

An Integer Programming Model for a Complex University Timetabling Problem: A Case Study

R. A. Ranga Prabodanie*

Department of Industrial Management, Faculty of Applied Sciences, Wayamba University of Sri Lanka, Kuliypitiya, Sri Lanka

(Received: May 26, 2016 / Revised: October 7, 2016 / Accepted: November 14, 2016)

ABSTRACT

A binary integer programming model is proposed for a complex timetabling problem in a university faculty which conducts various degree programs. The decision variables are defined with fewer dimensions to economize the model size of large scale problems and to improve modeling efficiency. Binary matrices are used to incorporate the relationships between the courses and students, and the courses and teachers. The model includes generally applicable constraints such as completeness, uniqueness, and consecutiveness; and case specific constraints. The model was coded and solved using Open Solver which is an open-source optimizer available as an Excel add-in. The results indicate that complicated timetabling problems with large numbers of courses and student groups can be formulated more efficiently with fewer numbers of variables and constraints using the proposed modeling framework. The model could effectively generate timetables with a significantly lower number of work hours per week compared to currently used timetables. The model results indicate that the particular timetabling problem is bounded by the student overlaps, and both human and physical resource constraints are insignificant.

Keywords: University, Timetabling, Integer Programming, Open Solver

* Corresponding Author, E-mail: ranga@wya.ac.lk

1. INTRODUCTION

Academic timetabling is a well-known scheduling problem discussed in operations research. Integer Programming (IP) models of timetabling problems belong to the family of NP-hard combinatorial optimization problems which are difficult to solve. With such problems, the time required to find an optimal solution grows rapidly with the size of the problem (Hochbaum, 1997). Nevertheless, owing to the rapid growth of algorithmic efficiency and computational power, efficient tools are now available to solve even large scale timetabling problems (Cacchiani *et al.*, 2013; Badoni *et al.*, 2014; Phillips *et al.*, 2015; Bellio *et al.*, 2016), but still applications of such optimization methods are not very common, particularly in the Sri Lankan context. University timetables are usually prepared by administrative officers based on pre-

vious years' timetables by making amendments to cater new developments. Sometimes these timetables are revised several times to avoid clashes, creating difficulties to both staff and students. Teachers' and students' preferences are rarely considered in preparing timetables. Tight timetables (with little or no free time) sometimes restrict the choice of optional courses that the students can take. Timetables with long work days are sometimes blamed for lack of student participation in sports and other extra-curricular activities. Given a large number of courses, a variety of subject combinations, and limited resources, it is reasonable to encounter such issues in manually prepared timetables. Lack of knowledge on scientific methods of scheduling, institutional politics, and poor management may have contributed to inefficient timetabling in universities.

Mathematicians' involvement in academic timetabl-

ing has a long history dating back to the nineteen sixties. A wide variety of methods such as simulated annealing (Ceschia *et al.*, 2012; Basir *et al.*, 2013; Bellio *et al.*, 2016), tabu search (Aladag *et al.*, 2009; Lü and Hao, 2010; Abdullah and Turabieh, 2012), genetic algorithms (Mirrazavi *et al.*, 2003; Agustin-Blas, 2009; Mahiba and Durai, 2012), constraint logic programming (Deris *et al.*, 1997; Panagiotis *et al.*, 1998) and graph colouring (Cangalovic and Schreuder, 1991; Burke *et al.*, 1994) have been used to formulate and solve timetabling problems. Most of these are heuristic search methods which may not produce globally optimal solutions. Hybrid algorithms combining various heuristic methods have also been proposed (Bolagi *et al.*, 2014; Badoni *et al.*, 2014).

Lawrie (1969), Akkoyunlu (1973) and Tripathy (1980) were among the pioneers of using IP models to solve academic timetabling problems. They modeled the basic requirements of academic timetables such as the number of teaching sessions required per week for each course (completeness constraint) and the need for avoiding simultaneous session assignments for students and teachers (uniqueness constraint). These models were later improved and applied by many others including Dimopoulou and Miliotis (2001), Martin (2004) and Boland *et al.* (2006).

Daskalaki *et al.* (2004) presented a comprehensive IP formulation for university timetabling including a wide range of applicable constraints such as collision, completeness, consecutive sessions, repetitive sessions, and pre-allocations. Their timetabling model was quite a general model which can be applied to a wide range of problems. Later, Schimmelpfeng and Helber (2007) reported on the application of an IP model to generate timetables for a larger scale problem with similar constraints. Ismayilova *et al.* (2007) applied a multi-objective IP model considering both the administration's and instructors' preferences to solve a faculty course scheduling problem. This model considered teacher-course assignments as well as course-time slot assignments. Bakir and Akshop (2008) also used an IP model similar to Daskalaki *et al.* (2004) to solve a complicated timetabling problem. They recognized the difficulties arising from the large size of the problem when applied to real-world situations and emphasized the need for reducing the number of variables. Broek *et al.* (2009) proposed an IP model to develop timetables by assigning students to courses based on student preference for courses and other constraints. Miranda *et al.* (2012) proposed a web based timetabling method (to get data online from users) which produced optimal timetables using IP. Cacchiani *et al.* (2013) proposed a new method to compute lower bounds for a curriculum-based course timetabling problem aiming the optimal weekly assignment of lectures to rooms and time slots. In curriculum-based timetabling, timetables are prepared before the students enroll for the courses or without

knowing the details of student enrolments, based on compulsory and optional courses for each student group. Phillips *et al.* (2015) introduced a novel generalized pattern-based IP formulation for the room assignment problem of university course timetabling. Domenech and Lusa (2015) developed an IP model to assign teachers to courses while balancing teachers' teaching load and maximizing teachers' preferences for courses. Vermuyten *et al.* (2015) presented a two-stage IP model to minimize student flows through the corridors and stairwells while satisfying other timetabling requirements.

The IP based timetabling models proposed and used to date vary in the scope and applicability. While some timetabling models handle course-teacher assignments (for ex., Domenech and Lusa, 2015) and/or student-course enrollments (for ex., Broek *et al.*, 2009), others handle course-time slot and course-lecture hall assignments assuming that course-teacher and course-student assignments have been fixed (for ex., Vermuyten *et al.*, 2015). In this paper, a timetabling problem of the latter type (i.e. post enrollment timetabling) was considered.

The main objective of this work was to develop an economical and manageable IP model of a complicated timetabling problem in the Faculty of Applied Sciences, Wayamba University of Sri Lanka. A slightly different modeling approach was used to economize the model size and to maintain the simplicity of the model. A freely available open-source optimizer, OpenSolver, was used together with MS-Excel to code and solve the problem. The results indicate that, large scale timetabling problems can be solved more efficiently using the proposed integer programming model. The model could effectively reproduce manually prepared timetables with a significantly lower number of work hours per week. For example, the results indicate that an extremely tight faculty timetable with 51 course-work hours per week can be effectively reduced to 43 course-work hours per week without violating any applicable constraint. The model simulations were further useful in identifying the binding factors. For example, it was observed that the Faculty of Applied Science timetabling problem is bounded by the student overlaps (student uniqueness constraint), and both human and physical resource constraints are not significant. The model can be modified and applied for other university timetabling problems as well as for other scheduling problems.

The rest of this paper is organized into three main sections. After a brief description of the case study, the proposed timetabling formulation and modeling strategies are discussed in section 2. Section 3 covers the results observed by applying the proposed model to the identified case study. Section 4 wraps up the paper with a comparison of the observed results with previous work and a discussion on possible extensions.

1.1 Faculty of Applied Sciences–Timetabling Problem

To formulate a timetabling problem accurately, a deep understanding of the problem domain is required together with a sound mathematical programming background. The timetabling problem in the Faculty of Applied Sciences (FAS), Wayamba University of Sri Lanka has some unique, favorable and unfavorable features compared to general timetabling problems. The faculty conducts 3-year and 4-year B. Sc. degree programs in various disciplines. Each internal student of the faculty can be identified with a combination of major subjects he or she is studying. For each combination, there are compulsory courses and optional courses. The students may have to take optional courses out of their majors to fulfill the credit requirements in each year. In a given academic year, each student is at a certain level of study which corresponds to the year of study (for example, the students who are in their first year in the university, are level I students, the students in their second year of study are level II students, and so on). Students in each of the four levels of study follow separate sets of courses, but the timetabling problem for each level is not independent owing to the common pool of resources shared: mainly lecture halls, laboratories, and teachers.

This study was carried out during the second semester of 2014 academic year scheduled from September 2014 to January 2015. In this semester, FAS had 409 internal undergraduates at levels I, II, and III studying two major subjects from Computing & Information Systems, Electronics, Industrial Management, Mathematics & Mathematical Modelling, and Statistics. All students at levels I and II have compulsory Mathematics courses despite their majors. Level-4 students go for a 6-month industrial training during the second semester and no course work is scheduled for them during this period. The teaching staff on duty consisted of 30 permanent lecturers and 21 temporary tutors. One or more lecturers and one or more tutors are assigned for each course by the respective department, and the details of these assignments are sent to the faculty office before preparing the timetable. For each course, both the assigned lecturer (s) and tutor (s) should be made available for all teaching/learning sessions. FAS has 9 lecture halls with capacities ranging from 60-200 students, an electronic laboratory, and a computer laboratory. In addition, FAS schedules certain practical sessions in the university's main Information Technology Center. FAS schedules no academic activity in the lunch hour from 12.30 to 1.30 pm, and therefore the consecutive sessions should be either in the morning session or in the afternoon session. Though the usual work hours are from 8.30 am to 4.30 pm, the faculty timetable runs from 7.30 am to 7.30 pm in some week days. It is extremely difficult to produce timetables without clashes, particularly for

upper levels of study. Over the years, long work days have frustrated the students as well as the staff. The primary objective of this work is to generate more rational timetables with only 8 work hours per day, or if an 8-hour timetable is not feasible, to minimize the number of working hours over the week.

2. MATERIALS AND METHODS

In literature, university timetabling problems similar to that of FAS, have been modeled with six-dimensional binary variables, i.e., with variables x_{ijklmn} taking the value of 1, if course m , taught by teacher l to the student group k , is scheduled for the j^{th} time period of day i in classroom n ; and 0 otherwise (Daskalaki *et al.*, 2004; Bakir and Akshop, 2008). Modern universities and faculties offer a wide variety of degree programs and courses and therefore millions of variables are required to model timetabling problems. To reduce the number of variables to a manageable level, the variables are usually redefined with relevant subsets of courses, teachers, students groups, and classrooms. In this work, instead of using subsets, a *Relation Matrix* was utilized to eliminate the need for some dimensions (indices) of the variables, thus reducing the total number of variables in the model.

For the FAS timetabling problem, five dimensional variables were used by excluding the indices student group and lecturer, and including another dimension as the session type. Each course offered by FAS can have lecture and tutorial sessions, lecture and practical sessions, or just practical sessions. Therefore the type of the sessions could be lecture, tutorial, or practical. Even for the same course, the scheduling requirements of lecture sessions and practical/tutorial sessions can be different. For example, a course may require 2-hour lecture sessions in a lecture hall and 3-hour practical sessions in a laboratory. Therefore the session type index is required. In FAS, since the number of students is relatively less, one course is usually taught by one of few previously assigned lecturers for all the students simultaneously. Therefore, apart from some practical sessions, the teaching sessions are not usually repeated for different student groups. Hence there is no need to index the decision variables over student groups and teachers, instead, a *Relation Matrix* is used to describe the relationships between courses and student groups and courses and teachers. This approach is more economical in terms of the size of the model because when the number of indices is reduced, the number of variables is also reduced.

Another matrix is used to indicate the hall feasibility: whether a certain course session can be conducted in a given hall depending on the course, session, number of students, hall capacity, and hall type. Since some course sessions have to be scheduled in only one pre-determined

location such as a laboratory, there is no need to define variables relative to other locations. On the other hand, there were some courses which require only practical sessions. For such courses, there is no need to define variables for lecture sessions also. A VBA (Visual Basic Application) programming code was used to create the set of variables based on feasible locations and required type of sessions only. The VBA code reads the course details and hall feasibility matrix and writes the set of required variables. This method further reduces the number of variables in the model. The proposed timetabling model is presented below.

Indices

- c = course: 1, 2, ..., C . A course is a taught module which belongs to one of the subjects.
- s = session type: 1, 2, ..., S . Since a course can have either tutorial or practical sessions, but not both, session types can be taken as lecture and tutorial/practical, $S=2$.
- h = hall: 1, 2, ..., H . Both lecture halls and laboratories are considered as halls.
- d = day: 1, 2, ..., D . Five days per week, $D=5$.
- t = time slot: 1, 2, ..., T . Eight 1-hour time slots per day, $T=8$.
- l = lecturer/tutor: 1, 2, ..., L .
- g = student group: 1, 2, ..., G . Student groups are defined based on the combinations of major subjects. So each student group consists of the students enrolled for a unique combination of major subjects.

Parameters

- Y_{gc} = 1 if at least one student of group g is registered for course c ; 0 otherwise.
- Z_{lc} = 1 if lecturer/tutor l teaches course c ; 0 otherwise.
- N_{cs} = total number of time slots required each week for session type s of course c .
- B_{cshdt} = Benefit of scheduling session s of course c for the t^{th} time slot of day d in hall h . Benefit values can be assigned based on student and teacher preferences.
- R_c = number of students registered for course c .
- E_{cs} = number of consecutive time slots required for session type s of course c . This indicates the length of the sessions. The length of lecture sessions can be either 2 hours or 3 hours. The length of tutorial sessions is 1 hour, and the length of practical sessions can be 2 hours, 3 hours, or 4 hours. Usually the courses do not have sessions of the same type with different lengths. (The only exception was some 3-credit courses requiring one 2-hour lecture session and another one

1-hour lecture session followed by a 1-hour tutorial session per week. The latter was considered as another 2-hour lecture session. This has no effect on the model accuracy because both the assigned lecturer(s) and the tutor(s), and all the students should be available for both lecture and tutorial sessions of these courses).

P_h = Capacity of hall h .

Decision variables

x_{cshdt} = 1 if a session s of course c is scheduled for the t^{th} time slot of day d in hall h ; 0 otherwise.

Model

$$\text{Maximize } \sum_c \sum_s \sum_h \sum_d \sum_t B_{cshdt} x_{cshdt} \tag{1}$$

As mentioned above, the primary objective of this work was to reduce the number of work hours per week to 40, or if it is not possible, to minimise the number of work hours per week. However the model is formulated with a secondary objective which maximise the total benefit derived from the assignments. The primary objective was achieved by iteratively solving this timetabling problem over an increasing number of available time slots, starting from 40 hours per week, until a feasible solution is obtained. Once the minimum required number of work hour per week is found, the above objective function assure that those hours are utilised in such a way that maximises the benefit to the students and/or the staff.

Subject to:

Completeness Constraint: for each session type of each course, the required number of weekly hours should be allocated.

$$\sum_h \sum_d \sum_t x_{cshdt} = N_{cs} \text{ for all } c \text{ \& } s. \tag{2}$$

Uniqueness constraint: each student, each lecturer/tutor and each hall can have at most one course session at a time.

$$\sum_c \sum_s \sum_h Y_{gc} x_{cshdt} \leq 1 \text{ for all } g, d \text{ \& } t. \tag{3}$$

$$\sum_c \sum_s \sum_h Z_{lc} x_{cshdt} \leq 1 \text{ for all } l, d \text{ \& } t. \tag{4}$$

$$\sum_c \sum_s x_{cshdt} \leq 1 \text{ for all } h, d \text{ \& } t. \tag{5}$$

In constraint 3, the student-course relation matrix (Y_{gc}) is used to calculate the number of course sessions that a student group has to attend in a given time slot. Similarly, the teacher-course relation matrix (Z_{lc}) is used in constraint 4 to avoid simultaneous session assignments for teachers.

Consecutiveness Constraint: consecutive time slots

should be assigned depending on the required length of the sessions, E_{cs} .

If two consecutive time slots are required for session type s of course c (if $E_{cs} = 2$):

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ for } t \in \{1, 5\} \text{ and all } h \& d. \quad (6)$$

$$x_{cshdt} - x_{cshd(t+1)} - x_{cshd(t-1)} \leq 0 \text{ for } t \in \{2, 3, 6, 7\} \text{ and all } h \& d. \quad (7)$$

$$x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ for } t = \{4, 8\} \text{ and all } h \& d. \quad (8)$$

Constraints 6-8 have been specified based on the fact that the lunch time will occur after the 4th time slot of each day (so the consecutive sessions should be either before or after lunch). Constraint 6 specifies that, if a session s of course c which has $E_{cs} = 2$ is allocated to the 1st or 5th time slot, then the next time slot (2nd or 6th) should also be allocated to the same session. If such a session is allocated to the 4th or 8th time slot, then the previous time slot should also be allocated to the same session (constraint 8). If such a session is allocated to any other time slot, then either the previous or the next time slot should also be allocated to the same session (constraint 7).

If three consecutive time slots are required for session type s of course c (if $E_{cs} = 3$):

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t+2)} \leq 0 \text{ for } t = \{1, 5\} \text{ and all } h \& d. \quad (9)$$

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-1)} - x_{cshd(t+2)} \leq 0 \text{ for } t = \{2, 6\} \text{ and all } h \& d. \quad (10)$$

$$x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t+1)} - x_{cshd(t-2)} \leq 0 \text{ for } t = \{3, 7\} \text{ and all } h \& d. \quad (11)$$

$$x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-2)} \leq 0 \text{ for } t = \{4, 8\} \text{ and all } h \& d. \quad (12)$$

For $E_{cs} = 3$, constraints 9-12 have been formulated based on the same concept discussed above. For example, constraint 9 specifies that, if a session s of course c which has $E_{cs} = 3$ is allocated to the 1st or 5th time slot, then the next two time slots should also be allocated to the same session.

If four consecutive time slots are required for session type s of course c (if $E_{cs} = 4$):

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t+2)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t+3)} \leq 0 \text{ for } t = \{1, 5\} \text{ and all } h \& d. \quad (13)$$

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t+2)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ for } t = \{2, 6\} \text{ and all } h \& d. \quad (14)$$

$$x_{cshdt} - x_{cshd(t+1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-2)} \leq 0 \text{ for } t = \{3, 7\} \text{ and all } h \& d. \quad (15)$$

$$x_{cshdt} - x_{cshd(t-1)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-2)} \leq 0 \text{ and } x_{cshdt} - x_{cshd(t-3)} \leq 0 \text{ for } t = \{4, 8\} \text{ and all } h \& d. \quad (16)$$

For $E_{cs} = 4$, above set of constraints (13-16) have also been formulated based on the same concept discussed previously.

Since the courses offered by FAS usually have fixed length sessions (i.e., a course can have either 2-hour lecture sessions or 3-hour lecture sessions but not both), the consecutiveness constraint could be modeled in a simpler manner compared to previous work (Daskalaki *et al.*, 2004) which allowed any split of weekly teaching hours as decided by the teachers. In this work, different sets of constraints were modeled based on the required length of the sessions, E_{cs} . Such conditional constraints can be easily modeled using looped IF functions in MS Excel.

Pre-assignment constraints: if a session s of course c is pre-assigned for time slot t of day d in hall h , the pre-assignment constraint can be given as:

$$x_{cshdt} = 1. \quad (17)$$

Certain courses such as English Language Proficiency have to be assigned to a given time slot of a given day as they are conducted by teachers from other faculties. If required, some pre-allocations can be made to reduce the size of the problem and thus the computational burden.

Binary constraint: all scheduling variables should be binary.

$$x_{cshdt} \text{ binary for all } c, s, h, d \& t. \quad (18)$$

Additional constraints

Completeness, uniqueness, consecutiveness, and pre-assignments are essential constraints usually included in any timetabling model. Additionally, there are some specific constraints applicable to the FAS timetabling problem.

For 4-credit courses which require two 2-hour lecture sessions per week, those two sessions should NOT be scheduled on the same day. Similarly some 4-credit courses require two 3-hour practical sessions per week which should not be scheduled on the same day. For such course sessions cs :

$$\sum_h \sum_t x_{cshdt} \leq E_{cs} \text{ for all } d. \quad (19)$$

For some courses, it is usually better if the lecture session and tutorial/practical session are scheduled in different weekdays. Otherwise a public holiday can have a significant impact on some courses. For such courses c :

$$\sum_h \sum_t x_{c1hdt} + \sum_h \sum_t x_{c2hdt} \leq \text{Max}(E_{c1}, E_{c2}) \text{ for all } d. \quad (20)$$

As a norm, FAS do not schedule more than one lecture session for any lecturer in either morning or afternoon sessions. Since the lecture sessions can be two or three hours long, this requirement can be formulated as below.

$$\sum_c \sum_h \sum_{t \leq A} Z_{lc} x_{c1hdt} \leq 3 \text{ for all } l \text{ and } d. \quad (21)$$

$$\sum_c \sum_h \sum_{t \geq S} Z_{lc} x_{c1hdt} \leq 3 \text{ for all } l \text{ and } d. \quad (22)$$

Since the lecture halls have limited capacity, the hall capacity constraint can be modeled as follows, but it is not required in this model because the variables are defined relative to feasible halls only as mentioned above.

$$R_c x_{cshdt} \leq P_h \text{ for all } c, s, h, d \text{ \& } t. \quad (23)$$

3. APPLICATION AND RESULTS

The proposed model was applied to generate Level I, level-II and Level-III timetables of the second semester of 2014 academic year (Sept-Jan 2014) and compared with the timetables already prepared by the faculty office. The required data were obtained from student prospectus, existing timetables, student course registration details, and personal observations. The number of students, the number of subject combinations and the number of courses to be scheduled in this semester are summarized in Table 1.

Table 1. Summary of data relevant to second semester timetabling in FAS

	No. of students	No. of subject combinations	No. of courses
Level-I	158	3	11
Level-II	120	3	13
Level-III	131	12	17

The number of subject combinations corresponds to the number of student groups to be considered (Figure 1).

Some laboratory practical sessions had to be scheduled 2-4 times a week due to resource limitations, and in such cases, the repetitive practical sessions were considered as different courses for different sets of student groups. After this expansion, altogether there were 22 level-III courses, 21 level-II courses and 18 level-I courses in the conceptual model (Figure 2). English courses were assumed to be pre-assigned as in the current timetable. Eight 1-hour time slots starting from 8.30 am in each week day were considered initially. As mentioned above, the benefit from assignments can be based on stu-

Stu Groups			Courses																																
Level of study	First Major	Second major	CMIS 3224	CMIS 3234	ELTN 3233	ELTN 3241	ELTN 3253	ELTN 3261	IMGT 3252	IMGT 3262	IMGT 3272	IMGT 3282	IMGT 3+34	MATH 3224	MMOD 3224	MMOD 3234	STAT 3234	AEP	BE	CMIS 2213/22	CMIS 2234	ELTN 2213	ELTN 2221	ELTN 2232	ELTN 2241	IMGT 2212	IMGT 2222	MATH 2213	STAT 2212	STAT 2222	ELPC 2+20	ICT 222			
3	CMIS	ELIN	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	CMIS	IMGT	1	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	CMIS	STAT	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	ELIN	CMIS	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	ELIN	IMGT	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	ELIN	MATH	0	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	IMGT	CMIS	1	1	0	0	0	0	1	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	IMGT	STAT	0	0	0	0	0	0	1	1	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	MATH	ELIN	0	0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	STAT	CMIS	1	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	STAT	IMGT	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	STAT	MATH	0	1	0	0	0	0	0	0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	CMIS	ELIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	
2	CMIS	IMGT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	
2	ELIN	IMGT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
1	CMIS	ELIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	CMIS	IMGT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	ELIN	IMGT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Total students			66	75	68	68	68	68	54	54	55	28	28	12	28	27	47	131	131	107	107	68	68	68	68	65	65	120	120	120	120	120	13		
Required L hours			4	4	4	0	4	0	2	2	2	2	0	3	3	4	4	2	2	3	4	4	0	2	0	2	2	4	2	2	2	2	3		
Required TP hours			2	2	0	2	0	2	1	1	1	1	6	3	3	1	1	0	0	2	2	0	2	0	2	1	1	0	1	1	0	0			
Length of L sessions			2	2	2	-	2	-	2	2	2	2	-	3	3	2	2	2	2	3	2	2	-	2	-	2	2	2	2	2	2	2	3		
Length of TP sessions			2	2	-	2	-	2	1	1	1	1	3	3	3	1	1	-	-	2	2	-	2	-	2	1	1	-	1	1	-	-			

Figure 1. Student-course matrix for level III and level II.

dent and teacher preferences, but in this illustration, assignments in the current timetable were considered as preferred assignments (with relatively larger values assigned for B_{cshdt}).

The IP model was coded on MS Excel and solved using an open source optimization program, OpenSolver (Mason, 2011) which is a free optimizer available as an Excel add-in. The OpenSolver program has the Cbc

(Coin-OR branch and cut) solver which execute a branch and cut algorithm written in C++ to solve integer programming problems. In this timetabling model, Excel serves as front-end for the solver program. Unlike the Excel's built-in solver, OpenSolver does not have any artificial limitations on the number of variables or constraints. Standard Excel formulas such as VLOOKUP and SUMIFS were extensively used in modeling the problem.

Level of study	Stu. Group		Courses																							No. of Students
	First Major	Second major	CMS 3224 (L)	CMS 3224 (P) - Gp 1	CMS 3224 (P) - Gp 2	CMS 3234 (L)	CMS 3234 (P) - Gp 1	CMS 3234 (P) - Gp 2	ELTN 3233	ELTN 3241	ELTN 3253	ELTN 3261	IMGT 3252	IMGT 3262	IMGT 3272	IMGT 3282	IMGT 3+34	MATH 3224	MMOD 3224	MMOD 3234	STAT 3234	AEP	BE - Gp I	BE - Gp II		
3	CMS	ELTN	1	0	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	8
3	CMS	IMGT	1	0	1	1	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	1	0	7	
3	CMS	STAT	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	2
3	ELTN	CMS	1	1	0	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	36
3	ELTN	IMGT	0	0	0	1	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1	0	10	
3	ELTN	MATH	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	0	1	12	
3	IMGT	CMS	1	0	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	0	0	1	0	1	9	
3	IMGT	STAT	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	1	1	0	1	19	
3	MATH	ELTN	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1	0	2	
3	STAT	CMS	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	4	
3	STAT	IMGT	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	1	0	1	9	
3	STAT	MATH	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	1	1	1	1	0	1	13	
			66	36	30	75	38	37	68	68	68	68	54	54	55	28	28	12	28	27	47	125	63	63	131	

Figure 2. Extended student-course matrix for level III.

Time	Monday	Tuesday	Wednesday	Thursday	Friday
8.30 AM - 9.30 AM	ICT 122 (P) ICT	CMIS 1221 (P-III) ELTN 1212 (P-I)	ELPC 1+10 (I,II,III,& IV) LR 1,2,3,& 4	MATH 1212 (L) MH	ELTN 1222 (L) MH
9.30 AM - 10.30 AM	ICT 122 (P) ICT	CMIS 1221 (P-II) ELTN 1212 (P-I)	ELPC 1+10 (I,II,III,& IV) LR 1,2,3,& 4	MATH 1212 (L) MH	ELTN 1222 (L) MH
10.30 AM - 11.30 AM	MATH 1222 (L) MH	CMIS 1221 (P-II) ELTN 1212 (P-I)	STAT 1213 (L) MH	IMGT 1212 (L) MH	IMGT 1222 (L) MH
11.30 AM - 12.30 AM	MATH 1222 (L) MH	CMIS 1221 (P-II) ELTN 1212 (P-I)	STAT 1213 (L) MH	IMGT 1212 (L) MH	IMGT 1222 (L) MH
12.30 AM - 1.30 AM	LUNCH				
1.30 AM - 2.30 AM	CMIS 1212 (L) MH	CMIS 1221 (P-II) ELTN 1212 (P-I)	ICT 122 MH	STAT 1213 (L) MH	IMGT 1212 (T-I&II) MH & LR5
2.30 AM - 3.30 AM	CMIS 1212 (L) MH	CMIS 1221 (P-II) ELTN 1212 (P-I)	ICT 122 MH	STAT 1213 (L) MH	
3.30 AM - 4.30 AM	IMGT 1222 (T-I&II) MH & LR5	CMIS 1221 (P-II) ELTN 1212 (P-I)	MATH 1222 (T) MH	ELTN 1222 (T) MH	
4.30 AM - 5.30 AM		CMIS 1221 (P-II) ELTN 1212 (P-I)	MATH 1222 (T) MH	MATH 1212 (T) MH	

Figure 3. Timetable for Level I. Session type and group are in brackets (L-lecture, T-tutorial, P-practical). The venue is in bold. All CMIS practical sessions are conducted in the CMIS lab and all ELTN practical sessions are conducted in the ELTN lab.

Initially the model was used to generate timetables for each level separately. The existing level-I timetable contained 36 course-work hours per week leaving Friday afternoon free. The model formulation for level-I contained 720 variables, and 1533 constraints. Given equal benefit values for all possible assignments, the model generated the timetable with 34 course hours, and several alternative optimal solutions were available. When a benefit value of 2 was given to current assignments against a benefit value of 1 for other possible assignments, the model generated exactly the current timetable (Figure 3). Since the level-I timetabling problem alone is simple and unbounded, it can be prepared manually. Hence in this work, the level-I timetabling problem was used just to validate the model.

The present level-II timetable was tight with 46 course-work hours per week starting from 7.30 am in most week days and ending at 6.30 pm on some week days (Figure 4). The level-II timetabling problem was formulated with 1320 variables and 2137 constraints. Initial results indicated that the problem was infeasible with 40 work hours per week and that at least 41 work

hours are required to produce a feasible timetable with no overlaps for the students. The proposed model could assign all course sessions into time slots from 8.30 am to 5.30 pm, with just one additional hour as shown in Figure 5.

The existing level-III timetable was even compact with 51 course-work hours per week scheduled from 7.30 am to 7.30 pm most weekdays (Figure 6). The main objective was to reduce the number of work hours. Since the number of students in each group and in each course is relatively small, all lecture halls other than LR7 and MH (which are usually allocated for level-II and level-I teaching sessions respectively) were considered to enable as many as possible parallel assignments. The formulation contained 4280 variables and 6461 constraints. Again the problem was infeasible with a 40 hour week. After some relaxations, it was observed that at least 43 work hours per week are required to generate a feasible timetable. When three course sessions were pre-assigned to three additional hours, the model could assign all the remaining teaching sessions to time slots from 8.30 am to 5.30 pm (Figure 7).

Time	Monday	Tuesday	Wednesday	Thursday	Friday
7.30 am – 8.30 am		STAT 2212 (T) LR 07	MATH 2213 (T) LR 07	ELPC 2†20 Gp. I, I & III LR 01,02,03	IMGT 2212 (T) Gp. I & II LR 07 & LR 08
8.30 am – 9.30 am	IMGT 2222 (L) LR 07	ELTN 2213 (L) LR 07	MATH 2213 (L) LR 07	ELPC 2†20 Gp. I, I & III LR 01,02,03	IMGT 2212 (L) LR 07 CMIS 2234 (P) Gp.I
9.30 am – 10.30 am	IMGT 2222 (L) LR 07	ELTN 2213 (L) LR 07	ELTN 2221 (P) CMIS 2213/2222 (P) Gp.I	CMIS 2213/ 2222 (L) LR 07	IMGT 2212 (L) LR 07 CMIS 2234 (P) Gp.I
10.30 am – 11.30 am	ELTN 2232 (L) LR 07	CMIS 2234 (L) LR 07	ELTN 2221 (P) CMIS 2213/2222 (P) Gp.I	CMIS 2213/2222 (L) LR 07	ELTN 2213 (L) LR 07
11.30 am – 12.30 pm	ELTN 2232 (L) LR 07	CMIS 2234 (L) LR 07	ELTN 2221 (P)	CMIS 2213/2222 (L) LR 07	ELTN 2213 (T) LR 07
12.30 pm - 1.30 p.m.					
1.30 p.m.- 2.30 p.m.	STAT 2212 (L) LR 07	ICT 222 LR 07	ELTN 2214 (P) CMIS 2234 (P)Gp.II	STAT 2222 (L) LR 07	MATH 2213 (L) LR 07
2.30 p.m.- 3.30 p.m.	STAT 2212 (L) LR 07	ICT 222 LR 07	ELTN 2214 (P) CMIS 2234 (P) Gp.II	STAT 2222 (L) LR 07	MATH 2213 (L) LR 07
3.30 p.m. – 4.30 p.m.	CMIS 2234 (L) LR 07	ICT 222 LR 07	ELTN 2214 (P)	CMIS 2213/2222 (P) Gp.III	CMIS 2213/2222 (P) Gp.II
4.30 pm – 5.30 p.m.	CMIS 2234 (L) LR 07	STAT 2222 (T) LR 07 4.40 pm – 5.40 p.m.	CMIS 2234 (P) Gp.III	CMIS 2213/2222 (P) Gp.III	CMIS 2213/2222 (P) Gp.II
5.30 pm – 6.30 p.m.			CMIS 2234 (P) Gp.III	IMGT 2222 (T) Gp. I & II LR 07 & 04	

Figure 4. Current level-II timetable.

Time	Monday	Tuesday	Wednesday	Thursday	Friday
7.30 AM - 8.30 AM				ELPC 2+20 (I,II,III) LR 1, 2, & 3	
8.30 AM - 9.30 AM	IMGT 2222 (L) LR7	ELTN 2213 (L) LR7	STAT 2212 (T) LR7	ELPC 2+20 (I, II, III) LR 1, 2, & 3	CMIS 2234 (P-I) IMGT 2212 (L) LR7
9.30 AM - 10.30 AM	IMGT 2222 (L) LR7	ELTN 2213 (L) LR7	CMIS 2213/22 (P-II) ELTN 2221 (P)	CMIS 2213/22 LR7 ICT 222 LR1	CMIS 2234 (P-I) IMGT 2212 (L) LR7
10.30 AM - 11.30 AM	ELTN 2232 (L) LR7	CMIS 2234 (L) LR7	CMIS 2213/22 (P-II) ELTN 2221 (P)	CMIS 2213/22 LR7 ICT 222 LR1	ELTN 2213 (L) LR7
11.30 AM - 12.30 AM	ELTN 2232 (L) LR7	CMIS 2234 (L) LR7	ELTN 2221 (P)	CMIS 2213/22 LR7 ICT 222 LR1	ELTN 2213 (T) LR7
12.30 AM - 1.30 AM	LUNCH				
1.30 AM- 2.30 AM	STAT 2212 (L) LR7	MATH 2213 (L) LR7	CMIS 2234 (P-II) ELTN 2241 (P)	STAT 2222 (L) LR7	MATH 2213 (L) LR7
2.30 AM- 3.30 AM	STAT 2212 (L) LR7	MATH 2213 (L) LR7	CMIS 2234 (P-II) ELTN 2241 (P)	STAT 2222 (L) LR7	MATH 2213 (T) LR7
3.30 AM- 4.30 AM	CMIS 2234 (L) LR7	CMIS 2234 (P-III)	ELTN 2241 (P)	CMIS 2213/22 (P-III)	CMIS 2213/22 (P-I) IMGT 2212 (T) LR7&4
4.30 AM- 5.30 AM	CMIS 2234(L) LR 7	CMIS 2234 (P-III)	STAT 2222 (T) LR7	CMIS 2213/22 (P-III)	CMIS 2213/22 (P-I) IMGT 2222 (T) LR7&4

Figure 5. Proposed level-II timetable.

7.30 am – 8.30 am		AEP Course <i>LR 03</i>	IMGT 3252 (T) <i>LR 08</i>	IMGT 3282 (T) <i>LR 08</i>	Business English Course Gp. I, & II <i>LR 05 & LR 04</i>
8.30 am – 9.30 am	CMIS 3234 (L) <i>LR 08</i>	AEP Course <i>LR 03</i>	ELTN 3253 (L) <i>LR 08</i>	ELTN 3253 (L) <i>LR 08</i> STAT 3234 (T) <i>LR 04</i>	Business English Course Gp. I, & II <i>LR 05 & LR 04</i>
9.30 am – 10.30 am	CMIS 3234 (L) <i>LR 08</i>	IMGT 3+34 MMOD 3224 (L) <i>LR 08</i>	MATH 3224 (P) IMGT 3262 (L) <i>LR 08</i>	ELTN 3253 (L) <i>LR 04</i>	MATH 3224 (P) MMOD 3224 (P)
10.30 am – 11.30 am	ELTN 3233 (L) <i>LR 08</i> STAT 3234 (L) <i>LR 04</i>	IMGT 3+34 MMOD 3224 (L) <i>LR 08</i>	MATH 3224 (P) IMGT 3262 (L) <i>LR 08</i>	ELTN 3241 (P) STAT 3234 (L) <i>LR 04</i>	MATH 3224 (P) CMIS 3234 (P) Gp. I MMOD 3224 (P)
11.30 am – 12.30 pm	ELTN 3233 (L) <i>LR 08</i> STAT 3234 (L) <i>LR 04</i>	IMGT 3+34 MMOD 3224 (L) <i>LR 08</i>	MATH 3224 (P) IMGT 3262 (T) <i>LR 08</i>	ELTN 3241 (P) STAT 3234 (L) <i>LR 04</i>	MATH 3224 (P) CMIS 3234 (P) Gp. I MMOD 3224 (P)(T)
12.30 pm - 1.30 p.m.					
1.30 p.m.- 2.30 p.m.	IMGT 3272 (L) <i>LR 08</i>	ELTN 3233 (L) <i>LR 08</i>	CMIS 3234 (L) <i>LR 08</i>	ELTN 3261 (P)	IMGT 3+34 MMOD 3234 (T) <i>LR 08</i>
2.30 p.m.- 3.30 p.m.	IMGT 3272 (L) <i>LR 08</i>	ELTN 3233 (T) <i>LR 08</i>	CMIS 3234 (L) <i>LR 08</i>	ELTN 3261 (P)	IMGT 3+34
3.30 p.m. – 4.30 p.m.	VMOD 3234 (L) <i>LR 04</i> IMGT 3282 (L) <i>LR 08</i>	CMIS 3224 (L) <i>LR 08</i>	MMOD 3234 (L) <i>LR 04</i> IMGT 3252 (L) <i>LR 08</i>	CMIS 3224 (L) <i>LR 08</i>	IMGT 3+34
4.30 pm – 5.30 p.m.	VMOD 3234 (L) <i>LR 04</i> IMGT 3282 (L) <i>LR 08</i>	CMIS 3224 (L) <i>LR 08</i>	MMOD 3234 (L) <i>LR 04</i> IMGT 3252 (L) <i>LR 08</i>	CMIS 3224 (L) <i>LR 08</i>	
5.30 pm. – 6.30 p.m.	CMIS 3234 (P) Gp. II	CMIS 3224 (P) Gp. I	ELTN 3253 (T) <i>LR 08</i>	CMIS 3224 (P) Gp. II	
6.30 pm. – 7.30 p.m.	CMIS 3234 (P) Gp. II	CMIS 3224 (P) Gp. I	IMGT 3272 (T) <i>LR 08</i>	CMIS 3224 (P) Gp. II	

Figure 6. Current level-III timetable.

Since more than two parallel sessions are not scheduled owing to student overlaps, the level-III timetabling model was refined considering only two lecture halls, LR4 and LR8. These two halls have sufficient capacity for any level-III course. Then the model contained only 2720 variables and 3521 constraints, and it generated the same result with only the locations changed. The usefulness of a computer based optimization model was clearly visible in this case, as it reduced the number of weekly work hours from 51 to 43. This is a significant saving of eight work hour per week, which is equivalent to a saving of a whole work day.

Finally, the proposed IP model was applied to generate the whole faculty timetable for levels I, II, and III simultaneously. The model then consisted of 61 courses requiring 180 teaching hours. All available lecture halls and laboratories were considered. When the timetabling problem was formulated separately for each level, the teacher overlap constraint was redundant because the student overlap constraint was sufficient to avoid parallel assignment of courses from the same subject stream. But for the generalized whole faculty timetabling problem, the teacher overlap constraint was required. Thus the number of constraints increased to 10,437 while the number of variables still remained economical at 6720. As expected, the model results indicated that at least 43 work hours are required to produce a feasible timetable. The generated timetable is shown in Figure 8. Again the proposed model could reduce a tight 7.30 am to 7.30 pm timetable to 7.30/8.30 am to 5.30 pm enabling the students to attend sports practice and other extra-curricular activities after

5.30 in all week days and before 8.30 in some weekdays. Overtime payments are also minimized.

The outcomes of the optimization indicate that student overlap is the binding factor in FAS timetabling, and both hall availability and teacher overlaps are not significant. In any time slot, only a maximum of five course sessions are scheduled in parallel leaving the majority of lecture halls and laboratories unoccupied. Owing to student overlaps, only a maximum of one course from one subject stream and one level of study can be scheduled in any time slot (for example, two level-II CMIS courses cannot be scheduled in parallel). The timetables for levels I, II, and III can be solved separately to reduce the model size thereby simplifying the models and increasing the speed of solving, but some manual adjustments are required to eliminate possible teacher overlaps. However, when level-IV courses are also included (particularly in semester 1), about 80 courses will have to be scheduled and then the hall and teacher overlaps can have a significant impact on scheduling. When the practical sessions are divided into several student groups, the grouping criteria have an impact on scheduling. The model can be further developed to find the optimal way of grouping (for practical sessions) so as to minimize scheduling conflicts.

The calculation time varied from few seconds to half an hour depending on the size of the problem, however if SolverStudio, which is the successor of OpenSolver, was used instead, even the largest model could have been solved faster. SolverStudio is also a free Excel add-in which provides access to other modeling tools such as AMPL and PuLP.

Time	Monday	Tuesday	Wednesday	Thursday	Friday
7.30 AM - 8.30 AM		AEP Course LR3	IMGT 3252 (T) LR8		Business English (I,II) LR4 & LR5
8.30 AM - 9.30 AM	CMIS 3234 (L) LR8	AEP Course LR3	IMGT 3262 (L) LR8 MMOD 3234 (P) LR2	ELTN 3253 (L) LR8 STAT 3234 (P) LR4	Business English (I,II) LR 4 & LR 5
9.30 AM - 10.30 AM	CMIS 3234 (L) LR8	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8	IMGT 3262 (L) LR8 MATH 3224 (P) ICT	ELTN 3253 (L) LR8	IMGT 3282 (T) LR1 MMOD 3224 (P) ICT
10.30 AM - 11.30 AM	ELTN 3233 (L) LR8 STAT 3234 (L) LR4	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8	CMIS 3224 (L) LR8 MATH 3224 (P) ICT	ELTN 3241 (P) STAT 3234 (L) LR4	CMIS 3234 (P-I) MMOD 3224 (P) ICT
11.30 AM - 12.30 AM	ELTN 3233 (L) LR8 STAT 3234 (L) LR4	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8	CMIS 3224 (L) LR8 MATH 3224 (P) ICT	ELTN 3241 (P) STAT 3234 (L) LR4	CMIS 3234 (P-I) MMOD 3224 (P) ICT
12.30 AM - 1.30 AM	LUNCH				
1.30 AM- 2.30 AM	CMIS 3224 (P-I) IMGT 3272 (L) LR8	ELTN 3233 (L) LR8 IMGT 3282 (L) LR2	CMIS 3234 (L) LR8	IMGT 3272 (T) LR2	ELTN 3261 (P) IMGT 3+34 (P) ICT
2.30 AM- 3.30 AM	CMIS 3224 (P-I) IMGT 3272 (L) LR8	ELTN 3233 (T) LR8 IMGT 3282 (L) LR2	CMIS 3234 (L) LR8	IMGT 3262 (T) LR8 MATH 3224 (L) LR3	ELTN 3261 (P) IMGT 3+34 (P)
3.30 AM - 4.30 AM	CMIS 3224 (P-II) MMOD 3234 (L) LR4	CMIS 3224 (L) LR8 MMOD 3234 (L) LR1	CMIS 3234 (P-II)	IMGT 3252 (L) LR8 MATH 3224 (L) LR3	ELTN 3253 (L) LR4 IMGT 3+34 (P) ICT
4.30 AM - 5.30 AM	CMIS 3224 (P-II) MMOD 3234 (L) LR4	CMIS 3224 (L) LR8 MMOD 3234 (L) LR1	CMIS 3234 (P-II)	IMGT 3252 (L) LR6 MATH 3224 (L) LR3	ELTN 3253 (L) LR4

Figure 7. Proposed level-III timetable.

4. DISCUSSION AND CONCLUSIONS

An IP model has been proposed for a case study of university timetabling. The model generated timetables appears to be more efficient and compact compared to existing timetables. The model was useful to identify factors which bind the timetabling problem and possible simplifications. For example, it was observed that student overlaps occurring from the presence of several optional courses (i.e., the students can take optional courses from non-major subject areas) have complicated the timetabling process and that the resource constraints (lecture halls and laboratories) were redundant.

The statistics of the timetabling problems and associated IP models discussed above are summarized in Table 2. To evaluate the efficiency of the proposed modeling

approach, the problem sizes and model sizes were compared with those of a relatively similar case study reported in Daskalaki *et al.* (2004). The whole faculty timetabling problem discussed in this work is comparable to Problem 2 of Daskalaki *et al.* in terms of both the number courses and the number of teaching hours required. Still, the number of variables and constraints used to model the whole faculty timetabling problem are respectively a half of the number of variables and eighty percent of the number of constraints used to model Problem 2 of Dalskalaki *et al.* (2004).

While the model size (relative to the problem size) provides a good indication of modeling efficiency, computational time is the key factor which determines the efficiency of the solution algorithm. The above optimal solutions have been obtained using the Cbc solver algo-

Time	Monday	Tuesday	Wednesday	Thursday	Friday
7.30 – 8.30		AEP/BE (L) LR 2		ELPC 2+20 (L) LR2 IMGT 3252 (T)	AEP/BE (L) LR2
8.30 – 9.30	CMIS 3234 (L) LR8 CMIS 2213 (P-I) IMGT 2222 (L) LR7 ICT 122 (P) ICT	AEP/BE (L) LR 2 ELTN 2213 (L) LR7 CMIS 1221 (P-III) ELTN 1212 (P-I)	IMGT 3272 (T) LR3 STAT 2212 (T) LR7 ELPC 1+10 (L) LR1	ELTN 3253 (L) LR8 STAT 3234 (P) LR4 ELPC 2+20 (L) LR2 MATH 1212 (L) MH	AEP/BE (L) LR2 CMIS 2234 (P-I) IMGT 2212 (L) LR7 ELTN 1222 (L) MH
9.30 – 10.30	CMIS 3234 (L) LR8 CMIS 2213 (P-I) IMGT 2222 (L) LR7 ICT 122 (P) ICT	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8 ELTN 2213 (L) LR7 CMIS 1221 (P-III) ELTN 1212 (P-I)	IMGT 3262 (L) LR8 MATH 3224 (P) ICT CMIS 2213 (P-II) ELTN 2221 (P) ELPC 1+10 (L) LR1	ELTN 3253 (L) LR8 IMGT 3282 (T) LR1 CMIS 2213/22 (L) LR7 MATH 1212 (L) MH	MMOD 3224 (P) ICT CMIS 2234 (P-I) IMGT 2212 (L) LR7 ELTN 1222 (L) MH
10.30 – 11.30	ELTN 3233 (L) LR8 STAT 3234 (L) LR4 ELTN 2232 (L) LR7 MATH 1222 (L) MH	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8 CMIS 2234 (L) LR7 CMIS 1221 (P-IV) ELTN 1212 (P-I)	IMGT 3262 (L) LR8 MATH 3224 (P) ICT CMIS 2213 (P-II) ELTN 2221 (P) STAT 1213 (L) MH	ELTN 3241 (P) STAT 3234 (L) LR4 CMIS 2213/22 (L) LR7 CMIS 1221 (P-II) IMGT 1212 (L) MH	CMIS 3234 (P-I) MMOD 3224 (P) ICT ELTN 2213 (L) LR7 IMGT 1222 (L) MH
11.30 – 12.30	ELTN 3233 (L) LR8 STAT 3234 (L) LR4 ELTN 2232 (L) LR7 MATH 1222 (L) MH	IMGT 3+34 (P) ICT MMOD 3224 (L) LR8 CMIS 2234 (L) LR7 CMIS 1221 (P-IV) ELTN 1212 (P-I)	IMGT 3262 (T) LR8 MATH 3224 (P) ICT ELTN 2221 (P) STAT 1213 (L) MH	ELTN 3241(P) STAT 3234 (L) LR4 CMIS 2213/22 (L) LR7 CMIS 1221 (P-II) IMGT 1212 (L) MH	CMIS 3234 (P-I) MMOD 3224 (P) ICT ELTN 2213(L) LR7 IMGT 1222 (L) MH
12.30 – 1.30	LUNCH				
1.30 – 2.30	IMGT 3272 (L) LR8 STAT 2212 (L) LR7 CMIS 1212 (L) MH	CMIS 3224 (L) LR4 MATH 3224 (L) LR5 CMIS 2234 (L) LR7 ICT 222 (L) LR1 ELTN 1212 (P-II)	CMIS 3234 (L) LR8 CMIS 2234 (P-II) ELTN 2241 (P) CMIS 1212 (T) MH ICT 122 (L) LR5	ELTN 3261 (P) IMGT 3282 (L) LR1 STAT 2222 (L) LR7 STAT 1213 (L) MH	ELTN 3233 (L) LR8 MATH 2213 (L) LR7 CMIS 1221 (P-I) IMGT 1212 (T) MH
2.30 – 3.30	IMGT 3272 (L) LR8 STAT 2212 (L) LR7 CMIS 1212 (L) MH	CMIS 3224 (L) LR4 MATH 3224 (L) LR5 CMIS 2234 (L) LR7 ICT 222 (L) LR1 ELTN 1212 (P-II)	CMIS 3234 (L) LR8 CMIS 2234 (P-II) ELTN 2241 (P) ICT 122 (L) LR5	ELTN 3261 (P) IMGT 3282 (L) LR1 STAT 2222 (L) LR7 STAT 1213 (L) MH	ELTN 3233 (T) LR8 IMGT 3+34 (P) ICT MATH 2213 (L) LR7 CMIS 1221 (P-I)
3.30 – 4.30	CMIS 3224 (P-II) MMOD 3234 (L) LR4 IMGT 2212 (T) LR8 IMGT 1222 (T) MH	CMIS 3224 (L) LR8 MATH 3224 (L) LR5 CMIS 2234 (P-III) ICT 222 (L) LR1 ELTN 1212 (P-II)	CMIS 3234 (P-II) ELTN 2241 (P) MATH 1222 (T) MH	CMIS 3224 (P-I) IMGT 3252 (L) LR7 MMOD 3234 (L) LR4 MATH 2213 (L) LR8 ELTN 1222 (T) MH	ELTN 3253 (L) LR7 IMGT 3+34 (P) ICT CMIS 2213 (P-III)
4.30 – 5.30	CMIS 3224 (P-II) MMOD 3234 (L) LR4 IMGT 2222 (T) LR7	CMIS 3224 (L) LR8 MMOD 3234 (P) LR5 CMIS 2234 (P-III) ELTN 1212 (P-II)	CMIS 3234 (P-II) STAT 2222 (T) LR7	CMIS 3224 (P-I) IMGT 3252(L) LR7 MMOD 3234 (L) LR4 MATH 2213 (L) LR8 MATH 1212 (T) MH	ELTN 3253 (T) LR7 IMGT 3+34 (P) ICT CMIS 2213 (P-III)

Figure 8. Proposed whole faculty timetable.

Table 2. Comparison of problem sizes and model sizes

Scope of the problem	Problem size			Model size		
	No. of courses	No. of lab courses	Required teach. hours	No. of halls/labs	No. of constraints	No. of variables
Level-I	18	7	50	6/3	1,533	720
Level-II	21	8	56	6/2	2,137	1,320
Level-III	22	9	74	2/3	3,521	2,720
Whole Faculty	61	24	180	9/4	10,437	6,720
Problems discussed in Daskalai <i>et al.</i> (1998)						
Problem #1	25	8	139	3/6	7,543	4,100
Problem #2	47	19	187	4/10	12,734	13,527
Problem #3	72	27	326	6/12	17,159	19,295

rithm (in OpenSolver), and a new solution algorithm has not been developed so far to solve the FAS timetabling problem. Hence, the computational time (obtained with an available algorithm) was not compared with that of other proposed solution algorithms for timetabling problems. This paper focuses on modeling and solving the problems using an available solver algorithm. The next stage of this study is to use SolverStudio which has more advanced solver algorithms. A new customized solution algorithms will be required only if the available algorithms appears to be inefficient.

Some specific features of the FAS timetabling problems (for ex., fixed-length lecture sessions) have played a favorable role in reducing the model size, and therefore the modeling efficiency also cannot be guaranteed in all other situations. Still, the use of a relation matrix may be more appropriate in modeling other timetabling problems and can be tested with standards datasets.

The primary focus of the work was to develop a manageable IP model of a complicated university timetabling problem. Owing to the presence of students studying a wide variety of subject combinations, generating a feasible timetable was the main challenge faced. Therefore, the student and teacher preferences were not considered in applying the models. However, the preferences and even other objective criteria such as the work load distribution over the week and utilization of resources can be considered as the objective criteria in the optimization model.

In general, the results generated from the model show that such computerized timetabling models can produce more efficient timetables meeting complicated requirements. The tools used, namely MS Excel and OpenSolver were very productive for modeling and solving such problems. The programming capability of MS Excel with VBA enabled automated generation of customized sets of variables based on specified requirements. Standard Excel formulas were particularly useful in modeling different sets of constraints for different parameter values. Since MS Excel serves as a user-friendly and familiar

interface, once the IP model is developed, no expert knowledge is required to use the model to generate timetables for different scenarios. The IP model can be further adopted to incorporate other requirements and to apply for similar scheduling problems.

REFERENCES

- Abdullah, S. and Turabieh, H. (2012), On the use of multi neighbourhood structures within a tabu-based memetic approach to university timetabling problems, *Inform, Sciences*, **191**, 146-168.
- Agustín-Blas, L. E., Salcedo-Sanz, S., Ortiz-García, E. G., Portilla-Figueras, A., and Pérez-Bellido, A. M. (2009), Hybrid grouping genetic algorithm for assigning students to preferred laboratory groups, *Expert System Applications*, **36**(3), 7234-7241.
- Akkoyunlu, E. A. (1973), A linear algorithm for computing the optimum university timetable, *The Computer Journal*, **16**(4), 347-350.
- Aladag, A. H., Hocaoglu, G., and Basaran, M. A. (2009), The effect of neighbourhood structures on tabu search algorithm in solving course timetabling problem, *Expert System Applications*, **36**(10), 12349-12356.
- Badoni, R. P., Gupta, D. K., and Mishra, P. (2014), A new hybrid algorithm for university course timetabling problem using events based on groupings of students, *Computers & Industrial Engineering*, **78**, 12-25.
- Bakir, M. A. and Akshop, C. (2008), A 0-1 integer programming approach to a university timetabling problem, *Hacettepe Journal of Mathematics and Statistics*, **37**(1), 41-55.
- Basir, N., Ismail, W., and Norwawi, N. (2013), A simulated annealing for Tahmidi course timetabling, *Procedia Technology*, **11**, 437-445.
- Bellio, R., Ceschia, S., Gaspero, L. D., Schaerf, A., and Urli, T. (2016), Feature-based tuning of simulated annealing applied to the curriculum-based course

- timetabling problem, *Computers & Operation Research*, **65**, 83-92.
- Bolaji, A. L., Khader, A. T., Al-Betar, M. A., and Awadallah, M. A. (2014), University course timetabling using hybridized artificial bee colony with hill climbing optimizer, *Journal of Computational Science*, **5**(5), 809-818.
- Boland, N., Hughes, B. D., Merlot, L. T. G., and Stuckey, P. J. (2006), New integer linear programming approaches for course timetabling, *Computers & Operation Research* **35**(7), 2209-2233.
- Broek, J. V. D., Hurkens, C., and Woeginger, G. (2009), Timetabling problems at the TU Eindhoven, *European Journal of Operational Research*, **196**(3), 877-885.
- Burke, E. K., Elliman, D. G., and Weare, R. F. (1994), A university timetabling system based on Graph colouring and constraint manipulation, *Journal of Research on Computing in Education*, **27**, 1-18.
- Cacchiani, V., Caprara, A., Roberti, R., and Toth, P. (2013), A new lower bound for curriculum-based course timetabling, *Computers & Operation Research*, **40**(10), 2466-2477.
- Cangalovic, M. and Schreuder, J. A. M. (1991), Exact coloring algorithms for weighted graphs applied to timetabling problems with lectures of different lengths, *European Journal of Operational Research*, **51**(2), 248-258.
- Ceschia, S., Gaspero, L. D., and Schaerf, A. (2012), Design, engineering, and experimental analysis of a simulated annealing approach to the post-enrolment course timetabling problem, *Computers & Operation Research*, **39**(7), 1615-1624.
- Daskalaki, S., Birbas, T., and Housos, E. (2004), An IP formulation for a case study in university timetabling, *European Journal of Operational Research*, **153**(1), 117-135.
- Deris, S. B., Omatu, S., Ohta, H., and Samat, P. A. B. D. (1997), University timetabling by Constraint based reasoning: A case study, *Journal of the Operational Research Society*, **48**(12), 1178-1190.
- Dimopoulou, M. and Miliotis, P. (2001), Implementation of a university course and examination timetabling system, *European Journal of Operational Research*, **130**(1), 202-213.
- Domenech, B. and Lusa, A. (2015), A MILP model for the teacher assignment problem considering teachers' preferences, *European Journal of Operational Research*, **249**(3), 1153-1160.
- Hochbaum, D. S. (1997), *Approximation algorithms for NP-hard problems*. PWS Publishing Company, Boston, MA.
- Ismayilova, A. A., Sağır, M., and Gasimov, R. N. (2007), A multiobjective faculty-course-time slot assignment problem with preferences, *Mathematical and Computer Modelling*, **46**(7-8), 1017-1029.
- Lawrie, N. L. (1969), An integer linear programming model of a school timetabling problem, *The Computer Journal*, **12**, 307-316.
- Lü, Z. and Hao, J. K. (2010), Adaptive tabu search for course timetabling, *European Journal of Operational Research*, **200**(1), 235-244.
- Mahiba, A. A. and Durai, C. A. D. (2012), Genetic Algorithm with Search Bank Strategies for University Course Timetabling Problem, *Procedia Engineering*, **38**, 253-263.
- Martin, C. H. (2004), Ohio University's college of business uses integer programming to schedule classes, *Interfaces*, **34**, 460-465.
- Mason, A. J. (2011), OpenSolver - An open source add-in to solve linear and integer programmes in Excel. In: Klatt D, Luthi HJ, Schmedders K (eds) Operations Research Proceedings, *Springer, Berlin Heidelberg*, 401-406.
- Miranda, J., Rey, P. A., and Robles, J. M. (2012), UdpSkeduler: A web architecture based decision support system for course and classroom scheduling, *Decision Support Systems*, **52**(2), 505-513.
- Mirrazavi, S. K., Mardle, S. J., and Tamiz, M. (2003), A two-phase multiple objective approach to university timetabling utilizing optimization and evolutionary solution methodologies, *Journal of the Operational Research Society*, **54**(11), 1155-1166.
- Panagiotis, S., Vigla, E., and Karaboyas, F. (1998), Nearly optimum timetable construction through CLP and intelligent search, *International Journal of Artificial Intelligence Tools*, **7**(4), 415-442.
- Phillips, A. E., Waterer, H., Ehrgott, M., and Ryan, D. M. (2015), Integer programming methods for large-scale practical classroom assignment problems, *Computers & Operations Research*, **53**, 42-53.
- Schimmelpfeng, K. and Helber, S. (2007), Application of a real-world university-course timetabling model solved by integer programming, *OR Spectrum*, **29**(4), 783-803.
- Tripathy, A. (1980), A Lagrangian relaxation approach to course timetabling, *Journal of the Operational Research Society*, **31**(7), 599-603.
- Vermuyten, H., Lemmens, S., Marques, I., and Belien, J. (2015), Developing compact course timetables with optimized student flows, *European Journal of Operational Research*, **251**(2), 651-661.