IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 7, 2021, accepted July 28, 2021, date of publication August 9, 2021, date of current version August 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3103495

A Mathematical Model for Course Timetabling Problem With Faculty-Course Assignment Constraints

H. ALGETHAMI[1], (Member, IEEE), AND W. LAESANKLANG[2], (Member, IEEE)
[1]Department of Computer Science, College of Computers and Information Technology, Taif University, Taif 21974, KSA
[2]Department of Mathematics, Faculty of Science, Mahidol University, Bangkok 10400, Thailand

Corresponding author: H. Algethami (hmgethami@tu.edu.sa)

This study was funded by the Deanship of Scientific Research, Taif University, KSA, under Grant 1- 441-10.

ABSTRACT University course timetabling problem (UCTP) includes the challenging task of generating an automated timetable for courses under resource limitations. Manual generated timetables might hold some errors and consume a long time to create feasible solutions. Thus, there is a need to find optimal and fast solutions for this problem. The difficulty of the timetabling problem is further increased when tackling faculty-related constraints, according to their requirements, preferences, and availability. Thus, student-related constraints are usually the focus of UCTP generated solutions, in which faculty constraints are limited to their teaching load or preferences. This paper proposes a multi-objective mixed-integer programming model for a preregistration UCTP, combined with faculty-related constraints. The goal is to maximize events assignments and faculty-members preferences satisfaction while balancing the university requirements. At the same time, student learning days and unassigned events are minimized. The model is tested with eight real-world instances. Computational experiments are carried out to show the efficiency of the model. The proposed method can generate optimal timetables for all problem instances that satisfy faculty constraints.

INDEX TERMS Combinatorial optimization, constraint satisfaction, faculty-course assignment, mathematical modeling, university course timetabling problem.

I. INTRODUCTION

The university course timetabling problem (UCTP) is an NP-hard combinatorial optimization [5], [14]. The problem has been the subject of several studies as it has many applications in real life [24], [26], [29], [32], [33].

UCTP includes the challenging task of generating an automated timetable for courses [5]. A set of entities, including courses, faculty-members, and students, must be allocated to limited resources such as campuses and teaching facilities [12]. There are several constraints for this problem, such as student numbers in a course, faculty-member availability, and room capacity. Solving this problem aims to find a solution satisfying all the constraints while minimizing the total costs.

Originally, course timetabling was planned manually by the administration team in the universities. There are many problems with the manual plan. For example, it can be time-consuming or generate inefficient resource allocations, affecting the institution's operational costs, such as allocating a large room to a class with a small group of students. A larger university usually has a more complicated problem. Thus, it requires more time to create a good assignment plan.

There is a need to implement an automated course timetabling tool that includes optimization to increase the assignment efficiency of resources shared by the events.

T This paper focuses on solving UCTP combined with a faculty-course assignment problem for real-world Taif University (TU) instances. The current timetabling method at TU requires administrators to manually generate the timetable, which must be completed before each semester's registration period. This manual process includes three primary stages:

- **Timeslots-Courses Assignment** Timeslots are allocated for each course, where each course has a specified number of timeslots per week. Conflicts must be

The associate editor coordinating the review of this manuscript and approving it for publication was Aniruddha Datta.

avoided, such as assigning two courses to one timeslot simultaneously.

- **Rooms-Courses Assignment** Rooms are allocated for each course at a specified timeslot, where each course is allocated in one room per event. Room requirements must be considered, such as capacity, suitability, availability, and other college-specific requirements. Conflicts must be avoided, such as assigning two courses to one room simultaneously.
- **Faculty-Courses Assignment** Faculty-members are allocated for each course, at a specified timeslot, and in a specified room. Faculty-member requirements must be considered, such as proficiency and other college-specific requirements, as well as course/campus preferences and availability. Conflicts must be avoided, such as assigning two courses to one faculty-member simultaneously.

UCTP handles the first two stages completely. However, the faculty-member assignment problem is often overlooked. By selecting a suitable faculty-member for each course, staff satisfaction will increase and enhance the learning experience [18]. Hence, this paper considers faculty-related constraints, including teaching load, requirements, preferences, and availability to teach at student campuses.

At TU, students are allocated to different campuses based on their gender. There are two campuses where faculty-member availability is restricted. Female faculty-members are not allowed to teach male students at the male campus. At the same time, male faculty-members can teach female students at the female campus, under some restrictions. This condition is not limited to TU or Saudi Arabia universities, but also many gender-basis academic institutions in different parts of the world, where staff capability to work or not at a specific location must be considered.

Solving the UCTP while focusing on faculty-related assignments was not widely investigated, especially when considering availability restrictions. The literature focused on the student-related constraints as a priority, in which faculty-member constraints are limited to their working hours or preferences. This paper presents a multi-objective mixed integer programming (MIP) model to solve the UCTP, combined with faculty-related constraints. Eight real-world instances are used as a case study from the B.Sc. Computer Science Department, College of Computers and Information Technology, TU, Saudi Arabia. This paper aims to generate a suitable timetable while satisfying all the constraints. To this end, the specific objectives of this study are:

- To present a MIP model for TU instances while considering faculty-related constraints.
- To carry out illustrative computational tests to show the utility of the proposed approach. A total of 8 problem instances from TU are used in the experimental study.

Following the introduction, the remainder of the paper is organized as follows: Section II reviews related work. Section III describes the problem and the instances used in this paper. Section IV presents the experimental study and discusses the obtained results. The paper is then concluded in Section V.

II. RELATED WORKS

Many models have been proposed to solve real-world applications of UCTP, such as [8], [13], [22], [30], [34], and [10]. However, researchers assumed that the faculty-related constraints had been tackled prior to solving the UCTP and focused on room capacity and student enrollment constraints [15], [23]. Thus, there are still some gaps between theoretical applications and real-world scenarios, especially in the case of prioritizing faculty-related constraints in UCTP [4].

Faculty-related constraints were usually tackled as part of the faculty-course assignment problem. It involved assigning courses to faculty-members at specific timeslots [2], [15], [28], [35]. The goal was to satisfy all constraints while balancing the teaching load.

Teaching-load constraints were the most common in UCTP problems. According to [26], faculty-member constraints that researchers used when tackling UCTP were in four forms: 1) faculty-member must deliver one event at a time, 2) one faculty-member of a given course must teach all events of that course during a week, 3) one faculty-member can teach different subjects, and 4) respect the maximum number of courses/hours they can teach in a week [3], [4], [8], [22], [24], [27], [31], [34].

In addition to the teaching load constraints, researchers considered faculty-related preferences constraints [11]. Such as having a day off or minimizing the number of working faculty-members per course [1]. The most common examples of faculty preferences constraints were related to teaching in a preferred room or at a preferred timeslot, as shown in work by [5], [9], [18], [21], [25], and [1]. Another example was to assign a faculty-member to preferred disciplines, as shown in work by [6], [13], and [18].

Faculty preferences constraints were usually addressed as a penalty function, i.e., penalizing not preferable assignments/day for a course taught by a faculty-member [16]. Hence, most UCTP problems that tackled faculty-members constraints were addressed as multi-objective problems to balance between the teaching load while maximizing their preferences [6], [20], [21], [32].

For example, the work by [32] demonstrated a solution for a multi-objective faculty-course assignment problem to replace the manual process. The aim was to assign faculty-members to courses based on competency, time availability, and teaching load. The objective function was to maximize faculty preferences on teaching a particular subject at a specific timeslot while minimizing working-day sections, i.e., morning, afternoon, or every day.

The faculty–course–timeslot assignment problem (FCT) was also studied by [20], using a multi-objective 0-1 linear programming model. In their work, both faculty-members and university administration preferences were considered. The aim was to minimize faculty preferences on teaching a course at a specific timeslot, faculty-members deviation from the upper load limits, university preference level on both the faculty–course assignments and the course timeslot assignments. However, faculty-members location availability constraints were not addressed. According to their preferences, the study had proven that obtaining feasible solutions can be challenging for some faculty-members, with a high satisfaction level. Thus, more considerations must be given to faculty-members-related constraints.

The work by [21] tackled a UCTP problem as a multi-objective mathematical model. Constraints tackled were related to timeslots-courses assignments, rooms-courses assignments, and faculty-members-courses assignments. They considered assigning courses to faculty-members with no overlapping assignments in one day and no overlapping courses assigned to one group in a semester. Two objectives were considered, maximize the preference level of faculty-member to work in a specific timeslot and minimize the total working days of faculty-members. Nevertheless, a case study of only 37 courses was considered without applying the faculty-members availability constraints.

Faculty location availability constraints were not widely used in the literature. The only location-related constraint considered was not having one faculty-member assigned in more than one place at the same time, i.e., one room, as in the research by [17]. They had addressed a collection of faculty-related constraints, such as assigning overlapping timeslots of the same course to different faculty-members and assigning specific courses of a specific timeslot to a specific faculty-member.

The MIP model proposed here tackles UCTP while focusing on faculty-members-related constraints. Including the teaching load and the university requirements considerations, i.e., their academic rank, contract type, and experience. Moreover, faculty preferences are considered, i.e., preferred courses and preferred campuses, i.e., location availability.

III. PROBLEM DESCRIPTION

This study focuses on solving the UCTP for Taif University (TU) real-world instances. There are unique features that differ TU-UCTP from the traditional UCTP, which are required by TU instances. A full description of the problem is explained in the following sections.

Each student at TU must follow an academic plan, where each plan holds information on the department courses. Such as course code, course name, prerequisite courses, and active course hours. Courses are ordered by the academic level, where each level holds the information of the courses in one semester. In this problem, S is a set of levels. A BSc. academic plan at TU has eight to ten levels, depending on the college. Table 1 shows the first four-level plan of the BSc. program in Computer Science, as used in the instances tackled in this paper.

TABLE 1. Four-level plan of the B.Sc. program in Computer Science.

Level	*Course Code*	*Course name*	*Prerequisite*	*Hours*
1	2004111-2	Islamic Culture (Ethics and Values)		2
	202126-3	Fundamentals of mathematics		3
	501110-2	Introduction to problem solving		2
	501112-2	Computer skills		2
	990311-2	University skills		2
	999805-2	English for intensive academic purposes 1		2
2	105115-2	Kingdom History		2
	201121-3	General biology		3
	204124-2	General chemistry		2
	501125-2	Scientific computing		2
	503121-1	Computer design		1
	990211-2	Arabic language skills		2
	999806-2	English for intensive academic purposes 2	999805-2	2
3	202261-3	Calculus 1		3
	203206-4	Physics 1 T		3
	203206-4	Physics 1 L		1
	501215-3	Discrete Structures		3
	501220-3	Computer Programming 1 T		2
	501220-3	Computer Programming 1 L		1
	999811-2	Special English for computers 1	999806-2	2
4	204112-2	Islamic culture (ethics and values)	2004111-2	2
	202263-3	Calculus 2	202261-3	3
	203207-4	Physics 2 T	203206-4	3
	203207-4	Physics 2 L	203206-4	1
	501222-3	Computer Programming 2 T	501220-3	2
	501222-3	Computer Programming 2 L	501220-3	1
	503221-4	Digital Logical Design T	501215-3	3
	503221-4	Digital Logical Design L	501215-3	1
	999808-2	Special English for computers 2	999811-2	2

A student must pass lower-level courses before registration of the higher-level courses. Courses at a higher level usually have course prerequisites from the lower-level courses where students must qualify or be graded as pass. Exceptions may apply to an individual who had failed prerequisites. Note, the prerequisite courses are not considered in this paper as they are not relevant to the generation of the general timetable while considering faculty-course assignments.

The administration team generates a department timetable at the start of each semester. The plan is for the weekly distribution of courses over 15 weeks. A complete college timetable holds all the levels information, with all the courses of the departments. It can be up to 50 to 60 pages long. Hence, an example of the timetable is shown in Table 2, which shows information for each course in one level for the female students. Including the student group number, course number, course name, duration-time, pattern type, the maximum number of students to be registered, current number of students, timeslots (by day and time), rooms, and faculty-members allocated to teach this course.

One level can be divided into more than one group of students that follows the same plan, where each group shares the same resources. Courses are available for student groups at different times. Courses must not be scheduled at the same timeslot unless there are for different groups or levels. This is because students in the same group must attend all the courses available at one level. If two courses in one group level are scheduled simultaneously, then it is a conflict. Not only that, but conflicts also include allocating two courses in a room at the same time and assigning a faculty-member to teach two courses at the same time [7]. Usually, there is at least one group for one course in one level per gender. This differs every year due to the number of students registered in a level. In this study, we focus on having only two groups in one level: one for male students S^M, and the other is for female students S^F.

Each course $c \in C$ requires h_c hours of study. This is defined as course duration time or activity hours. Courses can

TABLE 2. An example of the manual computer science department timetable.

Course							Time			Room	Faculty-member
Group Code	*Course Code*	*Course name*	*Course activity hours*	*Pattern*	*Course Capacity*	*Registered students*	*Day*	*From*	*To*	*Id*	*Id*
5730	2004111-2	Origins of Islamic Culture	2	*Lecture*	250	244	Sun	13:00	15:00	room1	F10
5731	2004111-2	Origins of Islamic Culture	2	*Lecture*	250	199	Sun	15:00	17:00	room1	F10
6813	202126-3	Fundamentals of mathematics	3	*Lecture*	40	29	Sun Mon	8:00 8:00	10:00 9:00	room3	F4
6814	202126-3	Fundamentals of mathematics	3	*Lecture*	40	40	Sun Mon	8:00 8:00	10:00 9:00	room2	F5
6866	501110-2	Introduction to problem solving	2	*Lecture*	40	35	Wed	15:00	17:00	room1	F10
6867	501110-2	Introduction to problem solving	2	*Lecture*	40	40	Mon	9:00	11:00	room1	F10
5711	501112-2	Computer skills	2	*Lecture*	15	15	Sun	15:00	17:00	room2	F10
5712	501112-2	Computer skills	2	*Lecture*	15	15	Wed	17:00	18:00	room2	F10
5719	990311-2	University skills	2	*Lecture*	30	30	Mon	14:00	16:00	room1	F8
5720	990311-2	University skills	2	*Lecture*	30	28	Mon	12:00	14:00	room1	F8
5751	999805-2	English for intensive academic purposes 1	2	*Lecture*	30	26	Wed	10:00	12:00	room1	F7
5752	999805-2	English for intensive academic purposes 1	2	*Lecture*	30	21	Sun	10:00	12:00	room1	F7

also be defined by their patterns, i.e., lecture courses (C^J) or laboratory courses (C^I).

Students number in a group is divided into two types: the maximum number of students, i.e., course capacity, and the current number of registered students. This information can benefit academic advisors who can recommend a course to a student. Nevertheless, in this paper, only course capacity, presented by parameter g_c, is used. This is because the goal is to generate a course timetable preregistration while addressing faculty-members constraints. Thus, information regarding registered students is not available, and it is not required at this stage.

In terms of facility, TU has a set of rooms R for conducting academic events. Each room $r \in R$ has a maximum capacity, denoted as b_r, that should not be exceeded. Since TU has two main campuses, one for female and one for male students, mixed-gender rooms are not allowed. Thus, at least two different rooms of the same course must be assigned.

Rooms are divided into lecture rooms and laboratory rooms, i.e., labs. Both identified as a set of room R^J and R^I respectfully. Each room is equipped with learning tools such as a projector, smart-board, video conference system. The learning tools used in a room are identified as features, where E is a set of all room features. However, not every feature is available in every room. For example, physics labs require different equipment than computer labs. Hence, $k_{r,e}$ is a binary parameter indicating that a room r is equipped with a feature e. Thus, courses require feature e may use that room. Note that $u_{c,e}$ is a binary parameter indicating that a course c requires a feature e.

TU has a set of faculty-members L, where L^M is a set of male faculty-members, and L^F is a set of female faculty-members. Each faculty-member $l \in L$ has maximum teaching hours per week, presented by the parameter w_l and maximum teaching hours per day, presented by the parameter k_l. Usually, male faculty-members are more in numbers in comparison to female faculty-members, especially in STEM departments. Thus, a female faculty-member can teach at the female campus only, while male faculty-members can teach at both campuses. Female campus assignment is allowed for male faculty-member if their academic rank, presented by the parameter a_l, is an assistant professor or above. Still, female faculty-members have the priority to be assigned at the female campus if they are available.

This paper aims to generate a timetable for the Computer Science department, considering the academic plan presented in Table 1. A mixed-integer programming model (MIP) is developed for the TU real-world instances, as MIP is commonly used in the literature [5], [7]. The model notation, constraints, and objectives are explained in the following section.

A. PROBLEM CONSTRAINTS AND OBJECTIVES

The solution for the UCTP is a weekly course schedule that allocates a set of courses C to a set of faculty-members L. Course assignment must be planned on a set of timeslots P, at a set of rooms R with a set of room features E. A good quality solution should assign all courses while satisfying all the constraints without any conflicts.

Table 3 shows all the constraints considered in this research. There are fifteen hard constraints and two soft constraints. The objective function is tackled as a multi-objective weighted sum method that measures the quality of the timetable. The goal is to maximize event and faculty assignments according to their preferences and university requirements, including academic ranks, experience level, and contract type. Additionally, a penalty cost is minimized, including the total number of active learning days per level and the total number of non-assigned events.

B. NOTATION

As is commonly practiced in Saudi Arabia, working days start by Sunday and end by Thursday, while the weekends are Friday and Saturday. Therefore, $D = \{1, 2, 3, 4, 5\}$ is the set of working days, where Sunday is one and Thursday is five. Each day $d \in D$ is divided into $|P|$ timeslots. Timeslot $p \in P$ can be categorized into two intervals: day periods – 8:00 AM to 12:00 PM, and evening periods – 1:00 PM to the end of the day. We define timeslots during the day as P_d and the timeslots during the evening period as P_e.

The notations used in the model are shown in Table 4 and described in the following sections. Event assignment, is represented by the decision variable $x_{d,p,l,c,r,s}$, where it is equal to one if a course $c \in C$, taken by a student group in a level $s \in S$, is scheduled to be taught by a faculty-member $l \in L$ on a day $d \in D$, at a timeslot $p \in P$ in a room $r \in R$.

TABLE 3. Objectives and Constraints in UCTP.

Objectives
Maximise event assignments (31). Maximise faculty-members course preferences in relation to the faculty-members university requirements, i.e., academic ranks, experience level and contract type (31). Minimise the total number of unassigned events (31). Minimise the total number of days that student must visit the campus per level (31).
Hard constraints
No two events included in a single level are taught at the same time (1). Each course must be assigned to only one faculty-member (2). Each course must be assigned to only one room (3). Two or less credit hour courses are scheduled in the afternoon/evening (4). Three or more credit hours courses are scheduled in the morning (5). Respect the plan sequence connectivity, (6),(11), (12) and (13) . Assign all the course teaching credits (10) Students in one level should not have more than two consecutive periods of the same course in the same day (8) and (9). An hour break must be scheduled on Wednesday between 12:00 and 14:00 (15). Respect course capacity for each room (16) Assign courses to suitable rooms (20) and (21). Assign courses to available faculty-member (24) and (25) Respect maximum credit hours per week for faculty members (26). Respect maximum credit hours per day for faculty members (27). Assign courses to suitable faculty members (30).
Soft constraints
Students in one level are not preferred to have more than two consecutive days (7). Respect maximum credit hours per day for each level (14).

TABLE 4. Notations for the UCTP model.

Notations	Description
Sets	
P	Set of timeslots for each day.
D	Set of days in a week.
L	Set of faculty-member.
L^M, L^F	Sets of male and female faculty-member.
C	Set of courses.
C^I, C^J	Set of laboratory courses and lecture courses.
R	Set of rooms.
R^I, R^J	Set of laboratory rooms and lecture rooms.
E	Set of room features.
S	Set of student levels.
S^M, S^F	Set of male students and female student levels.
Parameters	
$k_{r,e}$	Binary parameter indicates that room r has feature e.
$u_{c,e}$	Binary parameter indicates that course c requires feature e.
$q_{l,s}$	Binary parameter indicates that faculty-member l can teach student level s.
$o_{c,l}$	Binary parameter indicates that course c is preferred be taught by faculty-member l.
b_r	Room r maximum capacity.
h_c	Course c active learning hours.
g_c	Course c maximum capacity.
w_l	Maximum working hours per week for faculty-member l.
k_l	Maximum working hours per day for faculty-member l.
β_s	Maximum active learning hours per day for student level s.
a_l	Academic rank of faculty-member l.
v_l	Experience level of faculty-member l.
α_l	Contract type of faculty-member l.
M, μ	Large constants.
ω	Constraint violation weight.
Variables	
$x_{d,p,l,c,r,s}$	Decision variable deciding that a faculty-member l is assigned to teach a course c in room r to student level s at period p on day d.
$y_{c,s}$	Integer variable for the number of non-assigned events for course c that is offering to student level s.
$\zeta_{s,d}$	Binary decision variable indicating student level s has courses to study on day d.
$\phi_{s,r}$	Binary decision variable indicating student level s uses room r.
$\tau_{l,c,s}$	Binary decision variable indicating student level s studies course c with faculty-member l.
$\delta_{d,c,s}$	Integer variable for timeslot differences of course c that offered to student level s on day d.

C. MATHEMATICAL MODEL FORMULATION

In the light of the above definitions, the proposed MIP model is given as follows. At most, one course must be assigned to only one faculty-member in one room at each timeslot. Hence, a faculty-member could not attend two events at the same time, as shown in (2). A course is taught once per timeslot/room for one class, as shown in (1), and could not be taught in two different rooms at the same time, as shown in (3).

$$\sum_{l\in L}\sum_{c\in C}\sum_{r\in R} x_{d,p,l,c,r,s} \leq 1, \quad \forall d \in D, p \in P, s \in S \quad (1)$$

$$\sum_{c\in C}\sum_{r\in R}\sum_{s\in S} x_{d,p,l,c,r,s} \leq 1, \quad \forall d \in D, p \in P, l \in L \quad (2)$$

$$\sum_{l\in L}\sum_{c\in C}\sum_{s\in S} x_{d,p,l,c,r,s} \leq 1, \quad \forall d \in D, p \in P, r \in R \quad (3)$$

Courses are composed of theory or lab sessions. Theory sessions are set to two-hour credits or more, and it is held in lecture rooms. Lab and practical sessions are set to one-hour credit, and it is held in a laboratory. Courses with three-hour credits or more are usually assigned in the morning because students are more focused than in the afternoon. It can also affect the student performance during the session. Hence, if a course c is a theory course with three-hour credits or more, it is required to have a timeslot $p \in P_d$. However, if a course is a practical course or has less than three-hour credits, it can be scheduled at a timeslot $p \in P_e$, as shown in (4) and (5).

$$x_{d,p,l,c,r,s} = 0$$
$$\forall d \in D, p \in P_e, l \in L, c \in C, r \in R, s \in S, h_c > 2 \quad (4)$$

$$x_{d,p,l,c,r,s} = 0$$
$$\forall d \in D, p \in P_d, l \in L, c \in C, r \in R, s \in S, h_c \leq 2 \quad (5)$$

Constraint (6) denotes the day that students must turn up for course sessions, i.e., classes. Binary variable $\zeta_{s,d}$ indicates that a student group in q level s must turn up for classes on a day d. In the objective function, the summation of $\zeta_{s,d}$ is minimized, i.e., the number of active learning days of student groups in a level.

$$\sum_{r\in R}\sum_{c\in C}\sum_{l\in L}\sum_{p\in P} x_{d,p,l,c,r,s} \leq |P| \cdot \zeta_{s,d}, \quad \forall d \in D, s \in S \quad (6)$$

For each day d, classes of the same course should be scheduled consecutively. However, a class cannot have the same course for more than two periods a day. Hence, Constraint (7) measures the maximum period differences of classes scheduled for the same course. Integer variable $\delta_{d,c,s}$ represents the maximum time differences between two periods. This constraint encourages assignments of the same course on consecutive timeslots.

$$(p_1 - p_2) \times (x_{d,p_1,l,c,r,s} + x_{d,p_2,l,c,r,s}) - 1 \leq \delta_{d,c,s},$$
$$\forall c \in C, l \in L, s \in S, d \in D, r \in R, p_1, p_2 \in P \quad (7)$$

A course with one-hour credit can only be scheduled in one timeslot, which refers to only one day. A course with two-hour credits must be scheduled using two consecutive timeslots on a day d, as shown in (8). A course with more than two-hour credits is divided into 1 or 2 hours meetings per day, and the residue of the credit hours is scheduled into a different day, as shown in (9). Constraint (10) guarantee that the number of assignments must be equal to the number of credit hours. The unassigned events $y_{c,s}$ must be minimized. Hence, in the objective function, the summation of $y_{c,s}$ is minimized, i.e., the total number of unassigned events.

$$\sum_{l\in L}\sum_{p\in P}\sum_{r\in R} x_{d,p,l,c,r,s} = 2 \cdot \delta_{d,c,s},$$
$$\forall c \in \{c_i | h_{c_i} \leq 2, c_i \in C\}, d \in D, s \in S \quad (8)$$

$$\sum_{l\in L}\sum_{p\in P}\sum_{r\in R} x_{d,p,l,c,r,s} \leq 2, \quad \forall c \in C, d \in D, s \in S \quad (9)$$

$$\sum_{d\in D}\sum_{l\in L}\sum_{p\in P}\sum_{r\in R} x_{d,p,l,c,r,s} + y_{c,s} = h_c, \quad \forall c \in C, s \in S \quad (10)$$

One faculty-member is assigned to one course for all timeslots to ensure the plan consecutively, as shown in (11) and (12), even if the course is divided into multiple periods or allocated on different days.

$$\sum_{d\in D}\sum_{r\in R}\sum_{p\in P} x_{d,p,l,c,r,s} \leq h_c \cdot \tau_{l,c,s}, \quad \forall c \in C, l \in L, s \in S \quad (11)$$

$$\sum_{l\in L} \tau_{l,c,s} \leq 1, \quad \forall c \in C, s \in S \quad (12)$$

Likewise, to ensure the plan consecutively, one room is selected for a course for all timeslots, as shown in (13). In one level, no more than 8 hours per day is scheduled, as shown in (14). Note that all student classes have a break period from 12:00 PM - 2:00 PM on Wednesdays, as shown in (15).

$$\sum_{d\in D}\sum_{l\in L}\sum_{p\in P}\sum_{c\in C} x_{d,p,l,c,r,s} \leq M \cdot \phi_{s,r}, \quad \forall r \in R, s \in S \quad (13)$$

$$\sum_{d\in D}\sum_{c\in C}\sum_{l\in L}\sum_{r\in R} x_{d,p,l,c,r,s} \cdot h_c \leq \beta_s, \quad \forall p \in P, s \in S \quad (14)$$

$$\sum_{r\in R}\sum_{l\in L}\sum_{c\in C} x_{d,p,l,c,r,s} = 0,$$
$$\forall s \in S, d = \text{“}Wed\text{”} \in D, p = \{5, 6\} \in P \quad (15)$$

Each room r has a capacity b_r that must be proportional to the maximum number of students g_c, who can attend a given course c. Thus, room capacity b_r must not be less than course capacity g_c, as shown in (16).

$$\sum_{c\in C}\sum_{l\in L}\sum_{s\in S} x_{d,p,l,c,r,s} \cdot g_c \leq b_r \quad \forall p \in P, d \in D, r \in R \quad (16)$$

In the same way, an assignment must consider the suitability of a room for a course. Each room r has a set of features E. Thus, the room-feature matrix $k_{r,e}$ is used to indicate that a room r has a feature e, as shown in (17). Each course c has also a set of required features E. The course-feature matrix $u_{c,e}$ is used to indicate that a course c require a feature e, as shown (18).

$$k_{r,e} = \begin{cases} 1 & \text{if a room } r \text{ has a feature } e \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

$$u_{c,e} = \begin{cases} 1 & \text{if a course } c \text{ requires a feature } e \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

Given a feature e, is required to be available to assign a course c to room r, then e must exists in a room r.

For example, a male faculty-member of the assistant professor's rank or above can teach at both campuses. If a male faculty-member is teaching in the female campus, the room must have a feature of a remote broadcast network, i.e., "LAN", as shown in (19). Other features might include whiteboards, projectors, computers, or physics equipment, as shown in (20). The room type must also be compatible with the course pattern of teaching. Hence, assigning laboratory courses to lecture rooms or assigning lecture courses to laboratory rooms is not allowed, as shown in (21) and (22).

$$\sum_{c\in C}\sum_{d\in D}\sum_{p\in P} x_{d,p,l,c,r,s} \leq k_{r,\text{``LAN''}}, \quad \forall r \in R, s \in S^F, l \in L^M \tag{19}$$

$$\sum_{l\in L}\sum_{s\in S} u_{c,e} \cdot x_{d,p,l,c,r,s} \leq \sum_{l\in L}\sum_{s\in S} k_{r,e} \cdot x_{d,p,l,c,r,s}, \quad \forall e \in E, p \in P, d \in D, r \in R, c \in C \tag{20}$$

$$\sum_{s\in S}\sum_{d\in D}\sum_{p\in P}\sum_{l\in L} x_{d,p,l,c,r,s} = 0, \quad \forall c \in C^I, r \in R^J \tag{21}$$

$$\sum_{s\in S}\sum_{d\in D}\sum_{p\in P}\sum_{l\in L} x_{d,p,l,c,r,s} = 0, \quad \forall c \in C^J, r \in R^I \tag{22}$$

Assigning faculty-members must consider their level availability, where each level has a set of available faculty-members, who can teach in the female campus ca^F, the male campus, or both. The $q_{l,s}$ is the faculty-level matrix element, that indicates if a faculty-member $l \in L$ can teach a course c in a level $s \in S$, as shown in (23).

$$q_{l,s} = \begin{cases} 1 & \text{if a faculty-member } l \text{ can teach in a level } s \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

The assignment must be less or equal to $\mu \cdot q_{l,s}$, to use the faculty-members who are available to teach a student group in a level, as shown in (24). Where μ is a large number and $q_{l,s}$ a binary parameter indicating that a faculty-member l is available to teach a course for a student group in a level s. A female faculty-member can only teach in female campus, as shown in (25). Thus, their availability is limited to female student groups S^F. On the other hand, male faculty-members can teach on both campuses, but not all of them can teach at the female campus. Only male faculty-members of an academic rank of an assistant professor or above can teach female student groups S^F.

$$\sum_{c\in C}\sum_{r\in R}\sum_{d\in D}\sum_{p\in P} x_{d,p,l,c,r,s} \leq \mu \cdot q_{l,s}, \quad \forall l \in L, s \in S \tag{24}$$

$$\sum_{d\in D}\sum_{p\in P}\sum_{c\in C}\sum_{r\in R} x_{d,p,l,c,r,s} = 0, \quad \forall l \in L^F, s \in S^M \tag{25}$$

Suppose a faculty-member is available to teach at a level S_i^F. In that case, the timetable assignment must consider their maximum working hours w_l per week, as shown in (26), and their maximum working hours k_l per day, as shown in (27).

$$\sum_{d\in D}\sum_{c\in C}\sum_{r\in R}\sum_{s\in S} x_{d,p,l,c,r,s} \cdot h_c \leq w_l, \quad \forall l \in L, p \in P \tag{26}$$

$$\sum_{p\in P}\sum_{c\in C}\sum_{r\in R}\sum_{s\in S} x_{d,p,l,c,r,s} \cdot h_c \leq k_l, \quad \forall d \in D, l \in L \tag{27}$$

University requirement constraints are also considered in the faculty-member assignments. Academic ranks a_l, years of experience in teaching v_l, contract type α_l, and preferred courses list. Academic rank credit hours per week are 16, 14, 12, and 10 for teaching assistants, assistant professors, associate professors, and professors. If many members of equal rank are interested in a course, teaching assignment is given according to seniority in the appointment to the scientific position, i.e., teaching experience. Note that faculty-member of academic rank "teacher assistant" must teach laboratory courses, as shown in (28).

$$\sum_{r\in R^J}\sum_{d\in D}\sum_{p\in P} x_{d,p,l,c,r,s} = 0, \quad \forall l \in L, c \in C^I, s \in S, a_l \geq 2 \tag{28}$$

Priority is also evaluated based on the contract type. There are two types of contracts, permanent and on-roll contracts, where permanent contract holders are preferred when assigning courses. On-roll contracted faculty-members are assigned to courses if permanent contract members are not available on assignment. Unless it is on a female campus, and there is a female faculty-member. Each faculty-member has a list of preferred courses that they can teach. The $o_{c,l}$ is the course-faculty matrix, where a course c is preferred to be taught by a faculty-member l, as shown in (29). Equation (30) shows the assignment based on course preferences. It is assumed that each faculty-member can give more than one specific course. However, each faculty-member may or may not be able to teach all the preferred courses.

$$o_{c,l} = \begin{cases} 1 & \text{if a course } c \text{ is preferred by a faculty-member } l \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

$$x_{d,p,l,c,r,s} \leq o_{c,l}, \quad \forall c \in C, l \in L, s \in S, d \in D, r \in R, p \in P \tag{30}$$

The quality of the timetable is measured by evaluating the objective function and penalizing each non-assigned event by '1', as shown in (31).

Objective function is constructed as follows:

- Maximizing:
  - Event assignments, which is represented by the decision variable $x_{d,p,l,c,r,s}$ which equals to one when course $c \in C$ is planned on day $d \in D$ at timeslot $p \in P$ in a room $r \in R$ to be taught by a faculty-member $l \in L$ of a student group level $s \in S$.
  - Faculty-members assignments, according to their preferences $o_{c,l}$ and in relation to the faculty-members requirements, including academic ranks a_l, experience level v_l and contract type α_l.

TABLE 5. Eight UCTP instances features and sizes.

		TUA1	*TUA2*	*TUB1*	*TUB2*	*TUC1*	*TUC2*	*TUD1*	*TUD2*
Number of Levels		2	2	4	4	6	6	8	8
Number of Courses		12	12	26	26	40	40	58	58
Number of Events		26	26	54	54	96	96	132	132
Number of Time-slots		6	6	8	8	10	10	12	12
Number of Rooms	*Total*	3	5	3	5	9	9	8	9
	Female Campus	2	2	2	2	5	4	4	4
	Male Campus	1	3	1	3	4	5	4	5
Number of Features		3	3	4	4	5	5	5	5
Number of Faculty-members	*Total*	8	10	14	16	17	20	19	23
	Female	2	4	3	9	5	10	8	12
	Male	6	6	11	7	12	10	11	11

- Minimizing:
  - The number of days that student must visit the campus per level $\zeta_{s,d}$.
  - The total number of non-assigned events $y_{c,s}$.

$$
\begin{aligned}
\text{Maximize} \sum_{r\in R}\sum_{c\in C}\sum_{l\in L}\sum_{p\in P}\sum_{d\in D}\sum_{s\in S} & \{x_{d,p,l,c,r,s} \\
& + \left(o_{c,l}\cdot a_l + o_{c,l}\cdot v_l + o_{c,l}\cdot \alpha_l\right) x_{d,p,l,c,r,s}\} \\
& - \omega \sum_{d\in D}\sum_{s\in S}(\zeta_{s,d} + y_{c,s}) \qquad (31)
\end{aligned}
$$

D. PROBLEM INSTANCES

In this study, eight TU real-world problem instances are used as instances of UCTP combined with a faculty-course assignment problem. Most departments at TU, from scientific and engineering colleges, share courses in the first two years, such as Islamic culture, English, Arabic, biology, history, chemistry, physics, and mathematics. Hence, the problem instances selected from the B.Sc. program of the Computer Science Department, College of Computers and Information Technology, are used as a case study of the first and second-year plan. Table 5 shows the main features of each problem instance. The TUA1 instance is considered the smallest, while the TUD2 instance is the largest.

In these instances, there are two main campuses. Female and male student groups are allocated to different campuses. Still, they have the same number of levels, courses, events, timeslots, and room features. For example, TUA1 and TUA2 have two levels and 12 courses. This means there is one level $s = 1$ for each student group. Also, there are six courses in level ${s_1}^F$ and six courses in level ${s_1}^M$.

The number of timeslots identifies the maximum number of timeslots per day available for that instance. For TUA1 and TUA2, there are 6 timeslots $\times$ 5 days $=$ 30 timeslots per week. Timeslots available per week for each instance are 30, 30, 40, 40, 50, 50, 60, and 60, respectively. The number of events shows the total number of course activity hours assigned for all courses. For example, a course with three timeslots requires three events to be scheduled. Hence, for 12 courses, there are 26 events.

Differences between instances in Table 5 are found in the form of the number of rooms and faculty-members, where there are different requirements for each campus. In the female campus, female and male faculty-members are utilized. However, in the male campus, male faculty-members are utilized only due to the university regulations. Female campus rooms require "LAN" connections to be used by male faculty-members. Thus, at least one room in the female campus has to be equipped with a "LAN" connection.

TABLE 6. Optimal results for eight UCTP instances, showing objective value, mean objective value, computational time (in seconds) and number of violations.

	TUA1	*TUA2*	*TUB1*	*TUB2*	*TUC1*	*TUC2*	*TUD1*	*TUD2*
Obj.	144	138	398	350	666	662	1026	987
Mean obj.	85.71	70.33	209.6	219.14	407	481.44.8	754.11	677.86
Cpt.(Sec)	1.16	2.56	27.27	541.75	15.59	51.58	150.33	172.09
Violations	0	0	0	0	0	0	0	0

IV. COMPUTATIONAL EXPERIMENTS

The implementation has been evaluated using Windows 10, 64 operating system with IBM ILOG CPLEX optimization studio version 12.10.0 [19]. Restricted to run on a single desktop PC processor equipped with Intel Core i7 at 2.21 GHz and 16 GB of RAM. Using the data in Table 5.

Table 6 presents optimal results by the proposed mixed-integer programming (MIP) model on eight TU different problem instances, shown in Table 5. For each problem instance, the table shows the solution quality, noted as *Obj*, the mean objective value, noted as *Mean Obj*, the computational time in seconds, noted as *Cpt*, in which the best solution was found, and the number of violations. From the table, it can be concluded that there were no course conflicts or any other violations on any of the problem instances. Also, all courses were assigned to timeslots, rooms, and faculty-members. Hence, the proposed method obtained optimal solutions for all TU instances. Closer inspection of the results indicated a slight increase in the objective values in instances TUA1, TUB1, TUC1, and TUD1 than instances TUA2, TUB2, TUC2, and TUD2. This was because fewer faculty-members were available under those instances. Hence, violations of faculty-members preferences and requirements were more computationally expensive and affected the objective function value, where more resources were available.

Table 6 also shows that the computational times were longer in instances TUA2, TUB2, TUC2, and TUD2 than instances TUA1, TUB1, TUC1, and TUD1. More time was spent on finding suitable faculty-members under the university regulations. More time was also spent avoiding course conflicts and maintain the timetable's feasibility by obtaining a higher satisfaction level. Hence, the solver spent more time searching for an optimal solution because more resources were available on those instances.

Figure 1 illustrates the number of used faculty-members. Each sub-figure corresponds to a problem set, and each bar presents the number of used faculty-members in the student group levels. Each bar color and pattern indicate the following: the number of female faculty-members is presented as yellow bars with horizontal lines for all the female levels and blue bars with dots for all the male

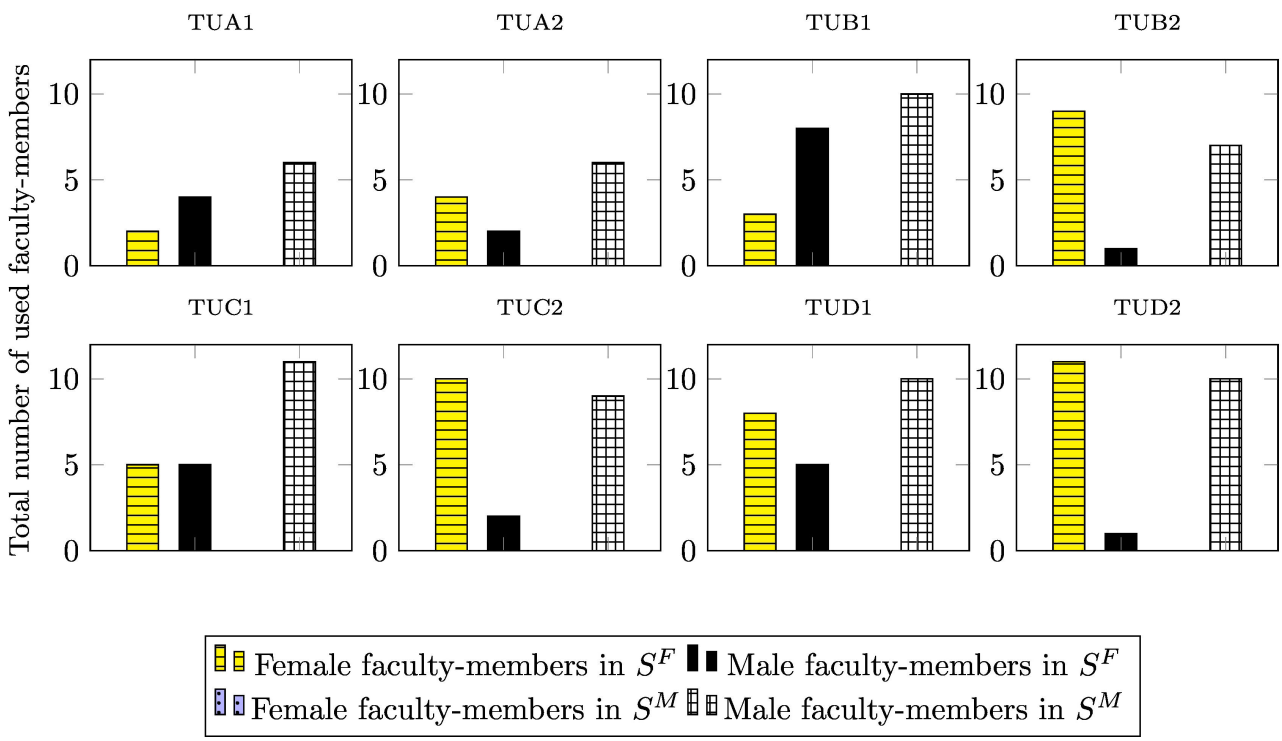


FIGURE 1. Number of utilized faculty-members for each student group level.

levels. Likewise, the number of male faculty-members used is presented as solid black bars for all the female levels and white bars with a grid for all the male levels. Instances TUA1, TUA2, TUB1, TUB2, TUC1, TUC2, and TUD1, fully utilized the total number of female faculty-members for all the female student levels. Hence, the number of female faculty-members was sufficient to obtain constraints satisfaction under those instances. Alternatively, there were no female faculty-members assigned to the male student group at all levels. Instances TUA1, TUA2, and TUB2 fully utilized the total number of male faculty-members for all the male student levels. At the same time, instances TUB1, TUC1, TUC2, TUD1, and TUD2 utilized a portion of the male faculty-members list for the female and male student group levels. Hence, there was no need for a large number of male faculty-members. Nevertheless, it can be concluded that all faculty-members were assigned according to their availability.

Generated timetables are visualized as Tables 7, 8, 9, 10, 11, 12, 13 and 14. Each table shows a solution for a problem set. Similar to Table 2, each row shows one course information. Including level, course, timeslot, room, and faculty-member information. Even though constraint (16) was satisfied for all instances, room capacity information was not presented since the total number of registered students is only available after the registration period. From the tables it is clear that there was one course event allocated in one room per timeslot (3) with one faculty-member (2). All daily events of the course were scheduled in the same

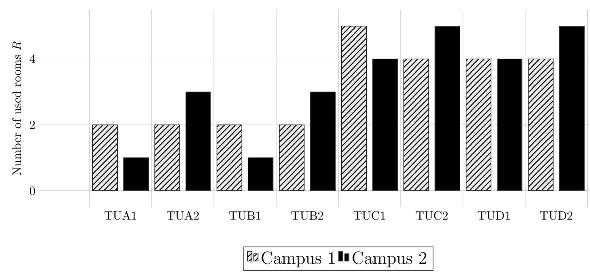


FIGURE 2. Number of utilized rooms, allocated in the female campus (campus 1) and the male campus (campus 2).

room (13). All events were scheduled in the required room type (21), with the required features (20). For example, it can be seen from the generated timetables that if a course pattern required a computer laboratory, a room of a type "Lab" was selected. Such as, rooms "f12102" and "m12101". Rooms "f4116" and "m4116" were of a type "Ph-Lab", i.e., physics laboratories. Rooms "f7101", "f7102", "f7104", "m7101", "m7102" and "m7103" were of a type "Lecture", i.e., lecture rooms. However, room "f7104" has the feature "LAN" and was used by male faculty-members at the female campus (19).

For further inspection, Figure 2 displays room utilization for all TU instances. Each bar in the X-axis illustrates the number of rooms used. Grey bars with striped lines present the number of rooms used in campus 1, and solid black bars present the number of rooms used in campus 2. Note that campus 1 is for the female campus, and campus 2 is for the

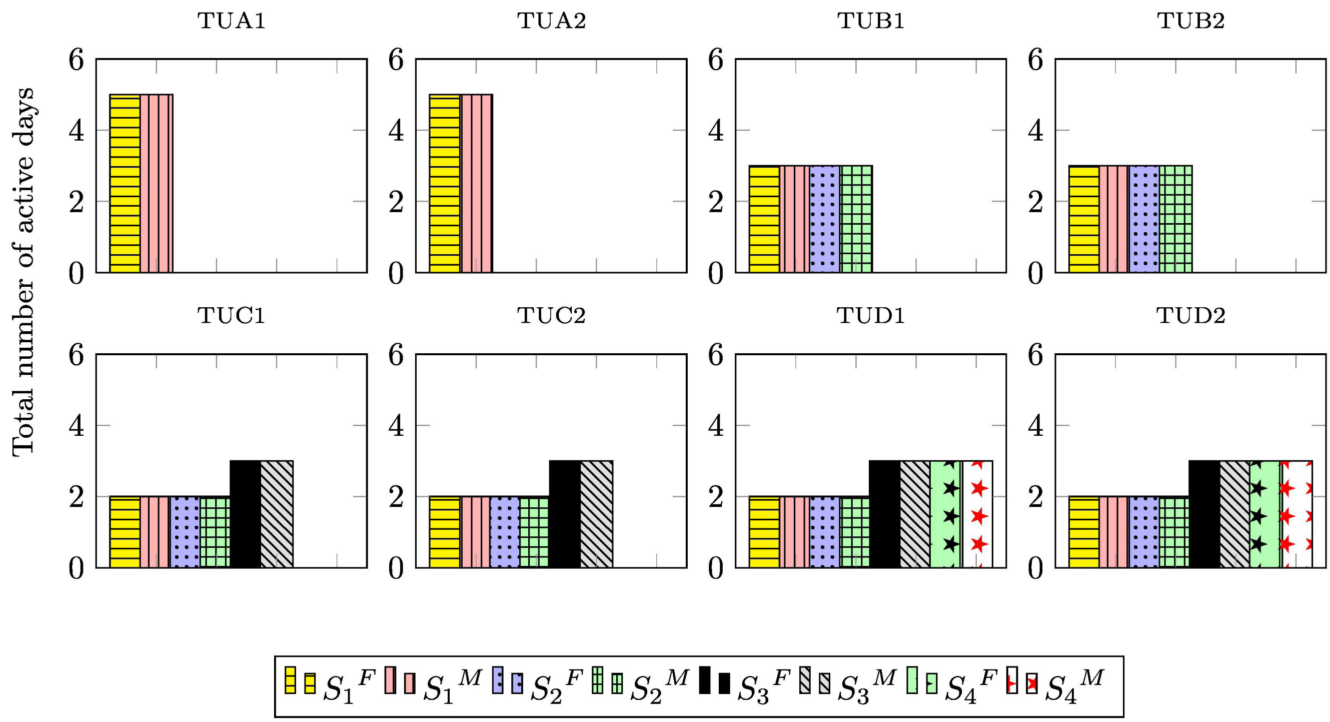


FIGURE 3. Number of active learning days for each student group level.

TABLE 7. Resulting timetable for all courses of TUA1 instance given to students at the computer science department at TU.

Course					*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Id*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Wed	11:00	13:00	f7104	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Sun	8:00	10:00	f7104	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Wed	8:00	10:00	f7104	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Mon	11:00	13:00	f7104	Lecture	M9	m
		Computer skills	2	Lecture	Sun	12:00	14:00	f7104	Lecture	F13	f
		University skills	2	Lecture	Tue	11:00	13:00	f7104	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Thu	12:00	14:00	f7102	Lecture	F17	f
	m	Islamic Culture (Ethics and Values)	2	Lecture	Tue	12:00	14:00	m7102	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Sun	10:00	11:00	m7102	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Tue	8:00	10:00	m7102	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Sun	11:00	13:00	m7102	Lecture	M9	m
		Computer skills	2	Lecture	Wed	12:00	14:00	m7102	Lecture	M15	m
		University skills	2	Lecture	Thu	11:00	13:00	m7102	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Mon	11:00	13:00	m7102	Lecture	M14	m

TABLE 8. Resulting timetable for all courses of TUA2 instance given to students at the computer science department at TU.

Course					*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Mon	11:00	13:00	f7102	Lecture	F2	f
		Fundamentals of mathematics	3	Lecture	Mon	10:00	11:00	f7104	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Tue	8:00	10:00	f7104	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Thu	12:00	14:00	f7102	Lecture	F15	f
		Computer skills	2	Lecture	Tue	12:00	14:00	f7102	Lecture	F13	f
		University skills	2	Lecture	Wed	12:00	14:00	f7104	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Sun	11:00	13:00	f7102	Lecture	F17	f
	m	Islamic Culture (Ethics and Values)	2	Lecture	Mon	12:00	14:00	m7103	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Sun	8:00	10:00	m7101	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Thu	9:00	10:00	m7101	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Tue	11:00	13:00	m7102	Lecture	M9	m
		Computer skills	2	Lecture	Sun	11:00	13:00	m7101	Lecture	M15	m
		University skills	2	Lecture	Thu	11:00	13:00	m7103	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Wed	11:00	13:00	m7103	Lecture	M14	m

TABLE 9. Resulting timetable for all courses of TUB1 instance given to students at the computer science department at TU.

		Course			*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Wed	12:00	14:00	f7104	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Sun	8:00	10:00	f7104	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Wed	8:00	9:00	f7104	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Tue	11:00	13:00	f7104	Lecture	M9	m
		Computer skills	2	Lecture	Sun	12:00	14;00	f7104	Lecture	M15	m
		University skills	2	Lecture	Wed	14:00	16:00	f7104	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Sun	14:00	16:00	f7104	Lecture	F17	f
2		Kingdom History	2	Lecture	Mon	14:00	16:00	f7104	Lecture	M1	m
		General biology	3	Lecture	Mon	10:00	11:00	f7104	Lecture	M4	m
		General biology	3	Lecture	Thu	9:00	11:00	f7104	Lecture	M4	m
		General chemistry	2	Lecture	Thu	13:00	15:00	f7102	Lecture	F6	f
		Scientific computing	2	Lecture	Wed	12:00	14:00	f7102	Lecture	F13	f
		Computer design	1	Lecture	Mon	13:00	14:00	f7104	Lecture	M9	m
		Arabic language skills	2	Lecture	Mon	11:00	13:00	f7104	Lecture	M16	m
		English for intensive academic purposes 2	2	Lecture	Thu	11:00	13:00	f7102	Lecture	F17	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Mon	13:00	15:00	m7102	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Mon	9:00	11:00	m7102	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Thu	10:00	11:00	m7102	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Thu	14:00	16:00	m7102	Lecture	M9	m
		Computer skills	2	Lecture	Sun	14:00	16:00	m7102	Lecture	M15	m
		University skills	2	Lecture	Thu	12:00	14:00	m7102	Lecture	M13	m
		English for intensive academic purposes 1	2	Lecture	Mon	11:00	13:00	m7102	Lecture	M14	m
2		Kingdom History	2	Lecture	Mon	13:00	15:00	m7102	Lecture	M1	m
		General biology	3	Lecture	Tue	8:00	10:00	m7102	Lecture	M4	m
		General biology	3	Lecture	Wed	9:00	10:00	m7102	Lecture	M4	m
		General chemistry	2	Lecture	Tue	11:00	13:00	m7102	Lecture	M5	m
		Scientific computing	2	Lecture	Wed	11:00	13:00	m7102	Lecture	M15	m
		Computer design	1	Lecture	Tue	15:00	16:00	m7102	Lecture	M9	m
		Arabic language skills	2	Lecture	Tue	13:00	15:00	m7102	Lecture	M16	m
		English for intensive academic purposes 2	2	Lecture	Sun	11:00	13:00	m7102	Lecture	M14	m

TABLE 10. Resulting timetable for all courses of TUB2 instance given to students at the computer science department at TU.

		Course			*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Thu	14:00	16:00	f7102	Lecture	F2	f
		Fundamentals of mathematics	3	Lecture	Mon	9:00	10:00	f7104	Lecture	F8	f
		Fundamentals of mathematics	3	Lecture	Thu	8:00	10:00	f7104	Lecture	F8	f
		Introduction to problem solving	2	Lecture	Mon	14:00	16:00	f7104	Lecture	F15	f
		Computer skills	2	Lecture	Mon	11:00	13:00	f7102	Lecture	F15	f
		University skills	2	Lecture	Thu	12:00	14:00	f7104	Lecture	F19	f
		English for intensive academic purposes 1	2	Lecture	Wed	11:00	13:00	f7104	Lecture	F17	f
2		Kingdom History	2	Lecture	Thu	14:00	16:00	f7104	Lecture	M1	m
		General biology	3	Lecture	Tue	9:00	10:00	f7104	Lecture	F7	f
		General biology	3	Lecture	Wed	8:00	10:00	f7104	Lecture	F7	f
		General chemistry	2	Lecture	Thu	11:00	13:00	f7102	Lecture	F6	f
		Scientific computing	2	Lecture	Wed	11:00	13:00	f7102	Lecture	F15	f
		Computer design	1	Lecture	Thu	13:00	14:00	f7102	Lecture	F13	f
		Arabic language skills	2	Lecture	Tue	12;00	14:00	f7104	Lecture	F20	f
		English for intensive academic purposes 2	2	Lecture	Wed	13:00	15;00	f7104	Lecture	F17	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Tue	12:00	14:00	m7101	Lecture	M3	m
		Fundamentals of mathematics	3	Lecture	Mon	8:00	9:00	m7101	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Tue	8:00	10:00	m7101	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Tue	14:00	16:00	m7103	Lecture	M11	m
		Computer skills	2	Lecture	Wed	12:00	14:00	m7103	Lecture	M11	m
		University skills	2	Lecture	Mon	11:00	13:00	m7101	Lecture	M1	m
		English for intensive academic purposes 1	2	Lecture	Mon	13:00	15:00	m7102	Lecture	M14	m
2		Kingdom History	2	Lecture	Mon	13:00	15:00	m7102	Lecture	M1	m
		General biology	3	Lecture	Mon	9:00	11:00	m7101	Lecture	M4	m
		General biology	3	Lecture	Wed	9:00	10:00	m7101	Lecture	M4	m
		General chemistry	2	Lecture	Wed	11:00	13:00	m7102	Lecture	M5	m
		Scientific computing	2	Lecture	Sun	11:00	13:00	m7103	Lecture	M11	m
		Computer design	1	Lecture	Wed	15:00	16:00	m7103	Lecture	M11	m
		Arabic language skills	2	Lecture	Mon	14:00	16:00	m7103	Lecture	M1	m
		English for intensive academic purposes 2	2	Lecture	Mon	11:00	13:00	m7103	Lecture	M14	m

TABLE 11. Resulting timetable for all courses of TUC1 instance given to students at the computer science department at TU.

Course					*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Sun	13:00	14:00	f7104	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Sun	8:00	10:00	f7104	Lecture	M7	m
		Fundamentals of mathematics	3	Lecture	Thu	8:00	9:00	f7104	Lecture	M7	m
		Introduction to problem solving	2	Lecture	Thu	11:00	13:00	f7104	Lecture	M15	m
		Computer skills	2	Lecture	Sun	11:00	13:00	f7101	Lecture	F18	f
		University skills	2	Lecture	Thu	16:00	18:00	f7104	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Sun	16:00	18:00	f7101	Lecture	F17	f
2		Kingdom History	2	Lecture	Wed	16:00	18:00	f7104	Lecture	M2	m
		General biology	3	Lecture	Wed	8:00	9:00	f7101	Lecture	F7	f
		General biology	3	Lecture	Thu	8:00	10:00	f7101	Lecture	F7	f
		General chemistry	2	Lecture	Thu	14:00	16:00	f7104	Lecture	F6	f
		Scientific computing	2	Lecture	Wed	13:00	15:00	f7104	Lecture	M15	m
		Computer design	1	Lecture	Wed	15:00	16:00	f7101	Lecture	F18	f
		Arabic language skills	2	Lecture	Wed	11:00	13:00	f7104	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Thu	11:00	13:00	f7101	Lecture	F17	f
3		Calculus 1	3	Lecture	Mon	8:00	10:00	f7104	Lecture	M7	m
		Calculus 1	3	Lecture	Wed	8:00	9:00	f7104	Lecture	M7	m
		Physics 1 T	3	Lecture	Tue	9:00	11:00	f7101	Lecture	F4	f
		Physics 1 T	3	Lecture	Wed	10:00	11:00	f7101	Lecture	F4	f
		Physics 1 L	1	PH-Lab	Wed	11:00	12:00	f4116	PH-Lab	F4	f
		Discrete Structures	3	Lecture	Mon	10:00	11:00	f7104	Lecture	M8	m
		Discrete Structures	3	Lecture	Tue	8:00	9:00	f7104	Lecture	M8	m
		Discrete Structures	3	Lecture	Wed	9:00	10:00	f7104	Lecture	M8	m
		Computer Programming 1 T	2	Lecture	Mon	16:00	18:00	f7104	Lecture	M8	m
		Computer Programming 1 L	1	Lab	Mon	14:00	15:00	f12102	Lab	F18	f
		Special English for computers 1	2	Lecture	Tue	15:00	17:00	f7101	Lecture	F17	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Thu	11:00	13:00	m7101	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Tue	8:00	10:00	m7101	Lecture	M6	m
		Fundamentals of mathematics	3	Lecture	Thu	9:00	10:00	m7101	Lecture	M6	m
		Introduction to problem solving	2	Lecture	Tue	12:00	14:00	m7102	Lecture	M15	m
		Computer skills	2	Lecture	Thu	16:00	18:00	m7102	Lecture	M11	m
		University skills	2	Lecture	Tue	15:00	17:00	m7101	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Thu	13:00	15:00	m7102	Lecture	M14	m
2		Kingdom History	2	Lecture	Tue	11:00	13:00	m7101	Lecture	M2	m
		General biology	3	Lecture	Tue	8:00	10:00	m7102	Lecture	M4	m
		General biology	3	Lecture	Wed	9:00	10:00	m7102	Lecture	M4	m
		General chemistry	2	Lecture	Wed	11:00	13:00	m7101	Lecture	M5	m
		Scientific computing	2	Lecture	Wed	16:00	18:00	m7102	Lecture	M15	m
		Computer design	1	Lecture	Wed	13:00	14:00	m7102	Lecture	M11	m
		Arabic language skills	2	Lecture	Wed	14:00	16:00	m7102	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Tue	16:00	18:00	m7102	Lecture	M14	m
3		Calculus 1	3	Lecture	Sun	10:00	11:00	m7102	Lecture	M7	m
		Calculus 1	3	Lecture	Mon	10:00	11:00	m7102	Lecture	M7	m
		Calculus 1	3	Lecture	Wed	8:00	9:00	m7102	Lecture	M7	m
		Physics 1 T	3	Lecture	Mon	9:00	10:00	m7101	Lecture	M17	m
		Physics 1 T	3	Lecture	Wed	9:00	11:00	m7101	Lecture	M17	m
		Physics 1 L	1	PH-Lab	Wed	12:00	13:00	m4116	PH-Lab	M17	m
		Discrete Structures	3	Lecture	Sun	8:00	10:00	m7102	Lecture	M8	m
		Discrete Structures	3	Lecture	Mon	8:00	9:00	m7102	Lecture	M8	m
		Computer Programming 1 T	2	Lecture	Mon	14:00	16:00	m7102	Lecture	M8	m
		Computer Programming 1 L	1	Lab	Mon	11:00	12:00	m12101	Lab	M11	m
		Special English for computers 1	2	Lecture	Wed	13:00	15:00	m7101	Lecture	M14	m

male campus. Results were compared with the number of available rooms, presented in Table 5. It can be seen from the results that all available rooms were fully utilized for all instances. Hence, facility resources were allocated efficiently.

It can be seen from the tables, that all courses were assigned to faculty-members, i.e., there were no unassigned courses. All faculty-related constraints were respected. The same faculty-member was assigned to all course events (11) for all time-slots (12), even if the events were divided into multiple periods or on different days. All courses were assigned according to faculty-members preferences (30), and university requirements (31), while respecting their availability (24). Female faculty-members were only assigned to female campus (25). All computer laboratories were assigned to a faculty-member of an academic rank teacher assistant (28). Such as F12, F13, F18, M11, and M14. Daily and weekly working hours constraints were respected (26,27).

Results also show that all events of each level were allocated to the required campus, with their required number of timeslots, and there were no course conflicts among levels (1). All events of one level were scheduled continuously among all timeslots for all active days (6). Course teaching credit hours were respected for each course (10). Two or fewer credit hour courses were scheduled one time in an active learning day (9), in the afternoon or the evening (4). Such as the "University Skills" course, in level one.

Three or more credit hours courses were divided among active days for each level (8) and were scheduled in the morning (5). Such as, "General Biology" course, in level two. Usually, a course with three credit hours was divided

TABLE 12. Resulting timetable for all courses of TUC2 instance given to students at the computer science department at TU.

		Course			*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Sun	16:00	18:00	f7101	Lecture	F2	f
		Fundamentals of mathematics	3	Lecture	Sun	8:00	9:00	f7104	Lecture	F8	f
		Fundamentals of mathematics	3	Lecture	Mon	9:00	11:00	f7104	Lecture	F8	f
		Introduction to problem solving	2	Lecture	Mon	15:00	17:00	f7101	Lecture	F11	f
		Computer skills	2	Lecture	Sun	14:00	16:00	f7104	Lecture	M15	m
		University skills	2	Lecture	Mon	11:00	13:00	f7104	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Mon	13:00	15:00	f7101	Lecture	F14	f
2		Kingdom History	2	Lecture	Tue	12:00	14:00	f7104	Lecture	M2	m
		General biology	3	Lecture	Tue	9:00	10:00	f7104	Lecture	F7	f
		General biology	3	Lecture	Wed	9:00	11:00	f7104	Lecture	F7	f
		General chemistry	2	Lecture	Tue	14:00	16:00	f7101	Lecture	F6	f
		Scientific computing	2	Lecture	Tue	16:00	18:00	f7101	Lecture	F11	f
		Computer design	1	Lecture	Wed	15:00	16;00	f7104	Lecture	F18	f
		Arabic language skills	2	Lecture	Wed	11:00	13:00	f7104	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Wed	13:00	15:00	f7101	Lecture	F17	f
3		Calculus 1	3	Lecture	Wed	10:00	11:00	f7101	Lecture	F5	f
		Calculus 1	3	Lecture	Thu	8:00	10:00	f7101	Lecture	F5	f
		Physics 1 T	3	Lecture	Sun	8:00	9:00	f7101	Lecture	F4	f
		Physics 1 T	3	Lecture	Wed	9:00	10:00	f7101	Lecture	F4	f
		Physics 1 T	3	Lecture	Thu	10:00	11:00	f7101	Lecture	F4	f
		Physics 1 L	1	PH-Lab	Sun	11:00	12:00	f4116	PH-Lab	F4	f
		Discrete Structures	3	Lecture	Sun	9:00	11:00	f7104	Lecture	F11	f
		Discrete Structures	3	Lecture	Wed	8:00	9:00	f7104	Lecture	F11	f
		Discrete Structures	3	Lecture	Thu	10:00	11:00	f7104	Lecture	F11	f
		Computer Programming 1 T	2	Lecture	Sun	14:00	16:00	f7101	Lecture	F11	f
		Computer Programming 1 L	1	Lab	Thu	17:00	18:00	f12102	Lab	F18	f
		Special English for computers 1	2	Lecture	Wed	15:00	17:00	f7101	Lecture	F17	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Wed	16:00	18:00	m7101	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Wed	8:00	10:00	m7101	Lecture	M6	m
		Fundamentals of mathematics	3	Lecture	Thu	9:00	10:00	m7101	Lecture	M6	m
		Introduction to problem solving	2	Lecture	Thu	13:00	15:00	m7102	Lecture	M15	m
		Computer skills	2	Lecture	Wed	12:00	14:00	m7101	Lecture	M15	m
		University skills	2	Lecture	Thu	11:00	13:00	m7101	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Wed	14:00	16:00	m7102	Lecture	M14	m
2		Kingdom History	2	Lecture	Mon	16:00	18:00	m7103	Lecture	M2	m
		General biology	3	Lecture	Mon	8:00	9:00	m7101	Lecture	M4	m
		General biology	3	Lecture	Tue	9:00	11:00	m7101	Lecture	M4	m
		General chemistry	2	Lecture	Tue	14:00	16:00	m7101	Lecture	M5	m
		Scientific computing	2	Lecture	Mon	11:00	13:00	m7103	Lecture	M11	m
		Computer design	1	Lecture	Tue	12:00	13:00	m7103	Lecture	M11	m
		Arabic language skills	2	Lecture	Tue	16:00	18:00	m7101	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Mon	14:00	16:00	m7103	Lecture	M14	m
3		Calculus 1	3	Lecture	Sun	9:00	10:00	m7102	Lecture	M6	m
		Calculus 1	3	Lecture	Tue	8:00	10:00	m7102	Lecture	M6	m
		Physics 1 T	3	Lecture	Sun	10:00	11:00	m7103	Lecture	M17	m
		Physics 1 T	3	Lecture	Tue	10:00	11:00	m7103	Lecture	M17	m
		Physics 1 T	3	Lecture	Thu	8:00	9:00	m7103	Lecture	M17	m
		Physics 1 L	1	PH-Lab	Sun	11:00	12:00	m4116	PH-Lab	M17	m
		Discrete Structures	3	Lecture	Sun	8:00	9:00	m7103	Lecture	M15	m
		Discrete Structures	3	Lecture	Thu	9:00	11:00	m7103	Lecture	M15	m
		Computer Programming 1 T	2	Lecture	Tue	11:00	12:00	m7101	Lecture	M15	m
		Computer Programming 1 L	1	Lab	Tue	17:00	18:00	m12101	Lab	M11	m
		Special English for computers 1	2	Lecture	Thu	12:00	14:00	m7103	Lecture	M14	m

over two days. However, in TUC1, TUC2, TUD1, and TUD2 instances, there were occurrences of three timeslots distributed over three different days. Especially for the following courses: calculus 1, calculus 2, physics 1, physics 2, and discrete structure. This was due to the number of timeslots available per day. Where earlier levels courses, with three credit hours, have been allocated first. Thus, morning timeslots were not available for further assignments while minimizing active learning days.

For further inspection, Figure 3 illustrates the number of active learning days for problem sets TUA1 to TUD2. Each sub-figure corresponds to a problem set from TUA1 to TUD2, and each bar illustrates the overall active learning days per level S. Bars color and pattern indicate the following: yellow bars with horizontal lines when ${S_1}^F$, red bars with vertical lines when ${S_1}^M$, blue bars with dots when ${S_2}^F$, green bars with the grid when ${S_2}^M$, solid black bars when ${S_3}^F$, grey bars with striped lines when ${S_3}^M$, green bars with black stars when ${S_4}^F$ and white bars with red stars when ${S_4}^M$. The lower the bar, the better, i.e., minimized the number of active learning days per level.

It can be seen that the number of levels per instance has affected the events' distribution over days. Instances with only one level, as in TUA1 and TUA2, had events distributed over five days. Hence, the number of active learning days were not minimized, and soft constraint (7) was not satisfied under these instances. On the other hand, more levels per instance indicated there were more events to be

TABLE 13. Resulting timetable for all courses of TUD1 instance given to students at the computer science department at TU.

		Special English for computers 2	2	Lecture	Tue	11:00	13:00	m7102	Lecture	M14	m
		Course				***Time***		***Room***		***Faculty-member***	
Level	***Campus***	***Name***	***Hours***	***Pattern***	***Day***	***From***	***To***	***Id***	***Type***	***Id***	***Gender***
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Thu	18:00	20:00	f7104	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Mon	10:00	11:00	f7104	Lecture	F5	f
		Fundamentals of mathematics	3	Lecture	Thu	9:00	11:00	f7104	Lecture	F5	f
		Introduction to problem solving	2	Lecture	Mon	15:00	17:00	f7101	Lecture	F10	f
		Computer skills	2	Lecture	Mon	13:00	15:00	f7101	Lecture	F10	f
		University skills	2	Lecture	Thu	13:00	15:00	f7104	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Thu	11:00	13:00	f7101	Lecture	F3	f
2		Kingdom History	2	Lecture	Thu	17:00	19:00	f7101	Lecture	F14	f
		General biology	3	Lecture	Sun	9:00	11:00	f7104	Lecture	M4	m
		General biology	3	Lecture	Thu	8:00	9:00	f7104	Lecture	M4	m
		General chemistry	2	Lecture	Sun	15:00	17:00	f7101	Lecture	F6	f
		Scientific computing	2	Lecture	Sun	11:00	13:00	f7104	Lecture	M8	m
		Computer design	1	Lecture	Sun	13:00	14:00	f7104	Lecture	M8	m
		Arabic language skills	2	Lecture	Thu	11:00	13:00	f7104	Lecture	F14	f
		English for intensive academic purposes 2	2	Lecture	Thu	13:00	15:00	f7101	Lecture	F3	f
3		Calculus 1	3	Lecture	Sun	10:00	11:00	f7101	Lecture	F5	f
		Calculus 1	3	Lecture	Wed	9:00	11:00	f7101	Lecture	F5	f
		Physics 1 T	3	Lecture	Sun	9:00	10:00	f7101	Lecture	F4	f
		Physics 1 T	3	Lecture	Mon	9:00	11:00	f7101	Lecture	F4	f
		Physics 1 L	1	PH-Lab	Wed	14:00	15:00	f4116	PH-Lab	F4	f
		Discrete Structures	3	Lecture	Sun	8:00	9:00	f7104	Lecture	F10	f
		Discrete Structures	3	Lecture	Mon	8:00	9:00	f7104	Lecture	F10	f
		Discrete Structures	3	Lecture	Wed	8:00	9:00	f7104	Lecture	F10	f
		Computer Programming 1 T	2	Lecture	Wed	11:00	13:00	f7101	Lecture	F10	f
		Computer Programming 1 L	1	Lab	Wed	17:00	18:00	f12102	Lab	F12	f
		Special English for computers 1	2	Lecture	Sun	11:00	13:00	f7101	Lecture	F3	f
4		Islamic culture (ethics and values)	2	Lecture	Tue	13:00	15:00	f7101	Lecture	F16	f
		Calculus 2	3	Lecture	Tue	9:00	11:00	f7101	Lecture	F5	f
		Calculus 2	3	Lecture	Thu	8:00	9:00	f7101	Lecture	F5	f
		Physics 2 T	3	Lecture	Wed	8:00	9:00	f7101	Lecture	F4	f
		Physics 2 T	3	Lecture	Thu	9:00	11:00	f7101	Lecture	F4	f
		Physics 2 L	1	PH-Lab	Thu	15:00	16:00	f4116	PH-Lab	F4	f
		Computer Programming 2 T	2	Lecture	Tue	11:00	13:00	f7104	Lecture	M8	m
		Computer Programming 2 L	1	Lab	Tue	15:00	16:00	f12102	Lab	F12	f
		Digital Logical Design T	3	Lecture	Tue	8:00	9:00	f7104	Lecture	M9	m
		Digital Logical Design T	3	Lecture	Wed	9:00	11:00	f7104	Lecture	M9	m
		Digital Logical Design L	1	Lab	Thu	12:00	13:00	f12102	Lab	F12	f
		Special English for computers 2	2	Lecture	Wed	13:00	15:00	f7104	Lecture	F3	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Sun	18:00	20:00	m7102	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Sun	8:00	10:00	m7102	Lecture	M6	m
		Fundamentals of mathematics	3	Lecture	Wed	8:00	9:00	m7102	Lecture	M6	m
		Introduction to problem solving	2	Lecture	Wed	14:00	16:00	m7101	Lecture	M9	m
		Computer skills	2	Lecture	Wed	11:00	13:00	m7102	Lecture	M9	m
		University skills	2	Lecture	Sun	11:00	13:00	m7101	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Sun	13:00	15:00	m7101	Lecture	M14	m
2		Kingdom History	2	Lecture	Wed	14:00	16:00	m7102	Lecture	M2	m
		General biology	3	Lecture	Mon	9:00	11:00	m7101	Lecture	M4	m
		General biology	3	Lecture	Wed	8:00	9:00	m7101	Lecture	M4	m
		General chemistry	2	Lecture	Mon	16:00	18:00	m7101	Lecture	M5	m
		Scientific computing	2	Lecture	Mon	11:00	13:00	m7102	Lecture	M8	m
		Computer design	1	Lecture	Wed	13:00	14:00	m7101	Lecture	M8	m
		Arabic language skills	2	Lecture	Mon	18:00	20:00	m7101	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Wed	11:00	13:00	m7101	Lecture	M14	m
3		Calculus 1	3	Lecture	Mon	8:00	9:00	m7101	Lecture	M6	m
		Calculus 1	3	Lecture	Tue	8:00	9:00	m7101	Lecture	M6	m
		Calculus 1	3	Lecture	Thu	8:00	9:00	m7101	Lecture	M6	m
		Physics 1 T	3	Lecture	Mon	10:00	11:00	m7102	Lecture	M12	m
		Physics 1 T	3	Lecture	Tue	10:00	11:00	m7102	Lecture	M12	m
		Physics 1 T	3	Lecture	Thu	10:00	11:00	m7102	Lecture	M12	m
		Physics 1 L	1	PH-Lab	Mon	11:00	12:00	m4116	PH-Lab	M12	m
		Discrete Structures	3	Lecture	Mon	9:00	10:00	m7102	Lecture	M9	m
		Discrete Structures	3	Lecture	Tue	9:00	10:00	m7102	Lecture	M9	m
		Discrete Structures	3	Lecture	Thu	9:00	10:00	m7102	Lecture	M9	m
		Computer Programming 1 T	2	Lecture	Tue	18:00	20:00	m7102	Lecture	M8	m
		Computer Programming 1 L	1	Lab	Mon	18:00	19:00	m12101	Lab	M11	m
		Special English for computers 1	2	Lecture	Tue	13:00	15:00	m7101	Lecture	M14	m
4		Islamic culture (ethics and values)	2	Lecture	Tue	13:00	15:00	m7101	Lecture	M1	m
		Calculus 2	3	Lecture	Tue	9:00	10:00	m7101	Lecture	M6	m
		Calculus 2	3	Lecture	Wed	10:00	11:00	m7101	Lecture	M6	m
		Calculus 2	3	Lecture	Thu	9:00	10:00	m7101	Lecture	M6	m
		Physics 2 T	3	Lecture	Tue	8:00	9:00	m7102	Lecture	M12	m
		Physics 2 T	3	Lecture	Wed	9:00	10:00	m7102	Lecture	M12	m
		Physics 2 T	3	Lecture	Thu	8:00	9:00	m7102	Lecture	M12	m
		Physics 2 L	1	PH-Lab	Tue	19:00	20:00	m4116	PH-Lab	M12	m
		Computer Programming 2 T	2	Lecture	Tue	14:00	16:00	m7102	Lecture	M8	m
		Computer Programming 2 L	1	Lab	Thu	14:00	15:00	m12101	Lab	M11	m
		Digital Logical Design T	3	Lecture	Tue	10:00	11:00	m7101	Lecture	M9	m
		Digital Logical Design T	3	Lecture	Wed	8:00	9:00	m7101	Lecture	M9	m
		Digital Logical Design T	3	Lecture	Thu	10:00	11:00	m7101	Lecture	M9	m
		Digital Logical Design L	1	Lab	Wed	14:00	15:00	m12101	Lab	M11	m
		Special English for computers 2	2	Lecture	Tue	11:00	13:00	m7102	Lecture	M14	m

TABLE 14. Resulting timetable for all courses of TUD2 instance given to students at the computer science department at TU.

		Course			*Time*			*Room*		*Faculty-member*	
Level	*Campus*	*Name*	*Hours*	*Pattern*	*Day*	*From*	*To*	*Id*	*Type*	*Id*	*Gender*
1	f	Islamic Culture (Ethics and Values)	2	Lecture	Mon	13:00	15:00	f7101	Lecture	F2	f
		Fundamentals of mathematics	3	Lecture	Mon	9:00	11:00	f7104	Lecture	F8	f
		Fundamentals of mathematics	3	Lecture	Wed	9:00	10:00	f7104	Lecture	F8	f
		Introduction to problem solving	2	Lecture	Mon	15:00	17:00	f7104	Lecture	F9	f
		Computer skills	2	Lecture	Mon	17:00	19:00	f7101	Lecture	F12	f
		University skills	2	Lecture	Wed	17:00	19:00	f7104	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Wed	11:00	13:00	f7101	Lecture	F3	f
2		Kingdom History	2	Lecture	Mon	11:00	13:00	f7104	Lecture	M2	m
		General biology	3	Lecture	Mon	8:00	9:00	f7104	Lecture	F7	f
		General biology	3	Lecture	Tue	8:00	10:00	f7104	Lecture	F7	f
		General chemistry	2	Lecture	Tue	11:00	13:00	f7101	Lecture	F6	f
		Scientific computing	2	Lecture	Tue	14:00	16:00	f7104	Lecture	F12	f
		Computer design	1	Lecture	Mon	14:00	15:00	f7104	Lecture	F11	f
		Arabic language skills	2	Lecture	Tue	17:00	19:00	f7104	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Mon	15:00	17:00	f7101	Lecture	F3	f
3		Calculus 1	3	Lecture	Sun	8:00	9:00	f7104	Lecture	F5	f
		Calculus 1	3	Lecture	Tue	10:00	11:00	f7104	Lecture	F5	f
		Calculus 1	3	Lecture	Thu	10:00	11:00	f7104	Lecture	F5	f
		Physics 1 T	3	Lecture	Tue	8:00	9:00	f7101	Lecture	F4	f
		Physics 1 T	3	Lecture	Thu	8:00	10:00	f7101	Lecture	F4	f
		Physics 1 L	1	PH-Lab	Tue	11:00	12:00	f4116	PH-Lab	F4	f
		Discrete Structures	3	Lecture	Sun	9:00	11:00	f7101	Lecture	F10	f
		Discrete Structures	3	Lecture	Tue	9:00	10:00	f7101	Lecture	F10	f
		Computer Programming 1 T	2	Lecture	Thu	15:00	17:00	f7101	Lecture	F10	f
		Computer Programming 1 L	1	Lab	Thu	18:00	19:00	f12102	Lab	F12	f
		Special English for computers 1	2	Lecture	Thu	11:00	13:00	f7104	Lecture	F3	f
4		Islamic culture (ethics and values)	2	Lecture	Wed	11:00	13:00	f7104	Lecture	F2	f
		Calculus 2	3	Lecture	Mon	8:00	10:00	f7101	Lecture	F5	f
		Calculus 2	3	Lecture	Wed	9:00	10:00	f7101	Lecture	F5	f
		Physics 2 T	3	Lecture	Sun	8:00	9:00	f7101	Lecture	F4	f
		Physics 2 T	3	Lecture	Mon	10:00	11:00	f7101	Lecture	F4	f
		Physics 2 T	3	Lecture	Wed	8:00	9:00	f7101	Lecture	F4	f
		Physics 2 L	1	PH-Lab	Sun	14:00	15:00	f4116	PH-Lab	F4	f
		Computer Programming 2 T	2	Lecture	Wed	16:00	18:00	f7101	Lecture	F10	f
		Computer Programming 2 L	1	Lab	Sun	11:00	12:00	f12102	Lab	F12	f
		Digital Logical Design T	3	Lecture	Sun	9:00	11:00	f7104	Lecture	F11	f
		Digital Logical Design T	3	Lecture	Wed	10:00	11:00	f7104	Lecture	F11	f
		Digital Logical Design L	1	Lab	Sun	18:00	19:00	f12102	Lab	F12	f
		Special English for computers 2	2	Lecture	Mon	11:00	13:00	f7101	Lecture	F3	f
1	m	Islamic Culture (Ethics and Values)	2	Lecture	Sun	13:00	15:00	m7103	Lecture	M1	m
		Fundamentals of mathematics	3	Lecture	Sun	9:00	11:00	m7102	Lecture	M6	m
		Fundamentals of mathematics	3	Lecture	Tue	9:00	10:00	m7102	Lecture	M6	m
		Introduction to problem solving	2	Lecture	Sun	11:00	13:00	m7101	Lecture	M9	m
		Computer skills	2	Lecture	Sun	17:00	19:00	m7103	Lecture	M9	m
		University skills	2	Lecture	Tue	15:00	17:00	m7102	Lecture	M2	m
		English for intensive academic purposes 1	2	Lecture	Tue	13:00	15:00	m7101	Lecture	M14	m
2		Kingdom History	2	Lecture	Sun	16:00	18:00	m7101	Lecture	M2	m
		General biology	3	Lecture	Sun	8:00	9:00	m7101	Lecture	M4	m
		General biology	3	Lecture	Tue	9:00	11:00	m7101	Lecture	M4	m
		General chemistry	2	Lecture	Sun	14:00	16:00	m7101	Lecture	M5	m
		Scientific computing	2	Lecture	Tue	13:00	15:00	m7102	Lecture	M8	m
		Computer design	1	Lecture	Sun	18:00	19:00	m7101	Lecture	M8	m
		Arabic language skills	2	Lecture	Tue	11:00	13:00	m7101	Lecture	M2	m
		English for intensive academic purposes 2	2	Lecture	Tue	16:00	18:00	m7101	Lecture	M14	m
3		Calculus 1	3	Lecture	Sun	8:00	9:00	m7102	Lecture	M6	m
		Calculus 1	3	Lecture	Mon	10:00	11:00	m7102	Lecture	M6	m
		Calculus 1	3	Lecture	Wed	10:00	11:00	m7102	Lecture	M6	m
		Physics 1 T	3	Lecture	Sun	10:00	11:00	m7103	Lecture	M12	m
		Physics 1 T	3	Lecture	Wed	8:00	10:00	m7103	Lecture	M12	m
		Physics 1 L	1	PH-Lab	Wed	19:00	20:00	m4116	PH-Lab	M12	m
		Discrete Structures	3	Lecture	Sun	9:00	10:00	m7101	Lecture	M9	m
		Discrete Structures	3	Lecture	Mon	8:00	10:00	m7101	Lecture	M9	m
		Computer Programming 1 T	2	Lecture	Sun	11:00	13:00	m7103	Lecture	M8	m
		Computer Programming 1 L	1	Lab	Wed	17:00	18:00	m12101	Lab	M11	m
		Special English for computers 1	2	Lecture	Sun	18:00	20:00	m7102	Lecture	M14	m
4		Islamic culture (ethics and values)	2	Lecture	Wed	13:00	15:00	m7101	Lecture	M1	m
		Calculus 2	3	Lecture	Mon	8:00	10:00	m7102	Lecture	M6	m
		Calculus 2	3	Lecture	Wed	8:00	9:00	m7102	Lecture	M6	m
		Physics 2 T	3	Lecture	Wed	10:00	11:00	m7101	Lecture	M12	m
		Physics 2 T	3	Lecture	Thu	8:00	10:00	m7101	Lecture	M12	m
		Physics 2 L	1	PH-Lab	Mon	13:00	14:00	m4116	PH-Lab	M12	m
		Computer Programming 2 T	2	Lecture	Thu	15:00	17:00	m7103	Lecture	M8	m
		Computer Programming 2 L	1	Lab	Mon	11:00	12:00	m12101	Lab	M11	m
		Digital Logical Design T	3	Lecture	Mon	10:00	11:00	m7101	Lecture	M9	m
		Digital Logical Design T	3	Lecture	Wed	9:00	10:00	m7101	Lecture	M9	m
		Digital Logical Design T	3	Lecture	Thu	10:00	11:00	m7101	Lecture	M9	m
		Digital Logical Design L	1	Lab	Mon	17:00	18:00	m12101	Lab	M11	m
		Special English for computers 2	2	Lecture	Wed	11:00	13:00	m7101	Lecture	M14	m

scheduled. Thus, a larger number of events were distributed over fewer days, as in the case of TUB1, TUB2, TUC1, TUC2, TUD1, and TUD2. This was related to the number of timeslots available for each instance. The more available timeslots per day, the less active learning days were required. Hence, the objective function was successful in minimizing the number of active learning days while satisfying soft constraints (7) under these instances.

In conclusion, results show that all hard constraints were fulfilled within a reasonable computational time for all problem instances. All soft constraints have also been met, as far as it was possible. For example, students in one level had some occurrences of two consecutive days (7). However, students still had at least one day off, as in TUB1, TUB2, TUC1, TUC2, TUD1, and TUD2. Also, maximum credit hours per week for each level were respected for all problem instances (14).

V. CONCLUSION

This research proposed a mixed-integer programming model for the course timetabling problem (UCTP), combined with the faculty-course assignment problem. Constraints such as faculty-member preferences, requirements, and availability were considered, in addition to timeslot and room assignment constraints. The aim was to maximize event and faculty assignments according to their preferences and requirements while minimizing student active learning days and non-assigned events. The proposed model has been implemented and then tested on eight different TU real-world instances of various sizes.Computational results showed that the method obtained optimal solutions on TU instances with an acceptable amount of time. This is an improvement from the time-consuming process of generating a timetable manually. However, solving more problem instances that were larger while using a commercial IP solver was challenging. According to [15], heuristics must be used to solve larger-size UCTP instances that have faculty-related constraints. Hence, future research directions will focus on exploring heuristic solution methods to solve larger TU instances. Future work is also required to investigate whether the proposed solution method would perform well on more realistic versions of these problem instances. For example, considering different groups per level or minimizing traveling times between campuses. More scenarios from various departments and colleges, with different sizes, should also be considered to show the model applicability.

REFERENCES

[1] S. Al-Negheimish, F. Alnuhait, H. Albrahim, S. Al-Mogherah, M. Alrajhi, and M. Hosny, "An intelligent bio-inspired algorithm for the faculty scheduling problem," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 5, pp. 151–159, 2018.

[2] S. M. Al-Yakoob and H. D. Sherali, "A column generation mathematical programming approach for a class-faculty assignment problem with preferences," *Comput. Manage. Sci.*, vol. 12, no. 2, pp. 297–318, Apr. 2015.

[3] C. H. Aladag, G. Hocaoglu, and M. A. Basaran, "The effect of neighborhood structures on Tabu search algorithm in solving course timetabling problem," *Expert Syst. Appl.*, vol. 36, no. 10, pp. 12349–12356, Dec. 2009.

[4] N. L. A. Aziz and N. A. H. Aizam, "A brief review on the features of university course timetabling problem," in *Proc. AIP Conf.*, vol. 2016, 2018, Art. no. 020001.

[5] H. Babaei, J. Karimpour, and A. Hadidi, "A survey of approaches for university course timetabling problem," *Comput. Ind. Eng.*, vol. 86, pp. 43–59, Aug. 2015.

[6] S. B. Bhoi and J. M. Dhodiya, "Multi-objective faculty course timeslot assignment problem with result-and feedback-based preferences," in *Ambient Communications and Computer Systems*. Singapore: Springer, 2020, pp. 105–119.

[7] A. Bonutti, F. De Cesco, L. Di Gaspero, and A. Schaerf, "Benchmarking curriculum-based course timetabling: Formulations, data formats, instances, validation, visualization, and results," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 59–70, Apr. 2012.

[8] E. K. Burke, J. Mareček, A. J. Parkes, and H. Rudová, "A branch-and-cut procedure for the udine course timetabling problem," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 71–87, Apr. 2012.

[9] V. E. Budi Darmawan, Y. W. Chen, A. Larasati, D. Prastyo, and A. Dwiastuti, "Multi-objective modeling for a course timetabling problem," in *Proc. Int. Conf. Creative Econ., Tourism Inf. Manage.*, 2019, pp. 10–14.

[10] P. R. D. L. Andrade, M. T. A. Steiner, and A. R. T. Góes, "Optimization in timetabling in schools using a mathematical model, local search and iterated local search procedures," *Gestão & Produção*, vol. 26, Dec. 2019. [Online]. Available: https://www.gestaoeproducao.com/journal/gp/ed/5dde67290e882546277b23c6 and https://search.crossref.org/?q=Optimization+in+timetabling+in+schools+using+a+mathematical+model%2C+local+search+and+Iterated+Local+Search+procedures&from_ui=yes#

[11] B. Domenech and A. Lusa, "A MILP model for the teacher assignment problem considering teachers' preferences," *Eur. J. Oper. Res.*, vol. 249, no. 3, pp. 1153–1160, 2016.

[12] M. Doulaty, M. F. Derakhshi, and M. Abdi, "Timetabling: A state-of-the-art evolutionary approach," *Int. J. Mach. Learn. Comput.*, vol. 3, no. 3, p. 255, 2013.

[13] T. El-Sakka, "University course timetable using constraint satisfaction and optimization," *Int. J. Comput.*, vol. 4, no. 3, pp. 83–95, 2015.

[14] S. Ghaemi, M. Taghi Vakili, and A. Aghagolzadeh, "Using a genetic algorithm optimizer tool to solve university timetable scheduling problem," in *Proc. 9th Int. Symp. Signal Process. Appl.*, Feb. 2007, pp. 1–4.

[15] A. Gunawan, K. M. Ng, and K. L. Poh, "Solving the teacher assignment-course scheduling problem by a hybrid algorithm," *Int. J. Comput., Inf., Syst. Sci., Eng.*, vol. 1, no. 2, p. 136, 2007.

[16] J. Havås, A. Olsson, J. Persson, and M. S. Schierscher, "Modeling and optimization of university timetabling-a case study in integer programming," Ph.D. dissertation, Kandidatarbete, Dept. Math. Sci., Univ. Gothenburg, Gothenburg, Sweden, 2013.

[17] A. Hmer and M. Mouhoub, "Teaching assignment problem solver," in *Proc. Int. Conf. Ind., Eng. Other Appl. Appl. Intell. Syst.* Berlin, Germany: Springer, 2010, pp. 298–307.

[18] M. I. Hosny, "A heuristic algorithm for solving the faculty assignment problem," *J. Commun. Comput.*, vol. 10, pp. 287–294, May 2013.

[19] IBM. *Ilog Cplex Optimization Studio–IBM*. Accessed: Nov. 30, 2020. [Online]. Available: https://www.ibm.com/products/ilog-cplex-optimization-studio

[20] N. A. Ismayilova, M. Sağir, and R. N. Gasimov, "A multiobjective faculty–course–time slot assignment problem with preferences," *Math. Comput. Model.*, vol. 46, nos. 7–8, pp. 1017–1029, 2007.

[21] A. Jamili, M. Hamid, H. Gharoun, and R. Khoshnoudi, "Developing a comprehensive and multi-objective mathematical model for university course timetabling problem: A real case study," in *Proc. Int. Conf. Ind. Eng. Oper. Manage.*, Paris, France, 2018, pp. 1–12.

[22] G. Lach and M. E. Lübbecke, "Curriculum based course timetabling: New solutions to udine benchmark instances," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 255–272, Apr. 2012.

[23] M. Lindahl, A. J. Mason, T. Stidsen, and M. Sørensen, "A strategic view of university timetabling," *Eur. J. Oper. Res.*, vol. 266, no. 1, pp. 35–45, Apr. 2018.

[24] M. Lindahl, M. Sørensen, and T. R. Stidsen, "A fix-and-optimize matheuristic for university timetabling," *J. Heuristics*, vol. 24, no. 4, pp. 645–665, Aug. 2018.

[25] B. B. Malik and S. Z. Nordin, "Mathematical model for timetabling problem in maximizing the preference level," in *Proc. AIP Conf.*, vol. 1974, Jun. 2018, Art. no. 020037. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.5041568, doi: 10.1063/1.5041568.

[26] S. A. MirHassani and F. Habibi, "Solution approaches to the course timetabling problem," *Artif. Intell. Rev.*, vol. 39, no. 2, pp. 133–149, Feb. 2013.

[27] T. Müller, "Constraint-based timetabling," Ph.D. dissertation, Dept. Math. Phys., Charles Univ., Prague, Czech Republic, 2005.

[28] E. Ongy, "Optimizing student learning: A faculty-course assignment problem using linear programming," *J. Educ. Hum. Resource Develop.*, vol. 5, pp. 1–14, Dec. 2017.

[29] Z. K. Öztürk and M. Sağir, "A new mathematical model and random key based metaheuristic solution approach for course-room-time assignment problem," *Eskişehir Osmangazi Üniversitesi Mühendislik ve Mimarlik Fakültesi Dergisi*, vol. 27, no. 2, pp. 67–76, Aug. 2019.

[30] A. E. Phillips, C. G. Walker, M. Ehrgott, and D. M. Ryan, "Integer programming for minimal perturbation problems in university course timetabling," *Ann. Oper. Res.*, vol. 252, no. 2, pp. 283–304, May 2017.

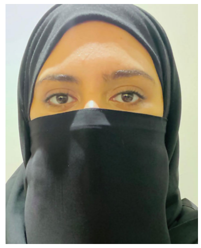
[31] A. Schaerf, "A survey of automated timetabling," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 87–127, 1999.

[32] S. T. Ngo, J. Jaafar, I. A. Aziz, and B. N. Anh, "A compromise programming for multi-objective task assignment problem," *Computers*, vol. 10, no. 2, p. 15, Jan. 2021.

[33] S. Timilsina, R. Negi, Y. Khurana, and J. Seth, "Genetically evolved solution to timetable scheduling problem," *Int. J. Comput. Appl.*, vol. 114, no. 18, pp. 12–17, Mar. 2015.

[34] A. Wasfy and F. Aloul, "Solving the University class scheduling problem using advanced ILP techniques," in *Proc. IEEE GCC Conf.*, Citeseer, Jan. 2007. [Online]. Available: https://www.researchgate.net/publication/228634502_Solving_the_University_Class_Scheduling_Problem_Using_Advanced_ILP_Techniques

[35] C. W. Yang and P. Y. Kim, "A simulation of the faculty-assignment problem: An integer programming approach," in *Proc. Acad. Marketing Sci. (AMS) Annu. Conf.* Springer, 2015, pp. 232–235.



H. ALGETHAMI (Member, IEEE) received the B.Sc. degree from Taif University, in 2006, and the M.Sc. degree in advanced computing science and the Ph.D. degree in computer science from the University of Nottingham, U.K., in 2012 and 2017, respectively. Since 2018, she has been an Assistant Professor with the Department of Computer Science, College of Computers and Information Technology, Taif University, Taif, KSA. Her research interests include real-world applications of combinatorial problems while using search algorithms and optimization techniques.

W. LAESANKLANG (Member, IEEE) received the B.S. degree in mathematics from Chiang Mai University, the M.Sc. degree in computational science from Chulalongkorn University, and the Ph.D. degree in computer science from the University of Nottingham. He is currently a Lecturer in industrial mathematics with the Department of Mathematics, Faculty of Science, Mahidol University. His research interests include vehicle routing problems, mathematical programming modeling for real-life problems, and heuristic algorithms.