Research Article

Improved Genetic Algorithm to Solve the Scheduling Problem of College English Courses

Jing Xu [1,2]

[1]*International Education School, Chifeng University, Chifeng 024000, China*
[2]*Phyllis J. Washington College of Education, The University of Montana, Missoula, MT 59812, USA*

Correspondence should be addressed to Jing Xu; jing1.xu@umontana.edu

In this paper, an improved genetic algorithm is designed to solve the above multiobjective optimization problem for the scheduling problem of college English courses. Firstly, a variable-length decimal coding scheme satisfying the same course that can be scheduled at different times, different classrooms, and different teaching weeks per week is proposed, which fully considers the flexibility of classrooms and time arrangements of the course and makes the scheduling problem more reasonable. Secondly, a problem-specific local search operator is designed to accelerate the convergence speed of the algorithm. Finally, under the framework of optimal individual retention, the selection operator, crossover operator, and variation operator are improved. It is experimentally demonstrated that the designed algorithm not only has a faster convergence speed but also improves the diversity of individuals to a certain extent to enhance the search space and jump out of the local optimum. Research shows that the improved genetic algorithm has improved average fitness value and time compared with traditional genetic algorithm. At the same time, the use of the largest fuzzy pattern algorithm effectively solves the conflict problem of college English lesson scheduling, thereby improving the solution of college English lesson scheduling. Through the research of this article, the management system of college English course scheduling has been made more intelligent, and the rational allocation of teaching resources and the completion of education and teaching plans have been improved.

1. Introduction

With the rapid development of science and technology, computers have become the backbone of various work applications with their excellent performance such as strong processing power and fast computing speed, and they have also become the main force in intelligent education and teaching work, especially in the field of scheduling lessons which has been widely used in recent years [1]. The use of computers for rational planning and scheduling of different courses enables quickly meeting different constraints and thus obtaining feasible results. The goal of the scheduling problem is to first aggregate all the courses offered and then to finally rationalize each teaching task, i.e., course, according to the current semester's teaching schedule, as well as school resources and teachers, to be able to optimize school resources, teachers to teach rationally, and students to learn efficiently. In the genetic algorithm, two genetic operations, crossover and variation, directly affect the overall effectiveness of the genetic algorithm. The genetic crossover and variation operators themselves are adaptive, and the individuals in the population evolve iteratively through the genetic operations, with changes in their crossover and variation rate values determined by the degree of population dispersion or concentration, as well as the size of the population [2]. The amount of population size increases as the crossover rate value becomes larger, but it also leads to an increase in the chance that the more outstanding individuals in the population are destroyed; the size of the variation rate value directly affects the number of newborn populations in the population, and the larger the value of the variation probability, the larger the value of the size of the newborn population, and the greater the possibility of the algorithm jumping out of local convergence to obtain the optimal solution [3].

The complexity of arranging classes increases exponentially with the increase in the school's teaching scale, which is not in direct proportion to the school's teaching scale. The essential problem of scheduling is a timetable problem, and the timetable problem is an important branch of operations research [4]. Through the analysis of the problem of scheduling, a mathematical model based on the scheduling is established to discuss the solution and the existence of the solution, but the desired result has not been achieved. With the development of science and technology, it has become possible to apply heuristic algorithms to the scheduling system. Heuristic algorithms include graph theory, greedy algorithms, simulated annealing algorithms, backtracking algorithms, etc. Heuristic algorithms can solve the problem of scheduling well [5], making the study of scheduling problems improve. With the continuous deepening of education reform, the number of students has continued to increase, the constraints have continued to increase, teaching resources are obviously insufficient, and the application of traditional algorithms is not enough to solve the complex problem of scheduling. Through the school's teaching plan analysis and establishment of a mathematical model of arranging courses, the corresponding function is designed according to the soft constraints in the constraint conditions to establish the final fitness function [6]. At the same time, the traditional genetic algorithm is improved, and an improved adaptive genetic algorithm is designed to solve the problem of scheduling. In addition, the maximum fuzzy pattern algorithm is used to solve the problem of scheduling conflict. Experiments show that, compared with the traditional genetic algorithm, the improved adaptive genetic algorithm has a great improvement in the efficiency of scheduling, and the use of the maximum fuzzy mode algorithm can solve the problem of scheduling conflicts.

The significance of this paper is to design a scheduling algorithm based on improving the existing artificial intelligence algorithm and for the scheduling problem, which applies to the current scheduling principles and can not only make reasonable arrangements for large-scale scheduling data but also improve the efficiency of the scheduling staff and reduce the redundant workload. Therefore, the use of computer scheduling has the advantages of saving time and labor and high quality and effectively reduces the tedious scheduling tasks, which plays a crucial role in the information and intelligent construction and development of universities. In summary, there has been great progress in solving the scheduling problem based on genetic algorithm, but there are still the following improvements: researchers usually consider fewer solution objectives for the scheduling problem, and the evaluation of class schedules in the actual teaching process is multifaceted, so a perfect solution objective should be established for the scheduling problem; because the genetic algorithm is random, the conflict of constraints arises from the scheduling. Because of the stochastic nature of genetic algorithms, the conflict of constraints arising from scheduling should be detected in time, and the constraints are interrelated, so the conflict characteristics should be fully considered and solved; many

researchers study the scheduling problem only based on the single schedule of the course but fail to relate the scheduling problem to the scheduling problem in actual teaching, such as the scheduling of different times and classrooms each week for courses with different weekly credit hours and the arrangement of the beginning and end weeks of teaching. More researchers in the genetic algorithm have made certain improvements to the coding design, initialized population scheme, and genetic operations, but no scholars have been able to propose a local search algorithm for the scheduling problem to improve the convergence speed and spatial search ability of the genetic algorithm.

2. Related Work

Deng et al. studied and analysed the causes of learning fatigue among college students and concluded that learning is an energetic process of physical and mental integration, and learning fatigue is an inevitable phenomenon, which is related to the psychological quality of students and is also the result of the influence of many external environments; for example, the school should arrange the curriculum in such a way that the courses of different natures are arranged in a time parity [7]. Barkaoui proposed that although scheduling involves various aspects, four main levels play a dominant role, i.e., curriculum level, teacher level, student level, and classroom level, and that scheduling should be carried out by adhering to the principles of scheduling that are conducive to improving students' learning efficiency [8]. Gabi proposed to conform to the principles of psychology, pedagogy, and health, based on students' psychological and physiological factors (biological clock, brain fatigue, study time, and preschool time) and teacher factors (teaching time interval, reasonable arrangement requirements, etc.), to help improve student learning and optimize teachers' teaching effectiveness in the classroom [9]. Lin proposed that the scheduling process should use system thinking, by understanding the school, teachers, students, and other related situations, to carry out comprehensive and balanced scheduling, to achieve "people-oriented", "focus on efficiency", "difficult to stagger", "efficiency-oriented", and balanced distribution [10]. On the premise of the complexity of class scheduling in colleges and universities, Nourmohammadi-Khiarak proposed the scheduling model of "teaching parallel classes" for public basic courses and analysed the key factors affecting class scheduling in colleges and universities, such as the instability of full-time and external faculty, the lack of students' awareness of course selection, and the tight classroom resources [11].

These systems have long response times and even deadlocks when the amount of input and the constraints developed to reach a certain amount, so these scheduling systems mentioned above are less adaptable, and the usual situation is to first perform automatic scheduling and then make manual adjustments to the unreasonable parts of the results, and the manual adjustment is not less than the workload of rearranging all the courses. So according to this view, these scheduling systems do not thoroughly help the scheduling staff if the amount of scheduling data is very large

[12]. As mentioned earlier, many researchers have already laid the research foundation for the scheduling problem, and through their extensive research on the scheduling problem, several heuristic algorithms have been used to solve the scheduling problem in recent years [13]. The mainstream algorithms for solving the scheduling problem are simulated annealing algorithm, expert system, ant colony algorithm, etc. More researchers have applied genetic algorithms to the scheduling problem, and many researchers have made some improvements to genetic algorithms for the characteristics of the scheduling problem. In Vannucci, in the scheduling problem based on a genetic algorithm, the analysis and design of the fitness function are carried out from five aspects, such as course session time slot superiority and course day combination degree [14].

The research objective of this paper is to propose an optimal solution applicable to the scheduling problem based on the principles of genetic algorithms. Firstly, by representing the experience of faculty schedulers in the scheduling problem and the principles of scheduling that exist in teaching work, the solution objective of scheduling is expressed using a mathematical model, and in the genetic operation, the coding scheme, initialized population scheme, scheduling conflict detection, and elimination, genetic operation and local search operator are designed. The algorithm is simulated to demonstrate the practicality and effectiveness of the improved algorithm. Based on the characteristics of the scheduling problem, the standard genetic algorithm is improved to design the coding scheme to meet the actual scheduling requirements, the initialization population scheme to meet the course day interval arrangement, the genetic operator for the optimal individual retention strategy, and the local search operator to accelerate the convergence speed and improve the searchability.

3. Analysis of Improved Genetic Algorithm for Automatic College English Scheduling

3.1. Improved Genetic Algorithm Scheduling Design. Genetic Algorithms (GA) is a computer programming algorithm based on the theory of biogenetics, which is a kind of search algorithm with a high degree of randomness. In the biological world, the best individuals are selected according to the survival of the fittest, and multiple individuals are combined to become a population, and through continuous natural selection, new individuals are evolved that can adapt to the environment [15]. When using genetic algorithms, researchers do not need to analyse and study in detail all the characteristics of the problem to be solved but only use the continuous "natural inheritance" to select the most suitable solution to various complex data structure problems, first genetic encoding, in the specific implementation process of the code to manipulate. The search is carried out according to the laws of natural selection. The main feature of the genetic algorithm is that it does not need to specify that the function must have continuity or rely on gradient information but rather encodes the model and parameters of the specific problem, such as transforming the problem into structural objects such as sequences, matrices, and chains. It

also does not need to follow specific rules but to guide the search based on certain probabilistic change conditions. Although it has certain randomness and blindness, it has a better global optimization capability by using a reasonable strategy.

The genetic algorithm is represented by a specified encoding that corresponds to the state space of the problem and the encoding space of the genetic algorithm, which depends to a large extent on the conditions of the problem itself, and the design of the encoding scheme, which will affect the genetic operation. This is because the optimization process of the genetic algorithm is not applied directly to the problem parameters themselves but is performed on the coding space corresponding to a certain coding scheme, and therefore, the design of the coding scheme is one of the most important factors affecting the performance of the algorithm and the efficiency of the algorithm. Different encoding lengths and binary encoding have a significant impact on the accuracy and efficiency of problem-solving [16]. Binary encoding represents the solution of the problem as a binary string and decimal encoding represents the solution of the problem as a decimal string; obviously, the encoding length will affect the accuracy of the algorithm and the algorithm will occupy large storage space. The real number encoding represents the solution of the problem as a real number, which solves the impact of encoding on the accuracy and storage space of the algorithm and helps to optimize the introduction of relevant information in the actual problem, which is widely used in high-dimensional complex optimization problems.

Genetic algorithms perform genetic operations from multiple given solutions to generate new solutions when searching for the optimal solution of a real problem, and the set of all solutions is called the initial population, which is a subset of the solution space of the problem to be solved. The quality of the initial population has a great impact on the evolutionary efficiency of the genetic algorithm, and a high-quality initial population is a strong guarantee that the genetic algorithm will eventually obtain a good result [17]. The genetic algorithm will perform genetic operations on the solution set, such as selection, crossover, mutation, etc. In each generation of inheritance, the set of selected solutions again forms a new population, and the next generation of the population is then generated by the continued reproduction of the new population, as shown in Figure 1.

Enhancing classroom seat utilization rate and course scheduling results in resource savings, and a low classroom seat utilization rate can lead to a decrease in teaching effectiveness. Then, the classroom seat utilization rate can be defined as

$$f_{\text{seat}}(i) = \frac{2C_{o_i}\text{student\_num}}{C_{o_i}\text{Roomseat\_num} + C_{o_i}\text{Roomseat\_num}},$$

$$f_1(s) = \frac{\sum_i^{\text{Co max}} f_{\text{seat}}(i)}{\text{Co max} + 1},$$

$$(1)$$

where $f_{\text{seat}}(i)$ is the classroom seat utilization rate of the i-th teaching task, $C_{o_i}\text{student\_num}$ is the number of students in
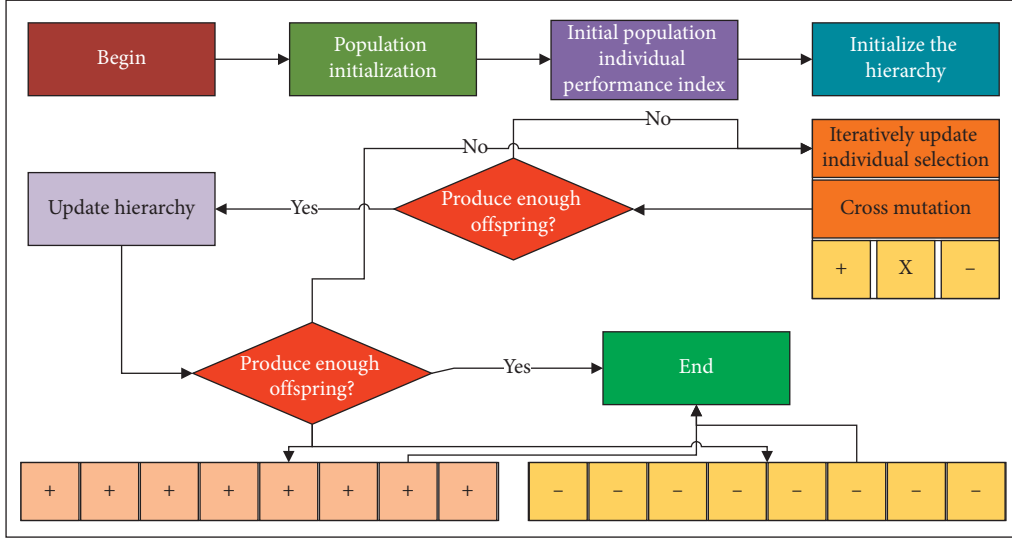
Figure 1: Improved genetic algorithm.

the i-th teaching task, C_{o_i} Roomseat_num is the number of seats in the classroom used for the i-th teaching task, $f_1(s)$ is the average classroom seat utilization rate of all teaching tasks, the larger $f_1(s)$, the higher the classroom, and the larger the $f_1(s)$, the higher the classroom seat utilization rate.

Teachers should schedule a maximum of six lessons a day. For teachers not to compromise their physical and mental health while achieving optimal teaching effectiveness, they should have no more than six lessons per day. The daily distribution of courses taught by teachers is calculated with the following objectives:

$$f_{\text{teach}}(i) \frac{\sum_{j=1}^{\text{Wen max}} \sum_{k=}^{5} f_{\text{tea\_week\_day}}(i, j, k)}{\text{Wen max} \times 5},$$
$$f_3(s) = \frac{\sum_{i}^{\text{To max}} f_{\text{teach}}(i)}{\text{Te max}}, \quad (2)$$

where the $f_{\text{tea\_week\_day}}(i, j, k)$ value indicates whether the number of daily lessons is satisfied with no more than four lessons per day, $f_{\text{teach}}(i)$ indicates the average number of lessons taught by a particular teacher during a workday, and $f_3(s)$ indicates the daily average number of lessons taught by each teacher during a workday. Larger values of $f_3(s)$ indicate the number of workdays in which a teacher meets the number of daily lessons with no more than four lessons. For classes, the daily lessons are evenly scheduled. The number of daily lessons for students should be in a relatively balanced state, trying to avoid a situation where there are too many lessons on one teaching day and too few, too light, or even no lessons on some days. For a week, the variance of the number of hours per day should be as small as possible. Define the average number of hours per day for week j of class i as

$$\overline{x_{ij}} = \frac{1}{5} \sum_{k=1}^{5} (x_{i,jk}). \quad (3)$$

Find the variance for the number of hours per day for a given week.

$$D(x_{ij}) = \sum_{k=1}^{5} \left(x_{i,jk} + \frac{1}{5} \sum_{k=1}^{5} (x_{i,jk}) \right). \quad (4)$$

The average daily course merit for class i for week j is

$$f_{\text{tea\_week\_day}}(i, j) = \begin{cases} 1, & 10 < D(x_{ij}) < 20, \\ 0.5, & D(x_{ij}) \geq 20 \\ 0, & \text{others}. \end{cases} \quad (5)$$

Then, the average number of daily class hours per week for all classes is superior.

$$f_7(s) = \frac{\sum_{j=1}^{\text{Wen max}} \sum_{k=}^{5} f_{\text{tea\_week\_day}}(i, j, k)}{\text{Wen max} \times \text{Te max}}. \quad (6)$$

The crossover probability is adaptively adjusted to recombine some of the structures encoded by the two parental genes to form new individuals. For example, after the selection operation is completed, everyone can randomly pair up two by two to generate five crossover pairs, at which point 10 chromosomes are available, and if two rows of parent chromosomes are randomly swapped, two new offspring are immediately created. In general, the searchability of the genetic algorithm is directly related to the crossover variation operation, and the improved adaptive crossover operation satisfies the condition that the adaptive crossover probability varies with the size of the fitness value of the individuals in the population, increasing when the average fitness value is larger than the fitness value of the individuals; when the average fitness value is smaller than the fitness value of the individuals, the adaptive crossover probability will become smaller. This method of adaptively adjusting the crossover probability improves the ability of the algorithm to search globally during the evolution of the previous population, avoiding the phenomenon of early convergence of

local search and saving the previous good individuals to the next generation.

$$Ci = \frac{1}{\lim_{n \longrightarrow \infty} \sqrt{\sum_{d=1}^{n} (Cs + Ct)^2}}, \tag{7}$$

The adaptive adjustment of the variation probability P_m uses the variation of two numbers generated by a random function, such as the variation of the time and teacher code in the schedule of a class. The performance of the genetic algorithm is affected by crossover and variation, and the magnitude of the adaptive variation probability is not a fixed value but varies with the crossover probability. When the crossover probability value is small, the adaptive variation probability increases, while when the crossover probability value increases, the adaptive variation probability value then decreases. These two operations, adaptive crossover and variation, coordinate with each other to ensure the global search capability of the genetic algorithm to obtain the global optimal solution, as shown in Figure 2.

In the basic genetic algorithm, the initial population is generated by the random search, and the initial solution is derived by a genetic operation, which is not very effective in terms of individual fitness. The size of the initial population has a certain impact on the efficiency of the genetic algorithm. If the initial population is too large, it will reduce the efficiency of the algorithm and increase the overall computation time of the large algorithm execution; on the contrary, if the initial population is too small, it will reduce the diversity of the population, the sample capacity will be reduced, and the overall performance of the algorithm will be poor, which will easily make the whole algorithm end prematurely and the phenomenon of premature convergence will occur. In the process of initializing the population, the value of the resulting individual fitness, if too small, will cause all solution spaces to shrink accordingly, and the result obtained will not be the global optimum, but a locally optimal solution.

The academic scheduler schedules the first class in the morning on Tuesday and Thursday for the weekly class number 2 because the academic scheduler's analysis shows that scheduling classes in the morning can help improve students' learning efficiency, but this is only a consideration in terms of time. The above simulated manual scheduling shows that although the faculty scheduler has made certain considerations and is more scientific and reasonable, the scheduling results can be optimized, and many soft constraints affect the scheduling results. After simulating the thinking process of manual scheduling, it can be found that the mathematical model of the scheduling problem is like combinatorial programming. The traditional approach to solving combinatorial planning problems relies on constraints that are sufficient and necessary for the optimal combination of solutions. It is theoretically feasible, but in practical application, the combinatorial planning problem becomes very complex as the number of combinatorial solutions increases dramatically with the increase of various factors.
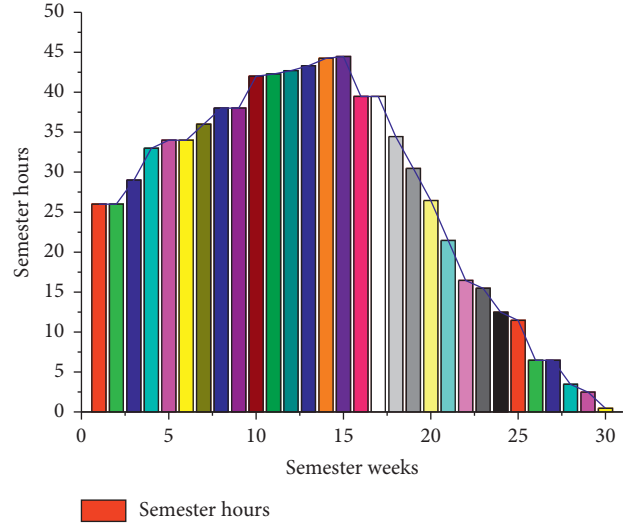

FIGURE 2: The weekly distribution of class hours is left-skewed.

3.2. Experimental Analysis of English Scheduling City in High Schools. The scheduling process begins with the teaching plan; i.e., the academic affairs department concerned with the scheduling process collects information on classes offered by each teaching unit for the following semester before the end of each semester. This information mainly includes classes (in this case, teaching classes), class size, course subjects, weekly credit hours, total credit hours, course nature, instructors, classroom type, and prerequisite course marks. After the academic affairs department summarizes the information of the classes, they will issue the teaching tasks to each teaching unit and make uniform arrangements according to the teaching resources of the school. This Figure shows how many classes are offered and which classes are taught by each teacher. For teachers, not only does the teacher's name have to be clarified, but also the teacher's number should be set to prevent ambiguity of the same name. For example, some teachers are pregnant or physically unfit, so they should not be assigned to high-floor classes, which should be marked with a special marker: 0 for no special circumstances and 1 for pregnant women or physically unfit.

For classes, two concepts need to be clear, namely, administrative classes and teaching classes [18–20]. Administrative classes, as the basic unit of the school, will be managed by the counsellors of each administrative class for the daily management of students; teaching classes are administrative classes in which the same curriculum needs are formed into a collective class, and the teachers of the classes are mainly responsible for the teaching and management of students. A teaching class contains at least one administrative class, and an administrative class can belong to different teaching classes, as shown in Figure 3.

The primary consideration of genetic algorithms is how to abstract the actual problem for coding, which in the case of the scheduling problem is to encode the teaching task. Binary encoding and Gray encoding are commonly used, but for the scheduling problem, the encoding is intuitive and
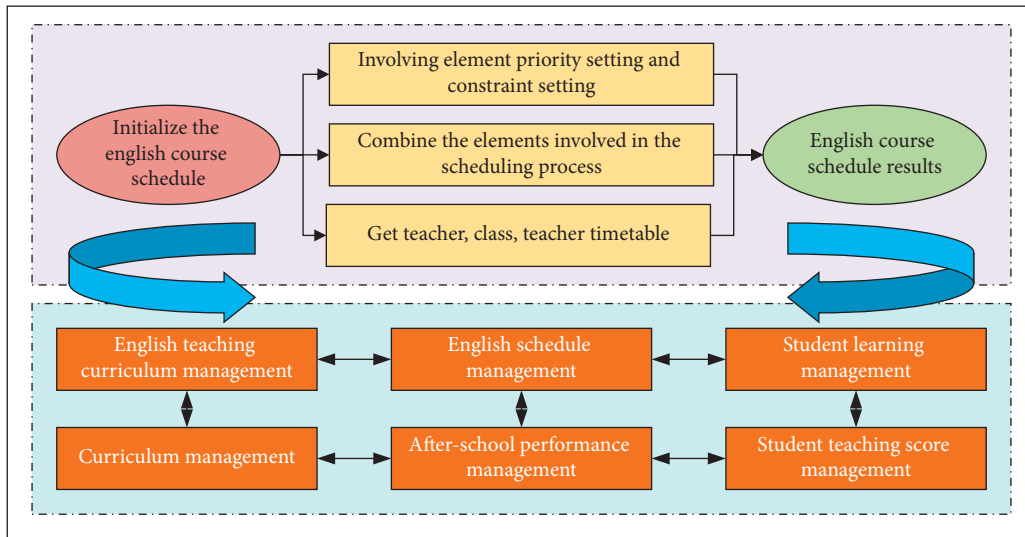
FIGURE 3: Experimental flow.

easy to understand. For the scheduling problem, the decimal variable-length coding scheme is used in this paper. Although, in actual teaching, there are cases that a class is taught by more than one teacher, and one teacher takes more than one course at the same time, for an administrative class or a teaching class, the courses that need to be scheduled and the teachers who teach them are decided before the start of the class, which is reflected in the teaching schedule. Therefore, the course, teacher, and class can be set as the same variable and correspond to the classroom and teaching time. The classroom and related information (classroom number, number of seats, classroom type, etc.) can be reflected by the classroom number in the classroom information table; the teaching time is composed of the teaching day, the starting week, and the ending week of the course.

The network required for the automatic class scheduling system in higher education is relatively simple, and only two servers are required, i.e., the database server and the application server, to support the scheduling function, as shown in Figure 4. Due to the different user rights, only the academic administrators have access rights to the database server, and students and teachers need to access the data information through the application server when they want to get the data information. Due to the development of the Internet and the continued popularity of mobile devices, when designing the network of automatic class scheduling system in colleges and universities, it is also necessary to consider that teachers and students use mobile devices (including cell phones, tablets, etc.) to log into the system for operations such as viewing class schedules. To protect the security of applications and data, it is also necessary to use security devices such as firewalls for protection, and teachers and students can access the college automatic class scheduling system through the Internet.

The system starts with user login, reads user information after successful login, records the number of failures if the login is unsuccessful, and automatically logs out of the system if the login fails more than 3 times in a row [21–23]. If the user logs in successfully, the user information is read and combined with the permission information obtained from the user to determine whether the user is logging in as a normal user or as a system administrator. If the user is judged to be a normal user, then the user has the right to modify personal information or to query and export the scheduling results according to their needs. If you are a system administrator, you can perform all functions. In the automatic class scheduling system built in this paper, the system administrator is mainly responsible for two aspects of management, i.e., the authority to perform management of class scheduling and the ability to perform management of class schedules.

The account written by the administrator is queried and tested to determine whether the account already exists in the database. If there is a saved record, the system will proceed to the next step and wait for the user to write the user's name and login password; if the account does not exist, the system will act as creating a new administrator user for the logged-in user and then wait for the user to write the user's name and login password after completion. After the login user writes the user's name and password, it is necessary to check the input content to determine whether the information entered is correct. If the information entered by the user is incorrect, the system has only three opportunities to provide, and if there is incorrect information entered three times in a row, the system will perform an automatic exit operation and return to the original system login screen again.

When the information entered by the system administrator is correct, it will enter the system to perform the relevant function and request to start the work of scheduling management. The specific management process is manifested in that the Academic Affairs Office will first issue the teaching assignment for the current academic year to clarify the teaching content and task objectives. Since the number of students, classrooms, teachers' resources, and training programs is different for different majors and grades, the teaching assignment is sent to each faculty on this basis to set
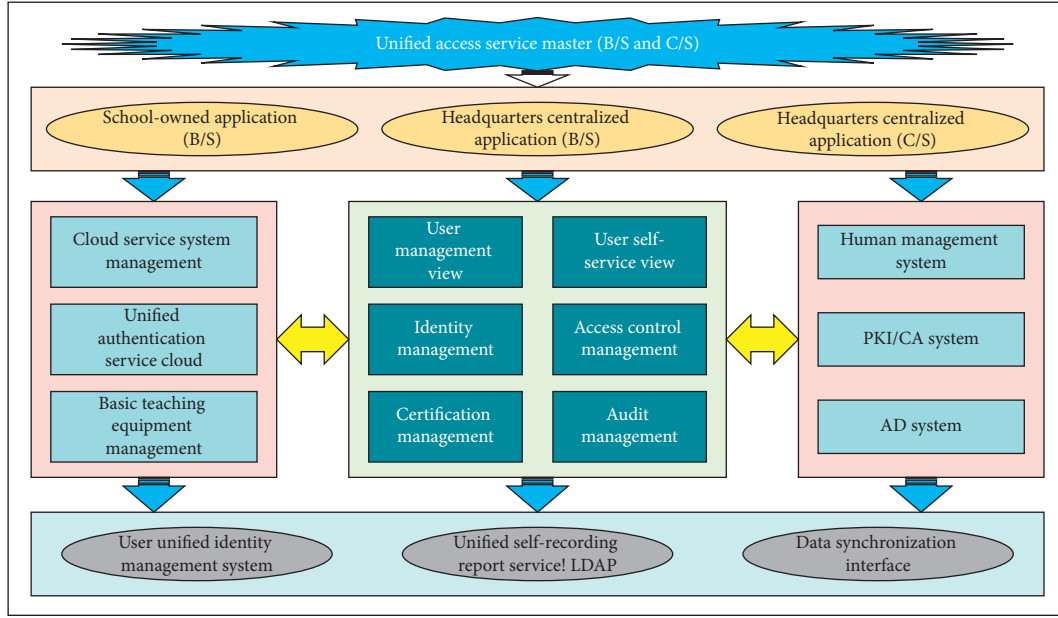
FIGURE 4: Network architecture.

the specific teaching requirements for each semester before the teaching tasks are issued [24–26]. The teaching assignments here will be reasonably adjusted before the start of each semester according to national policies, school situation, graduation situation, etc. Each college arranges the corresponding teachers according to the appropriately adjusted teaching assignment according to the specific actual training program and teaching needs, etc., and puts forward a series of requirements for the teaching tasks, including whether experimental courses need to be arranged, whether there are special requirements for teaching venues, etc.

4. Results and Discussion

4.1. Improved Algorithm Performance Results. For the search technology for overall optimization problems such as the scheduling and arrangement of the curriculum, genetic algorithm is a kind of evolutionary algorithm, but evolutionary algorithm and its branches are heuristic calculations and random search methods, so genetic algorithm is also heuristic; it calculates and searches methods immediately. In practical applications, various optimization scheduling problems with high computational complexity have appeared. The objective function may be noncontinuous, nondifferentiable, nonconvex, multipeak, and environmental noise. Under the class form, this kind of complex scheduling optimization problem is not suitable for the use of general analytical techniques, and if it is solved by the traditional search calculation method, it will also encounter many obstacles.

The local search operator designed in this paper aims to accelerate the convergence speed and enhance the local searchability. Meanwhile, this paper adopts the optimal individual retention strategy, and if the initial population can have a higher value of fitness function after the local

algorithm, then it can be considered to have better results in the later genetics. For the local search operator performance analysis experiments are done only after initializing the population, conflict detection, conflict removal, and local search operator operations, with no genetic operator (selection, crossover, and mutation) operations; this is because currently we only want to consider the effect of the use of local search operator, for this initialization of 200 chromosomes for the experiment. Figure 5 shows the fitness values for the initialized populations with and without the local search operator, with 100 chromosomes in each population.

As can be seen from Figure 5, the use of the local search operator does not guarantee that the fitness of each chromosome is higher than that of the initialization without the local search operator because there is a certain amount of randomness in the initialized population. The average fitness values of many chromosomes were analysed, and the fitness values of chromosomes in the initialized population using the local search operator were slightly higher than those in the initialized population.

From the observation and analysis in Figure 6, it can be seen that the standard genetic algorithm reaches convergence at about 600 generations; in the literature, it enters convergence at about 800 generations, and although the convergence speed is slower, the value of the fitness function is much higher than that of the standard genetic algorithm; the improved genetic algorithm designed in this paper converges at about 350 generations, and it has the characteristics of fast convergence and strong global search ability. The improved genetic algorithm designed in this paper converges at about 350 generations and has fast convergence and strong global search capability, which is due to the local search operator that optimizes the local search for some solution objectives in the specific problem of scheduling. For
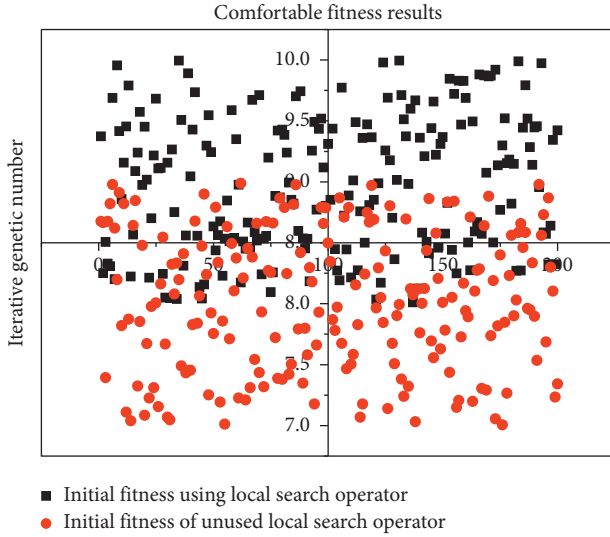
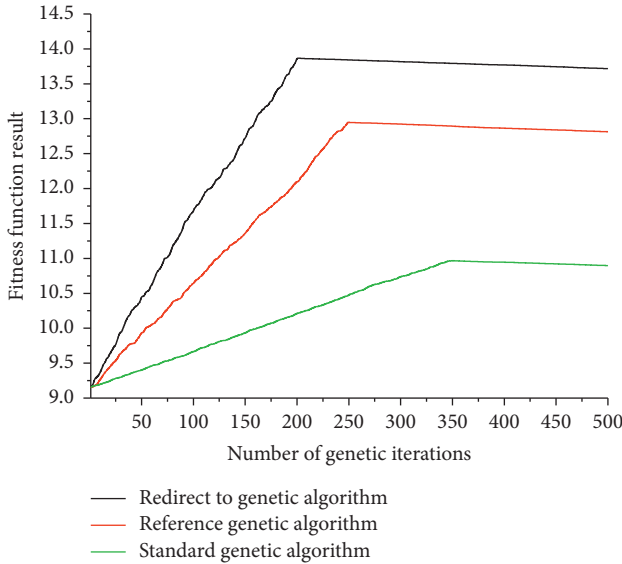Figure 5: Comparison of initialized chromosome adaptations.

Figure 6: Plot of the values of the fitness function for the three genetic algorithms.

the performance of the algorithm, there are two indicators: global search capability and local search capability. The global search capability refers to the ability to find the optimal solution in the global context, while the local search capability refers to the ability to analyse a specific problem from a local context and come up with a better solution algorithm based on the existing solution. In practical problems, the result value often has many extreme points in the set of all search spaces, and it is easy to fall into local extreme points, in which the local optimal situation is described above, for which the local search algorithm designed in this paper is dependent on the solution space for search.

This is due to the improvement of the genetic operator and the design of local search in this algorithm. In terms of the fitness function value of "easy for counsellors to listen to lectures", the algorithm in this paper is slightly lower than

the other two algorithms, which is because college counsellors usually do not have the same major or even the same grade, so the objective function of "easy for counsellors to listen to lectures" is probably different from the objective functions of "classroom seat utilization rate," " staggered course nature," "English class scheduling time," etc. The objective function of "easy for counsellors to listen to lectures" may be contrary to the objective functions of "classroom seat utilization rate," "staggered course nature" and "English class scheduling time," etc. To increase the value of the objective function of "easy for tutors to attend lectures," the weight value of the function can be increased, but it may also cause the value of other objective functions to decrease.

In the analysis of the scheduling results from the teacher's perspective, both algorithms are very close to each other in terms of the availability of teachers for one day of the teaching week. However, for the teacher's daily class hours that cannot exceed 6 hours, one of the classes in the literature has 8 hours, which is slightly better than the algorithm in this paper. In the analysis of the scheduling results from the class perspective, the difference between the two algorithms is not significant. However, in the textual scheduling algorithm, it can be seen from the following points that the scheduling results are more focused on student learning efficiency. First of all, the scheduling of courses is better by using the time with a high degree of superiority in the class period; then the courses in the adjacent time are staggered and the classrooms are relatively close to each other; secondly, it is more reasonable to arrange the physical education courses in the seventh and eighth periods; meanwhile, for the first time, the scheduling results of the algorithm in this paper are not more courses at the beginning of the semester, which is consistent with the scheduling principles of cognitive psychology, etc. The scheduling scheme designed in this paper can be upgraded for different weekly courses, for example, the courses involved in this paper are courses with 2, 4, and 6 hours per week, and for more weekly courses, we only need to add the corresponding teaching time and classroom positions in the designated positions in the code. The interval priority table is designed like the course interval priority table. After the schedule is generated, it can also be fine-tuned according to the actual situation; for example, some teachers want to teach at a specific time, etc. Since the coding is intuitive and easy to understand due to the decimal coding scheme, the relevant gene bits in the chromosome can be modified directly, and then conflict detection and conflict elimination can be performed.

4.2. Scheduling System Performance Test Results. The so-called load stress test is to check the number of concurrent users and the amount of data that can be carried by the system when it reaches the maximum number of users, the time required for the system to run normally. In the load stress test of the smart scheduling system, LoadRunner professional testing tool is used to test the monitoring of access to the system during the smart selection and
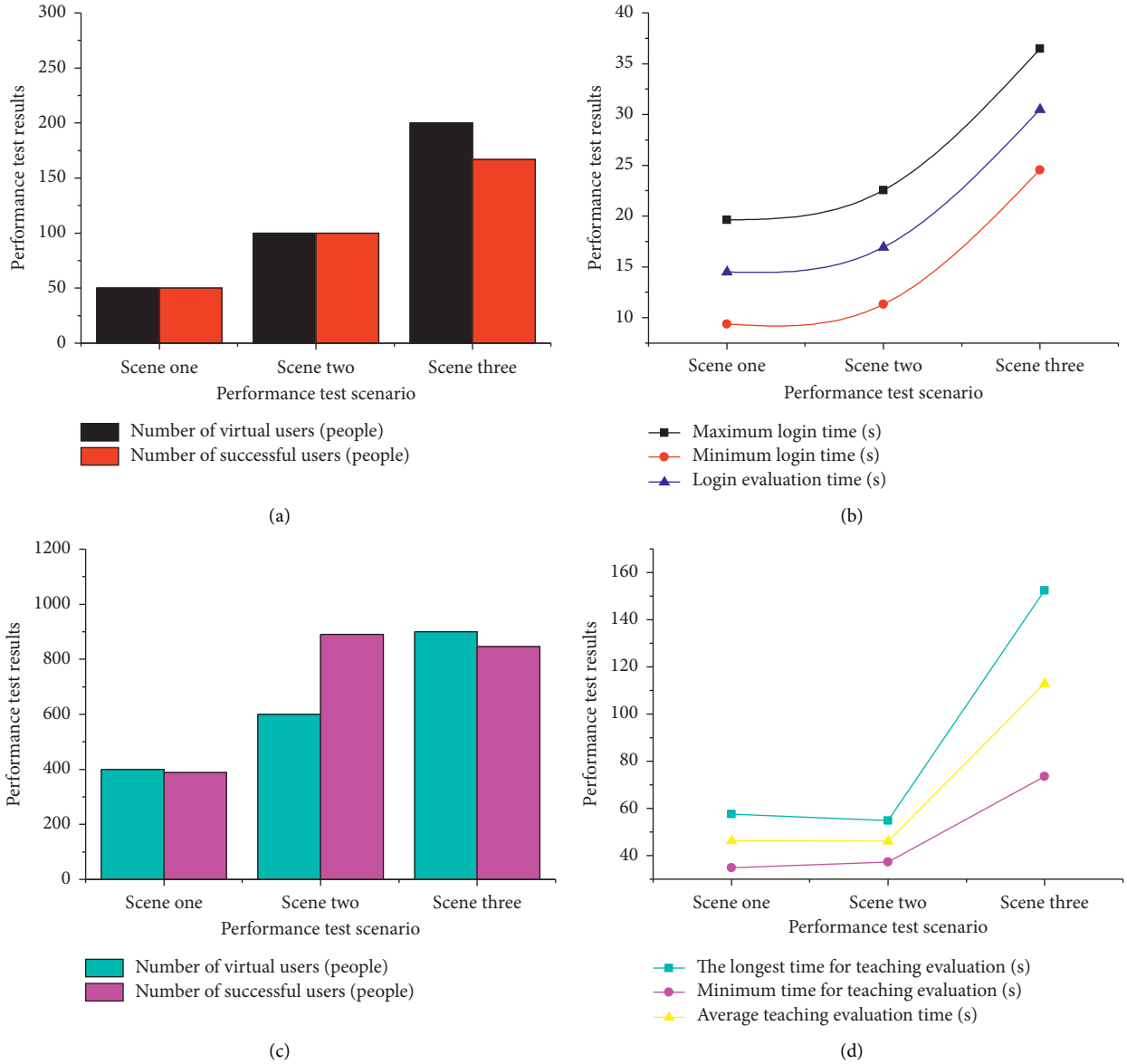
(a)

(b)

(c)

(d)

Figure 7: Stress load test results. (a) Log in system. (b) Log in system. (c) Teaching evaluation system. (d) Teaching evaluation system.

scheduling process by varying the number of users logged in each time. Generally, when a course selection activity is carried out and the course is popular, many students will log into the system for course selection at the same time once the system is opened, which is a challenge and test for the concurrent user capacity of the scheduling system. In the intelligent scheduling system of this paper, the maximum number of target online users set is 3000, and the number of people available for normal login access to the system is executed according to the number of concurrent users of 30% of the system online, so the maximum number of concurrent users selected for submitting teaching evaluation data is 900. The results of the system stress load test are shown in Figure 7.

From the results of the system stress load test in Figure 7, when the number of users logging into the system is less than or equal to one hundred, the system logging success rate

reaches 100%; when the number of concurrent users logging into the system is greater than or equal to two hundred, the logging success rate is only 95%, and there is a problem that access is denied when a small number of users log into the system. For teaching evaluation, when the concurrent number of logged-in users is less than or equal to four hundred, the success rate of submitting data reaches 100%; when the concurrent number of logged-in users is greater than or equal to six hundred, the success rate reaches 97%; when the concurrent number of users is greater than or equal to nine hundred, the success rate of submitting evaluation drops to 90%.

Intelligent scheduling system response time is one of the scheduling performance indicators. System response time is the user to the computer after a certain input or request; the computer made the response or output of the time. For the calculation of system response time, the number of system
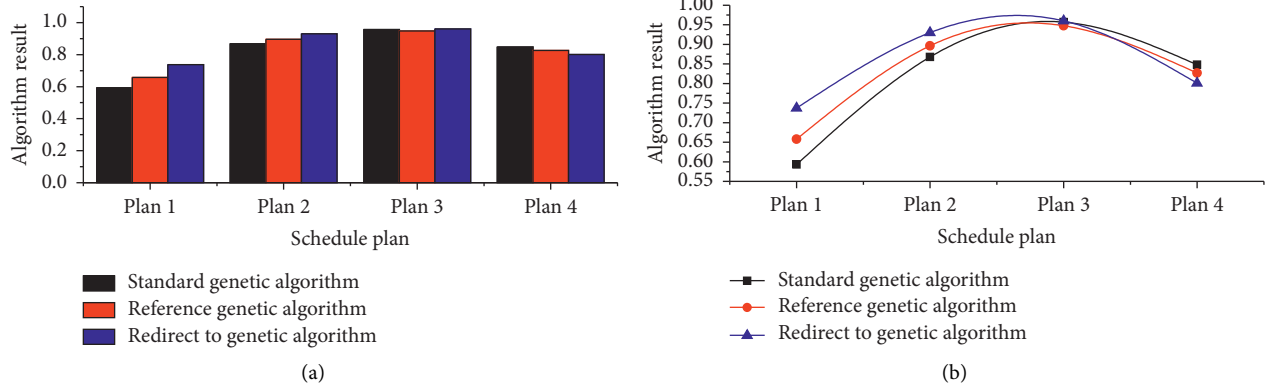
(a)

(b)

Figure 8: Value of the fitness function considered from the teacher's perspective.

users must be considered; when the number of system users is large, then the system response time is required to achieve high speed; otherwise, it is not guaranteed that each system user can accept the system response time. In this paper, the intelligent scheduling system is user/browser architecture; when the user opens the browser for access, the system page loading cache time, data transfer, and execution time, etc., will have a certain impact on the system response speed. When conducting the system response speed test, the different configurations of different users' computers, the smoothness of the network, the type of accessing browser, or different versions of the same browser will also have an impact on the system test results. These are objective influencing factors; these influences will be controlled to a minimum while controlling the system processing time as much as possible, and compressed to the shortest possible time, the results of the intelligent scheduling system performance operation indicators are shown in Figure 8.

During the test, the total number of courses in each class is 602, and there are 85 classrooms in the school. 474 scheduling results are obtained (this is because there are courses that do not need to be scheduled, such as graduation design, etc.), which takes less than two minutes and can meet the performance requirements of the automatic scheduling system in universities. The results of testing in a simulated environment show that the automatic class scheduling system designed in this paper has a time delay of 128 ms and a throughput of 800 Mb/s for 100 concurrent users, and the system testing results show that the automatic class scheduling system designed in this paper can automatically schedule classes reasonably and effectively according to the course requirements, teachers' resources, classrooms, and laboratories of universities.

5. Conclusion

Through the analysis of the traditional genetic algorithm, combined with the actual needs of college English curriculum, the genetic algorithm is improved. First, the decimal coding method is used for coding. Before the initial population is generated, the courses are sorted according to the weight of the course and the position of the teacher. The local optimal solution is generated prematurely, and adaptive crossover and mutation operations are used in the crossover and mutation operations. Finally, the improved genetic algorithm has been greatly improved in the value of fitness and time efficiency through experimental analysis. Through the analysis of the conflict problem of college English course arrangement, the maximum fuzzy pattern algorithm is introduced, the related terms of the maximum fuzzy pattern algorithm are defined, and the elements required by the item header table of the maximum fuzzy pattern tree are given, as well as the maximum fuzzy pattern mining algorithm. Experiments show that the largest fuzzy model tree can accurately find conflicting elements and solve the conflict problem in college English class arrangement. Although the improved genetic algorithm has achieved certain results, the experiment still has certain limitations: First, the data scale corresponding to the college English course arrangement is not large, and the amount of calculation is relatively low. The second is that the multicampus problem is not considered in the process of college English course scheduling. Therefore, in the future research work, it is necessary to seek a larger scale of data, while considering the problem of multiple campuses, so that the college English curriculum system can be improved.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] G. Janes, M. Perinic, and Z. Jurkovic, "Applying improved genetic algorithm for solving job shop scheduling Problems/ Ucinkoviti genetski algoritam za planiranje proizvodnje," *Tehnicki Vjesnik-Technical Gazette*, vol. 24, no. 4, pp. 1243–1248, 2017.

[2] A. Amrane, F. Debbat, and K. Yahyaoui, "GPU-based hybrid cellular genetic algorithm for job-shop scheduling problem," *International Journal of Applied Metaheuristic Computing*, vol. 12, no. 2, pp. 1–15, 2021.

[3] J. Arias-Osorio and A. Mora-Esquivel, "A solution to the university course timetabling problem using a hybrid method based on genetic algorithms," *DYNA: revista de la Facultad de Minas. Universidad Nacional de Colombia*, vol. 87, no. 215, pp. 47–56, 2020.

[4] H. Hu, J. He, X. He, W. Yang, J. Nie, and B. Ran, "Emergency material scheduling optimization model and algorithms: a review," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 6, no. 5, pp. 441–454, 2019.

[5] C. Contaldi, F. Vafaee, and P. C. Nelson, "Bayesian network hybrid learning using an elite-guided genetic algorithm," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 245–272, 2019.

[6] Q. Tian and X. Duan, "Teaching expansion analysis based on computer assisted diversified teaching," *Cluster Computing*, vol. 22, no. 6, pp. 13739–13744, 2019.

[7] W. Deng, J. Xu, X.-Z. Gao, and H. Zhao, "An enhanced MSIQDE algorithm with novel multiple strategies for global optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–10, 2020.

[8] M. Barkaoui and J. Berger, "A new hybrid genetic algorithm for the collection scheduling problem for a satellite constellation," *Journal of the Operational Research Society*, vol. 71, no. 9, pp. 1390–1410, 2020.

[9] D. Gabi, A. S. Ismail, A. Zainal, Z. Zakaria, and A. Al-Khasawneh, "Hybrid cat swarm optimization and simulated annealing for dynamic task scheduling on cloud computing environment," *Journal of Information and Communication Technology*, vol. 17, no. 3, pp. 435–467, 2018.

[10] W. Lin, L. Wang, Y. Zhang, and G. Tian, "MTLBO: a multi-objective multi-course teaching-learning-based optimization algorithm," *Journal of Applied Science and Engineering*, vol. 21, no. 3, pp. 331–342, 2018.

[11] J. Nourmohammadi-Khiarak, Y. Zamani-Harghalani and M.-R. Feizi-Derakhshi, Combined multi-agent method to control inter-department common events collision for university courses timetabling," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 110–126, 2020.

[12] J. H. Rosłon and J. Edward Kulejewski, "A hybrid approach for solving multi-mode resource-constrained project scheduling problem in construction," *Open Engineering*, vol. 9, no. 1, pp. 7–13, 2019.

[13] Y. Liu, S. Shahbazzade, and J. Wang, "The AFSA-GA algorithm for the quay crane scheduling problem of the loading and unloading operations," *International Journal of Software Science and Computational Intelligence*, vol. 9, no. 3, pp. 59–71, 2017.

[14] M. Vannucci, V. Colla, S. Dettori, and V. Iannino, "Fuzzy adaptive genetic algorithm for improving the solution of industrial optimization problems," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 409–422, 2020.

[15] S. K. Dey, D. P. Dash, and M. Basu, "Multi-objective economic environmental dispatch of variable hydro-wind-thermal power system," *International Journal of Applied Metaheuristic Computing*, vol. 12, no. 2, pp. 16–35, 2021.

[16] M. L. Seisarrina, I. Cholissodin, and H. Nurwarsito, "Invigilator examination scheduling using partial random injection and adaptive time variant genetic algorithm," *Journal of Information Technology and Computer Science*, vol. 3, no. 2, pp. 113–119, 2018.

[17] L. Calvet, J. D. Armas, D. Masip, and A. A. Juan, "Learnheuristics: hybridizing metaheuristics with machine learning for optimization with dynamic inputs," *Open Mathematics*, vol. 15, no. 1, pp. 261–280, 2017.

[18] A. A. Ewees and M. Abd Elaziz, "Improved adaptive neuro-fuzzy inference system using gray wolf optimization: a case study in predicting biochar yield," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 924–940, 2020.

[19] S. Rastbin, M. G. Shahbandi, and P. Soudmand, "A fuzzy based genetic algorithm for optimizing the pedestrian walking network; case study historic district of Tehran," *Journal of Industrial Engineering and Management Studies*, vol. 7, no. 2, pp. 119–138, 2020.

[20] I. Maatouk, I. Jarkass, E. Châtelet, and N. Chebbo, "Preventive maintenance optimization and Comparison of genetic algorithm models in a series-parallel multi-state system," *Journal of Intelligent Systems*, vol. 28, no. 2, pp. 219–230, 2019.

[21] Z. Qian, D. Hong, Z. Chang, J. Zhu, and Z. Zhu, "A strategy for multi-target paths coverage by improving individual information sharing," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 13, no. 11, pp. 5464–5488, 2019.

[22] X. Cai, H. Zhao, S. Shang et al., "An improved quantum-inspired cooperative co-evolution algorithm with muli-strategy and its application," *Expert Systems with Applications*, vol. 171, Article ID 114629, 2021.

[23] B. Jiao, Q. Chen, and S. Yan, "A cooperative Co-evolution PSO for flow shop scheduling problem with uncertainty," *Journal of Computers*, vol. 6, no. 9, pp. 1955–1961, 2011.

[24] X. Peng and Y. Wu, "Large-scale cooperative co-evolution using niching-based multi-modal optimization and adaptive fast clustering," *Swarm and Evolutionary Computation*, vol. 35, pp. 65–77, 2017.

[25] S. Yang, J. Wang, X. Hao et al., "BiCoSS: toward large-scale cognition brain with multigranular neuromorphic architecture," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.

[26] W. Wei, Q. Ke, J. Nowak, M. Korytkowski, R. Scherer, and M. Woźniak, "Accurate and fast URL phishing detector: a convolutional neural network approach," *Computer Networks*, vol. 178, Article ID 107275, 2020.