The 7th International Symposium on Operations Research and Its Applications (ISORA'08)
Lijiang, China, October 31–Novemver 3, 2008
Copyright © 2008 ORSC & APORC, pp. 266–273

Application to a University Course Timetabling Problem by a General Project Scheduler

Masanori Horio[1] Atsuo Suzuki[2]

[1]Department of Visual Media, Nagoya University of Arts and Sciences, Nissin, 470-0196, Japan
[2]Department of Mathematical Sciences, Nanzan University, Seto,489-0863,Japan

Abstract In this research, we solved the university timetabling problem by our general project scheduler. It is difficult to make the timetable of a university in general, because a lot of constraints exist. In the case of Nagoya University of Arts and Sciences, administrators who make timetables have to spend several days. The system has been able to obtain feasible solutions for actual data within 1.8 seconds. We found that our system was very useful as the result.

Keywords Scheduling; RCPSP; Timetabling

1 Introduction

We have developed a general-purpose project scheduler which is named SEES (Scheduling Expert Engine System)[1]. SEES can find solutions to RCPSP/τ (Resource-Constrained Project Scheduling Problem with Time-dependent resource parameters). RCPSP/τ is a generalized version of RCPSP [2]. In RCPSP/τ, both the resource consumption of a scheduling activity and the resource availability change depending on time. RCPSP/τ is a framework where various scheduling problems should be included[3]. Actually, SEES has been applied to several problems such as a scheduling of supplementary lecture class and the nurse scheduling problem[4,5]

In this research, we formulated the actual university timetabling problem for School of Human Care Studies of Nagoya University of Arts and Sciences as RCPSP/τ, and solved it by SEES.

2 Processing outline of SEES

2.1 Formulation of RCPSP/τ

In RCPSP/τ, each n resources has available amount which changes depending on unit-time. Each scheduling activities also consumes resources depending on its process time. All activities have to be executed in scheduling period without exceeding the available resource at any time. RCPSP/τ that we handle can be formulated as follows [5].

$\boldsymbol{R} = \{1,2,\ldots,r,\ldots,R\}$: set of resource constraints

$\boldsymbol{J} = \{1,2,\ldots,j,\ldots,J\}$: set of scheduling activities

$A_{r,t}$: r resource availability on unit time $[t\text{-}1,t]$

$\boldsymbol{J}_{r,t}$: set of activities which consume r at t

$u_{j,r,l}$: consumption r of j at l unit passage time from its process start time

s_j: allocation point time of j

l_j: processing time of j

$$\sum_{j \in J_{r,t}} u_{j,r,(t-s_j)} \leq A_{r,t} \ , r \in R, t = 1, 2, \cdots \tag{1}$$

$$s_j + l_j \leq s_k \ , j \prec k \tag{2}$$

(1) shows the resource constraints. (2) means the order constrains, because it expresses that k must be executed after j completed. In general, satisfying these constraints and completing all activities in target period becomes the scheduling purposes.

2.2 The basic idea of SEES

We set r available pattern which has X-axis as time and has Y-axis as available $r$amount. To similar, the r consumption pattern of j which is expressed from the processing start time is prepared (Refer to Figure 1). The outline of processing is as follows.

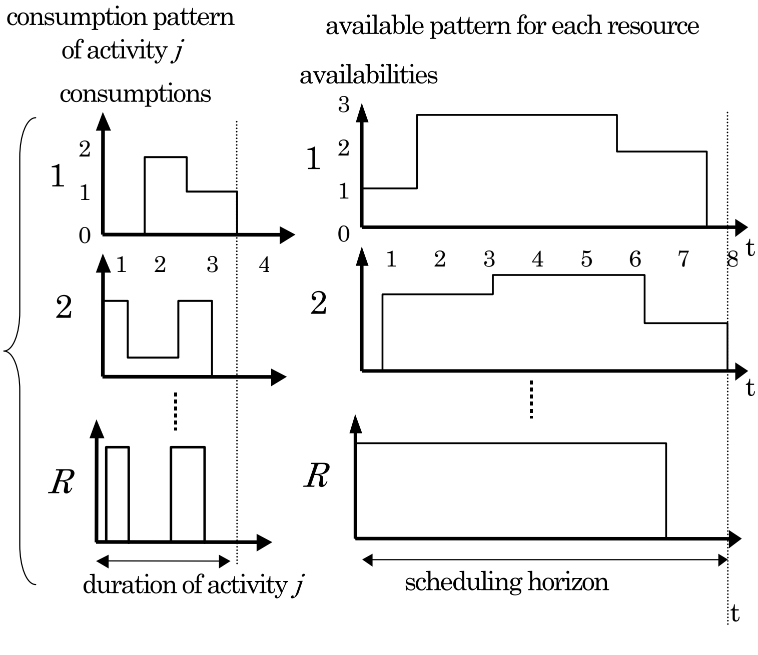

Figure 1: Examples of the pattern expression for amount of consumption and availability.

step1

Activity j is selected by using the priority rule.

step2

SEES calculates all time zones at when the r consumption pattern of j is included to r available pattern.

After applying to all r as same method , the solver calculates common parts of these zones. Those zones are satisfied of resource constraints of $j$in current states (Refer to Figure 2).

step3

If SEES cannot find common zones, scheduling condition is returned to past condition. It selects another activity, and go to step2 (refer to 2.3).

step4

The solver selects a allocating point time using scheduling evaluation rule etc, from zones of Step 2 . The r pattern is pulled out from the r available pattern at the selected time, to build a new resource environment (Refer to Figure 3).

step5

If processing ends in all j, it ends. Otherwise, to step1

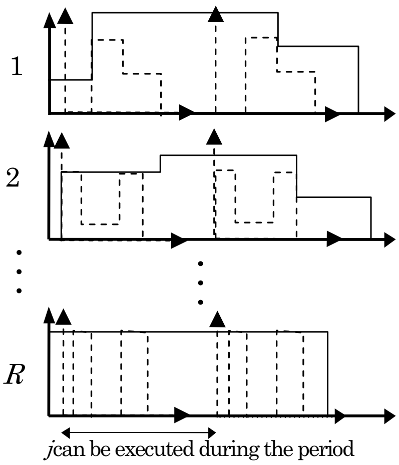

Figure 2: Calculated time periods

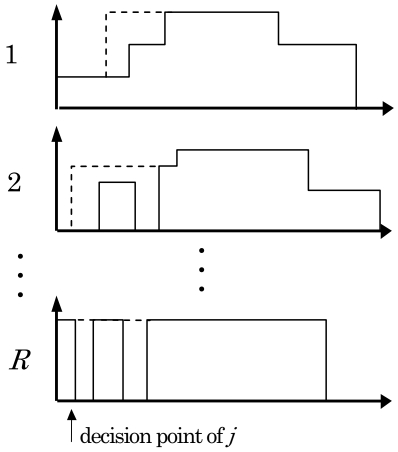

Figure 3: Calculation of the new environment

2.3 Backtracking Algorithm

In the tabu search, if various system parameters are not appropriately set, the system can not necessarily escape from the local optimal solution [6]. To avoid such the unfavorable situation, we use two lists that are tabu list and enumeration list of SSGS (Serial Scheduling Generation Scheme).

By our backtracking algorithm, the allocating stage of SSGS for the activity that cannot be scheduled will be early gradually. The outline is as follows.

We consider that an activity j_n is determined in nth (=1,2,3...,J) stage of SSGS. Then eligible set E_n ,i.e., those activities the predecessors of which are already scheduled, is computed. If an activity j_m (we call it failure-activity) cannot be determined within known better solution, then a back-track procedure would run. We use following notations and shows the algorithm

c_n : the activity which share resource ofj_n or precedes j_n , it has nearest decision point for j_n.(we call it cause-activity)

Tj_n: Tabu list that recorded activities exchanged forj_n

```
$n \leftarrow m$– 1;
while($n \neq 0$)
{
   if ( $j_m \in E_n$ and $j_m \notin Tj_n$) then
   {
      $Tj_n \leftarrow Tj_n \cup j_m$ ;
      $E_n \leftarrow E_n \setminus j_m$ ;
      returns to stage $n$ ;
      break;
   }
   $n \leftarrow$ stage number of $C_n$ ;
}
schedule $j_m$ at stage $n$ ;
```

In this algorithm, it is guaranteed that the same allocating is not repeated by the eligible set of SSGS. In addition, exchanging j_m for the same activities frequently in a local search space can be controlled through tabu list Tj_m, because the list has the activities which was exchanged for j_m .

3 Current state in the university

Nagoya University of Arts and Sciences has four faculties. School of Human Care Studies is one of these, and the school has two courses. Each student belongs to a class and the total number of classes is sixteen. The number of students, teachers and rooms are six hundred and seventy-three, fifty-five and twenty-six, respectively. The number of lectures to be scheduled is one hundred and thirty-five. Administrators should make twenty-four timetables for a semester. Because a lot of following constraints exist, they consume at least several days to make the timetables every half year[7].

• There are a lot of lectures such as seminars or laboratory classes that consist of two consecutive time periods.

• A lot of teachers have to teach various types of classes for many classes.

• The university is employing a lot of part-time teachers who can teach only on their convenient days of the week.

• Two or more lectures of a part-time teacher must be assigned consecutively in a day. Those lectures must be moved on a different day respectively, if it is impossible.

Moreover, to satisfy the requirements of other departments, the administrators have to modify the timetable many times. As a result, a lot of difficult adjustments cause following unfair conditions to many students.

• Some classes have less chances to study optional subjects. It is that the classes cannot be assigned to the optional subjects though the other classes can be assigned, because they were assigned to the other required subject at the same time.

• The administrators cannot forecast the number of the participating classes in a optional subject because many classes' condition varies every year. So the assignment of classroom of adequate size is quite difficult.

So they need the automatic timetabling system which can solve many cases in short time.

4 Considering of the problem from RCPSP

Petr Šlechta has researched Multi-resource Timetabling Problem Model that considered the student and the teacher to resources for the problem[8]. We advance this idea further, and consider this problem to be RCPSP/τ by regarding time as project period and by regarding teachers, students and classrooms as resources. Therefore, we relate this problem as follows:

j : scheduling event ,such as lecture or subject etc.

l_j : length of j (example 1, 2 or 4, etc) .

L : A continuous number allocated at a each period on Monday to Saturday. (Example in this Univ. , L=30 : 5 periods/day * 6 days).

$A_{p,t}$: If teacher p can teach at time t,1. Otherwise 0.

$A_{m,t}$: If room m can be used at time t,1. Otherwise 0.

$A_{g,t}$: The number of students in groups g that needs their time-table,G.

$u_{j,p,lj}$: If teacher p teachs$j$,1. Otherwise 0.

$u_{j,m,lj}$: If j uses room m,1. Otherwise 0.

$u_{j,g,lj}$: If students of g must attend in required subject$j$, G . If students of g may attend in optional subject$j$,1. Otherwise 0.

As a result, we can make a schedule of which two or more required subjects for G do not start at same period, because students resource lack.

Moreover, we adopt following constraints instead of (2), because in this problem constraints (2) does not exist.

$J_{d,p}$ĄFSet of j which is taught by a part-time docents p on d-day, however, elements of $\boldsymbol{J}$ are two or more.

$$\exists\,(j,k) \in J_{d,p}\,,\ |s_j - s_k| \ = \ 1 \qquad (3)$$

5 Outline of our school-timetabling system

We made the scheduling condition data by using a spreadsheet. Our developed front-end program converts the data to SEES's source file. SEES solves the converted data, and outputs the feasible solution using CSV format. Our scheduling system configurations are shown in Figure 2.

The flowing data of each part is as follows:

1. The user inputs following scheduling condition into Condition Management Sheets;

- All teachers, the using classrooms (even the input only of the size and the building location is possible), and participating student groups for each scheduling.

- The prior time period at which each teacher wishes to lecture.

- All the names, the sizes, and the building of each classroom.

- All the student group name and those properties (such as school year, the faculty, the course, the class number and total number of students).

2. The Condition Management Sheets outputs the scheduling data in CSV format.

3. The Data Converter Program changes 2 data to the original format of SEES.

4. SEES outputs the result in CSV formats.

5. The Condition Management Sheets outputs the data of 4 as each result for the student group, the teacher, and the room.

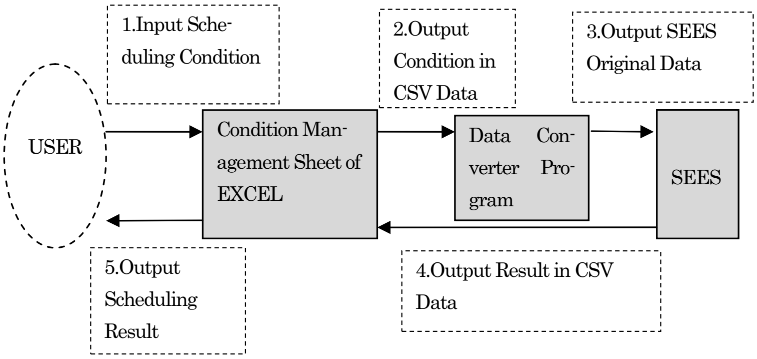
1.Input Scheduling Condition
2.Output Condition in CSV Data
3.Output SEES Original Data
USER
Condition Management Sheet of EXCEL
Data Converter Program
SEES
5.Output Scheduling Result
4.Output Result in CSV Data

Figure4: Our Scheduling System configuration

6 Result of experiment

We employ the experimental data that were used in the first term of the 2007 and 2008 fiscal year by Human Caring Faculty (Refer to Table1). In this subject, each class is divided into two groups for oral communication or a piano practice, etc,. Such groups are made in a freshman and a sophomore. The administrators should make the schedule for each group. Therefore, they needed the timetable of 24 kinds (2 graders * 2 courses * 2 classes * 2 groups + 2 graders * 2 courses * 2 classes) in the 2008 fiscal year.

Table 1: Experimental data

	2007s	2008s	Remark
subjects	102	134	* Because the fourth grader doesn’t exist in 2007, the data scale has become small.
students	20 groups	24 groups	
classrooms	26	26	
teachers	46	60	

We made ten problems in total through the following three cases for the experiment.

Case 1 (Actual situation):

Two actual problems were made from data of fiscal year 2007 and 2008.

Case 2 (Problems in which two or three optional subjects can be assigned at the same time):

Optional subjects cannot be assigned at the same time in case1. Because most optional subjects are needed for many licenses such as kindergarten teachers and school counselor that students should acquire, actual time-tables are made by such a rule. To verify our approach for optional subjects' constraints, we made two problems by using the data of the 2007 fiscal year. In each problem, we can assign two and three optional subjects at the same time in maximum.

Case 3 (Problems in which the time that a teacher can lecture changes by random):

In real situations the administrators have to schedule their time-tables again in order to adjust the possible assignments of part-time teachers. So we made six problems by using the data of the 2008 fiscal year as the following way.

1. We researched the information of six part time teachers whose assignments have to be adjusted in actual scheduling.

2. We generate their new assignment randomly.

3. If the assignment by 2 is the same day or holiday, we repeat 2.

We use a computer where OS is Windows Vista, CPU is Core2 DUO 6850(3.0GHz) and memory size is 2G byte. Table 2 shows the result.

Table 2: Results in the experiment

case	problems	results sec
Case1	The real data in fiscal year 2007	072
	The real data in fiscal year 2008	1.84
Case2	Problem that two optional subjects can be started at same time	0.06
	Problem that three optional subjects can be started at same time	0.06
Case3	Possible teaching time of teacher A Tuesday → Wednesday	2.02
	Possible teaching time of teacher B Wednesday →Friday	228.22
	Possible teaching time of teacher C Thursday → Friday	5.88
	Possible teaching time of teacher D Tuesday → Thursday	14.46
	Possible teaching time of teacher E Monday → Wednesday	6.46
	Possible teaching time of teacher F Tuesday → Monday	0.57

In the problem of case 1, our system was able to obtain the solution for the 2007 data and 2008 data in 0.72 seconds and in 1.84 seconds, respectively. From the case 2, we find that the problem becomes easy if the optional subjects are allowed to be assigned at same time. In case 3, our system was able to find the solutions in 43.9 seconds on the average. However, the problem of teacher B took much computing time compared with the other problems. It is by cause that the changed day was other teachers' priority day. Naturally, the problem that cannot be scheduled is caused by the lack of resources, such as teachers, students and classrooms etc. Fortunately, those infeasible problems did not exist in this experiment. In such the problems, we will not be able to find any feasible solution. In order to make more practicable system, the system needs the ability that can show the reasons and approach of such infeasibility.

The administrators require several days to make time-tables manually. We can conclude that making a time table of a faculty by current system is very practicable, because it is able to obtain the solution for this experiment in 33.15 seconds on the average.

References

[1] M. Horio and A. Suzuki. Development of the general scheduler using RCPSP/τ with time constraints (in Japanese). Journal of Japan Industrial Management Association, 542003

[2] S. Harthman. Project Scheduling under Limited Resources.Springer, 1999

[3] M. Horio and A. Suzuki. Modeling of the resource constrained project scheduling problem, and its application to a school timetable generation system (in Japanese). Abstracts of the 2000 Spring National Conference of ORSJ Abstract 2000

[4] M. Horio and A. Suzuki:Solution method research of the nurse scheduling problem by RCPSP (in Japanese). The 2001 Fall National Conference of ORSJ Abstracts, 2000

[5] K. Nonobe and T. Ibaraki. General-purpose scheduler -Approach by RCPSP-Ąiin Japanese). Communication of the Operations Research Society of Japan, 45 ,2000

[6] R. Klein. Scheduling of Resource-Constrained ProjectsKluwer Academic Publishers, 2000.

[7] M. Horio. Application to a University Course Timetabling Problem by a General Project Scheduler(in Japanese). Nagoya University of Arts and Sciences The Journal of Liberal Arts,2007

[8] Petr Šlechta. Decomposition and Parallelization of Multi-resource Timetabling Problems,In Burke,E.,Trick,M.,(eds.):Practice and Theory of Automated Timetabling 5,Lecture Notes in Computer Science, vol.3616, 2005