

# Application of Genetic Algorithm and Simulated Annealing Algorithm for Course Scheduling Problem

Deng-yu-hui LI\*, Jia-ji SHEN, Hui-zhu DONG, Yi-ran SU and Zhi-gang ZHANG

University of Science and Technology Beijing, China

\*Corresponding author

**Keywords:** Timetabling problem, Genetic algorithm, Simulated annealing algorithm.

**Abstract.** In this paper, we propose a special course scheduling problem with class combination. By dismantling the restrictions of given classes, students with similar foundations are reorganized into new classes. It helps to improve the efficiency of the class and the acceptance of students. However, it is difficult to arrange new classes and time artificially, considering class capacity and the conflict with other courses. And as the number of class increases, the calculation time shows an exponential growth, which is an NP-hard problem. In order to solve this problem, we use genetic algorithm and simulated annealing algorithm to get the optimal solution.

## Introduction

As one of the core contents of teaching management, class scheduling is an important indicator to measure the level of teaching management.[1] The intelligent automatic course arrangement is a NP-hard problem.[2] Nowadays, many universities face the problem of class recombination. That is, due to the different original capabilities of students, it is necessary to reorganize classes just for a certain course, to help the new class become more efficient and more active. In this paper, we use genetic algorithm and simulated annealing algorithm to discuss the course scheduling problem with class recombination, and analyze their results respectively. Our experiments come from real situation in the University of Science and Technology Beijing, taking the subject of English as an example.

## Problem Description

We need to arrange English classes for 120 given classes. There are seven feasible periods, which are Tuesday's 1st, 2nd and 3rd lessons and Wednesday's 1st, 2nd, 3rd and 4th lessons. Before the English class, the Mathematics class time has been arranged, which is available for the 1st, 2nd and 3rd lessons on Monday, Wednesday, and Friday. According to the existing schedule, the time of the English class and the Mathematics class cannot be conflicted, and the course arrangement of the English class also has the following requirements:

1. The students are divided into five levels, each of which accounts for 10%, 10%, 20%, 40%, 20%. So there are five levels in each class. But the proportion of five levels in these classes is different, and it should be ensured that the sum of students of each level is equal to the pre-set proportion.
2. There are seven optional periods. Because the number of students in classes varies, we should pay attention to combine large and small classes, so that the number of new classes in each period is similar.
3. All students are divided into 7 periods. The number of students in each period should be equal to 1/7 of total students. At the same time, the number of students in each period is divided into 2-5 according to English level. and the ratio should be 1:2:4:2.

As we can see, this problem belongs to integer programming, the equality of the above requirements can not be fully satisfied, so in the actual solution, we weaken it to the minimum error, that is, the difference between the final results and the above requirements is the smallest.

### Parameters and Constraints

Based on the problem of course scheduling in this paper, some parameters are involved, class set:  $C = \{c_1, c_2, \dots, c_{120}\}$ , and period set:  $T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$ . Variables  $X_{c,t}$  and  $Y_{c,t}$  are defined:

$$X_{c,t} = \begin{cases} 1, & \text{if students in class } c \text{ take English class during period } t \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$Y_{c,t} = \begin{cases} 1, & \text{if students in class } c \text{ take Mathematics class during period } t \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

There are two strict constraints in this problem: The same class can not take English and Mathematics simultaneously, so there is  $X_{c_m t_r} + Y_{c_m t_r} \leq 1$ . And one and only one of periods in the same class is selected, so there is  $\sum_{i=1}^7 X_{c_m t_i} = 1$ . Soft constraints includes three parts. Since they are all related to the number of students in each English level, a matrix  $A_{120 \times 6}$  is needed, whose rows store the information of different class. And the number of students with different English level respectively.

$$\min y = \sum_{j=1}^7 \left| \sum_{i=1}^{120} X_{c_i t_j} - 17 \right| \tag{3}$$

$$\min z = \sum_{j=1}^7 \left| \sum_{i=1}^{120} A(i,1)X_{c_i t_j} - 487 \right| + \sum_{j=1}^7 \left| \sum_{i=1}^{120} A(i,3)X_{c_i t_j} - 49 \right| + \sum_{j=1}^7 \left| \sum_{i=1}^{120} A(i,4)X_{c_i t_j} - 97 \right| + \sum_{j=1}^7 \left| \sum_{i=1}^{120} A(i,5)X_{c_i t_j} - 195 \right| + \sum_{j=1}^7 \left| \sum_{i=1}^{120} A(i,1)X_{c_i t_j} - 97 \right| \tag{4}$$

### Precise Method

#### Based on Lingo

Firstly, we prepare a data file to store the number information of each class, that is matrix  $A$ . And matrix  $Y$  stores the information of Mathematics class. It takes 37 seconds as shown in the second picture, without soft constraints.

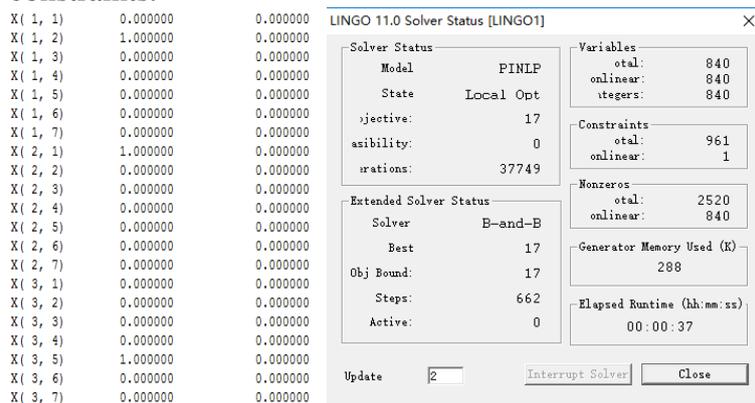


Figure 1. The results of Lingo

### Traversal Method

Traversal method is to list all the situations, and then choose the ones that satisfy the conditions. In this problem, there are 120 classes and 7 feasible periods, so we need to generate a matrix of  $120 \times 7$ , in which each row only have one 1, and the rest are 0. After the initial matrix  $A$  is generated, it is added to the matrix  $B$  which stores the information of Mathematics class. The size of matrix  $B$  is  $120 \times 7$ . If a class has a Math class during a certain period, the corresponding element is -1, and 0,

otherwise. If the number of non-zero elements in the summation matrix is 240, it means there is no conflict between English class and Math class. Then, the preliminary screening is performed so that the number of classes in each period is roughly equal. The initial screening method is to sum the matrix  $A$  by column, and then to find the variance between these numbers, and take the  $A$  with minimum variance. Since the method takes too much time, following results just consider 7 classes.

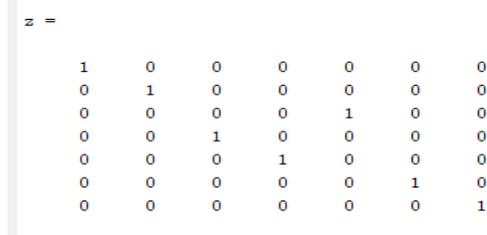


Figure 2. The results of traversal method.

Because there are only 7 classes here, the preference function is simplified. For 120 classes and 7 periods in this problem, the time-consuming is mainly in the generation of the initial matrix, and there are  $7^{120}$  initial matrices. So the method is unrealistic when there is more data.

### Solution of Genetic Algorithm

#### Coding

In this problem, we take the real number coding method, and each gene represents a selected period of one class. There are seven optional periods correspond to the real numbers 1-7 as follows:

Table 1. The coding of GA

period	Tuesday			Wednesday			
	1st class	2nd class	3rd class	1st lesson	2nd class	3rd class	4th class
coding	1	2	3	4	5	6	7

Each class corresponds to a number, and the problem is transformed into finding out the optimal sequence. The sequence consists of numbers from 1 to 7, and the number of people in each period is as average as possible, in order to optimize the use of resources. After that, we define the global population  $pop$ .  $pop(i,j)$  represents the time of English class selected by  $j$ -th class of the  $i$ -th individual.

Table 2. The structure of chromosomes

2	3	4	3	4	5	6	...	7	5	4	1	2
---	---	---	---	---	---	---	-----	---	---	---	---	---

#### Initialization

First, a reasonable set of solutions is obtained by observation, while the other sets of solutions are randomly generated by the random number method, forming the first generation together.

#### Fitness Function

The fitness function is the key to evaluate the pros and cons of individuals. It is divided into two parts, the difference between the number of people and the predicted one for each period, and the variance of the number of students in each period according to each level.

#### Selection

We take the roulette method, that is, according to the ratio of the local fitness function to the global fitness function, assign choose probability to each individual. Elite selection can be used, that is, the two parents with the greatest fitness function are always retained to the next generation.

### Crossover

First, randomly pair for individuals in the group. Secondly, randomly set the intersection position, including two-point intersection and three-point intersection. Finally, the genes between the positions of the paired chromosome intersections are exchanged, and the other positions remain unchanged.

Take the intersection of twelve classes as an example: (arrows indicate the locations of crossover)

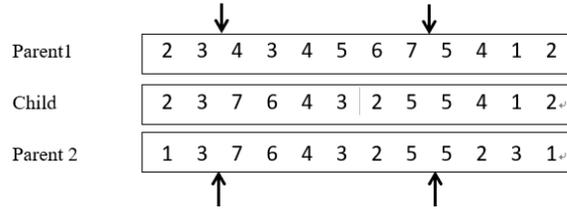


Figure 3. Two-point crossover

It is necessary to judge whether there is a conflict with the Math class before the exchange.

### Mutation

We take single point mutation method. Randomly select a position of an individual with the period  $t$ . There are two possible ways: Change the time to  $7-t$  to get a new individual, or randomly select another one in 1-7 to replace the original gene. It is also necessary to judge whether the selection time conflicts with Math class.

Table 3. Results of GA

class	result										
1	3	21	3	41	2	61	3	81	4	101	2
2	6	22	1	42	1	62	6	82	4	102	1
3	6	23	2	43	3	63	2	83	4	103	2
4	1	24	1	44	4	64	2	84	3	104	5
5	3	25	3	45	4	65	6	85	7	105	1
6	3	26	2	46	7	66	6	86	7	106	2
7	6	27	6	47	3	67	7	87	7	107	7
8	4	28	3	48	2	68	6	88	2	108	7
9	2	29	2	49	2	69	1	89	2	109	1
10	4	30	6	50	5	70	4	90	6	110	6
11	6	31	3	51	6	71	7	91	4	111	2
12	7	32	1	52	7	72	6	92	6	112	6
13	6	33	2	53	2	73	1	93	2	113	6
14	6	34	1	54	6	74	7	94	5	114	5
15	1	35	5	55	5	75	7	95	2	115	1
16	3	36	4	56	3	76	3	96	1	116	3
17	1	37	4	57	5	77	5	97	2	117	6
18	6	38	6	58	3	78	1	98	3	118	3
19	3	39	6	59	2	79	3	99	4	119	2
20	2	40	4	60	6	80	7	100	2	120	4

### Results Analysis

After Selection, Crossover, and Mutation, we get a new generation of populations, and continue the above operations. Until the maximum genetic algebra is reached, it stops. Table 3 shows the optimal descendant sequence, which is the English class time selected. Compared with the previous methods, it runs faster. After many experiments, the optimal fitness is stable at around 0.0882.

## Solution of Simulated Annealing Algorithm

### Initialization

We take the initial temperature as 500, the termination temperature as 1, and the temperature change rate as 0.95. Temperature is an important parameter, which decreases gradually with the iteration of the algorithm to simulate the cooling process during solid annealing.

### Objective Function

Simulated annealing algorithm optimizes the minimum of objective function.[4] In this model, the objective function is the variance of the number of students in each period and the deviation of the number of students in each period according to different levels. The two factors work together.

### Metropolis Criterion

It refers to the probability that it accepts a new solution. For the minimizing problem, for the current solution  $x$  and the new solution  $x'$ , there are 3 states. If  $f(x') < f(x)$  the new solution will be accepted as the current solution. If  $\exp\left[-\frac{f(x') - f(x)}{T}\right]$  is greater than a random number in (0,1) interval, the new solution will still be accepted as the current solution. Otherwise, the new solution will be rejected, and the current solution will be retained. It can be seen that the temperature is high at the beginning, which gives the algorithm a greater chance to jump out of the local optimal solution. As it proceeds, the temperature gradually decreases and the probability of accepting the worse solution decreases.

### Results Analysis

It is a randomly selected of one-time running result as shown in Table 4. When the temperature is higher than the termination temperature, it stops. This is the outermost cycle, which has 123 times, and the inner cycle has 1000 times. In general, the number of cycles is at least  $123 \times 1000 = 123\ 000$ , except for some cycles which have certain requirements for random solutions and initial solutions.

Table 4. Results of simulated annealing algorithms

	Result
Number of examinations in each period	(17 17 21 13 12 21 19)
Number of persons in each period	(472 501 596 365 346 605 527)
The variance	8.9414e+03
The sum of the deviations	575.7400

### Summary

We propose a special problem of courses arrangement, with class recombination. Therefore, students with similar foundations can form a new class. We consider about the required number of students in different period and the conflict of other courses, it is very difficult to arrange them manually. We use specific methods to solve small-scale cases first, to verify the model. And then, we give the solutions based on genetic algorithm and simulated annealing algorithm.

### Acknowledgement

This paper was funded by the Beijing Education Union.

### References

- [1] GONG Cheng, CHEN Gaoyun, LIU Yintian, LI Daiwei, Solution to Course Scheduling in Colleges and Universities Based on Improved Genetic Algorithm, *J. Software Engineering*. 2018, 21(3):1-4.

- [2] ZONG Wei, Research and Realization of University Timetable System Algorithm, *J. Computer Simulation*, 2011, 28(12): 389-392.
- [3] CHEN Lu, WANG Xiu, Solution to Course-Timetabling Problem of Mobile Learning System Based on Improved Genetic Algorithm, *J. Computer Engineering and Applications*, 2019, 55(6):218-223.
- [4] Zhu H. D. and Zhong Y., An improved simulated annealing algorithm, *J. Computer Technology and Development*, 2009, 6(19): 32-35.