Intelligent Computing Technology and Automation
Z. Hou (Ed.)

doi:10.3233/ATDE231224

Pharmaceutical Laboratory Course Scheduling System Optimization Based on Genetic Algorithm

Shaohua CHEN[a,1]
[a] *Minxi Vocational and Technical College, Longyan, 364021, China*

Abstract. This article analyzes and assumes the construction of laboratory platforms in response to the specific requirements for intelligent management of pharmaceutical experiments and the experimental teaching requirements of pharmaceutical concepts. Then, the influencing factors and main constraints in the course scheduling problem are systematically discussed, a mathematical model of the course scheduling process is provided, and an overall framework for the solution method is also proposed. On this basis, the improvements are proposed to the algorithm, which mixed simulated annealing algorithm to improve the accuracy and convergence speed of the algorithm. Finally, the improved adaptive genetic algorithm is applied to the intelligent course scheduling system in universities, and the usability of the system is tested through empirical analysis. The comparison and analysis results between the improved genetic algorithm and the classical genetic algorithm indicate that our strategy can achieve good global optimal solution search ability, which provides better service for teaching management system of pharmaceutical laboratories.

Keywords. *GA*; course scheduling system; mutation; cross; chromosome; pharmaceutical laboratory

1. Introduction

Due to the difference between the cultivation of pharmaceutical talents and other disciplines, corresponding scientific laboratory preparations should be made according to the characteristics of pharmaceutical cultivation. Practical teaching is an important content of pharmaceutical teaching, and pharmaceutical talent cultivation should pay more attention to the construction of laboratory platforms. Practical platforms play an indispensable role in pharmaceutical experimental teaching. Therefore, more rigorous practical operations should be applied to cultivate students' innovative ability to observe and discover problems, At the same time, one must have the ability to solve problems [1]. The pharmaceutical research laboratory management system is a product of the combination of science and modern information technology. The pharmacological mechanism of action must rely on rigorous scientific experimental verification, and it utilizes computer network technology, data storage technology, rapid data processing technology, and other technologies to comprehensively manage the laboratory. With the rapid development of artificial intelligence, big data, information technology, and the Internet of Things technology, constructing a

[1] Corresponding Author: Shaohua CHEN; Minxi Vocational and Technical College, Longyan, 364021, China; cehns@163.com

laboratory management system that is in line with medical colleges has become an important means to improve the management level of pharmaceutical laboratories, optimize the allocation of scientific research and teaching resources in pharmaceutical laboratories, and enhance the safety management ability of laboratories. The purpose of establishing a course scheduling management system for pharmaceutical research laboratories is to improve testing efficiency and efficiency, provide accurate data for scientific research, manage pharmaceutical laboratories in an orderly manner, and improve the overall research level of a comprehensive intelligent laboratory management system.

Genetic algorithm(GA) is a heuristic search algorithm that can be used to solve problems in course scheduling systems. In the experimental course scheduling system, genetic algorithms can be used to optimize course scheduling to meet various constraints and optimization objectives [2]. The usual approach is to first define a fitness function, which evaluates the quality of each scheduling plan. The fitness function can consider factors such as classroom utilization, teacher time utilization, and student course conflicts. Then, it is necessary to define the gene representation, that is, how to represent a scheduling plan. Binary encoding, integer encoding, or other suitable encoding methods can be used. Next, it is necessary to define the operations of genetic algorithms, including selection, crossover, and mutation. The selection operation is used to select individuals with high fitness, the crossover operation is used to generate new individuals, and the mutation operation is used to introduce randomness to avoid falling into local optima. In each generation, new individuals are generated through selection, crossover, and mutation operations, and their fitness is calculated. Then, individuals are sorted based on their fitness values and the individuals with higher fitness are selected as the next generation's population. Repeat the evolution process for multiple generations until the stop condition is reached. Therefore, in practical applications, it may be necessary to combine other optimization methods or heuristic algorithms to improve the efficiency and performance of the algorithm.

In order to meet the special needs of teaching management in pharmaceutical laboratories, this article introduces genetic algorithms into the scheduling algorithm, which can search for a wider solution space and find better course scheduling solutions. Encoding the timetable is a way to represent it as a chromosome object. Chromosome objects are composed of genes, each representing a time slot or a course. In the fitness function, the optimization of course duration, the fitness of course day combination, the fitness of weekly class hour distribution, and the uniformity of classroom day distribution are set. Randomly select a gene from a chromosome and perform mutation operations. For example, exchanging course schedules between two time slots, or randomly selecting a course for replacement. In order to make the algorithm more stable and efficient, a GA optimization mechanism based on simulated annealing algorithm is also proposed. At the end of the algorithm, the individual with the highest fitness is output as the optimal solution, which is the best course arrangement plan. The simulation experimental results show that the proposed scheduling mechanism has the optimal solution and can be used to improve the efficiency of online scheduling services in pharmaceutical laboratory management systems.

2. Requirement Analysis and Problem Description of Course Scheduling System for Pharmaceutical Laboratories

2.1 Demand Analysis

The opening of the laboratory has greatly increased the workload and management difficulty of the laboratory. The Pharmaceutical Experimental Center follows the gradually advancing experimental teaching model of "foundation synthesis design innovation" and follows an open experimental teaching method that combines "teaching research". After the opening of the laboratory, students conduct independent experiments and choose experiments based on their interests and professional directions. The diversity of experimental content and the continuity of experimental time bring certain difficulties to the management of the laboratory, and students may raise more and more complex questions in independent experiments [3,4]; At the same time, due to the varying levels of students' proficiency and familiarity with the operation of instruments and equipment, it has brought certain difficulties to the management and maintenance of the laboratory and its instruments and equipment. In order to ensure the effective and orderly conduct of teaching and research work, as well as greatly improve students' innovative research and practical abilities, the experimental center implements a gradually transitional laboratory open management model of "teacher to teacher and student to student". Before discussing the issue, we generally assume that:

- Non overlapping course times: Assuming that each course can only be scheduled once within a specific time period to avoid time conflicts.
- Classroom capacity limit: Assuming that each classroom has a certain capacity limit to ensure that the number of students arranged does not exceed the capacity of the classroom.
- Teacher availability: Assuming that each teacher can only teach one course during a specific time period, to ensure teacher availability and avoid time conflicts.

These assumptions can be adjusted and expanded based on specific scheduling issues.

2.2 Mathematical Description of Course Scheduling Problems

The scheduling factors mainly include: (teaching) time, (teaching) class, (teaching) course, (teaching) teacher, and (teaching) location. Course scheduling mainly aims to solve the contradictions and conflicts that may occur in the arrangement and combination of these resources, and find a better and reasonable combination that does not conflict. Suppose there is a set of courses, each with a fixed time period and duration. We need to schedule these courses at different times of the day to maximize the number of courses arranged and avoid time conflicts.

We can use the following variables to represent the scheduling problem:

Course collection: $C=\{c_1, c_2,..., c_n\}$, where ci represents the i-th course.

Time period set: $T=\{t_1, t_2,..., t_m\}$, where ti represents the i-th time period.

Course time period mapping: $M=\{(c, t) \mid c \in C, t \in T\}$, indicating which time period each course is scheduled in.

We need to meet the following constraints:

Each course can only be arranged within one time period: for each course $c \in C$, there exists a unique time period $t \in T$, such that $(c, t) \in M$.

There cannot be time conflicts between courses: for any two courses c_i, $c_j \in C$, if they are arranged in the same time period $t \in T$, then the time periods of c_i and c_j cannot be overlapped.

Our object is to find a course time period mapping M that meets the above constraints, in order to maximize the number of scheduled courses.

3. Genetic Algorithm Optimization Process and Its Implementation in Course Scheduling System

3.1 The Basic Process of GA Applying to Course Scheduling System

When designing a course scheduling system using genetic algorithms, it is necessary to comprehensively analyze the course scheduling business processes, school resources, and personalized needs of different universities, and develop an intelligent course scheduling system that meets the rules of university course scheduling. It will automatically generate a course scheduling plan that meets the needs of teaching management, in order to meet the performance requirements of universities for the intelligent course scheduling system. The main indicators of its performance requirements are the algorithm's execution efficiency, The stability, scalability, and maintainability of the system [5]. The key mechanism design of the course scheduling system is described as follows:

(1) Establish initial population

We use a two-dimensional array to represent the student information of a class. Each student's information includes name, age, gender, grades, etc. Chromosomes are composed of multiple genes, each representing a time slot or a course. For example, a chromosome can represent a weekly course schedule, including daily time slots and corresponding courses. Each element represents a time slot and contains course information for that time slot. Here is an example of how to use a two-dimensional array to represent a class schedule

```
# Define a two-dimensional array of class schedules
class_schedule = [
    ["Experimental Pharmaceutics I", " Experimental Organic Chemistry Ⅱ", " Experimental Instrumental Analysis", " Experimental Medicinal Chemistry"],
    ["Experimental Medicinal Chemistry", " Experimental Pharmaceutics II", " Experimental Physical Chemistry", "Math"],
    ["Experimental Synthetic Medicinal Chemistry", "Chemistry", "Math", "English"],
    ["Experimental Natural Products Chemistry ", " The Integrated Experiments of Pharmacy", " Experimental Pharmacognosy I", " Experimental Pharmacognosy I"]
]
#The rows of a two-dimensional array represent the day of the week, and the columns represent the time slot
#Obtain course schedule for Monday
monday_schedule = class_schedule[0]
#Obtain the course for the first class on Mondayfirst_class = class_schedule[0][0]
#Revise the course for the third class on Tuesdayclass_schedule[1][2] = " Experimental Pharmaceutics I "
```

(2) Fitness function Design

Considering the soft constraints mentioned earlier, we define the cost of a gene as:

$$Cost(Timetble) = \sum_{i=1}^{classes} p_i * w_i \tag{1}$$

where w_i is the weights of penalty values. Then the fitness function is

$$F(Chromosome) = 1 / Cost(Chromosome) \tag{2}$$

The value of w_i can be flexible since it can be changed according to course scheduling requirements and different fitness function, which leads to the final orientation of curriculum.

(3) Design of genetic operators

Usually, the cumulative probability of an individual is determined based on their fitness or priority. Individuals with higher fitness or priority will have a higher probability of being selected during the scheduling process, thereby increasing their cumulative probability. Set the number of parent populations as $F = \{x_1, x_2, ..., x_n\}$ and the fitness as $F = \{f(x_1), f(x_2), ..., f(x_n)\}$, and then calculate the probability of an individual being selected according to the following equation:

$$P(x_i) = \frac{f(x_i)}{\sum_{i=1}^{n} f(x_i)} \tag{3}$$

(4) Conflict re detection

After generating the course schedule, detect and repair conflicts between courses. Traverse the curriculum to detect types of conflicts such as time conflicts, classroom conflicts, and teacher conflicts. If conflicts are found, fix them according to the specified rules [6]. The methods of repair can include adjusting course schedules, changing classrooms, or rearranging teachers. After completing the above round of genetic algorithm steps, an approximate global optimal solution is obtained, which completes the evolutionary process of the genetic algorithm.

3.2 GA Optimization

The specific application process of this scheme is shown in figure 1.

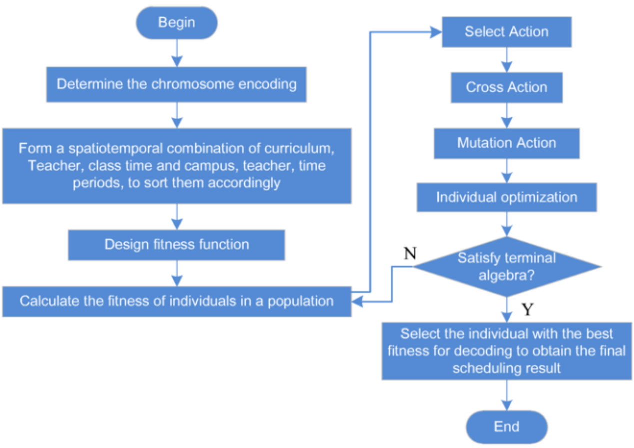

Figure 1. Basic process of course scheduling algorithm based on GA optimization.

Genetic algorithm and simulated annealing algorithm are two commonly used optimization algorithms that can be combined to improve optimization performance [7,8]. The following is an algorithm that combines genetic algorithm with simulated annealing, called the Genetic Simulated Annealing(SA-GA) Algorithm:

Initialize population: Firstly, randomly generate a set of initial individuals as the initial population of the genetic algorithm.

Step 1: Evaluate fitness: For each individual, calculate their fitness value to measure their ability to solve problems.

Step 2: Selecting parents: Use a selection algorithm to select a portion of individuals from the population as parents, and select individuals based on fitness values, so that individuals with higher fitness have a higher probability of being selected.

Step 3: Cross and mutation: Generate new offspring individuals from selected parent individuals through cross and mutation operations. Cross operation can generate new solutions by exchanging certain parts of an individual, while mutation operation randomly changes certain genes of the individual.

Step 4: Replace Parent: Replace some of the parent individuals with the generated offspring to maintain the same population size. Replacement operations can be selected based on fitness values, such as selecting individuals with lower fitness for replacement.

Step 5: Simulated annealing: For each offspring individual, use simulated annealing algorithm for local search. The simulated annealing algorithm increases global search ability by accepting a certain probability of poor solutions to avoid falling into local optima.

Step 6: Update population: Compare the simulated annealing offspring individuals with the original population, and select individuals with higher fitness as the next generation population.

Step 7: Output optimal solution: After reaching the stop condition, output the individual with the highest fitness, which is the optimal solution.

Repeat iteration: Repeat steps 2 to 7 until the stop condition is reached. The stopping condition can be reaching a predetermined number of iterations, reaching a certain threshold for fitness, or not significantly improving after a certain period of time.

By combining genetic algorithm and simulated annealing algorithm, genetic simulated annealing algorithm can find a balance between global search and local search, thereby improving the optimization effect.

4. Case Analysis

4.1 Functional Testing of Course Scheduling System

After the algorithm design is completed, this article uses 2021 course scheduling data from a third year of a medical school as test data for detection. There are a total of 2000 students, 25 teaching classes, and 30 experimental teachers. The course scheduling system case adopts a B/S structure of JSP+MSSQL developed using MVC mode. The management subsystem adopts improved GA proposed in this article for automatic course scheduling in pharmaceutical experiments. The system uses independent division and management for each class, and all teachers can serve any class. Each class is automatically sorted based on the number of courses per day, and each teacher is allowed to teach multiple courses. The course scheduling management subsystem adopts genetic algorithms for automatic course scheduling. The login subsystem distinguishes the different identities of course schedulers, teachers, and students, provides different permissions, and judges their corresponding functions on the page based on their identities to use this system. The course scheduling subsystem is mainly used by course schedulers, who can conduct all activities related to course scheduling here. Referring to auxiliary elements, fill the class bound to the teacher with unique color blocks to a specific fixed time period in the classroom. In principle, all experimental courses are scheduled in the academic management system, with designated experimental teachers and hours to complete teaching tasks. For independent course experiments with a requirement for continuous scheduling throughout the week, priority will be given to course scheduling; The same class cannot add the same course repeatedly in the teaching plan. Once all information is confirmed to be correct, one click scheduling can proceed. Figure 2 shows the result interface of the course scheduling plan: the teacher's schedule can be viewed in a table format or in a list format. The schedule can also be output in Excel or PDF format and displayed in a list format; Click on the intelligent scheduling button for class schedule query, and find the class schedule query in the left function bar. Class schedule query can be queried by three dimensions: class, teacher, and venue. In the analysis window of teacher teaching situation, click a certain teacher in the left teacher statistics list and the table on the right will display the specific teaching situation and daily class hour distribution of the teacher.

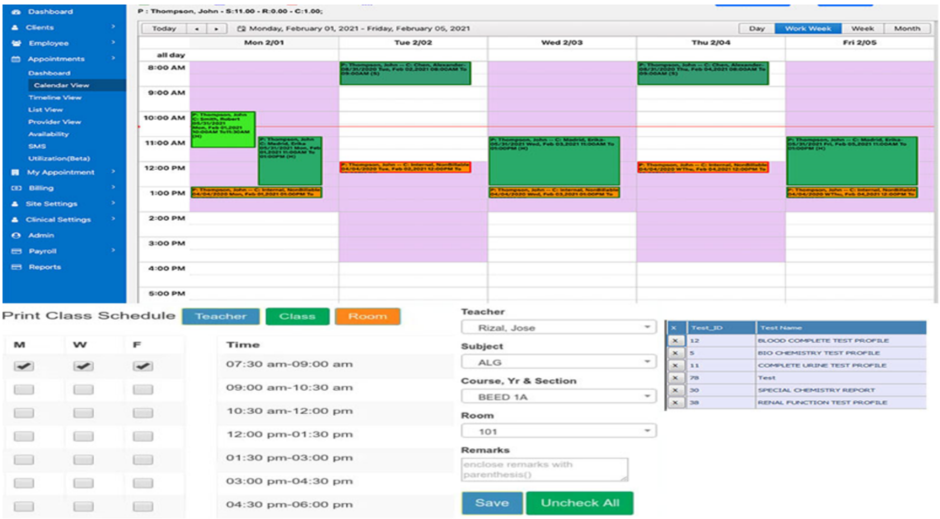

Figure 2. Course scheduling plan result interface diagram

4.2 Performance Testing of Course Scheduling System

To test the performance of the algorithm, we first set the size of the population to remain unchanged and create an empty list to store the fitness function values of each individual. For each individual, calculate their fitness function value and add it to the list. Calculate the average of all fitness function values in the list. Perform 500 tests on both the genetic algorithm and the basic genetic algorithm designed in this article, and calculate the average fitness function of the initial population, the average number of iterations at algorithm termination, and the evolution calculation time, as shown in Table 1. Calculate the performance consumption of the optimal solution through basic GA and improved GA algorithm. From the results in the table, it can be seen that the definition of the fitness function of the initial population in this article is more reasonable and accurate, which can better evaluate the individual's strengths and weaknesses. It has a significant impact on the iteration time and average fitness value of the course scheduling process: the average number of iterations has been reduced by 15%; The calculation time has been increased by 25%.

Table 1. Comprehensive comparison of course finding and scheduling performance between this algorithm and traditional algorithms

Algorithm	Average population fitness	Average number of iterations	Average time cost(s)
SA-GA	6105.3	1418.6	620.58
Classic GA	5524.7	1536.5	817.94

In the iterative process of genetic algorithms, the change in the number of conflicts among the optimal individuals can provide information about algorithm performance and solution quality. If the number of conflicts for the optimal individual gradually decreases during the iteration process, it indicates that the algorithm is effectively optimizing the course scheduling plan. However, if there are significant fluctuations or no significant improvements in the number of conflicts among the optimal individuals during the iteration process, it may be necessary to adjust algorithm parameters or take other optimization measures. Figure 3 shows the variation of the conflict number of the optimal individual in the SA-GA and classical GA populations during the iteration process. It can be seen that as the iteration continues, the algorithm in this article quickly continues to search and gradually converges to a better solution, ultimately reaching a lower number of conflicts; The latter has significant fluctuations or no significant improvement in the number of conflicts during the iteration process. This change is due to the operation of selection, crossover, and mutation in the improved genetic algorithm. In each generation, the selection operation selects individuals with higher fitness as parents, which helps to preserve better solutions. The introduction of new individuals through crossover and mutation operations may improve the quality of course scheduling plans.

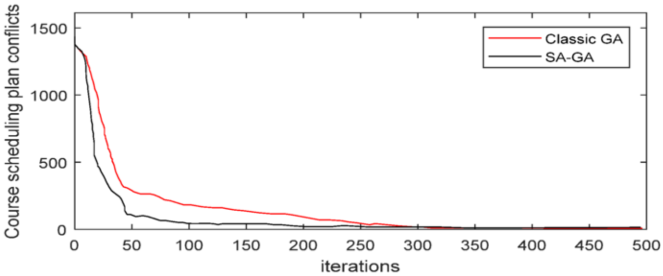

Figure 3. Changes in the number of conflicts in course scheduling plans

The optimal values for mutation rate and crossover rate are problem specific and may vary depending on the complexity of the problem, population size, and other factors. Usually, experiments and adjustments are needed to find the optimal parameter values. We observe the impact of these two main parameter values on the accuracy of the algorithm by adjusting their thresholds through transformation. From the experimental results in figures 4, it can be seen that if the crossover rate is low, there is less gene exchange between individuals. This may lead to weak exploration ability of the algorithm, low population diversity, and difficulty in finding better solutions. The result may cause the algorithm to fall into local optima; If the mutation rate is high, the individual's genes will undergo frequent changes. This may lead to the algorithm overexploiting the search space, resulting in high population diversity, but it may also lead to the algorithm losing convergence and making it difficult to find optimal solutions. When the number of iterations of the algorithm is 100, the mutation rate is 0.06, and the crossover rate is set to 0.9, the average fitness value obtained by the algorithm is the highest. Therefore, it can be considered that the algorithm has achieved the optimal solution, which is the most suitable experimental management schedule.

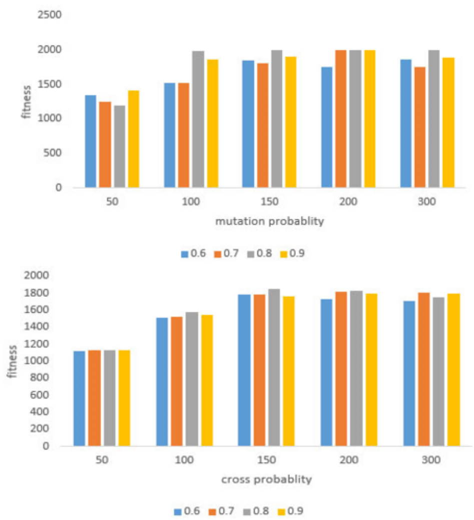

Figure 4. The influence of crossover probability and iteration times on fitness.

5. Conclusion

Genetic algorithm, as a self-organizing and adaptive artificial intelligence technology for solving problems, provides an effective tool for general combinatorial optimization problems. This article practices and explores the open operation mechanism of pharmaceutical professional laboratories, proposes the idea of applying improved GA algorithm for course scheduling, and provides a system implementation plan. In response to the actual needs of pharmaceutical laboratories, based on the framework steps of genetic algorithms and the specific characteristics of course scheduling problems, targeted coding, population initialization, genetic operator design, hybrid simulated annealing algorithm, and other tasks are completed. Subsequently, a detailed design of the course scheduling system was carried out. Finally, a comparative experiment was conducted between the improved genetic algorithm and the basic genetic algorithm, and it was found that the improved genetic algorithm has high efficiency. The test also proved that the course scheduling system can effectively complete course scheduling and use.

References

[1] Jingjing Z. Analysis of pharmaceutical laboratory platform construction. Jiangsu Science & Technology Information, 2018, 13(5): 63-65
[2] Wang, Yen Zen. Using genetic algorithm methods to solve course scheduling problems. Expert Systems with Applications, 2003, 25(1):39-50.
[3] Yu G, Xiao-Yong L, Guo-Tao T, et al. Exploration in the Open Operation Mechanism of Pharmaceutical Laboratories. Northwest Medical Education, 2011, 19(5): 1025-1027.
[4] Zhu Liangxue. Research on the Design of Curriculum Scheduling System in Vocational Colleges Based on Genetic Algorithm. Journal of Hubei Normal University (Philosophy and Social Science), 2021, 41(3):64-99
[5] Li-Li C, Ning H U. On Design of Intelligent Classroom Scheduling System Based on Genetic Algorithm. Computer Knowledge and Technology, 2019, 15(6): 159-161.
[6] CHEN Shoujia, FU Xia,ZHOU Xin, et al.Solution to course scheduling problems based on genetic algorithm and tabu search. computer application, 2007, 27(7):1806-1808.
[7] Iranmanesh A, Naji H R. DCHG-TS: a deadline-constrained and cost-effective hybrid genetic algorithm for scientific workflow scheduling in cloud computing. Cluster computing, 2021, 2:24.
[8] Liu C, Liu Y, Wang J. A revised imperialist competition algorithm for cellular manufacturing optimization based on product line design. Journal of Industrial and Management Optimization, 2023, 19(1):69-104.