Timetabling for Satisfying Professors' Requirements and Students' Desires using Genetic Algorithm

Ikuo YOSHIHARA[1] Yoshiyuki SAKAGUCHI[2] Kunihito YAMAMORI[3]

Abstract

Timetabling is to allocate the lectures in the time slot of a week, so as to fulfill various constraints. Timetable is re-created every year due to alternate professors, revision of a curriculum and so on. However, creating timetable is complicated because of a variety of constraints. This paper proposes a technique to create timetables using genetic algorithms. Timetabling problem is formulated as an optimization problem which satisfies students' desires as much as possible and professors' requirements are considered as constraints. A university timetable is created using the proposed technique and compared with an actual timetable. The experiment shows that the proposed technique can create a timetable with same or higher quality as an actual timetable.

Key Words:
Timetabling Problem, Genetic Algorithm, Optimization Problem

1 Introduction

Timetable is re-created every time by reasons of alternation of professors and revision of a curriculum and so on. Timetabling problem has various conditions, so manual fashion is very difficult and creator's responsibility becomes large. Therefore timetable creation by the computer has been expectanted.

Timetabling methods developed so far are simulated annealing[1],tabu search[1],random search[2],Monte Carlo method[3],best first search algorithm[4],genetic algorithm[1, 5] etc.

Most conventional methods solve timetabling problem as a constraints satisfaction problem. We propose a novel technique that adds newly an element "students' desires" and define the problem as an optimization problem which maximizes students' desires.

The university timetable is created using this technique and compared with an actual timetable for validation.

[1]Professor, Dept. of Computer Science and Systems Engineering
[2]Undergraduate Student, Dept. of Computer Science and Systems Engineering
[3]Associate Professor, Dept. of Computer Science and Systems Engineering

2 Timetabling problem

2.1 Conventional university timetabling problem

There is a difference type of constraints between the junior high school and high school timetabling problem and University one. For example, a junior high school and a high school do not almost have the relation between a grade and lectures and has less lectures and teachers compared with a university. On the other hand, at a university, the relation is also strong between a grade and lectures and the relation exists between faculty and department. University has many lectures and professors, so the candidates of solution are increase compared to the timetable of a junior high school and high school. Thus, university timetabling problem is difficult from junior high school and high school one.

2.2 Extended university timetabling problem

As mentioned above, in case of creating the timetable of a university, we have to consider many constraints. These constraints are categorized into two types. They are "essential conditions" and "desirable conditions". "Essential conditions" are indispensable restrictions when creating a timetable and "desirable conditions" are constraints with desirable filling as far as possible. Many of timetabling problems are solved as a restrictions sufficiency problem which only fills constraints. However, the timetabling problem in this paper is defined as the problem which solves the optimization problem of fulfilling more students' desires P, with filling constraints.

- Objective(Students' desires)(P) :

$$P = \sum_i S_i(\mathbf{x}) \rightarrow max \quad (1)$$

$\mathbf{x}$: Timetable
S_i : The i-th students' desire

- Constraints(F_c) :

Constraints are the sum of the essential conditions and the desirable conditions. In fact, weight factor is attached to each conditions. Therefore, constraints F_c are defined as follows,

$$F_c = F_c^{(e)} + F_c^{(d)}. \quad (2)$$

- Essential condition($F_c^{(e)}$) :

$$F_c^{(e)} = \sum_{i=1}^{n} C_i(\mathbf{x}) \rightarrow max \quad (3)$$

- Desire condition($F_c^{(d)}$) :

$$F_c^{(d)} = \sum_{i=1}^{m} H_i(\mathbf{x}) \rightarrow max \quad (4)$$

$\mathbf{x}$: Timetable
C_i : The i-th essential condition
H_i : The i-th desirable condition
(In this paper, refer to the Table2, 3 for $C_1, C_2 \ldots C_n, H_1, H_2 \ldots H_m$.)

2.3 Problem setting details

We compose the timetable in the first semester of our university and this is created using a genetic algorithm and comparison with an actual timetable is performed. In order to take more practical timetable, we send out questionnaires about the period to assign the lecture etc. to the professors who take charge of the lecture before the experiment. In the result, the essential conditions and the desirable conditions are determines as shown in Table 2 and Table 3. Additionally, about the students' desires are determines as shown in Table 1.

Table. 1 The Students' Desires

S_1	A compulsory lecture is needed without assigning in the 1st period.
S_2	The period which is free between lectures is reduced.
S_3	Same professors' lecture does not continue.

Table. 2 The Essential Conditions

C_1	The lecture cannot be assigned to the period which the cultural lecture is assigned.
C_2	A professor can perform only one at the same time.
C_3	The compulsory lecture of consecutive grade cannot be allocated to the same period.
C_4	Exercises and experiments are assigned in the 4th and 5th periods.
C_5	A normal lecture is not assigned in the 5th period.
C_6	The lecture cannot be assigned at the period which the professor cannot take charge of.

Table. 3 The Desirable Conditions

H_1	We allocate the lecture to the period which is wished by professor.
H_2	If possible, we do not allocate the lecture to the period which the professor does not want to put the lecture.
H_3	If possible, we allocate the lecture so that the professor may not take charge of the lecture continuously.
H_4	If possible, the lecture is not allocated in the 1st period.
H_5	If possible, the lecture is allocated so that the number of lectures may not biased by the day of the week.
H_6	If possible, the period which is free is not made between lectures.
H_7	We allocate the lecture as a priority to 2nd period and 3rd period.

3 Timetabling using Genetic Algorithms

Genetic Algorithm is the solution search algorithm imitating the process of natural evolution of a living thing[6]. It is a technique which expresses a solution as a gene and the solution improve by genetic operation as crossover, mutation and selection. Figure 1 shows the procedure of a genetic algorithm used in this paper.

3.1 Genetic Coding

The gene loci are put on the position which shows the 5th period on Friday from the 1st period on Mon-

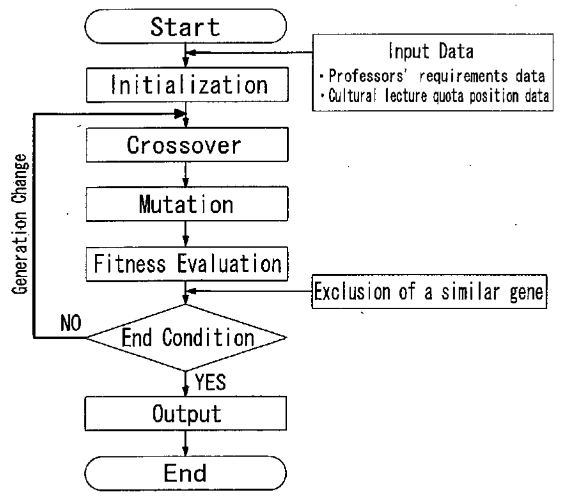

Fig. 1 Flow Chart of Genetic Algorithm used in This Paper

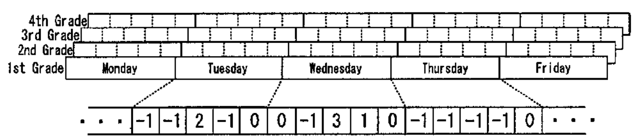

Fig. 2 Composition of a Gene

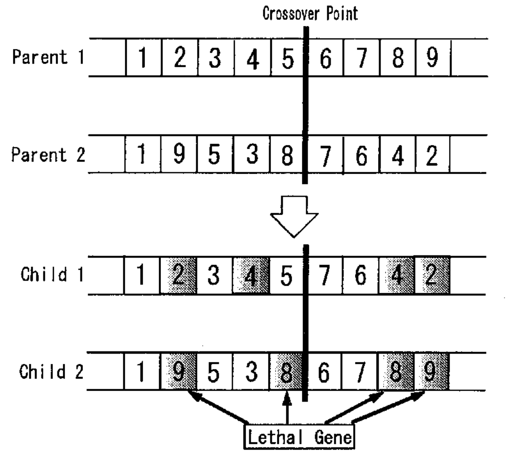

Fig. 3 Example of The Lethal Gene Generated One-Point Crossover

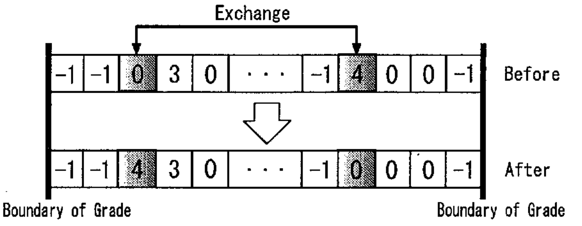

Fig. 4 Method of Mutation

day(Figure 2). The value which shows the lecture "N"(In case of N lectures in one grade: "1" - "N"), "-1" showing a cultural lecture and "0" that shows a non-assigned position. This is made from 1st grade to 4th grade.

3.2 Genetic Operators

1) Crossover

Crossover is performed for every grade and PMX (Partially Matched Crossover) is adopted. Figure 3 shows, if one-point crossover is performed, a gene with the same value occurs and the gene which should exist is lost. If same value occurs, the same lecture will performe twice at one week and when the value which should exist is lost, it becomes impossible to perform the lecture which should originally performe. Such a gene which is called lethal gene and it is not composed as a timetable. For these reasons, PMX which a lethal gene does not generate is used in this paper. Although two or more values "0" which show a non-assigned position exist in a gene, this "0" ignores at the time of crossover.

2) Mutation

As shown in Figure 4, mutation as well as crossover is performed for each grade and mutation is choosing two points at random and exchanging the value of gene loci.

3) Fitness Evaluation

The fitness is defined as follows,

$$F_c = \alpha \sum_g \sum_i c_i + \beta \sum_g \sum_i h_i, \tag{5}$$

$$fitness = P + \frac{1}{F_c + \varepsilon}. \tag{6}$$

g : grade
i : number of conditions
c_i : number of breach parts of the i-th essential condition
h_i : number of breach parts of the i-th desirable condition
α , β : weight factor
ε : constant(small value)

4) Selection

The first generation advances the generation for the purpose of fulfilling essential conditions completely. If it is attained, the generation is advanced for the purpose of fulfilling more desirable conditions, in this paper. If it fills until it can also fulfill desirable conditions, a generation is advanced for the purpose of fulfilling a students' desires as much as possible, making it not affect the degree of achievement of essential conditions and desirable conditions finally. After that, ranking selection is performed in order with the high degree of conformity from both parents and children. And only the same number as parents survives the next generation and the gene which remained disappears.

3.3 Exclusion of a similar gene

It is operated to keep a gene with the high degree of fitness in this paper. While repeating the genetic operations, there is a possibility that the similar gene may increase. The gene becomes converged to local minima. The result, global search becomes impossible, then search of the optimal solution become difficult. Consequently, in order to avoid it, a similar gene is eliminated as follows.

If change of the elite's degree of fitness is lost, it is compare with the genes which fitness resembled. In the result, when judged with it being the similar gene, other gene eliminated and replaces a random gene.

It is defined that gene A is similar to gene B when $L_{A,B}$ is smaller than some value. $L_{A,B}$ is defined as follows,

$$L_{A,B} = \sum_i d(a_i, b_i), \tag{7}$$

$$d(a_i, b_i) = \begin{cases} 1 & (a_i \neq b_i) \\ 0 & (a_i = b_i) \end{cases}. \tag{8}$$

a_i : The gene A to the i-th element

b_i : The gene B to the i-th element

4 The comparison experiment based on actual data

4.1 Experimental conditions

The experimental conditions are as follows.

- The number of population : 100
- Crossover rate : 100%
- Mutation rate : 5%
- The end condition : There is no change in 500 generation elite's fitness.

4.2 Results

- About constraints.

The timetable optimized using the genetic algorithm is Table 7 and this is the solution whose degree of fitness is highest. The actual timetable for comparison is shown in Table 6. In a total of 20 experiments, all the solutions are able to fulfill the essential condition. Comparison of the desirable conditions which are not fulfilled has become as it is shown in Table 4.

Table. 4 The Desirable Conditions Which Are Not Fulfilled

Conditions	Best solution	Actual timetable
Not yet achievement of professors' requirements. (The breach of condition H_1,H_2.)	2 places	——
A professors' continuation lecture. (The breach of condition H_3.)	0 place	1 place
Assignment to the 1st period. (The breach of condition H_4.)	1 places	1 place
Deviation of the number of lectures by the day of a week. (The breach of condition H_5.)	8 lectures	6 lectures
Free time between lectures. (The breach of condition H_6.)	1 place	2 places
Breach of the 2nd period and the 3rd period priority. (The breach of condition H_7.)	0 place	2 places
(The sum total of the number of breach.)	12	12

- About students' desires.

A students' desires which is not filled becomes as it is shown in Table 5.

Table. 5 The Students' Desires Which Are Not Fulfilled

Conditions	Best solution	Actual timetable
The compulsory lecture is assigned in the 1st period. (The breach of condition S_1.)	1 place	1 place
Free time is between lectures. (The breach of condition S_2.)	1 place	2 places
The same professors' lectures do not continue. (The breach of condition S_3.)	0 place	1 place
(The sum total of the number of breach.)	2	4

Almost all the created timetables have been created without generating an professors' continuation lecture. Compared with the result to which others were outputted, the best timetable filled professors' requirements. Additionally, about the constraints, the created timetable is the almost same level compared to the actual timetable. As

Table. 6 The Actual Timetable for University

Grade	Period	Monday		Tuesday		Wednesday		Thursday		Friday	
1st Grade	1	Cultural Lecture		Cultural Lecture		Introduction to University Study	Someone	Cultural Lecture			
	2					Cultural Lecture					
	3	Introduction to Computer Science	Yamamori			Mathematical Analysis I	Hatsukade			Introduction to Programming	Furutani
	4			Cultural Lecture						Laboratory for Information Processing I	Furutani S. Katayama
	5										
2nd Grade	1							Cultural Lecture		Analysis of Dynamic System	Sato
	2	Applied Analysis I	Yazaki			Mathematical Programming	Ikeda			Probability and Statistics	Yokomichi
	3	Graph and Network	T. Katayama	Cultural Lecture		Numerical Analysis	Yoshihara				
	4	Laboratory for Information Processing III	Yokomichi					Experiments in General Physics	Matsuda T. Katayama		
	5										
3rd Grade	1										
	2	Liner System and Signal Processing	Kono	Algorithms and Data Structures	Yoshihara	Introduction to Modern Physics	Saito	Information Network	Okazaki	Introduction to Applied Chemistry	Yui・Shiomori Hayashi
	3	Discrete-event Systems	Tomita			Compilers	Sakamoto	Fundamentals of Knowledge Processing	Tomita	Probability Theory and Information Theory	Ikeda
	4	Analysis of Measurement data	Hatsukade			Advanced Laboratory for Information Processing I	Someone	Robotics	Sato		
	5							Introduction Civil and Environmental Engineering	Someone		
4th Grade	1										
	2					Parallel Computing and Distributed Processing	Yamamori				
	3										
	4										
	5										

:The part with which a students' desires was not fulfilled.

Table. 7 The Result of Timetable for University

Grade	Period	Monday		Tuesday		Wednesday		Thursday		Friday	
1st Grade	1	Cultural Lecture		Cultural Lecture		Introduction to University Study	Someone	Cultural Lecture			
	2					Cultural Lecture					
	3	Introduction to Computer Science	Yamamori			Introduction to Programming	Furutani			Mathematical Analysis I	Hatsukade
	4			Cultural Lecture						Laboratory for Information Processing I	Furutani S. Katayama
	5										
2nd Grade	1							Cultural Lecture			
	2	Analysis of Dynamic System	Sato	Applied Analysis I	Yazaki	Mathematical Programming	Ikeda			Probability and Statistics	Yokomichi
	3	Graph and Network	T. Katayama	Cultural Lecture				Numerical Analysis	Yoshihara		
	4	Laboratory for Information Processing III	Yokomichi					Experiments in General Physics	Matsuda T. Katayama		
	5										
3rd Grade	1							Introduction Civil and Environmental Engineering	Someone		
	2	Fundamentals of Knowledge Processing	Tomita	Liner System and Signal Processing	Kono	Introduction to Modern Physics	Saito	Algorithms and Data Structures	Yoshihara	Introduction to Applied Chemistry	Yui・Shiomori Hayashi
	3	Analysis of Measurement data	Hatsukade			Compilers	Sakamoto	Discrete-event Systems	Tomita	Information Network	Okazaki
	4	Probability Theory and Information Theory	Ikeda			Advanced Laboratory for Information Processing I	Someone			Robotics	Sato
	5										
4th Grade	1										
	2	Parallel Computing and Distributed Processing	Yamamori								
	3										
	4										
	5										

:The part with which a students' desires was not fulfilled.

compared with the actually used timetable, the result has improved about a students' desires.

5 Conclusion

The technique of creating the timetable as an optimization problem was developed using genetic algorithms. And we experimented using the proposed technique. In the result, about constraints, the elite solution fulfills all essential conditions and the desirable conditions are also attained considerably. About the students' desires, the result timetable better than an actual timetable was obtained. The experiments lead us for the conclusion that the proposed method gives a timetable with same or higher quality as handmade.

Future investigation is developing the high system of the flexibility which can take into consideration multiple departments.

References

[1] H. Ueda, D. Ouchi, K. Takahashi and T. Miyahara: "Comparisons of Genetic Algorithms for Timetabling Problems", Journal of IEICE, Vol. J86-D-I, No. 9, pp. 691–701 (2003) (in Japanese).

[2] K. Fujino: "Timetabling Problem using Random", Journal of IPSJ, Vol. 5, No. 2, pp. 68–76 (1964) (in Japanese).

[3] S. Kitahara and I. Iguchi: "A Monte Carlo approach to timetabling problems", Journal of JSEI, Vol. 17, No. 3, pp. 21–31 (2002) (in Japanese).

[4] Y. Fujiwara, T. Matsunishi, H. Goto and T. Kuromaru: "Time Table Scheduling by a Local Best First Search Method with a Circumventing Function", Journal of IEICE, Vol. J76-D-II, No. 3, pp. 605–614 (1993) (in Japanese).

[5] T. Kanamori, T. Matsui and T. Okamoto: "TimeTable Generation System Using Genetic Algorithm", Technical report of IEICE, Vol. 97, No. 340, pp. 25–32 (1997) (in Japanese).

[6] H. Kitano ed.: "Genetic Algorithm", Sangyo Tosho Publishing , 1993 (in Japanese).