of assignments in subjects assigned to the home; Ef_{max}—maximum efficiency, a_N—task efficiency coefficient obtained using a neural network, kol—number of tasks.

It is necessary to impose the condition on the parameter to be optimized with a constant average time for completing homework

$$a \leq \mu(t) = \mu \leq b,$$

where a and b are determined by the decision-maker.

Here is an example of optimizing the performance of individual homework by one student in-class learning. We have the following data: $M = 1$—the number of students; $Q = 5$—the length of the queue of subjects for which homework is given; $\lambda = 1/15 \, min^{-1}$—the intensity of the input flow of incoming applications; $T = 2.5$ h. The accuracy of solving the system of Equations (8)–(10) by the Runge–Kutta method is 4–5 orders of magnitude of accuracy $\Delta t = 0.1$ min; $\mu_i \in [0.01; 1] \, min^{-1}, i = \overline{1, n}$.

The solution of this problem by the method using the genetic algorithm is presented in Table 1, and the corresponding Pareto-optimal set is shown in Figure 2. In this example, we assume the size of the population $N\_pop = 25$ and the number of generations $N\_pok = 20$, the probability of crossover and mutation is 0.01.

Table 1. Optimization results by the method using the genetic algorithm.

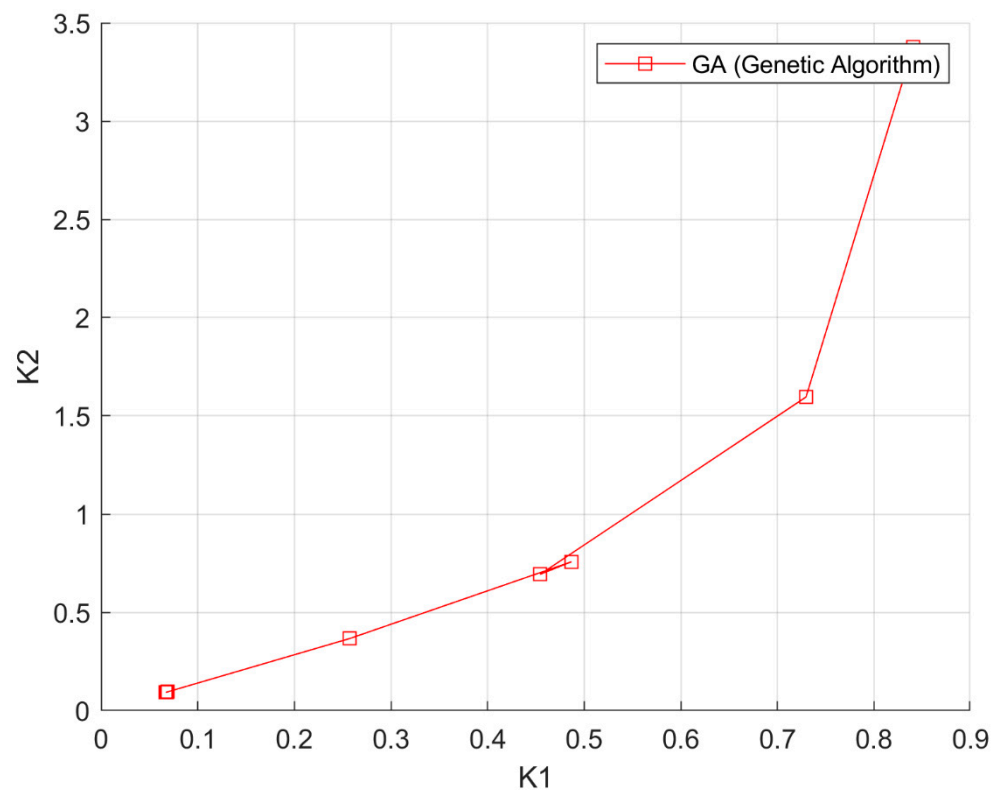| ς	K_1	K_2	μ, min^{-1}
0.0	0.0682	0.0943	0.9869
0.1	0.0688	0.0951	0.9721
0.2	0.0670	0.0927	0.9867
0.3	0.0680	0.0941	0.9731
0.4	0.0676	0.0934	0.9798
0.5	0.0679	0.0939	0.9749
0.6	0.2570	0.3661	0.25
0.7	0.4869	0.7573	0.1209
0.8	0.4544	0.6933	0.1477
0.9	0.7296	1.5957	0.0574
1.0	0.8406	3.3763	0.0271

Figure 2. Pareto-optimal set for solving a problem by a method using a genetic algorithm.

The solution to the optimization problem using the particle swarm algorithm (PSO) is presented in Table 2, and the corresponding Pareto-optimal set is presented in Figure 3.

In this example, the size of the swarm particlesize = 30, the maximum number of iterations MaxNum = 100, inertia $\alpha = 0.6$, the individual learning factor for each particle $\beta = 2$, social learning factor per particle $\gamma = 2$.

Using the backtracking algorithm, the problem of optimizing the time for completing homework is solved. The results of the work are presented in Table 3, and the corresponding Pareto-optimal set is presented in Figure 4. Let us set the starting point equal to the value of half of the segment under study, the initial step length $\lambda = 0.2$, the finite number of failed attempts $R = 20$, and the search end condition $\varepsilon = 0.01$.

In this problem, we chose the following values from the Pareto set, obtained with the genetic algorithm: $\zeta = 0.9$, $K_1 = 0.7296$, $K_2 = 1.5957$. With these values, the time to complete homework is 17.43 min. From the Pareto set obtained by optimizing the swarm of particles, the chosen values are: $\zeta = 0.9$, $K_1 = 0.7498$, $K_2 = 1.7080$. For these values, the homework time is 18.65 min. Using backtracking method, we chose the solution $\zeta = 0.9$, $K_1 = 0.7578$, $K_2 = 1.8140$, Solution 2 was selected using backtracking method, which corresponds to 19.84 min. Of the values obtained by various methods, the minimum is the value obtained using the genetic algorithm. The resulting value is 10.3% more effective than the current time for completing homework by students and does not contradict the norms of the learning load regulated by the SanPiN.

Table 2. Optimization results by the method using a particle swarm algorithm.

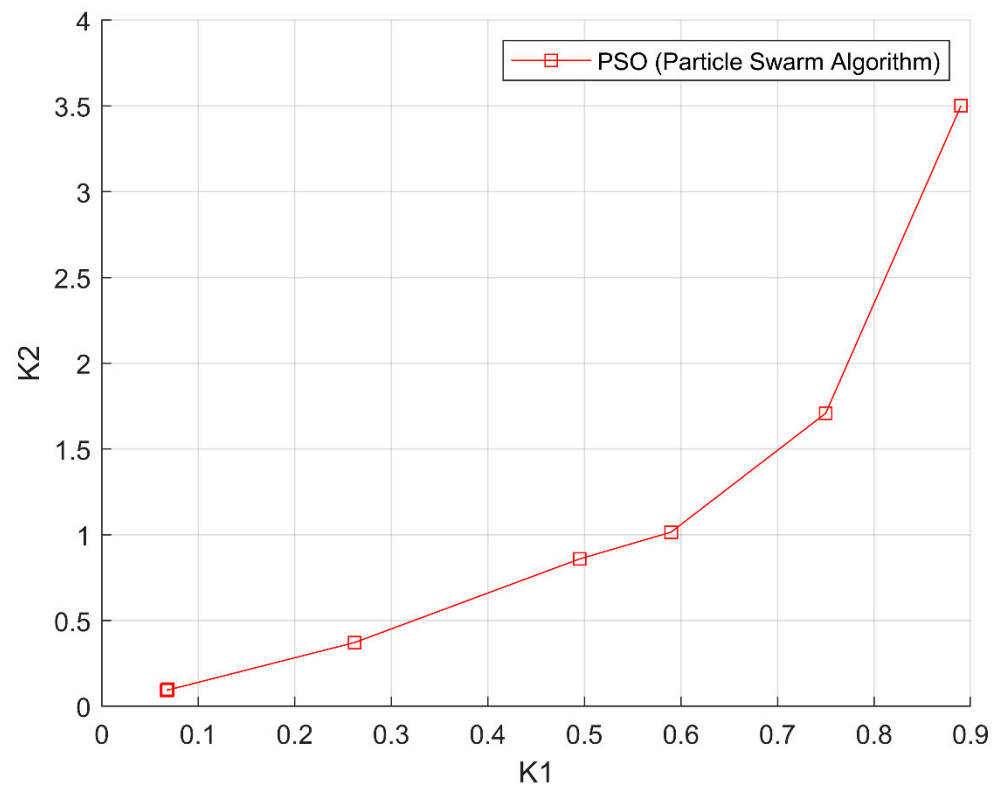| ξ	K_1	K_2	$\mu, \mathbf{min}^{-1}$
0.0	0.0673	0.0932	0.9979
0.1	0.0678	0.0935	0.9979
0.2	0.0678	0.0935	0.9979
0.3	0.0682	0.0946	0.9912
0.4	0.0685	0.0998	0.9785
0.5	0.0809	0.0940	0.9749
0.6	0.2620	0.3719	0.2468
0.7	0.4950	0.8591	0.1209
0.8	0.5896	1.0150	0.1002
0.9	0.7498	1.7080	0.0536
1.0	0.8900	3.5012	0.0105

Figure 3. Pareto-optimal set for solving a problem by a method using a particle swarm algorithm.

Table 3. Optimization results by the method using a backtracking search algorithm.

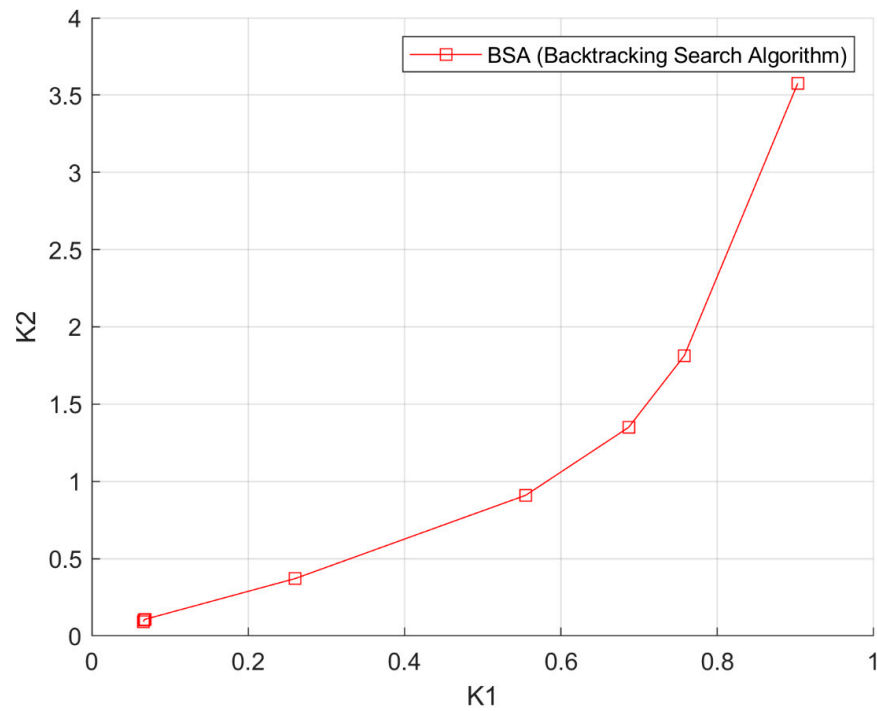| ξ	K_1	K_2	$\mu, \mathbf{min}^{-1}$
0.0	0.0663	0.0917	0.9988
0.1	0.0663	0.0917	0.9988
0.2	0.0663	0.0917	0.9988
0.3	0.0663	0.0917	0.9988
0.4	0.0668	0.1020	0.8975
0.5	0.0679	0.1070	0.8790
0.6	0.2600	0.3709	0.2468
0.7	0.5550	0.9103	0.1005
0.8	0.6866	1.3500	0.0677
0.9	0.7578	1.8140	0.0504
1.0	0.9030	3.5776	0.0255

Figure 4. Pareto-optimal set for solving a problem by a method using a backtracking search algorithm.

Consider an example of optimizing the performance of individual homework by one student in a distance learning form. We will leave the input parameters of the system the same as in the example with in-class training. In this task, only the task efficiency coefficient obtained using the neural network will change. As a result of calculations, the results of which are presented in Figure 5, the genetic algorithm showed the most optimal result.
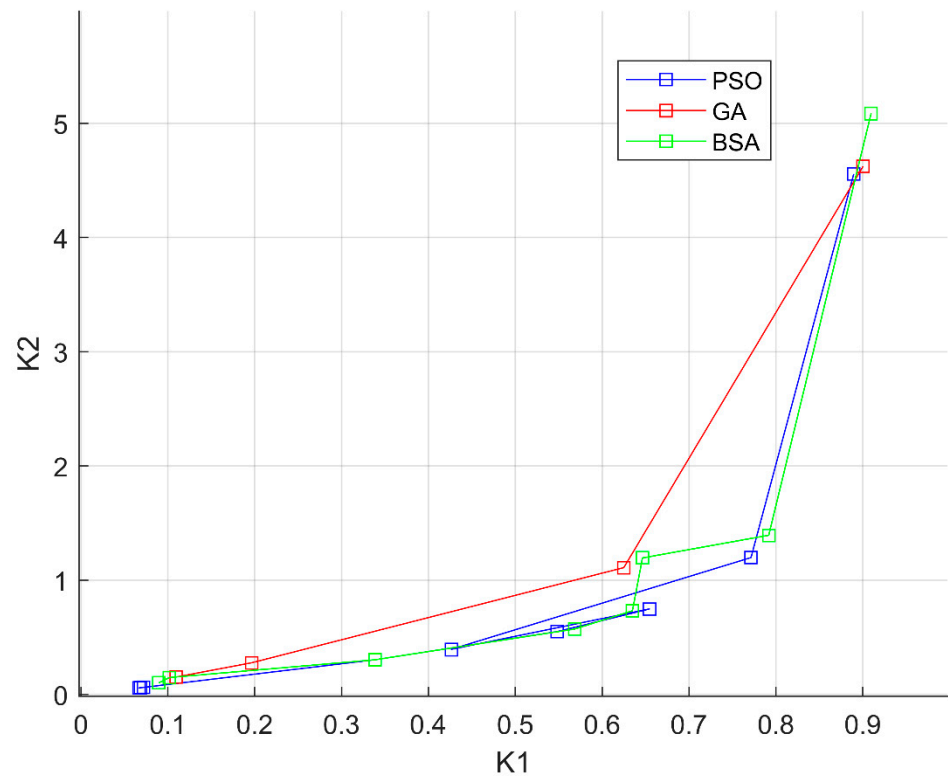

Figure 5. Pareto-optimal set for solving a problem by three methods.

As a result of the genetic algorithm, the following values of the optimality criteria were obtained: $\zeta = 0.9$, $K_1 = 0.7713$, $K_2 = 1.1979$, which corresponds to 21.32 min. The resulting value is 9.67% higher than the norms allotted for the time of homework by grade 6 students. Therefore, it is advisable to conclude that a complete transition to distance learning yields results that exceed the maximum permissible norms. The obtained solution was achieved while minimizing the time for completing homework, provided that maximum efficiency was obtained from the assignments performed.

A questionnaire on distance learning for schoolchildren was developed for feedback. The questionnaire included the following questions:

- Has the time you spend on your homework changed when you switched to distance learning?

  Options for reply: Increased, decreased, unchanged.

- If changed, select by how much.

  Options for reply: 10 min, 20 min, 30 min, 40 min, 50 min, 60 min, 70 min, 80 min, your own answer.

The survey was attended by 452 school students between the ages of 11 and 17. Of respondents, 3% indicated that time spent on homework decreased; 23%—not changed; 74%—increased. According to the results of the questionnaire, it was found that the time for completing homework increased by 18.3% on average, which is 23 min.

An experimental study of optimization methods is presented in Table 4. The test functions make it possible to check the quality of the search for extrema for functions with different topography of the search space. Two test functions: spherical $F_1(x) = \sum\limits_{i=1}^{n} x_i^2$ and Rosenbrock function $F_2(x) = \sum\limits_{i=1}^{n-1} (100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1))$ are unimodal; and function De Jong 2 $F_3(x) = \frac{-100}{100 \cdot (x_1^2 - x_2) + (1 - x_1)^2 + 1}$ and Rastrigin function $F_4(x) = 10 \cdot n + \sum\limits_{i=1}^{n} (x_i^2 - 10 \cdot \cos(2 \cdot \pi \cdot x_i))$ are complex multimodal. For all the functions, the dimension of the coordinate space is $n = 10$. Table 4 also shows the average values of these indicators. The best value on each line is shown in bold.

Table 4. Indicators of effectiveness of various methods.

Function	Indicators	PSO	BSA	GA
$F_1(x)$	*efficiency*	**100%**	**100%**	**100%**
	number of iterations	523.4	492.1	**483.1**
	solution time	0.311	0.707	**0.309**
$F_2(x)$	*efficiency*	**100%**	**100%**	**100%**
	number of iterations	591.2	527.7	**497.8**
	solution time	0.408	0.785	**0.3906**
$F_3(x)$	*efficiency*	90.1%	**100%**	**100%**
	number of iterations	754.5	692.7	**684.4**
	solution time	**0.634**	0.867	0.631
$F_4(x)$	*efficiency*	92.5%	**100%**	**100%**
	number of iterations	712.9	647.0	**640.0**
	solution time	0.612	0.823	**0.5906**
Average value	*efficiency*	95.7%	**100%**	**100%**
	number of iterations	645.4	589.8	**576.3**
	solution time	0.491	0.795	**0.481**

Of the algorithms of optimization methods proposed for consideration, the genetic algorithm showed the best result in terms of efficiency, solution time, and the number of iterations. Therefore, we can conclude that this algorithm is more suitable for the investigated problem of multicriteria optimization. However, the PSO and BSA algorithms showed quite good results when considering the problem of full-time education.

All the proposed algorithms, including the genetic one, have some limitations associated with the nature of the mathematical model of the system. In addition, to obtain the best convergence, it is necessary to know the range in which the optimized parameter should be presented, which is not always possible. However, the results obtained using the genetic algorithm are optimal in evaluating various forms of learning. In addition, the choice of this optimization method is associated with such features of the mathematical model of the problem as the dimension of the modeled system and the presence of discrete components in the vector of the optimized parameters.

4. Conclusions

The COVID-19 pandemic and its corresponding online learning have caused great difficulties for pupils in mastering school material. This manifested itself in the inability of schoolchildren to complete the tasks set out in the program at a given time. Having studied the reasons for this phenomenon, we identified 12 criteria that affect the effectiveness of solving school problems. Five of these criteria relate to the problem itself, and seven to the conditions under which it is solved. We used these criteria to build an MLP neural network that was trained by Resilient Propagation. The neural network shows a significant decline in the effectiveness of solving school problems in online learning. The process of finding the optimal time for students to complete their homework is based on a genetic algorithm, a backtracking search optimization (BSA) algorithm, and a particle swarm optimization (PSO) algorithm. The best results for the studied objective function were shown by a hybrid model that includes a genetic algorithm. We used queuing theory and a neural network to determine the matching function. This approach allows solving problems with a large system dimension and representing the optimized parameters in the form of a scalar and a vector. Based on the results obtained using the genetic algorithm, it was found that students spend 19.97% more time when solving school assignments in online learning conditions than in classroom learning. Allocation of extra time is not possible due to SanPin restrictions.

References

1. Sanitary Rules SP 2.4.3648-20. Sanitary and Epidemiological Requirements for the Organization of Education and Training, Recreation and Health Improvement of Children and Youth; Approved by the Chief State Sanitary Doctor Russian Federation Dated 28 September 2020 N 28. Available online: https://docs.cntd.ru/document/566085656 (accessed on 1 June 2021).
2. Wang, J. An International Comparative Study on Exercise Questions in Middle School Mathematics Textbooks. In *School Mathematics Textbooks in China. Comparative Studies and Beyond*; Wang, J., Ed.; East China Normal University Scientific Reports; World Scientific Publishing Co. Pte. Ltd.: Singapore, 2021; Chapter 10, Volume 2, pp. 347–373.
3. Mohseny, M.; Zamani, Z.; Basti, S.A.; Sohrabi, M.-R.; Najafi, A.; Tajdini, F. Exposure to Cyberbullying, Cybervictimization, and Related Factors Among Junior High School Students. *Iran. J. Psychiatry Behav. Sci.* **2020**, *14*, e99357. [CrossRef]
4. Masci, C.; Ieva, F.; Agasisti, T.; Paganoni, A.M. Evaluating class and school effects on the joint student achievements in different subjects: A bivariate semiparametric model with random coefficients. *Comput. Stat.* **2021**, *11*, 1–41. [CrossRef]

5. Febriana, M.D.; Zainuddin, Z.; Nurtanio, I. School zoning system using K-Means algorithm for high school students in Makassar City. In Proceedings of the International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) 2019, Yogyakarta, Indonesia, 5–6 December 2019; pp. 368–372. [CrossRef]

6. Marcenaro-Gutierrez, O.D.; Gonzalez-Gallardo, S.; Luque, M. Evaluating the potential trade-off between students' satisfaction and school performance using evolutionary multiobjective optimization. *RAIRO Oper. Res.* **2021**, *55*, S1051–S1067. [CrossRef]

7. Shehab, T.; Haghighat, R.; Sajjan, K.K.; Balali., V. Prioritization of K-12 school maintenance construction projects using genetic algorithm and dynamic programming models. *J. Inf. Technol. Constr.* **2021**, *26*, 112–127. [CrossRef]

8. Shukhman, A.E.; Bolodurina, I.P.; Polezhaev, P.N.; Ushakov, Y.A.; Legashev, L.V. Adaptive technology to support talented secondary school students with the educational IT infrastructure. In Proceedings of the IEEE Global Engineering Education Conference (EDUCON), Santa Cruz de Tenerife, Spain, 18–20 April 2018; pp. 993–998. [CrossRef]

9. Petrosov, D.A.; Lomazov, V.A.; Petrosova, N.V. Model of an Artificial Neural Network for Solving the Problem of Controlling a Genetic Algorithm Using the Mathematical Apparatus of the Theory of Petri Nets. *Appl. Sci.* **2021**, *11*, 3899. [CrossRef]

10. Khasanshin, I. Application of an artificial neural network to automate the measurement of kinematic characteristics of punches in boxing. *Appl. Sci.* **2021**, *11*, 1223. [CrossRef]

11. Ivanyuk, V.A.; Pashchenko, F.F. Neural networks and their application in forecasting problems. *J. Phys. Conf.* **2020**, *1703*, 012033. [CrossRef]

12. Makarov, M.A.; Obidin, V.V.; Osipov, A.V.; Filimonov, A.V. Neural Excel VBA Extension Pack: Certificate of State Registration of Computer Programs. Available online: https://new.fips.ru/registers-doc-view/fips_servlet?DB=EVM&DocNumber=2019617073&TypeFile=html (accessed on 1 June 2021).

13. Dogadina, E.P.; Smirnov, M.V.; Osipov, A.V.; Suvorov, S.V. Formation of the Optimal Load of High School Students Using a Genetic Algorithm and a Neural Network. *Appl. Sci.* **2021**, *11*, 5263. [CrossRef]

14. Amjad, M.K.; Butt, S.I.; Kousar, R.; Ahmad, R.; Agha, M.H.; Faping, Z.; Anjum, N.; Asgher, U. Recent Research Trends in Genetic Algorithm Based Flexible Job Shop Scheduling Problems. *Math. Probl. Eng.* **2018**, *2018*, 9270802. [CrossRef]

15. Viana, M.S.; Junior, O.M.; Contreras, R.C. A Modified Genetic Algorithm with Local Search Strategies and Multi-Crossover Operator for Job Shop Scheduling Problem. *Sensors* **2020**, *20*, 5440. [CrossRef] [PubMed]

16. Rarità, L.; Stamova, I.; Tomasiello, S. Numerical schemes and genetic algorithms for the optimal control of a continuous model of supply chains. *Appl. Math. Comput.* **2021**, *388*, 125464. [CrossRef]

17. Hu, Y.; Li, J.; Hong, M.; Liu, M.; Man, Y. Short term electric load forecasting model and its verification for process industrial enterprises based on hybrid GA-PSO-BPNN algorithm—A case study of papermaking process. *Energy* **2019**, *170*, 1215–1227. [CrossRef]

18. Sedak, M.; Rosić, B. Multi-Objective Optimization of Planetary Gearbox with Adaptive Hybrid Particle Swarm Differential Evolution Algorithm. *Appl. Sci.* **2021**, *11*, 1107. [CrossRef]

19. Wang, L.; Peng, L.; Wang, S.; Liu, S. Advanced backtracking search optimization algorithm for a new joint replenishment problem under trade credit with grouping constraint. *Appl. Soft Comput.* **2020**, *86*, 105953. [CrossRef]

20. Park, K.; Shin, D.; Chi, S. Variable Chromosome Genetic Algorithm for Structure Learning in Neural Networks to Imitate Human Brain. *Appl. Sci.* **2019**, *9*, 3176. [CrossRef]

21. Yerznkyan, B.; Bychkova, S.; Gataullin, T.; Gataullin, S. The sufficiency principle as the ideas quintessence of the club of Rome. *Montenegrin J. Econ.* **2019**, *15*, 21–29. [CrossRef]

22. Korchagin, S.; Serdechny, D.; Kim, R.; Terin, D.; Bey, M. The use of machine learning methods in the diagnosis of diseases of crops. *E3S Web Conf. EPD Sci.* **2020**, *176*, 04011. [CrossRef]

23. Dogadina, E.P.; Konoplev, A.N.; Belov, A.A.; Kropotov, Y.A.; Proskuryakov, A.Y. Automated Control Systems for Industrial Production. Available online: https://www.directmedia.ru/book_454164_avtomatizirovannyie_sistemyi_upravleniya_promyishlennyimi_proizvodstvami/ (accessed on 15 July 2021).