Hacettepe Journal of Mathematics and Statistics
Volume 37 (1) (2008), 41 – 55

A 0-1 INTEGER PROGRAMMING APPROACH TO A UNIVERSITY TIMETABLING PROBLEM

M Akif Bakır* and Cihan Aksop*

Received 12 : 04 : 2007 : Accepted 30 : 06 : 2008

Abstract

One of the major problems with course scheduling - a particular type of timetabling - is the difficulty that arises when trying to suitably co-ordinate lectures, students and classrooms according to a set of operational rules within a framework of certain constraints. The assignment of courses and lectures to periods and classrooms is an important administrative task that must be performed each term. The primary purpose of this paper is to solve an existing course scheduling problem. Organizing courses and lectures according to periods and availability of classrooms is a difficult course scheduling problem, which we are currently experiencing within the Department of Statistics at Gazi University. We have therefore formulated the problem as a 0-1 integer programming model. The aim of this model is to minimize the dissatisfaction of students and lecturers whilst at the same time implementing rules bounded by a set of constraints. The model produced has flexibility in terms of embracing new rules and/or criteria.

Keywords: Timetabling, Course scheduling, 0-1 integer programming.

2000 AMS Classification: 90 B 35.

1. Introduction

Scheduling involves devising a plan to carry out a number of activities over a period of time when the activities require resources which are limited, and there are various constraints and one or more objectives to be optimized [19]. In this paper we formulate a 0-1 integer programming model for the problem of course scheduling for the Department of Statistics at Gazi University, Turkey. The proposed model assigns courses offered to periods and classrooms. The objective function consists of terms that measure the dissatisfaction of lecturers and students. The case considered in this paper has many

*Gazi University, Department of Statistics, 06500 Teknikokullar, Ankara, Turkey. E-mail (M. A. Bakır) bakirakif@yahoo.com.tr (C. Aksop) entelpi@yahoo.com

interesting and complicated characteristics such as parallel programs for normal and secondary education student groups, some academic rules imposed by the Faculty Office, and constraints for non-faculty lecturers. Hence, it requires the implementation of case specific modeling logic. The model produced has flexibility in terms of embracing new rules and/or criteria that may arise in the future.

According to the literature, time scheduling problems belong to the family of NP-complete problems. Time scheduling for university courses, alongside vehicle scheduling for travel agencies and the scheduling of league fixtures, is one of the most interesting combinatorial problems. Integer programming is widely used for the solution of time scheduling problems which involve the construction of flexible schedules to meet criteria given as a set. Various algorithmic approaches for the solution of the educational timetabling problem have been suggested in the literature. Among them, graph theory, integer programming, simulated annealing, constraint logic programming and some heuristics such as tabu search are the most widely known approaches. A detailed review of the models and proposed solutions to the problem of timetabling is presented by Schaerf [18]. Besides that, the studies of Welsh and Powell [19], Balakrishnan [5], de Werra et al. [10] and Cangolovic and Schreuder [6] provide examples of graph coloring approaches.

Looking closer into the literature we should cite Lawrie [17] as being among the first to present a mathematical programming approach to timetabling problems, while Akkoyunlu [2] applied a formula that appoints courses to time slots. The model of Hultberg and Cardoso [16] is based on MIP. Akkoyunlu formulated the problem as an assignment problem, Hulberg and Cardoso as a fixed charge problem and Badri et al. [4] as a goal programming problem.

As mentioned above, all timetabling problems are NP-complete. That is to say, finding an optimum solution gets more and more difficult as the model expands in terms of size. Therefore, meta-heuristic approaches have been extensively applied in order to arrive at a reasonable solution to the problem within a reasonable period of time. Hertz [15], De Werra and Hertz [11], Costa [8], Azimi Z.N. [3] solved the university timetabling problem using Tabu search. On the other hand, while Abramson [1] used a simulated annealing algorithm for the solution of a school timetabling problem, Carrasco and Rato [7] employed a genetic algorithm.

Constrained logic programming is also a popular method, since it provides enough flexibility through the declarative language used in constrained logic programming which can be adapted to different sets of constraints. The topic is well elaborated by Gupta [14] and Deris et al. [12].

Amongst the most interesting approaches using mathematical programming are those of Dimopoulou [13] and Daskalaki, et al. [9]. The latter constitutes the core of our model, although there are a few differences stemming from specific rules and regulations in our case study. We also changed the modeling approach for a few rules common to the two cases. Moreover, while Daskalaki et al. used two different decision variables to assign consecutive periods of the same day for a course, in our case we were able to build a model with only one type of decision variable, since each course has at most two parts, theory and practice. Thus, we consider the theory and practice parts of a course as if they were different courses. This results in a model of more moderate size.

Additionally, in our model we also introduced a new criterion in order to ensure that the consecutive periods of a given course take place in the same classroom. We refer to this as “classroom consistency”.

2. Characteristics of the Department of Statistics

The Department of Statistics at Gazi University has two tracks: One is the normal, the other the secondary. Here it is necessary to give a brief explanation of the system implemented within the Department: Students are accepted into the programme and are streamed into groups (A (normal) and B (secondary)). Both groups follow the same programme, but students of Group A enter university with higher grades and are therefore not required to pay a fee. Students in Group B, however, have lower grades upon entry and must pay a fee. Any junior student enrolled in one of these two tracks is expected to graduate within eight semesters. The students of these tracks are obliged to abide by the same academic rules. Students of Group A and B cannot take the same course together. This is a rule enforced by the Higher Education Council (HEC) and consequently implemented by Gazi University. Nominally students in group B are taught in the evenings, but in case of the availability of classrooms and lecturers the HEC recommends that their courses be taught in the afternoon.

In addition to having two tracks, all courses are divided into two categories; core and elective. Core courses are obligatory for each student, whereas the electives comprise a list of courses from which a student is allowed to pick freely according to his/her credit requirements. Elective courses for a given semester are partitioned according to the credits awarded, courses with the same number of credits being "exact alternatives." A student must choose one and only one elective course from each set of exactly alternative courses. For both tracks, there are core and elective courses. Clearly, the number of students enrolled in core courses can be significantly different from those in electives, hence leading to a need for classrooms of different capacities.

Finally, the courses are divided into three categories in a different way. The majority of courses are offered by lecturers of the Department of Statistic or lecturers of other departments of the same Faculty such as Mathematics and History. Twenty eight different courses fall into this category. In the second category, there are four courses, for example Turkish Language, which are mandatory for all students of the university. The third category is called "service", and these courses are offered by non-faculty lecturers; there are five such courses, for example Economics and Finance.

Twenty six lecturers from the Departments of Statistics and Mathematics, and five lecturers from other faculties offer courses.

In the faculty building, there are six rooms allotted for usage. Most courses are held in any of the four classrooms, which have different capacities. From the remaining two, the first (called Z-10) is used to hold courses appointed by the Faculty Office and the second (called LAB-1) to hold courses requiring special equipment like computers. If a course is taken only by a few students and does not require special equipment, it can be conducted in any classroom. This is an important assumption that is used to reduce the size of our model.

The total number of periods in a day is accepted to be 14, between 8:30 and 21:05. Each period is 45 minutes long and there are 10-minute breaks between periods.

3. Basic Rules for Designing a Timetable

Any timetable suitable for the Department of Statistics has to meet certain criteria. These are:

3.1. No Overlapping. There are three cases for this criterion, regarding (i) the students, (ii) the lecturers and (iii) the classrooms. The first one requires that a student group should not be assigned more than one course for a given period, unless they are "exact alternatives". For a lecturer, if he offers more than one course, they should not overlap. For the last case, there should not be more than one course taking place in a classroom at the same time.

3.2. Meeting Credit Requirements. This is the set of constraints which ensures that the students can take the number of courses for the semester, as dictated by the Department of Statistics.

3.3. Consecutiveness. The periods of any course should be consecutive, as long as it requires no more than three periods per week. If the course is composed of theory and practice/recitation sessions, we treat them as two separate courses (theory and practice), instead of dealing only with one course. If a course requires more than 3 periods per week then it can be divided in two sessions. For example: a course taking 5 hours could be divided thus: One session composed of 2 periods on a Monday and another session of 3 periods on a Wednesday.

3.4. Classroom Consistency. All periods of a given course should be held in the same classroom. Note that for all courses, a session consists of at least two periods, which should be taught consecutively and in the same classroom.

3.5. Theory-Practice Precedence. If a course has theory and practice requirements, then theory should be held before practice; yet they both should be held in the same week.

3.6. Courses Appointed by the Faculty Office. Some courses are scheduled by the Dean's Office. Hence, the time-table of the Department should enable the students affected to take them.

3.7. Time Constraints. No lecturer can offer more than two periods on any weekday after 17:35 for the secondary classes, while normal classes cannot be scheduled for periods after 17:35.

3.8. Non-Faculty Lecturers. As far as possible the number of periods between Group A and B sessions should be kept to a minimum for the non-faculty lecturer's courses since non-faculty buildings are not in the same campus. As a matter of course the same course for Group A and B is given by the same lecturer.

3.9. Failed Students. Failed students refer to students who have failed a course taken during a previous year. It is desired, if at all possible, that courses with failed students do not overlap with courses belonging to subsequent semesters.

4. 0-1 Integer Programming Model for an Optimum Time-Table

For our model we use the first seven rules above as constraints. We will develop the objective function to determine the total non-satisfaction of the students and the lecturers, and at the same time we will include the 8th and 9th rules in the objective function.

4.1. Definition of the required sets.

I	$=\{1,\dots,5\}$	Set of days in a week in which the courses are offered
J	$=\{1,\dots,14\}$	Set of time slots in a day
K	$=\{1,\dots,8\}$	Sets of student groups. The first 4 refer to Group A and the remaining 4 to Group B
L	$=\{1,\dots,28\}$	Set of lecturers
M	$=\{1,\dots,95\}$	Set of courses. Note that if a class is divided in two parts, theory and practice, these are defined as separate elements of the set. Similarly, the same course offered for Group A and Group B is declared with separate elements
N	$=\{1,2,3,4\}$	Set of classrooms

Building a model using the sets above results in approximately 6 million decision variables, which obviously creates difficulties. In order to economize but not to loose insight, new sets will be used in place of the sets K, L, M and N. These are defined below.

K_l	Set of students for which lecturer l offers at least one course
K_m	Set of students which take course m
L_k	Set of lecturers offering at least one course for the group of students k
L_{km}	Set of lecturers teaching course m for the group of students k
L_{out}	Set of lecturers not from the Faculty of Science and Arts
M_{kl}	Set of courses offered by lecturer l for the group of students k
M_{kln}	Set of courses offered by lecturer l for the group of students k that can be held in classroom n
$M_{k,l}^{\text{core}}$	Set of core courses offered by lecturer l for the group of students k that can be held in classroom n
$M_{k,l}^{\text{elec}}$	Set of elective courses offered by lecturer l for the group of students k can be held in classroom n
M_{pra}	Set of practice courses
N_m	Set of classrooms that may accommodate the students registered for course m

4.2. The Decision Variables of the Model. The model is constructed with two sets of decision variables defined below:

$$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, k \in K,\ \forall\, l \in L_k,\ \forall\, m \in M_{kl},\ \forall\, n \in N_m$$

$$x_{i,j,k,l,m,n} = \begin{cases} 1, & \text{if course } m \text{ taught by teacher } l \text{ to the group of student } k \\ & \text{is assigned to the } j\text{th time slot of day } i \text{ in classroom } n \\ 0, & \text{otherwise} \end{cases}$$

$$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, k \in K - \{4,8\}$$

$$z_{i,j,k} = \begin{cases} 1, & \text{when core courses of the student group } k \text{ overlap with} \\ & \text{courses taught to the upper class in the } j\text{-th period of} \\ & \text{day } i \\ 0, & \text{otherwise} \end{cases}$$

The variables $z_{i,j,k}$ are used to handle the situation with the failed students explained in Subsection 3.9.

4.3. The Constraints of the model. Having defined the first seven basic rules for timetabling we can now develop the mathematical form of these rules.

4.3.1. *The courses of the student groups cannot overlap with their exact alternatives.* Within the Department of Statistics, the elective courses are split into two or three-credit courses. As mentioned earlier, the two-credit courses for a given semester are regarded as "exact alternatives." A student cannot take more than one of these during a given semester, but can of course take different ones during different semesters. The same is true for the three-credit courses, and in this way the students can take the number of two and three-credit elective courses specified by the Department of Statistics.

The following constraints refer to the situation discussed above:

$$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, k \in K$$
$$\sum_{\substack{m \in M_k \\ \text{alternative}(m)=0}} \ \sum_{l \in L_{km}} \ \sum_{n \in N_m} x_{i,j,k,l,m,n} \leq 1$$

$$\forall i \in I, \forall j \in J, \forall k \in K,$$
$$\forall\ m_1, m_2 \in M_k, \text{alternative}\,(m_1) \neq \text{alternative}\,(m_2)$$
$$\sum_{l \in L_{km_1}} \ \sum_{n \in N_{m_1}} x_{i,j,k,l,m_1,n} + \sum_{l \in L_{m_2}} \ \sum_{n \in N_{m_2}} x_{i,j,k,l,m_2,n} \leq 1$$

Here, alternative (m) is an indicator function for exact alternative courses. Hence, alternative $(m) = 0$ refers to core courses and it means that there is no exact alternative for course m; otherwise alternative $(m_1) =$ alternative (m_2) means that the courses m_1 and m_2 are exact alternatives of each other.

4.3.2. *Lecturers and Overlapping.* Each lecturer can be assigned to at most one course for any given period.

$$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, l \in L$$
$$\sum_{k \in K_l} \ \sum_{m \in M_{kl}} \ \sum_{n \in N_m} x_{i,j,k,l,m,n} \leq 1$$

4.3.3. *Classrooms and Overlapping.* Each classroom can hold at most one course during any period.

$$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, n \in N$$
$$\sum_{k \in K} \ \sum_{l \in L_k} \ \sum_{m \in M_{kln}} x_{i,j,k,l,m,n} \leq 1$$

4.3.4. *Meeting the Credit Requirement.* Each student must fulfill a minimum credit requirement for each semester in order to graduate. The number of credits is equivalent to the number of periods per week. In this case, credit (k) is a function giving the total number of credits of the elective courses which must be taken by the student group k.

$$\forall\, k \in K$$
$$\sum_{i \in I} \sum_{j \in J} \sum_{l \in L_k} \ \sum_{m \in M_{kl}^{\text{elec}}} \ \sum_{n \in N_m} x_{i,j,k,l,m,n} = \text{credit}\,(k).$$

In order to ensure that all courses are accommodated in the timetable, the following constraint must be added.

$$\forall\, m \in M$$
$$\sum_{i \in I} \sum_{j \in J} \sum_{k \in K_m} \ \sum_{l \in L_{km}} \ \sum_{n \in N_m} x_{i,j,k,l,m,n} = h\,(m)$$

Here, $h\,(m)$ is a function that gives the number of periods required for the course m.

4.3.5. *Consecutiveness.* Each and every course should be taught in consecutive periods that match its number of credits. To fulfill this requirement, we need to set the following three constraints. The first one ensures the consecutiveness for a course, if the first period of that course is assigned to the first time slot of a day. The third one plays the same role if the last period of a course is assigned to the last time slot of a day. The second constraint fulfills consecutiveness for the remaining situations.

$$\forall i \in I,\ \forall k \in K,\ \forall l \in L_k,\ \forall m \in M_k,\ \forall n \in N_m,\ \forall t \in \{2, 3, \ldots, h(m)\}$$
$$x_{i,1,k,l,m,n} - x_{i,t,k,l,m,n} \leq 0$$

$$\forall i \in I,\ \forall k \in K,\ \forall l \in L_k,\ \forall m \in M_{kl},\ \forall n \in N_m,$$
$$\forall p \in \{1, 2, \ldots, 13\},\ \forall t \in \{2, 3, \ldots, h(m)\},\ \text{and } t + p \leq 14$$
$$-x_{i,j,k,l,m,n} + x_{i,p+1,k,l,m,n} - x_{i,p+t,k,l,m,n} \leq 0$$

$$\forall i \in I,\ \forall k \in K,\ \forall l \in L_k,\ \forall m \in M_{kl},\ \forall n \in N_m,\ \forall t \in \{2, \ldots, h(m)\}$$
$$x_{i,14,k,l,m,n} - x_{i,14-t,k,l,m,n} \leq 0$$

4.3.6. *Consistency of Classrooms.* All periods of a course in a day should be held in the same classroom.

$$\forall j \in J,\ \forall m \in M,\ \forall n \in N_m$$
$$h(m) * \sum_{i \in I} \sum_{k \in K_m} \sum_{l \in L_{km}} x_{i,j,k,l,m,n} \leq \sum_{i \in I} \sum_{p \in J} \sum_{k \in K_m} \sum_{l \in L_{km}} x_{i,p,k,l,m,n}$$

4.3.7. *Theory-Practice Precedence.* If a course requires theory and practice, then within a given week, the theory part should be scheduled before the practice. Here, $tu(m)$ is the function that gives an index for the theory part of course m within the set M of courses.

$$\forall m \in M^{\text{pra}}$$
$$\sum_{i \in I} \sum_{j \in J} \sum_{k \in K_m} \sum_{l \in L_{km}} \sum_{n \in N_m} i * x_{i,j,k,l,m,n} / h(m) >$$
$$\sum_{i \in I} \sum_{j \in J} \sum_{k \in K_{tu(m)}} \sum_{n \in N_{tu(m)}} i * x_{i,j,k,l,tu(m),n} / h(tu(m))$$

4.3.8. *Courses assigned by the Faculty Office should fit into the schedule.* To meet this requirement, the student groups taking the courses assigned by the Faculty Office should have no other course during the pre-assigned periods for those courses.

$$\forall p \in \{8, 9\}$$
$$\sum_{m \in M_1} \sum_{l \in L_{1m}} \sum_{n \in N_m} x_{3,p,1,l,m,n} = 0$$

$$\forall p \in \{9, 10\}$$
$$\sum_{m \in M_5} \sum_{l \in L_{1m}} \sum_{n \in N_m} x_{2,p,5,l,m,n} = 0$$

$$\forall p \in \{1, 2\}$$
$$\sum_{m \in M_1} \sum_{l \in L_{1m}} \sum_{n \in N_m} x_{5,p,1,l,m,n} = 0$$

$$\forall p \in \{6, 7\}$$
$$\sum_{m \in M_5} \sum_{l \in L_{5m}} \sum_{n \in N_m} x_{3,p,5,l,m,n} = 0$$

$$\forall p \in \{1, 2, 6, 7\}$$
$$\sum_{m \in M_1} \sum_{l \in L_{1m}} \sum_{n \in N_m} x_{3,p,1,l,m,n} = 0$$

$\forall\, p \in \{3, 4, 8, 9\}$

$$\sum_{m\in M_5}\ \sum_{l\in L_{5m}}\ \sum_{n\in N_m} x_{3,p,5,l,m,n} = 0$$

$\forall\, p \in \{3, 4\}$

$$\sum_{m\in M_1}\ \sum_{l\in L_{1m}}\ \sum_{n\in N_m} x_{5,p,1,l,m,n} = 0$$

$\forall\, p \in \{9, 10\}$

$$\sum_{m\in M_5}\ \sum_{l\in L_{5m}}\ \sum_{n\in N_m} x_{4,p,5,l,m,n} = 0$$

4.3.9. *Time Rules and Regulations.* According to the university rules and regulations, no courses can be assigned after 17:35 for students in the normal group. Moreover, no lecturer can have more than two periods for students of the secondary group after 17:35.

$\forall\, l \in L$

$$\sum_{i\in I}\ \sum_{\substack{j\in J\\ j\geq 11}}\ \sum_{\substack{k\in K_l\\ k\geq 5}}\ \sum_{m\in M_{kl}}\ \sum_{n\in N_m} x_{i,j,k,l,m,n} \leq 2$$

$\forall\, k \leq 4$

$$\sum_{i\in I}\ \sum_{\substack{j\in J\\ j\geq 11}}\ \sum_{l\in L_k}\ \sum_{m\in M_{kl}}\ \sum_{n\in N_m} x_{i,j,k,l,m,n} = 0$$

4.3.10. *Consideration for failed students.* The following constraint assigns a 0 or 1 value to the variable $z_{i,j,k}$ defined in subsection 4.2. In case core courses of failed students overlap with their current courses, the variable $z_{i,j,k}$ negatively affects the objective function in the form of a penalty.

$\forall\, i \in I,\ \forall\, j \in J,\ \forall\, k \in K - \{4, 8\}$

$$\sum_{l\in L_k}\ \sum_{m\in M_{kl}^{\text{core}}}\ \sum_{n\in N_m} x_{i,j,k,l,m,n} + \sum_{l\in L_{k+1}}\ \sum_{m\in M_{k+1,l}^{\text{core}}}\ \sum_{n\in N_m} x_{i,j,k+1,l,m,n} \leq 1 + z_{i,j,k}$$

The 4 and 8 in $\{4, 8\}$ refers to the final year students for Group A and B, respectively. There is no further semester following the final year. That is why we have extracted these final year students.

4.4. Objective Function. The objective function of the model is constructed to minimize total dissatisfaction, and consists of five terms. The total dissatisfaction can be defined as:

$$\begin{aligned}\{\text{ total dissatisfaction}\} \quad &= (\text{dissatisfaction of students in the normal group})\\ &+ (\text{dissatisfaction of students in the secondary group})\\ &+ (\text{dissatisfaction of lecturers})\\ &+ (\text{dissatisfaction of non-faculty lecturers})\\ &+ (\text{dissatisfaction of failed students})\end{aligned}$$

Here, the dissatisfaction of the normal group students is measured by the classes offered during evening hours; that of the secondary group by the classes offered in morning hours. At the beginning of this work, we asked the lecturers to specify the days and periods that they do not want to lecture. In line with this, dissatisfaction of lecturers stems from assignments of days and periods when they do not want to lecture. The dissatisfaction of a non-faculty lecturer is measured by the assignment of courses on different days and by a long spare-time between periods. The mathematical format of the objective function is:

$$\begin{aligned}
\min z = \; & \sum_{j\in J}\sum_{k=1}^{4} a_{1j}\left(\sum_{i\in I}\sum_{l\in L_k}\sum_{m\in M_{kl}}\sum_{n\in N_m} x_{i,j,k,l,m,n}\right)\\
& + \sum_{j\in J}\sum_{k=5}^{8} a_{2j}\left(\sum_{i\in I}\sum_{l\in L_k}\sum_{m\in M_{kl}}\sum_{n\in N_m} x_{i,j,k,l,m,n}\right)\\
& + \sum_{i\in I}\sum_{j\in J}\sum_{l\in L} b_{i,j,k} * \left(\sum_{k\in K_l}\sum_{m\in M_{kl}}\sum_{n\in N_m} x_{i,j,k,l,m,n}\right)\\
& + \sum_{l\in L_{\text{out}}}\sum_{\substack{k\in K_l\\ k\le 4}}\sum_{m\in M_{kl}} 10 * \left(\sum_{i\in I}\sum_{j\in J}\sum_{n\in N_m} i * x_{i,j,k,l,m,n}\right)\\
& - \sum_{l\in L_{\text{out}}}\sum_{\substack{k\in K_l\\ k\ge 5}}\sum_{m\in M_{kl}} 10 * \left(\sum_{i\in I}\sum_{j\in J}\sum_{n\in N_m} i * x_{i,j,k,l,m,n}\right)\\
& + \sum_{l\in L_{\text{out}}}\sum_{\substack{k\in K_l\\ k\le 4}}\sum_{m\in M_{kl}} 5 * \left(\sum_{i\in I}\sum_{j\in J}\sum_{n\in N_m} j * x_{i,j,k,l,m,n}\right)\\
& - \sum_{l\in L_{\text{out}}}\sum_{\substack{k\in K_l\\ k\ge 5}}\sum_{m\in M_{kl}} 5 * \left(\sum_{i\in I}\sum_{j\in J}\sum_{n\in N_m} j * x_{i,j,k,l,m,n}\right)\\
& + 50 * \sum_{m\in M}\sum_{i\in I}\sum_{j\in J}\sum_{k\in K_m}\sum_{l\in L_{km}}\sum_{n\in N_m} \frac{i*x_{i,j,k,l,m,n}}{h(m)}\\
& - 50 * \sum_{m\in M}\sum_{i\in I}\sum_{j\in J}\sum_{k\in K_{tu(m)}}\sum_{l\in L_{ktu(m)}}\sum_{n\in N_{tu(m)}} \frac{i*x_{i,j,k,l,tu(m),n}}{h(tu(m))}\\
& + 100 * \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} z_{i,j,k}
\end{aligned}$$

In addition, we should add the criteria below to ensure the satisfaction of the non-faculty lecturers:

$\forall\, l \in L^{\text{out}},\ \forall\, m \in M_{kl}$

$$\sum_{i\in I}\sum_{j\in J}\sum_{\substack{k\in K_l\\ k\le 4}}\sum_{n\in N_m} i * x_{i,j,k,l,m,n} \le \sum_{i\in I}\sum_{j\in J}\sum_{\substack{k\in K_l\\ k\ge 5}}\sum_{n\in N_m} i * x_{i,j,k,l,m,n}$$

$$\sum_{i\in I}\sum_{j\in J}\sum_{\substack{k\in K_l\\ k\le 4}}\sum_{n\in N_m} j * x_{i,j,k,l,m,n} \le \sum_{i\in I}\sum_{j\in J}\sum_{\substack{k\in K_l\\ k\ge 5}}\sum_{n\in N_m} j * x_{i,j,k,l,m,n}$$

Here, a_{1j} and a_{2j} are defined as the willingness of the students in Group A and Group B, respectively, to have courses consecutively at period j. These coefficients are defined by the dissatisfaction functions below.

Figure 1. The function that generates the coefficients a_{1j}

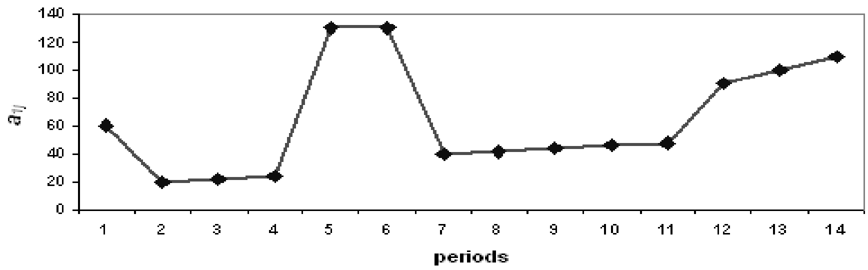
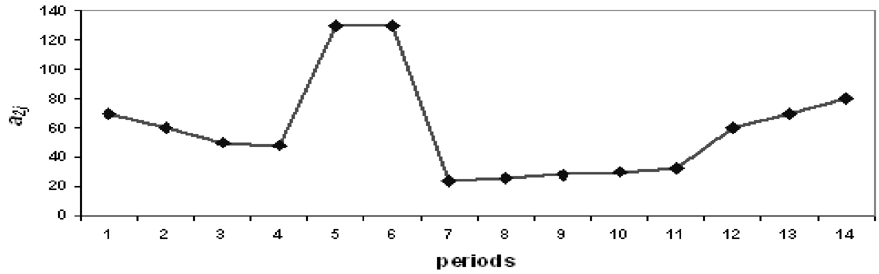


Figure 2. The function that generates the coefficients a_{2j}



The dissatisfaction coefficients for the lecturers, $b_{i,j,k}$, are defined as:

$$b_{i,j,l} = \begin{cases} 15, & \text{if lecturer } l \text{ is not willing to lecture at the } j\text{th period of day } i \\ 0, & \text{otherwise} \end{cases}$$

The value of $b_{i,j,k}$ is chosen arbitrary so as to give a higher importance to a lecturers' dissatisfaction than to a students'.

5. Reducing the Size of the Model

Since our model as such is still very large, we deflated it without causing any crucial harm by ignoring the classroom index for the decision variable x. To avoid overlapping in classrooms, we should add the following constraints to the model:

$$\forall i \in I,\ \forall j \in J$$
$$\sum_{k\in K} \sum_{l\in L_k} \sum_{\substack{m\in M_{kl} \\ n(m)\neq 0 \\ n(m)\neq 4}} x_{i,j,k,l,m} \leq 4$$

$$\forall i \in I,\ \forall j \in J$$
$$\sum_{k\in K} \sum_{l\in L_k} \sum_{\substack{m\in M_{kl} \\ n(m)\neq 4 \\ n(m)\neq 0}} n\left(m\right) * x_{i,j,k,l,m} \leq 8$$

$$\forall i \in I,\ \forall j \in J$$
$$\sum_{k\in K} \sum_{l\in L_k} \sum_{\substack{m\in M_{kl} \\ n(m)=4}} x_{i,j,k,l,m} \leq 1$$

The first constraint limits the assignment of courses at any period to 4, the total number of classrooms that may be used for regular courses. The second constraint replaces the capacity requirements for the classrooms. Lastly, we can make assignments to classroom LAB-1 through the third set of constraints. There is no need to include the assignment to classroom Z-10 in the above constraint, since the Faculty Office makes the assignment.

The $n\left(m\right)$ function in the constraints above is defined as:

$$n\left(m\right) = \begin{cases} 0, & \text{course } m \text{ can be taught in classroon Z-10} \\ 1, & \text{course } m \text{ can be taught in classrooms 210, 211, 205 or 206} \\ 3, & \text{course } m \text{ can be taught in classrooms 210 or 211} \\ 4, & \text{course } m \text{ can be taught only in LAB-1} \end{cases}$$

After adding these new constraints to the model, ignoring all classroom indices of x and deleting all summations that refer to these indices, we obtained a new model with a total of 7610 decision variables and 34978 constraints. This is significantly more moderate in size than the first version of the model in which the variable x had 5 indices.

6. Solution of the problem and an optimum schedule

The above model formulated for the Department of Statistics is a large scale 0-1 integer programming problem. The LINGO 8.0 software program on a Pentium 4, 2.6 MHz and 256 MB Ram computer platform was used for the solution. The branch and bound algorithm was used to solve the problem. However, since Karmarkar's interior point algorithm is faster than the simplex algorithm, it was preferred when solving the linear programming part of the problem. It took 5 minutes to reach a feasible solution. It took a long time to obtain an exact optimum because the computer platform was not adequate for solving such a large combinatorial problem. Usually, such problems require computers with parallel processing to obtain a solution within a reasonable period of time.

The solution of the problem provided an optimum timetable that minimizes the total dissatisfaction function, while meeting the rules imposed on the model. Since our aim is to illustrate the main points in the solution, it will suffice to present the timetable for the final year of both the normal and the secondary groups, and a lecturer's schedule as an example.

Table 1. The course schedule for the final year student groups

The normal group of students

	Monday	Tuesday	Wednesday	Thursday	Friday
1	SQC.T210	LM.T 205	MVS.T 206	MVS.P 206	IP.P 205
2	SQC.T210	LM.T 205	MVS.T 206	MVS.P 206	IP.P 205
3	SP.T 206	CDS.T 205	OS 205	LM.P 206	CDA.P205
4	SP.T 206	CDA.T 205	OS 205	LM.P 206	CDA.P205
5					
6					
7	IP.T 206	SP.P 205	SQC.P 211		
8	IP.T 206	SP.P 205	SQC.P 211		
9	SC 206				
10	SC 206				
11					
12					
13					
14					

The secondary group of students

	Monday	Tuesday	Wednesday	Thursday	Friday
1					
2					
3	OS 206				
4	OS 206				
5					
6					
7	CDA.T 205	MVS.T 206	ID 206	LM.U 206	IP.U 206
8	CDA.T 205	MVS.T 206	ID 206	LM.U 206	IP.U 206
9	SQC.T 205	LM.T 206	CDA.U 205	SP.U 205	SQC.U206
10	SQC.T 205	LM.T 206	CDA.U 205	SP.U 205	SQC.U206
11	IP.T 206	SP.T 206	MVS.U 206		
12	IP.T 206	SP.T 206	MVS.U 206		

Table 2. A lecturer's schedule

	Monday	Tuesday	Wednesday	Thursday	Friday
1		LM-N-T 205	MS-N-P 210		
2		LM-N-T 205	MS-N-P 210		
3				LM-N-P 206	
4		MS-N-T 210		LM-N-P 206	
5		MS-N-T 210			
6		MS-N-T 210			
7			MS-S-T 210	LM-S-P 206	
8			MS-S-T 210	LM-S-P 206	
9		LM-S-T 206	MS-S-T 210		
10		LM-S-T 206			
11				MS-S-P 210	
12				MS-S-P 210	
13					
14					

Table 3. The occupancy of classroom 210

	Monday	Tuesday	Wednesday	Thursday	Friday
1	XXXXX	XXXXX		XXXXX	XXXXX
2	XXXXX	XXXXX		XXXXX	XXXXX
3	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
4	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
5		XXXXX		XXXXX	XXXXX
6	XXXXX	XXXXX		XXXXX	XXXXX
7	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
8	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
9	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
10	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
11	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
12	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
13					
14					

We can summarize some characteristics of the solution which are consistent with the rules as below:

(1) From the tables above we can see that both Group A and Group B schedules are composed of courses with sessions that are more or less subsequent. Such a non-stop-like schedule is preferred by the students, because they want to have their free hours en bloc. Spare time diffused throughout the day is not fertile.
(2) All sessions are assigned without overlapping.
(3) All course blocks enable subsequent sessions without changing classrooms.
(4) The optimal solution assigned only a minimum number of hours during the periods that had been defined as undesirable by the lecturers.
(5) It provides an optimum usage of the limited number of classrooms, so as to leave the classrooms empty as little as possible. (See. Table 4, a clear cell means that the classroom is empty during the corresponding periods)
(6) The courses for normal students are assigned mostly to morning periods while those for the secondary students are assigned mostly to afternoon periods.
(7) The theory session of any course is assigned before the corresponding practice session of that course within the same week.
(8) No lecturer is assigned more than two periods for the secondary group of students after 17:35.
(9) An optimum timetable that also meets all other rules has been created.

7. Conclusions

An optimum course scheduling timetable for the Department of Statistics at Gazi University is achieved by using a 0-1 integer programming model, in which both students' and lecturers' dissatisfaction is minimized. Our case study, presented in this article, carries many interesting features. Therefore, although we used the model in Daskalaki et al. [9] as a guide we had to proceed with several adaptations and extra formulations emanating from our specific requirements.

The problem was solved successfully for the spring semester. The flexibility of the model permits its extension to the fall semester, and its use for other Departments as well. However, the size of the problem creates certain difficulties in terms of attaining an exact optimum solution. It is therefore necessary to find a way of decreasing CPU time. In order to achieve this, further research is needed into the use of heuristic algorithms and a reduction in the number of variables.

We developed a model for our case-study and solved only one instance. Future versions of this model are planned to include optimal schedules for irregular students (double major and minor students), to reduce further the idle times between sessions as well as the number of days between theory and practice sessions. Moreover, we plan to extend our study in order to cover the whole Faculty and to develop a flexible and user-friendly prototype that includes modules for data inputs and outputs, and a database for storing information.

Acknowledgements We would like to thank the Associate Editor and anonymous referee for very helpful comments that have led to a substantial improvement of the paper.

References

[1] Abramson, D. *Constructing school timetables using simulated annealing: sequential and parallel algorithms*, Management Science **37**, 98–113, 1991.

[2] Akkoyunlu, E. A. *A linear algorithm for computing the optimum university timetable*, The Computer Journal **16** (4), 347–350, 1973.

[3] Azimi, Z. N. *Hybrid heuristics for examination timetabling problem*, Applied Mathematics and Computation **163**, 705–733, 2005.

[4] Badri, M. A., Davis, D. L., Davis, D. F. and Hollingsworth, J. *A multi-objective course scheduling Model: combining faculty preferences for courses and times*, Computers and Operations Research **25** (4), 303–316, 1998.

[5] Balakrishnan, N. *Examination scheduling : A computerized application*, International Journal of Management Science **19**, 37–41, 1991.

[6] Cangolovic, M. and Schreuder, J. A. M. *Exact coloring algorithms for weighed graphs applied to timetabling problems with lectures of different lenghths*, European Journal of Operational Research **51**, 248–258, 1991.

[7] Carrasco, M. P. and Rato, M. V. *A multiobjective genetic algorithm for the class/teacher timetabling problem*, In Burke E, Erben W. (Eds.) (Practice and theory of timetabling III. Lecture Notes in Computer Science. Vol. 2079, Springer-Verlag, 2001), 3–17.

[8] Costa, D. *A tabu search algorithm for computing an operational timetable*, European Journal of Operational Research **76**, 98–110, 1994.

[9] Daskalaki, S., Birbas T. and Housos, E. *An integer programming formulation for a case study in university timetabling*, European Journal of Operational Research, 117–135, 2004.

[10] De Werra, D., Mahadev, N. V. R. and Peled, U. N. *Edge chromatic scheduling with simultaneous constraints*, SIAM Journal on Discrete Mathematics **6**, 175–182, 1993.

[11] De Werra, D. and Hertz, A. *Tabu search techniques: a tutorial and an application to neural networks*, OR Spectrum **11**, 131–141, 1989.

[12] Deris, S. B., Ormatu, S., Ohta, H. and Samat, P. A. B. D. *University timetabling by constraint-based reasoning: a case study*, Journal of the Operational Research Society **48**, 1178–1190, 1997.

[13] Dimopoulou, M. and Milliotis, P. *Implementation of a university course and examination timetabling system*, European Journal of Operational Research **130**, 2–213, 2001.

[14] Gupta, G. *Practical Aspects of Declarative Languages* (LNCS 1551, Springer-Verlag 1999).

[15] Hertz, A. *Find a feasible course schedule using tabu search*, Discrete Applied Mathematics **35**, 255–270, 1992.

[16] Hultberg, T. H. and Cardoso, D. M. *The teacher assignment problem: a special case of fixed charge transportation problem*, European Journal of Operational Research **101**, 463–473, 1997.

[17] Lawrie, N. L. *An integer linear programming model of a school timetabling problem*, The Computer Journal **12**, 307–316, 1969.

[18] Schaerf, A. *A Survey of automated timetabling*, Artificial Intelligence Review **13**, 87–127, 1999.

[19] Welsh, D. J. A. and Powell, M. B. *An upper bound to the choromatic number of a graph and its application to timetabling problem*, The Computer Journal **10**, 85–86, 1967.