RESEARCH ARTICLE

A General Mathematical Model for University Courses Timetabling: Implementation to a Public University in Malaysia

Zahidah Mohd Zaulir[a], Nurul Liyana Abdul Aziz[b], Nur Aidya Hanum Aizam[c],*

[a,c] Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu; [b] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Negeri Sembilan, Kampus Seremban, 70300 Seremban, Negeri Sembilan.

Abstract University course timetabling is a well-known management problem amongst researchers, thus the rich body of literature. However, published articles are mainly on improved solution approaches which lead to presentation on different constraints used and ignoring human preferences. This however, limits the model application to other universities. The research aims to bridge the gap by acknowledging these varieties of demands. In the process of generating our mathematical model, we have gone through meticulously researches that have been carried out in the past years to determine the demands of individuals involved directly with the timetable. The varieties of demands were clarified from surveys conducted. An improvised university course timetabling problem model was developed, which involves a superset of constraints that also includes the users' preferences. However, we will extensively discuss on the list of requirements obtained from the survey and demonstrate the requirements that were found acceptable to be considered in a general mathematical model. To verify the compatibility of our mathematical model, we illustrate with real data from a university in Malaysia. The experimental result confirms the applicability of our mathematical model towards real problem. We expect that this model could be in favor of solving other university course timetabling problem with slight modifications.

Keywords: University course timetabling problem, Scheduling, Management problem, Integer programming, Mathematical model.

Introduction

The university course timetabling problem (UCTP) involves assigning courses taken by a group of students, taught by a specified lecturer, to a limited number of timeslots into appropriate classroom. The assignment is carried out in such way that there are no conflicts between rooms, students, and lecturers, as well as fulfilling range of other requirements or what is defined as constraints. These constraints are generally the rules and policies of a university. Constraints can be divided into two categories, either hard or soft constraints. Hard constraints must be taken into serious consideration without allowing any violation to occur. Timetable that violates at least one of these hard constraints will cause an infeasible solution, where it contradicts to the purpose of a timetable. These constraints are usually the universities' policies. In contrast to hard constraints, a timetable that violates soft constraints is still usable however does not achieve the quality it needs in a timetable that is a much friendlier for either lecturers or students. Basically the soft constraints are the constraints that are considered to be producing a better timetable

that fulfills demands which are categorized not too critical but does give a higher level of acceptance and satisfactory. These two different types of constraints conclude the policies and requirements employed by researchers to form a mathematical model. Note that most universities will only consider the constraints that reflect the policies of its' own. From this point forward only then we can observe other researchers exploiting the requirements to produce a more effective timetable. It can thereafter be seen that the differences of articles exist in the literature in terms of variations of requirements. From the rich body of literature, we can classify five main requirements that have always been used by researchers as stated below:

- *Completeness*: each event either lectures, tutorials or lab classes included in the curriculum must be assigned in the timetable.
- *Conflict of resources*: no conflict of resources should occur in a timeslot. Resources in this refer to the lecturers, student groups, and rooms.
- *Work load*: this is similar to a type of distribution constraint where lecturers and student groups have a limited number of teaching and learning hours daily.
- *Availability of resources*: this constraint is related to the availability of lecturers, rooms, and timeslots. Example of availability constraint, the lecturer may not be available on a given day or during certain timeslot.
- *Meeting patterns*: this constraint stipulates on how the lecture, tutorial and lab classes are to be assigned and usually determined by the nature of their courses.

In line with the many different authors published their work in the literature on UCTP, numerous approaches have been introduced to solve this particular problem. Integer programming is one of the established methods that are used in solving the timetable problem. Lawrie [1] is the earliest researcher who used integer programming (IP) to solve timetabling problem. A university timetabling problem is formulated in terms of costs associated with timetable features, including the interaction between courses. Another most referred research is by Daskalaki et al [2]. They employed a novel 0-1 integer programming formulation of the university-timetabling problem. In the study, they minimize a linear cost function by considering the demands regarding timeslots, days or rooms for specific course. Following the results obtained, again they have used IP in other research but with different approach. Two-stage relaxation procedure is used, which involves the relaxation of computationally heavier constraint in the first stage and solving the whole problem in the next stage (Daskalaki and Birbas [3]). They have expressed that this procedure is better compared to single stage procedure in terms of time reduction and that additional features could be included in the model. Besides that, the two-phase approach to model the timetabling problem is also presented by Ribić and Konjicija [4]. In the initial stage, classes were assigned to days and then followed by assignments to slots of that specific day. Other studies that have also employed IP with other strategies include Oladokun and Badmus [5], MirHassani [6], Colajanni and Daniele [7] and Lemos et al., [8]. Besides integer programming, graph coloring (Samarasekara [9]), simulated annealing (Gunawan and Ng [10]), genetic algorithm (Modibbo et al. [11]), tabu search (Chen et al. [12]), ant colony optimization (Mahmud [13]) and constraint programming (Junn et al. [14]) are among the other well-known approaches that have been presented in the literature. There are various other techniques which involve hybrid techniques. These approaches will not be covered in this research; however, it can be observed that the problem is tailored by improving one method over the other. Covering a different aspect of the problem, we improved the previous model in Aizam and Caccetta [15] of determining more common requirements considered by different universities. This process is done by identifying the requirements over the literature, own observations, and through survey distributions. Detailed results of the survey can be found in Abdul Aziz and Aizam [16]. Thus, from both works carried out, we have come up with a brief report (Aziz and Aizam [17]). The requirements obtained will be included in university course timetabling model that will not only capture the most constraints used but to also highlight the demands of all parties involved.

The main objective of this research is to construct a general mathematical model that is able to suit most university course timetabling problems, where the existing models developed are based solely on their specified requirements. This is to emphasize the idea of saving up administration time in finding a way to produce a timetable that benefits its main users. The research is organized as follows. Discussion on

the list of requirements is given in the early part of the methodology section. We present the basic requirements used by most researchers and also the inclusion of additional requirements that are raised by individuals related to timetable. This is to portray different kind of requirements needed in different universities. However, not all requirements are agreed. Arguments and some point of views on the discussion are also reported. The new list of constraints that are used in the general mathematical model will complete the initial part of our methodology. With the determination of the requirements, presentation on the mathematical model, where all notation and mathematical formulation are given. Following the model developed, we have run through experimental tests on real data of a Malaysian university. Overall results focusing on the technical analysis together with extensive discussion based on four different key elements are provided. Concluding remarks and some perspectives for future research are given in the final section.

Methodology

Prior constructing a mathematical model of a university course timetabling problem, a list of requirements that consists of rules, policies and demands is needed. These requirements are then formulated and categorized as the constraints of the model. Thus, in order to formulate a general mathematical model that can represent the whole problem that produces a university course timetable, it is necessary to observe and investigate thoroughly the possible requirements that exist in a university. As mentioned previously, the requirements employed in universities differ from one another. This is completely depending on the policies made by the top management, some of which is in determining the way classes are assigned. The way classes are assign could also be defined as meeting patterns. These meeting pattern requirements are much related to human factor since various group of individuals favor distinctive types of assignment combinations. Some take this issue lightly while some consider this as an important element to be acknowledged as it is a basic step in producing the well-being of graduates. The demands coming from the users are somewhat essential in pertaining this. This is the main reason of the research and it was motivated by Aizam and Caccetta [15]. The authors have successfully constructed a university course timetabling model that includes constraints that are found to be essential. As a continuation from the work, Abdul Aziz and Aizam [16] have conducted a survey on other requirements, which consist on the demands from users to be considered in the model. They observed, analyzed research articles and did their survey in the process of determination. However, the survey was conducted amongst universities in the East-Coast of the peninsular. Setting aside the sample of respondent, few more requirements were listed apart from the original that could be looked into. Below is the list of requirements that were gathered portraying the essential requirements that are mostly included in one's model, also the demands stated from the survey distributed:

a) Completeness: All events (lectures, tutorials, labs) are being assigned to specific slot and venue.
b) Room size limitation: Number of student of an event cannot exceed the capacity of a specific room.
c) Availability of the resources (time, rooms, and lecturers): Resources those are unavailable to be assigned of any events.
d) Conflict of resources (student groups, rooms and lecturers): Avoiding clashes between events for specific resources.
e) Working load (lecturers and students) in a day: Limitation of having to teach/attend more than the allowable number of events.
f) Maximum consecutive (lecturers and students) per day: Restriction in having number of consecutive events.
g) Meeting pattern: Assigning certain events in the same day.
h) Meeting pattern: Assigning certain events in different days.
i) Meeting pattern: Assigning events consecutively to one another.
j) Meeting pattern: Assigning events non-consecutively to one another.
k) Meeting pattern: Assigning certain events in morning and evening sessions on the same day (interval classes).

l) Meeting pattern: Assigning multiple events simultaneously (parallel assignments for large number of student enroll).
m) Meeting pattern: Avoiding assignments in late hours
n) Meeting pattern: Assigning one event before the other (precedence assignment for theory to happen before practical events).
o) Meeting pattern: Applying a day off between events of the same course.
p) Meeting pattern: Avoiding assignments during prayer times (applies to Muslim countries).
q) Meeting pattern: Assigning specific events (theory courses) to the morning sessions.
r) Meeting pattern: Assigning specific events (practical courses) to the evening sessions.
s) Meeting pattern: Maximum number of total events a lecturer can teach in a semester.
t) Meeting pattern: A room cannot have more than the maximum number of its capacity.
u) Meeting pattern: Assigning to at least a minimum number of events in a day.
v) Meeting pattern: Assigning more events earlier in the week. (Distribution: monotonically decreases throughout the week).

With the list of requirements presented, we will state a new list of requirements that we consider are more suitable to be included in the general model. We will eliminate some requirements that we thought are somehow contradicts, redundant or illogical. The constraints are related to the meeting patterns. Hence, 4 constraints have been removed:

I.   Maximum number of total subject a lecturer can teach in a semester.
II.   A room that is prohibited to have more than its maximum capacity.
III. Assigning to at least a minimum number of events in a day.
IV.   Monotonically decreases throughout the week.

In every university, each lecturer and programs have been designed beforehand in some specific manner. These numbers of events are distributed accordingly amongst the academic staffs and students to teach/attend weekly for the whole semester. Thus, constraint (i) is irrelevant in our opinion. We eliminate constraint (ii) as it is a similar definition to the earlier constraint that could be found in (b). When the datasets are large in numbers, the requirement in having the minimum number of events in a day (iii) is irrelevant. The large number will force the distribution of allocation to have more assignments of events in a specific day. Due to the limited resources, constraint (iv) is unable to be fulfilled. By having a large number of datasets, this type of constraint is too an irrelevant requirement as the assignments will be scattered throughout the week. Therefore, the new list of requirements excluding (s), (t), (u) and (v) and the inclusion of conflict constraint for rooms in (d) are used to be presented as a general model in our research.

Mixed Integer Linear Programming (MILP) approach

In Aizam and Caccetta [15], the authors have successfully constructed a university course timetabling model that includes essential constraints. These constraints are stated as the requirements that are most used in universities. However, with some arguments and concrete reasons, we have come up with a new list of requirements as given in the previous subsection. Before discussing the model in detail, the notation that will be used in the mathematical model is presented.

Notation

Sets and indices

C Courses offered

$r \in R$ Room type of different capacities and facilities

$l \in L$ Lecturers

$g \in G$ Student groups

$t \in T$ Timeslots available

$d \in D$	Days of the week
$c_b \in C$	Laboratory courses
C_l	Courses that are taught by lecturer $l, \forall l \in L$
C_g	Courses that have the same student group $g, \forall g \in G$
T_{lunch}	Timeslots for lunch break
T_d	Set of timeslots in day $d, \forall d \in D$
H	Set of courses in pairs (c_m, c_n) that needs to be assigned simultaneously in a timeslot, $\forall (c_m, c_n) \in C$
I	Set of courses in pairs (c_m, c_n) that needs to be assigned consecutively and in the same day, $\forall (c_m, c_n) \in C$
I'	Set of courses in pairs (c_m, c_n) that should not be assigned consecutively and not in the same day, $\forall (c_m, c_n) \in C$
O	Set of courses in pairs (c_m, c_n) that needs to be assigned on the same day, $\forall (c_m, c_n) \in C$
O'	Set of courses in pairs (c_m, c_n) that should not be assigned on the same day, $\forall (c_m, c_n) \in C$
J	Set of courses in pairs (c_m, c_n) that needs to be assigned one after another, $\forall (c_m, c_n) \in C$
K	Set of courses in pairs (c_m, c_n) that needs to be assigned in the morning and afternoon sessions, $\forall (c_m, c_n) \in C$
T_{eve}	Set of timeslots for evening sessions
T_{morn}	Set of timeslots for morning sessions
T_{late}	Set of timeslots for late evening sessions
T_l	Set of unavailable timeslots for certain lecturers
C_{prac}	Set of practical lectures
C_{theo}	Set of theory lectures

Parameter

RC_r	Capacity of room r
CS_c	Size of course c
$U_{\max}$	Maximum number of courses per day scheduled for lecturer l
$V_{\max}$	Maximum number of courses per day scheduled for student group g
MC_l	Maximum number of consecutive lectures per day for lecturer l
MC_g	Maximum number of consecutive lectures per day for student group g
$P_{c,t,r}$	Lecturers' preferences on having course c at timeslot t and at room r
$Q_{c,r}$	Course c are assigned at room r

Decision variable

$$X_{c,t,r} = \begin{cases} 1, \text{if a class } c \text{ is assigned to timeslot } t \text{ in room } r \\ 0, \text{otherwise} \end{cases} \qquad \forall c, \forall t, \forall r$$

Objective function

In this research, we chose to optimize the preferences of allocating classes to rooms and timeslots. According to the approach of assigning values of $P_{c,t,r}$, all lecturers will provide different levels of preferences for the time periods and rooms required, whereby value 5 is given to the most preferred slots and rooms, and value 1 otherwise. This information is the pre-processing data received before the scheduling process begins. In the model, these parameters are the values that reflect the priority of allocation of classes to the desired timeslots and rooms. These preferences are considered as soft constraints, namely the desirable type of constraint that can be treated as less significant; however, if fulfills could increase the level of satisfactory.

$$\text{Maximize } Z = \sum_c \sum_t \sum_r (P_{c,t,r} \cdot X_{c,t,r})$$

Mathematical model formulation

In UCTP, there are both basic and additional constraints considered. Basic constraints are the ones that are mostly used by researchers in their respective models. Three basic constraints commonly used in university course timetabling models found in the literature. These include requirements such as completeness, conflict and availabilities of resources. On the other hand, additional constraints are constraints that are closely related to meeting patterns. In this general mathematical formulation, we will gather and formulate both basics and meeting patterns type of constraints that arise in various applications in the literature. The objective is usually to optimize an objective function subject to these constraints which can be written mathematically as follows. The constraints for general model can be listed as in Table 1 below:

Table 1. Constraints in the general model

Requirement	Constraint		Eq.
All lectures are assigned to the respective timeslot and room (Completeness)	$\sum_t \sum_r X_{c,t,r} = 1$	$\forall c$	(1)
Number of students cannot exceed the room capacity (Room capacity)	$CS_c \cdot X_{c,t,r} \leq RC_r$	$\forall c, \forall t, \forall r$	(2)
Some timeslots are unavailable for the assignment of courses (Availability of timeslot)	$\sum_{c_b \in C} \sum_{t \in T_{un}} X_{c,t,r} = 0$	$\forall r$	(3)
Certain rooms are unavailable for the assignment of courses (Availability of room)	$\sum_t X_{c,t,r} = 0$	$\forall (c,r) \in Q_{c,r}$	(4)
Some lecturers are unavailable at a certain timeslot (Availability of lecturer)	$\sum_{t \in T_l} \sum_{c \in C_l} X_{c,t,r} = 0$	$\forall r, \forall l$	(5)
No student should attend more than one lecture in any timeslot (Conflict of student groups)	$\sum_{c \in C_g} \sum_r X_{c,t,r} \leq 1$	$\forall t, \forall g$	(6)
No room should be used for more than one lecture in any timeslot (Conflict of rooms)	$\sum_c X_{c,t,r} \leq 1$	$\forall t, \forall r$	(7)

No lecturer should teach more than one lecture in any timeslot (Conflict of lecturer)	$\displaystyle\sum_{c\in C_l}\sum_r X_{c,t,r}\le 1 \qquad\qquad \forall t,\forall l$	(8)
Each lecturer cannot teach more than the limited number of their workload per day (Workload of lecturers per day)	$\displaystyle\sum_{c\in C_l}\sum_{t\in T_d}\sum_r X_{c,t,r}\le U_{\max} \qquad \forall d,\forall l$	(9)
Each student group cannot attend more than the limited number of their workload per day (Workload of student groups per day)	$\displaystyle\sum_{c\in C_g}\sum_{t\in T_d}\sum_r X_{c,t,r}\le V_{\max} \qquad \forall d,\forall g$	(10)
Each lecturer cannot have more than the maximum number of consecutive lectures per day	$\displaystyle\sum_r\sum_{c\in C_l}(X_{c,t,r}+X_{c,t+1,r}+...+X_{c,t+MC_l,r})\le MC_l$ $\forall t\in\{t_1,t_2,...,t_{n_d-MC_l}\},\forall l$	(11)
Each student group cannot have more than the maximum number of consecutive lectures per day	$\displaystyle\sum_r\sum_{c\in C_g}(X_{c,t,r}+X_{c,t+1,r}+...+X_{c,t+MC_g,r})\le MC_g$ $\forall t\in\{t_1,t_2,...,t_{n_d-MC_g}\},\forall g$	(12)
Some lectures of the same course are to be scheduled on the same day (Same day)	$\displaystyle\sum_{t\in T_d}\sum_r(X_{c_m,t,r}-X_{c_n,t,r})=0 \qquad \forall(c_m,c_n)\in O,\forall d$	(13)
Some lectures of the same course must not be scheduled on the same day (Not same day)	$\displaystyle\sum_{t\in T_d}\sum_r(X_{c_m,t,r}+X_{c_n,t,r})\le 1 \qquad \forall(c_m,c_n)\in O',\forall d$	(14)
Some lectures of the same course are to be scheduled consecutively (Consecutive courses)	$X_{c_m,t,r}-X_{c_n,t+1,r}=0 \qquad\qquad \forall(c_m,c_n)\in I,\forall t,\forall r$	(15)
Some lectures cannot be scheduled consecutively (Non-consecutive lectures)	$\displaystyle\sum_r(X_{c_m,t,r}+X_{c_n,t+1,r})\le 1$ $\forall(c_m,c_n)\in I',\forall t\in\{1,2,...,n_d-1\},\forall d$	(16)
Interval between two lectures (morning and afternoon sessions)	$\displaystyle\sum_{t\in T_{mom}}X_{c_m,t,r}=\sum_{t\in T_{eve}}X_{c_n,t,r} \qquad \forall(c_m,c_n)\in K,\forall r$	(17)
Lectures with a large number of students are to be scheduled simultaneously (Simultaneously)	$\displaystyle\sum_r(X_{c_m,t,r}-X_{c_n,t,r})=0 \qquad\qquad \forall(c_m,c_n)\in H,\forall t$	(18)
Avoid lectures in late evening sessions	$\displaystyle\sum_r\sum_{t\in T_{late}}X_{c,t,r}=0 \qquad\qquad \forall c$	(19)
Some course must be assigned one after another (Precedence)	$\displaystyle X_{c_m,t,r}-\sum_{t=t+1}X_{c_n,t,r}=0 \qquad \forall(c_m,c_n)\in J,\forall t\in\{1,2,...,n-1\},\forall r$	(20)
A day off between two lectures of the same course (Gap)	$\displaystyle\sum_{t\in T_d}X_{c_i,t,r}+\sum_{t\in T_d+T_{d+1}}X_{c_j,t,r}\le 1 \quad \forall(c_m,c_n)\in O',\forall d,\forall r$	(21)
Some timeslots are set as the break hour and are unavailable for the assignment of courses (ex: prayer times)	$\displaystyle\sum_{t\in T_{lunch}}X_{c,t,r}=0 \qquad\qquad \forall c,\forall r$	(22)
Theoretical lectures must be scheduled in the morning session	$\displaystyle\sum_r\sum_{t\in T_{eve}}X_{c,t,r}=0 \qquad\qquad \forall c\in C_{theo}$	(23)
Practical lectures must be scheduled in the evening session	$\displaystyle\sum_r\sum_{t\in T_{mom}}X_{c,t,r}=0 \qquad\qquad \forall c\in C_{prac}$	(24)

Thus, the general model for university course timetabling problem can be written as below:

$$\text{Maximize } Z = \sum_{c}\sum_{t}\sum_{r}(P_{c,t,r} \cdot X_{c,t,r})$$

subject to:

Constraints (1) to (24)

and

$$X_{c,t,r} \in \{0,1\} \qquad \forall c, \forall r, \forall t$$

In this research, the general model is tested with a real Malaysian university dataset. The dataset comprises of courses offered in semester 1, 2017/2018 session and the list of rooms' capacities and facilities. However, only program core courses along as its lab requirements from four schools are considered in this case study. The overall data includes of 1,098 lectures (courses that are broken into number of elements according to the credit hours), 141 rooms (72 lecture rooms and 69 laboratory rooms), 55 timeslots (11 timeslots per day including lunch hours), 194 lecturers that taught the courses under the four schools and 124 student groups that enrolled into the same courses. Some data for certain constraints are randomly assigned, but with reference to other research done. The results obtained are discussed in the next section.

Results and discussion

The general model was tested with a real datasets and was solved using AIMMS. The next two subsections will respectively explain in detail regarding the AIMMS computational results and the performance analysis of the general model. The model's output will be analyzed based on four key themes (lecturers' preferences, classes, timeslots, and rooms).

Computational results

The general model was solved on a Core i7 computer with 3.40 GHz speed and 16GB of RAM using AIMMS optimization software and CPLEX 12.9 as a solver. The optimal solution of 5,353 was achieved after 19,085 iterations within 1,340.19 seconds. The relative gap between the 'Best LP Bound' and the 'Best Solution' is 0% which means that the assignments made are to the best possible time and rooms. Figure 1 shows the progress window of AIMMS towards the problem.
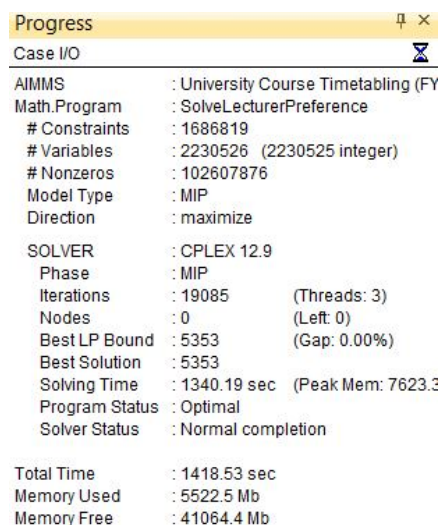

Figure 1. Progress window of AIMMS result.

Performance analysis

AIMMS assigns lecturer's preferences at uniformly random distribution. The preferences are set as an integer range from 1 (least preferred) to 5 (most preferred). Since the objective of the problem is to maximize the lecturers' preferences, an optimal solution of 5,353 from a total of 5,490 (if all courses were assigned to the most preferred timeslot, with 5 as its value) was achieved for the total of lecturers' preferences in the assignment of courses to timeslots and rooms. This shows that the majority of courses were assigned to the most preferred timeslots and rooms. Figure 2 shows the percentage for the lecturers' preferences. Mixed-integer linear programming (MILP) approach satisfies almost all of the lecturers' preferences for allocating courses to timeslots and rooms. 1,006 out of the 1,098 lectures (91.8%) were allocated to the most preferred timeslots. Meanwhile, 5.5% of lectures are allocated to the second most preferred timeslots. 1.7%, 0.6% and 0.4% of lectures are allocated to the 'no preference', not preferred and least preferred timeslots, respectively. The constructed course timetable satisfies the lecturers' preferences while still adhering to the model's requirements. Table 2 shows an example of one selected program's timetable generated by the general model. For the other programs, similar findings were obtained.
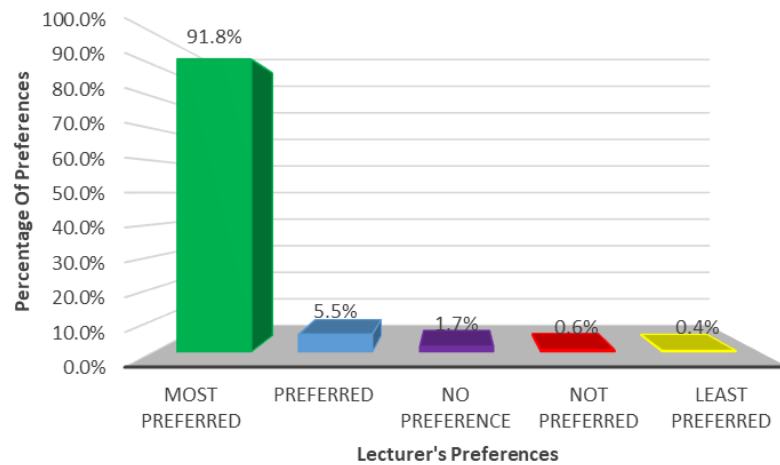

Figure 2. Percentage of lecturers' preferences in assignment of courses to timeslots and rooms.

Table 2 shows the course timetable generated by the general model for a program offered. The discussion and analysis will be referred solely to Table 2. All main requirements listed have been fulfilled. This alone has fit the purpose of having a timetable. We will highlight each requirement used and the outcome obtained systematically from the basic to the additional constraints. All courses were assigned in a timeslot and its room without exceeding the capacities with constraints (1) and (2). The conflict-free timetable was a result from constraints (6), (7) and (8) where a student group, a room and a lecturer must be assigned with only one course at a time. Constraints (3), (4), (5), (19) and (22) prevent the assignment of courses to the unavailable timeslots, rooms and lecturers. We can see that there are no courses assigned during the break hour and at the late slots, every day. Lectures were allocated to the rooms available, while lab courses were assigned to a specific lab rooms. To detail out the unavailable slots for a lecturer, one can see that there are no assignments made on Sunday (12 to 1 p.m.); Wednesday (8 a.m. to 12 p.m.); and Thursday (2 to 7 p.m.) for MTK3700 as the lecturer is unavailable at these timeslots. These assignments mentioned, sum up the basic requirements for the surveyed and most universities.

Similar approach of demonstrating is used in analyzing the additional constraints. These requirements are usually the meeting patterns according to each course's requirement. Some courses must be

Table 2. Timetable generated by the general model for a program.

Day	Year	8.00 am	9.00 am	10.00 am	11.00 am	12.00 pm	1.00 pm	2.00 pm	3.00 pm	4.00 pm	5.00 pm	6.00 pm
Sunday	1				MTK3100 (G3) AU 1-01	MTK3100 (G3) AU 1-01			MTK3100 (G3) CERMAT	MTK3100 (G3) CERMAT	MTK3100 (G3) CERMAT	
	2		MKP3100 (G2) BK 3-01	MKP3100 (G2) BK 3-01	MKP3100 (G1) KK 12	MKP3100 (G1) KK 12				MTK3102 (G3) BS	MTK3102 (G3) BS	
	3			MKP4300 (G1) BK 5-03	MKP4300 (G1) BK 5-03			MKP4600 (G1) BK 5-01	MKP4600 (G1) BK 5-01			
Monday	1						BREAK HOUR	MTK3200 (G3) IBH 9	MTK3200 (G3) IBH 9	MTK3700 (G3) BK 2-01	MTK3700 (G3) BK 2-01	
	2				MKP3200 (G1) BK 4 SMS & (G2) BK 4-04	MKP3200 (G1) BK 4 SMS & (G2) BK 4-04				MTK3701 (G3) KK 11	MTK3701 (G3) KK 11	
	3								MKP4300 (G1) CISCO	MKP4300 (G1) CISCO	MKP4300 (G1) CISCO	
Tuesday	1									MTK3400 (G3) BK 3-07	MTK3400 (G3) BK 3-07	
	2											
	3				MKP4300 (G2) IBH 6	MKP4300 (G2) IBH 6				MTK4700 (G3) KK 11	MTK4700 (G3) KK 11	
Wednesday	1							MTK3700 (G3) KK 12	MTK3400 (G3) CERMAT	MTK3400 (G3) CERMAT	MTK3400 (G3) CERMAT	
	2				MKP3100 (G1) KK 12	MKP3100 (G2) DS 1-01						
	3			MKP4300 (G1) KK 1								
Thursday	1	MTK3100 (G3) BK 4-01		MTK3700 (G3) MP 3	MTK3700 (G3) MP 3	MTK3700 (G3) MP 3		MTK3400 (G3) BK 3-04			MTK3200 (G3) IBH 9	
	2	MKP3200 (G1) BI 2-01 & (G2) BK 4-05							MTK3102 (G3) BK 5-02		MTK3701 (G3) BTB 4 PM	
	3		MKP4300 (G2) KK 13							MTK4700 (G3) AU 1-02	MKP4600 (G1) BK 5-04	

Indicator for the types of lecture:

▮ (teal)	Lectures of program core courses
▮ (orange)	Lab works

Indicator for the assigned lecture:

STM3107 (G1) →	Course code (Student Group)
MPRO →	Room for the assigned lecture

assigned consecutively for two, three or six hours on the same day. Constraints (13) and (15) ensure that the requirements are fulfilled. Combined with constraints (11) and (12), this is to ensure that only the maximum numbers of consecutive sessions are allowed. As for our research, it is limited to 4 hours maximum. This can be clearly seen in Table 2. The lectures that have to be assigned consecutively for two hours, MTK3100 are assigned from 11 a.m. to 1 p.m. on Sunday. Unlike constraint (13), constraint (14) prevents lectures of the same course from being assigned on the same day and for some, there is also demand of having at least a day-off in between that can be captured by constraint (21). This is displayed clearly in the timetable that the two lectures of MTK3100 were assigned separately to Sunday and Thursday. For the specific course, it is needed to be assigned in such way that the 2-hour slots must be assigned before the 1-hour slot. The precedence constraint in (20) will ensure this as the assignment made are on Sundays (2-hour) and Thursdays (1-hour). Some courses are prevented from being assigned consecutively. This occurs to pairs of courses for MKP3100 and MKP3200; MKP4300 and MKP3600. The requirement is fulfilled by constraint (16). Courses with a large number of students are divided into several student groups. These groups of students will have simultaneous lecture scheduled to different rooms. This situation is represented in constraint (18). As to demonstrate this, MKP3200 is divided into two student groups, (G1) and (G2), taught by two different lecturers. Both are assigned on Monday from 11 a.m. to 12 p.m. in two different rooms. As for differentiating the courses to theory and practical based courses for morning and afternoon slots respectively, constraints (23) and (24) will guarantee the assignments accordingly. Taking MKP3200 and MTK3700 as examples, both courses are assigned to their requirement, where the theory courses are placed in the morning session, while MTK3700 are placed in the afternoon. A requirement of having interval between two lectures in constraint (17) simply means to have both the theories and practical classes in the same day but separated between two sessions. This usually occurs to a course that consists of practical labs as can be seen for MTK3100 (G1). Constraint (13) is used together as to fulfill the same day demand. The 2-hour lectures and 3-hour lab were set to be assigned on the same day. In some institution, setting up the workload for lecturers and students are necessary. In this research, they have set-up workload restriction for lecturers and students. Lecturers are given a maximum of 5 slots lecturing per day, while the students are set to 7 hours of lecture to attend. These two requirements are captured in constraints (9) and (10). In the timetable produced, there are no days that have more than the allowed workload assigned for both lecturer and student group.

Conclusions

This research discussed an implementation of general mathematical model to the dataset of a Malaysian public university. A total of 24 requirements managed to be fulfilled in less than half an hour. The positive result of our case study can be viewed in terms of the program's timetable, solving time and lecturers' satisfaction level in assigning courses to timeslots and rooms. Conclusion can be made that the mathematical model developed are capable in solving real UCTP. The timetables produced were not only conflict-free, but managed to attain within a reasonable period of time, even for a large number of events. We have introduced general model with MILP approach for UCTP which includes every fundamental constraint required in various universities. Note that different institutions may need different features. Hence, the work done eases other users by extracting unrelated constraints. We modified the model developed in Aizam and Caccetta [15] through various ways in order to improve the current timetabling problem which considers only specific type of constraints to specific institution. Throughout the process, we had gone through intensive discussion to conclude in removing some of the requirements listed. With the outcome achieved, we expect that this model could be in favor of solving other university course timetabling problem with slight modifications. The ongoing research includes observing and investigating more requirements to be included or detailed out and further test to the compatibility of the model. In our opinion, we thought that the eliminated requirements in the methodology section can be detailed out and focused for each student groups and therefore be incorporated into the model. This also applies to the prayer break requirement whereby it should refer to specific individuals. Instead of categorizing the 'avoiding to have a late-hour assignments' under the hard constraints, we figured that it could be best considering under the optional constraint to reduce computational time. With these adjustments, the general mathematical model can somehow represent more of the requirements used by universities. A more concrete result can therefore be presented.

Data availability

All data used in this research were obtained from the Center for Academic Management and Quality (PPAK), UMT. The data includes offered courses of semester 1 2017/2018 session and list of rooms' capacities and facilities. In 2018, UMT is made up of schools, departments, institutes and a central administration. Eight schools provide a total of 27 degree programs. There have been 508 courses offered which are then broken into 2,895 lectures according to the credit hours. Only 1,098 lectures are used in this paper. However, please write to the author for more information on the data used.

Conflicts of interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

Funding statement

References

[1] N. L. Lawrie, "An Integer Linear Programming Model of a School Timetabling Problem," The Computer Journal, vol. 12, no. 4, pp. 307-316, 1969.

[2] S. Daskalaki, T. Birbas, and E. Housos, "An Integer Programming Formulation for a Case Study in University Timetabling," European Journal of Operational Research, vol. 153, no. 1, pp. 117-135, 2004.

[3] S. Daskalaki, and T. Birbas, "Efficient Solutions for a University Timetabling Problem Through Integer Programming," European Journal of Operational Research, vol. 160, no. 1, pp. 106-120, 2005.

[4] S. Ribić, and S. Konjicija, "A Two Phase Integer Linear Programming Approach to Solving the School Timetable Problem," in Proceedings of the ITI 2010, 32nd International Conference on Information Technology Interfaces, IEEE, pp. 651-656, 2010.

[5] V. O. Oladokun, and S. O. Badmus, "An Integer Linear Programming Model of a University Course Timetabling Problem," The Pacific Journal of Science and Technology, vol. 9, no. 2, 2008.

[6] S. A. Mirhassani, "A Computational Approach to Enhancing Course Timetabling with Integer Programming," Applied Mathematics and Computation, vol. 175, no. 1, pp. 814-822, 2006.

[7] G. Colajanni, and P. Daniele, "A New Model for Curriculum-Based University Course Timetabling", Optimization Letters, pp. 1-16, 2020.

[8] A. Lemos, P. T. Monteiro, and I. Lynce, "Disruptions in Timetables: A Case Study at Universidade de Lisboa", Journal of Scheduling, vol. 24, no. 1, pp. 35-48, 2021.

[9] W. Samarasekara, "An Application of Graph Coloring Model to Course Timetabling Problem", 2019.

[10] A. Gunawan, and K. M. Ng, "Solving the Teacher Assignment Problem by Two Metaheuristics", International Journal of Information and Management Sciences, vol. 22, no. 1, pp. 73, 2011.

[11] U. M. Modibbo, I. Umar, M. Mijinyawa, and R. Hafisu, "Genetic Algorithm for Solving University Timetabling Problem", Amity Journal of Computational Sciences, vol. 3, no. 1, pp. 43-50, 2019.

[12] M. Chen, X. Tang, T. Song, C. Wu, S. Liu, and X. Peng, "A Tabu Search Algorithm with Controlled Randomization for Constructing Feasible University Course Timetables", Computers & Operations Research, vol. 123, pp. 105007, 2020.

[13] A. Mahmud, "Highly Constrained University Class Scheduling using Ant Colony Optimization", International Journal of Computer Science & Information Technology, vol. 13, 2021.

[14] K. Y. Junn, J. H. Obit, and R. Alfred, "A Constraint Programming Approach to Solving University Course Timetabling Problem (UCTP)", Advanced Science Letters, vol. 23, no. 11, pp. 11023-11026, 2017.

[15] N. A. H. Aizam, and L. Caccetta, "Computational Model for Timetabling Problem," Numerical

Algebra, Control and Optimization, vol. 4, no. 1, pp. 269-285, 2014.

[16]   N. L. Abdul Aziz, and N. A. H. Aizam, "University Course Timetabling and the Requirements: Survey in Several Universities in the East-Coast of Malaysia," in AIP Conference Proceedings, AIP Publishing, vol. 1870, no. 1, pp. 040013, 2017.

[17]   N. L. A. Aziz, and N. A. H. Aizam, "A Brief Review on the Features of University Course Timetabling Problem," in AIP Conference Proceedings, AIP Publishing LLC, vol. 2016, no. 1, pp. 020001, 2018.