INFORMATICA, 2017, Vol. 28, No. 4, 583–608

DOI: http://dx.doi.org/10.15388/Informatica.2017.147

A Column Generation Mathematical Model for a Teaching Assistant Workload Assignment Problem

Salem M. AL-YAKOOB[1]*, Hanif D. SHERALI[2]

[1]*Department of Mathematics, College of Science, Kuwait University P.O. Box: 5969, Safat 13060, State of Kuwait*

[2]*Grado Department of Industrial and Systems Engineering (0118), Virginia Tech, Blacksburg VA 24061, USA*

e-mail: salem@al-yakoob.com, hanifs@vt.edu

Received: January 2017; accepted: September 2017

Abstract. This paper presents a column generation-based modelling and solution approach for a teaching assistant workload scheduling problem that arises at academic institutions. A typical weekly workload schedule involves teaching deficiency classes, instructing problem-solving tutorial sessions, and allocating help-hours for students. For this purpose, a mixed-integer programming model that selects valid combinations of weekly schedules from the set of all feasible schedules is formulated. Due to the overwhelming number of variables in this model, an effective column generation procedure is developed. To illustrate the proof-of-concept along with modelling and algorithmic constructs, a case study related to the Department of Mathematics at Kuwait University is addressed. Computational results based on real data indicate that the generated schedules using the proposed model and solution procedure yield improved weekly workloads for teaching assistants in terms of fairness, and achieve enhanced satisfaction levels among assistants, as compared to schedules obtained using ad-hoc manual approaches.

Key words: academic timetabling, scheduling, mathematical programming, column generation.

1. Introduction

1.1. *Overview and Motivation*

In its broadest context, an academic scheduling and timetabling problem deals with classes, sections of classes, tutorial and lab sessions, faculty members, teaching assistants, midterm and final exams, available time-slots, and available facility resources, in addition to certain enhancing features such as preferences of faculty members and teaching assistants, while providing conflict-free class schedules. The intricacy and combinatorial nature of such problems for relatively large universities highlight the need for developing efficient quantitative approaches for generating acceptable, flexible, and robust class schedules.

*Corresponding author.

In this paper, we address a novel problem related to generating weekly workload schedules for teaching assistants at academic institutions. In order to illustrate the proof-of-concept along with modelling and algorithmic constructs, we focus on analysing a case study pertaining to the Department of Mathematics at Kuwait University. However, the proposed modelling and solution approach can be readily adapted to likewise study problems having similar intrinsic structures as faced by many relatively large-sized academic institutions worldwide.

In previous research (Al-Yakoob and Sherali, 2006, 2007, 2015), we designed a two-stage approach to address such a problem at Kuwait University (KU). Stage I of this approach (see Al-Yakoob and Sherali, 2007) deals with the generation of an efficient class timetable that provides flexible class and time schedules, while considering available resources such as classrooms, laboratories, and parking facilities, as well as related traffic issues. Stage II (see Al-Yakoob and Sherali, 2006; Al-Yakoob *et al.*, 2010) subsequently assigns faculty members to sections of offered classes within individual departments, while permitting at most a 15% rescheduling of classes as mandated by the Office of the Registrar.

More specifically, the Mathematics Department at KU offers a number of sections for a deficiency Pre-Calculus class (Math91-1) for students majoring in science or engineering who have not passed the Mathematics Placement Aptitude Test. Similar deficiency Pre-Calculus classes oriented toward business students (Math91-2) and social science students (Math91-3) are also offered. In the sequel, we will use Math91 to jointly refer to Math91-1, Math91-2, and Math91-3. Moreover, certain freshman and sophomore level mathematics classes are offered along with problem-solving tutorial sessions (Al-Yakoob and Sherali, 2006). For ease in reference, we will refer to these problem-solving tutorial sessions simply as *tutorials*. A solution to the Stage II problem alluded to above specifies subsets of days on which certain tutorials are to be offered, without specifying time-slots and instructors, such that each tutorial is offered on a day that is disjoint from those on which the corresponding class is taught. Hence, another timetabling problem that emerges from this two-stage approach is the *Teaching Assistant Workload Assignment Problem*, denoted TAP, which is mainly concerned with assigning sections of Math91 and tutorials to available teaching assistants, and also with specifying required help-hours for each assistant. Note that the two-stage approach described in Al-Yakoob and Sherali (2006, 2007, 2015) handles the assignment of classes to faculty members, classrooms, and time-slots but it does not deal with Problem TAP, which is the principal focus of the present paper.

The classes that are offered along with tutorials in the Mathematics Department at KU are Calculus I (Math101), Calculus II (Math102), Calculus for Biology (Math103), Calculus for Social Studies (Math108), Linear Algebra (Math111), Discrete Math (Math115), Calculus III (Math211), Differential Equations (Math240), and Numerical Analysis (Math352). Furthermore, there are two types of help-hours offered for students: a) regular office-hours where a teaching assistant allocates a certain number of instructional hours for students, and b) other types of help-hours related to the Mathematics Laboratory (*MathLab*) to assist freshman and sophomore students in general. Note that MathLab is typically open for students on a daily basis from 9:00 a.m. to 5:00 p.m. with at least two

Table 1
Weekly workload activity distribution.

Class		Number of hours per week					Total hours per week	*Campus
Title	Index *c* for classes with tutorials	Teaching or problem solving	Material preparation	Grading	Help-hours			
					First section	Each repeated section		
Math91-1		3	2	0	3	5	8	Science
Math91-2		3	2	0	3	5	8	College of Girls
Math91-3		3	2	0	3	5	8	Business
Math101	1	1.5	1.5	2.5	2.5	4	8	Science
Math102	2	1.5	1.5	2.5	2.5	4	8	Science
Math103	3	1.5	1	0	1.5	2.5	4	Science
Math108	4	1.5	1.5	2.5	2.5	4	8	College of Girls
Math111	5	1.5	1.5	2.5	2.5	4	8	Science
Math115	6	1.5	1	0	1.5	2.5	4	Science
Math211	7	1.5	1.5	2.5	2.5	4	8	Science
Math240	8	1.5	1.5	2.5	2.5	4	8	Science
Math352	9	1.5	2.5	3.5	2.5	5	10	Science

*MathLab is located in the College of Science.

assistants on duty during any given hour. MathLab is designed to complement the regular office-hours by providing students with convenient individual and group study environments, with available on-the-spot assistance as necessary.

1.2. *Weekly Workload Schedule Requirements*

In constructing schedules for assistants, certain essential weekly workload requirements must be satisfied, as detailed below with specific information displayed in Table 1:

(a) Each section of Math91 is for three credit hours, and these courses involve three midterm exams and a final exam, where these exams are multiple-choice tests that are graded by a scanning machine. Hence, as displayed in Table 1, no grading hours are allocated for Math91.

(b) Each section of the following classes is offered along with a 75-minute tutorial: Math101, Math102, Math103, Math108, Math111, Math115, Math211, Math240, and Math352. Except for tutorials associated with Math103 and Math115, an assistant is required to administer and grade seven quizzes throughout the semester.

(c) The weekly load-range for assistants (in terms of hours) must lie within [40, 48], which consists of teaching sections of Math91, instructing tutorials, preparing material, grading quizzes, and providing help-hours. A detailed distribution of these activities is presented in Table 1.

(d) Each assistant is required to allocate at least eight office-hours during the school week.

(e) As indicated in Table 1, the mandated help-hours are specified according to the number required for covering a single section of the particular class as well as the help-hours required for each repeated section of this class. For example, an assistant teaching a section of Math91 is required to allocate three help-hours, but when teaching a second section of Math91, the two hour preparation time for this section is re-allocated toward providing help-hours. Hence, in this case, the total number of help-hours that need to be allocated for teaching a second section of Math91 is five.
(f) The annual workload of an assistant over Fall and Winter semesters must cover at least four tutorials, each pertaining to *different* classes.
(g) The daily work requirement for an assistant is at least three hours.
(h) Since it takes about 20 minutes by car to transit between any two of the three campuses as indicated in Table 1, assigning teaching activities over consecutive periods within two distinct campuses is not permitted.

The remainder of this paper is organized as follows. Section 2 presents a discussion of the literature related to Problem TAP. Section 3 introduces our notation along with certain preliminary modelling constructs. Section 4 formulates a mixed-integer program (denoted TAM), which selects valid combinations of weekly schedules for assistants from the set of all feasible weekly schedules. Sections 5 and 6 describe constraints that are used to characterize columns of Model TAM. A column generation algorithm is designed in Section 7 to solve the linear relaxation of Model TAM, based on which, a sequential variable-fixing heuristic is devised to solve Model TAM. Computational results and analyses related to solving Model TAM are presented in Section 8, and we close the paper in Section 9 with a summary, concluding remarks, and directions for future research.

2. Related Research

All academic institutions handle timetabling tasks, where there are many activities that need to be scheduled subject to the availability of existing resources and other constraining limitations. For example, within a university environment, decision makers grapple with challenging issues such as: assigning sections of different classes to various time-slots, faculty members, and classrooms; assigning tutorials and lab sessions to teaching assistants, and generating conflict-free exam schedules. Another well-known example arises within a high school environment, where concerned administrators need to generate yearly or half-yearly schedules that assign teachers to grade-levels, groups of students, specific classes, and to time-slots while taking into account available manpower and physical resources, and school-specific requirements. Due to the idiosyncrasies of each individual problem, there is no standard modelling approach or solution methodology that can be utilized to solve all such problems.

Timetabling problems that address several critical features are typically proven in the literature to be NP-hard (see, for example, de Werra *et al.*, 2002; Eikelder and Willemen, 2001; Even *et al.*, 1976). As evident from the existing academic timetabling literature, the last three decades have witnessed a great interest in this area. Sandhu (2001) provides a comprehensive review of this enormous body of literature until 2001, giving a chronological presentation of timetabling along with insights into the evolution of approaches from the first manual heuristic procedure to the state-of-art computer-based methods. Several other surveys related to academic timetabling problems appear in Burke and Petrovic (2002), Lewis (2007), McCollum (2007), McCollum *et al.* (2010), Petrovic and Burke (2004), Qu *et al.* (2009), Schaerf (1999). For a more recent overview of academic timetabling, we refer the reader to Burke *et al.* (2010). Furthermore, McCollum (2007) and McCollum *et al.* (2010) have presented an insightful discussion on bridging the gap between research and practice in this area.

The existing literature is rich with approaches that have been used to solve academic timetabling problems such as mathematical programming methods, local search algorithms, tabu search, constraint-based reasoning and logic programming, genetic algorithms, decision-support systems and goal programming, simulated annealing, neural networks, and metaheuristics. Our proposed research effort falls into the mathematical programming category, and many academic timetabling problems have been modelled and solved using this approach (see Al-Yakoob and Sherali, 2006, 2007; Al-Yakoob *et al.*, 2010; Avella and Vasil'Ev, 2005; Baker and Aksop, 2008; Birbas *et al.*, 1997; Boland *et al.*, 2008; Burke and Gendreau, 2008; Daskalaki *et al.*, 2004; Dimopoulou and Miliotis, 2001; Ismayilova *et al.*, 2007; MirHassani, 2006; Ozdemir and Gasimov, 2004; Papoutsis *et al.*, 2003; Santos *et al.*, 2008; Tripathy, 1984; Valouxis and Houso, 2003; Yuqiang, 2007). The present paper contributes toward the foregoing body of literature by addressing a novel teaching assistant timetabling problem of the type encountered by many academic institutions in the world. The problem incorporates various load activities over separate campus locations, and deals with many problem-specific constraining issues along with user-desirable features that serve to enhance the quality of solutions produced. Although many academic timetabling problems have been investigated in the literature, none of them have tackled the particular type of teaching assistant timetabling problem addressed herein and therefore the modelling and solution approaches presented in this paper afford a useful addition to the academic timetabling and scheduling literature. Moreover, from both modelling and algorithmic perspectives, several aspects of our methodology could benefit timetabling efforts in general.

3. Notation and Modelling Preliminaries

In this section, we present our notation along with basic modelling constructs that will be used to formulate and solve Model TAM. We also use this discussion to describe the problem at hand in more detail.

Let A denote the set of all available teaching assistants, indexed by $a = 1, \dots, |A|$. Let C be the set of classes that are offered with tutorials, indexed by $c = 1, \dots, 9$, as indicated in Table 1. The set C is partitioned into four subsets, based on common total weekly workloads and campus locations as follows: $C^1 = \{1, 2, 5, 7, 8\}$, $C^2 = \{3, 6\}$, $C^3 = \{4\}$, and $C^4 = \{9\}$. Let $l = 1, 2$, and 3 respectively index the College of Science, College of Girls, and College of Business, where we denote $L = \{1, 2, 3\}$. Accordingly, we alternatively partition C into three subsets based on the campus location as follows: $C_1 = \{1, 2, 3, 5, 6, 7, 8, 9\}$ and $C_2 = \{4\}$; recall also that the Math91-l sessions are respectively instructed in location $l \in \{1, 2, 3\}$.

A section of Math91 is for three hours and each section is scheduled either on Sunday (S), Tuesday (T), and Thursday (*Th*), or on Monday (M) and Wednesday (W). Let $STT \equiv \{S, T, Th\}$, $MW \equiv \{M, W\}$, and $D \equiv \{S, M, T, W, Th\}$. Also, note that the duration of the time-slots for sections of Math91 offered on days in *STT* and *MW* are respectively 50-minutes (followed by a 10-minute break) and 75-minutes (followed by a 15-minute break). A section of Math91 may be offered between 8:00 a.m. and 8:00 p.m. on any day of the week. Hence, there are 12 time-slots for days in *STT* and 8 time-slots for days in *MW*. Each tutorial is a one-day-a-week 75-minute session that can be offered during certain time-slots of the school week, but it must not be held on the same day as its corresponding class-section. In the current practice, tutorials are not offered on days in *STT* during the period 8:00 a.m.–2:00 p.m.

Next, we define notation related to time-slots, which are indexed chronologically based on the durations of the different types of sessions offered on days in *STT* and *MW*.

1. $T1_{STT} = \{1, \ldots, 12\}$ and $T1_{MW} = \{13, \ldots, 20\}$: Math91 time-slot index sets for days in *STT* and *MW*, respectively.
2. $T1 = T1_{STT} \cup T1_{MW}$.
3. $T2_S = \{1, \ldots, 4\}$, $T2_T = \{5, \ldots, 8\}$, and $T2_{Th} = \{9, \ldots, 12\}$: respectively represent the Sunday, Tuesday, and Thursday tutorial time-slot index sets, where $T2_{STT} = \{1, \ldots, 12\}$.
4. $T2_M = \{13, \ldots, 20\}$ and $T2_W = \{21, \ldots, 28\}$: respectively represent the Monday and Wednesday tutorial time-slot index sets, where $T2_{MW} = \{13, \ldots, 28\}$.
5. $T2 = T2_{STT} \cup T2_{MW}$.
6. $T3_S = \{1, \ldots, 12\}$, $T3_T = \{13, \ldots, 24\}$, and $T3_{Th} = \{25, \ldots, 36\}$: respectively represent the Sunday, Tuesday, and Thursday MathLab time-slot index sets, where $T3_{STT} = \{1, \ldots, 36\}$.
7. $T3_M = \{37, \ldots, 44\}$ and $T3_W = \{45, \ldots, 52\}$: respectively represent the Monday and Wednesday MathLab time-slot index sets, where $T3_{MW} = \{37, \ldots, 52\}$.
8. $T3 = T3_{STT} \cup T3_{MW}$.
9. $T4_S = \{1, \ldots, 12\}$, $T4_T = \{13, \ldots, 24\}$, and $T4_{Th} = \{25, \ldots, 36\}$: respectively represent the Sunday, Tuesday, and Thursday office-hour time-slot index sets, where $T4_{STT} = \{1, \ldots, 36\}$.
10. $T4_M = \{37, \ldots, 44\}$ and $T4_W = \{45, \ldots, 52\}$: respectively represent the Monday and Wednesday office-hour time-slot index sets, where $T4_{MW} = \{37, \ldots, 52\}$.
11. $T4 = T4_{STT} \cup T4_{MW}$.

4. Formulation of Model TAM

In this section, we present a teaching assistant model (TAM) based on selecting a feasible weekly schedule from all possible valid schedules for assistants. Accordingly, let S be the set of all feasible schedules for assistants, indexed by $s = 1, \ldots, |S|$, where any such schedule satisfies all the specified requirements discussed in the foregoing sections. Note that we simply need to generate $|A|$ such schedules, not necessarily distinct, without keeping track of individual assistants, which automatically defeats symmetry in the model. The requisite integer variables and related parameters for Model TAM are introduced in Section 4.1, and the problem constraints are formulated in Section 4.2. The objective function and the overall model are then presented in Section 4.3.

4.1. *Decision Variables and Related Parameters*

Define the following set of integer decision variables:

$$x_s = \text{ number of times schedule } s \in S \text{ is selected for assignment to assistants.}$$

Also, we define the following sets of parameters, whose values are known *a priori* for any given assistant's schedule; these parameters define each column in Model TAM:

$$\delta_{s,t}^k = \begin{cases} 1 & \text{if the assistant teaches a section of Math91-}k, \\ & \text{for } k \in K, \text{ during time-slot } t \in T1 \text{ within schedule } s \in S, \\ 0 & \text{otherwise,} \end{cases}$$

$$\lambda_{s,t}^c = \begin{cases} 1 & \text{if the assistant instructs a tutorial associated with class } c \text{ during} \\ & \text{time-slot } t \in T2 \text{ within schedule } s \in S, \\ 0 & \text{otherwise,} \end{cases}$$

$$\pi_{s,t} = \begin{cases} 1 & \text{if the assistant serves in the MathLab during time-slot } t \in T3 \\ & \text{within schedule } s \in S, \\ 0 & \text{otherwise.} \end{cases}$$

4.2. *Problem Constraints*

The various problem constraints are formulated in turn next.

A) Assigning assistants to Math91

Offered sections of Math91 must be covered by the assistants, as enforced by constraint (4.1) below:

$$\sum_{s \in S} \sum_{t \in T1} \delta_{s,t}^k x_s = N^k, \quad \forall k \in K. \tag{4.1}$$

To avoid clustering sections of Math91 during certain time-slots within $T1$, the following constraint sets upper bounds on the number of sections of these classes that are offered during any such time-slots:

$$\sum_{s \in S} \delta_{s,t}^k x_s \leqslant U_t^k, \quad \forall t \in T1, \ k \in K. \tag{4.2}$$

B) Tutorial sessions

Similar to Math91, tutorials are assigned to assistants as follows:

$$\sum_{s\in S}\sum_{t\in T2_{STT}} \lambda_{s,t}^{c} x_s = N_{STT}^{c}, \quad \forall c, \tag{4.3}$$

$$\sum_{s\in S}\sum_{t\in T2_{MW}} \lambda_{s,t}^{c} x_s = N_{MW}^{c}, \quad \forall c. \tag{4.4}$$

The following constraint spreads tutorial offerings by imposing upper bounds on the number of tutorials that are offered during each of the time-slots in $T2$:

$$\sum_{s\in S} \lambda_{s,t}^{c} x_s \leqslant U_t^{c+3}, \quad \forall t \in T2, c. \tag{4.5}$$

C) MathLab hours

Staffing the MathLab by assistants is achieved by setting lower and upper bounds on the number of assistants that are assigned during any given time-slot as follows:

$$\sum_{s\in S} \pi_{s,t} x_s \geqslant L_t^{ML}, \quad \forall t \in T3, \tag{4.6}$$

$$\sum_{s\in S} \pi_{s,t} x_s \leqslant U_t^{ML}, \quad \forall t \in T3. \tag{4.7}$$

D) Schedule selection

The following constraint ensures that the requisite number of valid schedules is selected for the set of teaching assistants:

$$\sum_{s\in S} x_s = |A|, \quad \forall a. \tag{4.8}$$

Note that the schedule columns generated for the model, as described in the sequel, ensure that the assistants are actually assigned valid weekly workloads.

4.3. *The Overall Model*

The objective function of Model TAM attempts to minimize the sum of the daily assignment time-spans for assistants. Letting c_s represent the time-span associated with schedule $s \in S$, the objective function of the proposed model TAM is given by $\sum_{s\in S} c_s x_s$, which yields the following formulation:

$$\textbf{TAM}: \text{Minimize}\left[\sum_{s\in S} c_s x_s : (4.1)\text{–}(4.8),\ x_s \text{ integer},\ \forall s \in S\right].$$

Next, we define a set of binary variables that will enable us to formulate suitable constraints in Sections 5 and 6 below, whose feasible region characterizes all valid schedules for assistants. This will facilitate the development of a column generation framework in Section 7 to solve Model TAM. Let $K = \{1, 2, 3\}$, and consider the following binary variables, where all indices are assumed to take on their respective values:

$$X^k_{a,t} = \begin{cases} 1 & \text{if assistant } a \text{ teaches a section of Math91-}k,\ k \in K, \\ & \text{during time-slot } t \in T1, \\ 0 & \text{otherwise,} \end{cases}$$

$$Y^c_{a,t} = \begin{cases} 1 & \text{if assistant } a \text{ instructs a tutorial associated with class } c \in C \\ & \text{during time-slot } t \in T2, \\ 0 & \text{otherwise,} \end{cases}$$

$$Z_{a,t} = \begin{cases} 1 & \text{if assistant } a \text{ serves in the MathLab during time-slot } t \in T3, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$W^l_{a,t} = \begin{cases} 1 & \text{if assistant } a \text{ allocates an office-hour during time-slot } t \in T4 \\ & \text{in college } l \in L, \\ 0 & \text{otherwise.} \end{cases}$$

Note that if a section of Math91 is held in college $l \in L$, then all or a certain specified number of office-hours associated with this section as specified in Table 1 must be held in the same college. The same holds for tutorials held in college $l \in \{1, 2\}$, as discussed further in Section 5 below.

5. Assignment of Help-Hours

The number of *help-hours* (MathLab-hours and office-hours) assigned to each assistant depends on the number of allocated sections of Math91, and the number of tutorials to be instructed by this assistant as discussed earlier in Section 1. Hence, the total number of help-hours assigned to each assistant depends on the specific composition of the allocated teaching duties. This is addressed in detail next.

A) Specification of help-hours

Based on Table 1, the following constraints determine the total number of weekly help-hours for a given assistant a:

$$\alpha^k_a = \sum_{t \in T1} X^k_{a,t}, \quad \forall a \text{ and } k \in K, \tag{5.1}$$

$$\beta^c_a = \sum_{t \in T2} Y^c_{a,t}, \quad \forall a,\ c \in C^1 \cup C^2 \cup C^4, \tag{5.2}$$

$$\gamma_a = \sum_{t \in T2} Y^c_{a,t}, \quad \forall a,\ c \in C^3, \tag{5.3}$$

$$h_a^1 = 3\alpha_a^1 + 2\max\{0, \alpha_a^1 - 1\} + \sum_{c\in C^1} \left(2.5\beta_a^c + 1.5\max\{0, \beta_a^c - 1\}\right)$$
$$+ \sum_{c\in C^2} \left(1.5\beta_a^c + \max\{0, \beta_a^c - 1\}\right) + \left(2.5\beta_a^9 + 2.5\max\{0, \beta_a^9 - 1\}\right), \quad \forall a, \tag{5.4}$$

$$h_a^2 = 3\alpha_a^2 + 2\max\{0, \alpha_a^2 - 1\} + 2.5\gamma_a + 1.5\max\{0, \gamma_a - 1\}, \quad \forall a, \tag{5.5}$$

$$h_a^3 = 3\alpha_a^3 + 2\max\{0, \alpha_a^3 - 1\}, \quad \forall a, \tag{5.6}$$

$$h_a = h_a^1 + h_a^2 + h_a^3, \quad \forall a, \tag{5.7}$$

$$h_a \geqslant m + 8, \quad \forall a, \tag{5.8}$$

$$L^{ML} \leqslant m \leqslant U^{ML}. \tag{5.9}$$

Consider any assistant a. Then constraints (5.1)–(5.9) compute the help-hours for assistant a as per the rules specified in Table 1 as follows: for $k \in K$, the variable α_a^k in constraint (5.1) represents the number of sections of Math91-k that are assigned to assistant a. The variable β_a^c in constraint (5.2) represents the number of tutorials associated with class c that are assigned to assistant a, where $c \in C^1 \cup C^2 \cup C^4$, meaning that these tutorials will be held in college $l = 1$. Likewise, constraint (5.3) counts the number of tutorials assigned to assistant a that are associated with class $c = 4$ in the College of Girls, where these tutorials will accordingly be held in college $l = 2$. Based on these class section and tutorial assignments, and according to Table 1, constraints (5.4)–(5.6) compute the total number of help-hours that are correspondingly assigned to assistant a in campuses $l = 1, 2$, and 3, respectively, with a total equal to h_a as given by Constraint (5.7). Constraint (5.8) sets a lower bound for the total number of help-hours assigned to assistant a during the week, where the auxiliary decision variable m represents the MathLab hours that are assigned uniformly to each assistant (see Part B below), and where each assistant is required to allocate at least eight office-hours during the week as indicated in Section 1. Constraint (5.9) restricts the MathLab hours variable m to lie within certain lower and upper bounds given by L^{ML} and U^{ML}, respectively, as specified by the Mathematics Department at KU.

Observe that the nonlinear terms in constraints (5.4)–(5.6) in concert with (5.7)–(5.9) yield nonconvex constraints. The max-operations in these constraints can be linearized using binary variables as generically delineated by the following proposition:

Proposition 1. *Let* $\tau^+ = \max\{0, \tau - 1\}$*, where* $0 \leqslant \tau \leqslant \tau_{\max}$*. Then this can be linearized via the following set of constraints:*

$$\tau - 1 \leqslant \tau^+ \leqslant \tau - \varepsilon_\tau, \tag{5.10}$$

$$\tau_{\max}\varepsilon_\tau \geqslant \tau, \tag{5.11}$$

$$\varepsilon_\tau \in \{0, 1\}, \quad \tau^+ \geqslant 0. \tag{5.12}$$

Proof. We consider three cases. First, suppose that $\tau = 0$. Then, (5.10) and (5.12) yield $0 \leqslant \tau^+ \leqslant -\varepsilon_\tau$, which forces $\varepsilon_\tau = 0$ and $\tau^+ = 0$. Second, suppose that $\tau = 1$. In this case, (5.11) leads to $\varepsilon_\tau = 1$, which with (5.10) results in $\tau^+ = 0$. Finally, when $2 \leqslant \tau \leqslant \tau_{\max}$, we again have that (5.11) and (5.10) respectively imply that $\varepsilon_\tau = 1$ and $\tau^+ = \tau - 1$. □

B) Assignment of MathLab hours and office-hours

Having specified the total number of help-hours for any assistant a as determined above, we can then assign the corresponding MathLab and office-hours to this assistant, noting that the MathLab is located in the College of Science (i.e. $l = 1$). Recall that $m \in [L^{ML}, U^{ML}]$ is a to-be-determined value for the total number of MathLab hours that will be served by each assistant. Hence, noting (5.8), the remaining hours $(h_a - m) \geqslant 8$ will be allocated for office-hours, where h_a is given by (5.7). The following constraints handle the assignment of the MathLab hours:

$$\sum_{t \in T3_{STT}} Z_{a,t} + \sum_{t \in T3_{MW}} (1.5)\ Z_{a,t} = m, \quad \forall a, \tag{5.13}$$

$$L_t^{ML} \leqslant \sum_a Z_{a,t} \leqslant U_t^{ML}, \quad \forall t \in T3. \tag{5.14}$$

Each assistant a is required to serve in the MathLab for a total of (to-be-determined) m hours as enforced by constraint (5.13), where note that each of the *STT* time-slots is of one hour duration while each of the *MW* time-slots is of an hour-and-half duration. Constraint (5.14) sets lower and upper bounds, respectively given by L_t^{ML} and U_t^{ML}, on the number of assistants that need to serve in the MathLab during any time-slot $t \in T3$.

Also, the weekly office-hours are assigned to assistants via the following constraints:

$$\sum_{t \in T4_{STT}} \sum_{l=1}^{3} W_{a,t}^l + \sum_{t \in T4_{MW}} \sum_{l=1}^{3} (1.5)\ W_{a,t}^l = (h_a - m), \quad \forall a, \tag{5.15}$$

$$L_{a,STT}^O \leqslant \sum_{t \in T4_{STT}} \sum_{l=1}^{3} W_{a,t}^l \leqslant U_{a,STT}^O, \quad \forall a, \tag{5.16}$$

$$L_{a,MW}^O \leqslant \sum_{t \in T4_{MW}} \sum_{l=1}^{3} W_{a,t}^l \leqslant U_{a,MW}^O, \quad \forall a, \tag{5.17}$$

$$\sum_{t \in T4_{STT}} W_{a,t}^2 + \sum_{t \in T4_{MW}} (1.5)\ W_{a,t}^2 = h_a^2, \quad \forall a, \tag{5.18}$$

$$\sum_{t \in T4_{STT}} W_{a,t}^3 + \sum_{t \in T4_{MW}} (1.5)\ W_{a,t}^3 = h_a^3, \quad \forall a. \tag{5.19}$$

The total weekly office-hours, $(h_a - m)$, are assigned to assistant a by constraint (5.15), where each *STT* office-hour is for one hour while each *MW* office-hour is for an hour-and-half. Constraints (5.16) and (5.17) distribute the office hours over days in *STT* and *MW* as desired. Constraint (5.18) guarantees that each assistant a allocates all office hours associated with Math91-2 and Math108 (i.e. $c = 4$), as computed via constraint (5.5), in college $l = 2$. In case, as specified by the Mathematics Department at KU, only the direct office-hours per section (without the incremental hours for replications) need to be allocated in college $l = 2$, then we can replace h_a^2 in constraint (5.18) by $3\alpha_a^2 + 2.5\gamma_a$ (see Table 1). Similarly, constraint (5.19) allocates the required office-hours for Math91-2 in college $l = 3$, where h_a^3 can similarly be replaced by $3\alpha_a^3$ as desired.

6. Workload, Commuting, Activity-Per-Time-Slot Restrictions, and Time-Spans

In this section, we formulate constraints related to workload requirements, commuting restrictions and the constraining of a single activity per time-slot.

6.1. *Daily and Weekly Workload Requirements and Features*

Certain daily and weekly workload requirements are discussed in this section.

A) Daily workload requirements

For each assistant, the daily minimum and maximum workload requirements imposed by the Department of Mathematics at KU are enforced by constraints (6.1) and (6.2) given below for days in *STT* and *MW*, respectively:

$$L^d \leqslant \sum_{t \in T1_{STT}} \sum_{k=1}^{3} X_{a,t}^k + \sum_c \sum_{t \in T2_d} Y_{a,t}^c + \sum_{t \in T3_d} Z_{a,t} + \sum_{t \in T4_d} \sum_{l=1}^{3} W_{a,t}^l \leqslant U^d, \quad \forall a \text{ and } d \in STT, \tag{6.1}$$

$$L^d \leqslant \sum_{t \in T1_{MW}} \sum_{k=1}^{3} X_{a,t}^k + \sum_c \sum_{t \in T2_d} Y_{a,t}^c + \sum_{t \in T3_d} Z_{a,t} + \sum_{t \in T4_d} \sum_{l=1}^{3} W_{a,t}^l \leqslant U^d, \quad \forall a \text{ and } d \in MW. \tag{6.2}$$

B) Distribution of teaching loads

For a given assistant, teaching duties pertaining to tutorials and to sections of Math91 should not be clustered on a given day and should be spread over the entire week as desired. Thus, constraints (6.3) and (6.4) below set lower and upper bounds as specified by the Department of Mathematics at KU on the total hours of teaching duties for days in *STT* and *MW*, respectively:

$$L_{STT}^d \leqslant \sum_{t \in T1_{STT}} \sum_{k=1}^{3} X_{a,t}^k + \sum_c \sum_{t \in T2_d} Y_{a,t}^c \leqslant U_{STT}^d, \quad \forall a \text{ and } d \in STT, \tag{6.3}$$

$$L_{MW}^{d} \leqslant \sum_{t \in T1_{MW}} \sum_{k=1}^{3} X_{a,t}^{k} + \sum_{c} \sum_{t \in T2_{d}} Y_{a,t}^{c} \leqslant U_{MW}^{d}, \quad \forall a \text{ and } d \in MW. \tag{6.4}$$

C) Weekly workload

The overall weekly hour-load for a given teaching assistant a is required to lie within some specified range $[L, U]$, as enforced by the constraint given next, where the weekly workload values are determined according to Table 1. In current practice, we have $[L, U] = [40, 48]$.

$$L \leqslant \left[8 \sum_{t \in T1} \sum_{k=1}^{3} X_{a,t}^{k} + 8 \sum_{c \in C^1 \cup C^3} \sum_{t \in T2} Y_{a,t}^{c} + 4 \sum_{t \in T2} \sum_{c \in C^2} Y_{a,t}^{4} + 10 \sum_{t \in T2} Y_{a,t}^{9} \right]$$
$$\leqslant U, \quad \forall a. \tag{6.5}$$

6.2. *Minimum and Maximum Number of Different Class Subjects*

Teaching duties of an assistant during a given term must cover different class subjects (i.e. combinations of sections of Math91, and tutorials associated with different classes). Currently, the minimum and maximum number of different class subjects are given by two and three, respectively, where these bounds are enforced as follows. First, the following constraint guarantees that the maximum number of sections of Math91 that are assigned to an assistant is three:

$$\sum_{t \in T1} \sum_{k=1}^{3} X_{a,t}^{k} \leqslant 3, \quad \forall a, \tag{6.6}$$

and the following constraint guarantees that the maximum number of tutorials associated with any particular class that are assigned to an assistant is three:

$$\sum_{t \in T2} Y_{a,t}^{c} \leqslant 3, \quad \forall a, c. \tag{6.7}$$

Second, the following constraints ensure that the teaching load of an assistant contains at most three different classes:

$$H_a \geqslant X_{a,t}^{k}, \quad \forall a,\ t \in T1,\ k \in K, \tag{6.8}$$

$$H_a^c \geqslant Y_{a,t}^{c}, \quad \forall a,\ t \in T2, c, \tag{6.9}$$

$$H_a + \sum_{c} H_a^c \leqslant 3, \quad \forall a. \tag{6.10}$$

Third, the minimum number of different subjects is modelled via the following constraints:

$$\Gamma_a \leqslant \sum_{t \in T1} \sum_{k=1}^{3} X_{a,t}^{k}, \quad \forall a, \tag{6.11}$$

$$\Gamma_a^c \leqslant \sum_{t \in T2} Y_{a,t}^c, \quad \forall a, \tag{6.12}$$

$$\Gamma_a + \sum_c \Gamma_a^c \geqslant 2, \quad \forall a, \tag{6.13}$$

$$0 \leqslant \Gamma_a \leqslant 1, \quad 0 \leqslant \Gamma_a^c \leqslant 1, \quad \forall a, c. \tag{6.14}$$

Note that Γ_a or Γ_a^c can be one only if at least one of variables on the right-hand side of (6.11) or (6.12), respectively, equals one, and are zero otherwise. Hence, (6.13) ensures the desired restriction, even with the Γ-variables declared to be continuous on [0, 1] as specified in (6.14).

6.3. *Commuting Between Campuses and at Most One Activity Per Time-sLOT*

In this section, we formulate constraints to ensure that no two duties are assigned over consecutive periods in two distinct colleges, and also that the maximum number of switches between locations on any given day is smaller than some pre-specified value N^{LS}. To achieve this, we first define the following binary variables, noting that $T1_d \equiv T1_{STT}$ if $d \in STT$ and $T1_d \equiv T1_{MW}$ if $d \in MW$:

$$g_{a,t}^{l,d} = \begin{cases} 1 & \text{if assistant } a \text{ is in college } l \in L \text{ during time-slot } t \in T1_d \text{ of day } d \in D, \\ 0 & \text{otherwise,} \end{cases}$$

$$f_{a,t}^{l,d} \equiv g_{a,t}^{l,d} g_{a,(t+1)}^{l,d}$$

$$= \begin{cases} 1 & \text{if assistant } a \text{ is in college } l \in L \text{ during time-slots } t \text{ and } t+1, \\ & \text{on day } d \in D \text{ where } t \in T1_d \backslash \{12, 20\}, \\ 0 & \text{otherwise.} \end{cases}$$

Then the following constraints enforce the switching restrictions between colleges:

$$\sum_{l=1}^{3} g_{a,t}^{l,d} = 1, \quad \forall a,\ t \in T1_d,\ d \in D, \tag{6.15}$$

$$\sum_{t \in T1_d \backslash \{12,20\}} \sum_{l=1}^{3} f_{a,t}^{l,d} \geqslant \left(|T1_d| - 1 - N^{LS}\right), \quad \forall a, d \in D, \tag{6.16}$$

$$f_{a,t}^{l,d} \leqslant g_{a,t}^{l,d}, \quad f_{a,t}^{l,d} \leqslant g_{a,t+1}^{l,d}, \quad f_{a,t}^{l,d} \geqslant g_{a,t}^{l,d} + g_{a,t+1}^{l,d} - 1,$$

$$f_{a,t}^{l,d} \geqslant 0, \qquad \forall a,\ t \in T1_d,\ d \in D,\ l \in L. \tag{6.17}$$

Constraint (6.15) assures that any assistant a is present in exactly one college during any time-slot $t \in T1_d$ of any given day d. Note that for a given day d, there are $(|T1_d| - 1)$ time-slot transitions, and it is required that at most N^{LS} of these transitions involve switching colleges. Hence, for any assistant a and day d, we need at least

$(|T1_d| - 1 - N^{LS})$ f-variables to be one, as enforced by constraint (6.16). The f-variables are linearized via the logical restrictions given in constraint (6.17), where the restriction $f_{a,t}^{l,d} \geqslant g_{a,t}^{l,d} + g_{a,t+1}^{ld} - 1$ can be dropped since (6.16) automatically prefers the corresponding f-variables to be one whenever possible. Finally, any duty assignment is permissible for an assistant at any college l during time-slot $t \in T1_d$ of day d, only if this assistant is available at location l during time-slot $(t-1) \in T1_d$ of day d. This is accomplished via the constraints in Parts A–C below.

A) Presence at colleges on days in *STT* during the period 8:00 a.m.–2:00 p.m.

Define the set $DJ = \{(S, 0), (T, 12), (Th, 24)\}$, where the first element of each pair in *DJ* represents some day $d \in STT$ and the second element represents an index $j = \{0, 12, 24\}$ that is used to model the various defined time-slot activities. The following constraints handle the availability restrictions for days in *STT* during the time-duration 8:00 a.m.–2:00 p.m., noting that tutorials are not offered during this time-duration:

$$X_{a,t}^{1} + Z_{a,t+j} + W_{a,t+j}^{l} \leqslant g_{a,t-1}^{1,d}, \quad \forall a,\ t \in \{2,\ldots,6\},\ (d,j) \in DJ, \tag{6.18}$$

$$X_{a,t}^{2} + W_{a,t}^{2} \leqslant g_{a,t-1}^{2,d}, \quad \forall a,\ t \in \{2,\ldots,6\},\ (d,j) \in DJ, \tag{6.19}$$

$$X_{a,t}^{3} + W_{a,t}^{3} \leqslant g_{a,t-1}^{3,d}, \quad \forall a,\ t \in \{2,\ldots,6\},\ (d,j) \in DJ. \tag{6.20}$$

B) Presence at colleges on days in *STT* for the period 2:00 p.m.–8:00 p.m.

Tutorials can be offered during the period 2:00 p.m.–8:00 p.m. on any day in *STT*, each of which is of one-hour-and-half duration, while other load activities on these days cover one-hour time-slots. Therefore, this case entails special attention since we are dealing with workload activities having time-slots of different durations. Note that the time duration of the hour-and-half tutorial time-slot $t = 1$ contains the one-hour time duration of the Math91 time-slot $t = 7$, the MathLab time-slot $t = 7$, and the office-hour time-slot $t = 7$. Moreover, the tutorial time-slot $t = 1$ partially overlaps with the one-hour time duration of the Math91 time-slot $t = 8$, the MathLab time-slot $t = 8$, and the office-hour time-slot $t = 8$. Similar comments apply for the tutorial time-slots $t = 2$, 3, and 4. For convenience in formulation, we define the following sets, each element of which is given by a quadruple (d, t, t_1, j), where $d \in D_{STT}$, $t \in \{7, \ldots, 12\}$ (one-hour MathLab time-slots), $t_1 \in T2_{STT}$ (an-hour-and-half tutorial time-slots), and where j is an index as before that will be used to appropriately model the various activity time-slots:

$$\begin{aligned}
DJ_S = &\big\{(S,7,1,0), (S,8,1,0), (S,8,2,0), (S,9,2,0), (S,10,3,0), (S,11,3,0),\\
&(S,11,4,0), (S,12,4,0)\big\},\\
DJ_T = &\big\{(T,7,5,12), (T,8,5,12), (T,8,6,12), (T,9,6,12), (T,10,7,12),\\
&(T,11,7,12), (T,11,8,12), (T,12,8,12)\big\},\\
DJ_{Th} = &\big\{(Th,7,9,24), (Th,8,9,24), (Th,8,10,24), (Th,9,10,24),\\
&(Th,10,11,24), (Th,11,11,24), (Th,11,12,24), (Th,12,12,24)\big\},
\end{aligned}$$

and

$$DJ_{STT} = DJ_S \cup DJ_T \cup DJ_{Th},$$

where, DJ_S, DJ_T, and DJ_{Th} are respectively associated with the days Sunday, Tuesday, and Thursday. The second and third terms of each quadruple in DJ_{STT} represent the overlapping or partially overlapping time-slots associated with Math91 and tutorials. For example, for $d = 1$, the Math91 time-slot $t = 8$ partially overlaps with the tutorial time-slots $t = 1$ and $t = 2$. Based on the above discussion, we formulate the following constraints:

$$X^1_{a,t} + \sum_{c \in C_1} Y^c_{a,t_1} + Z_{a,t+j} + W^1_{a,t+j} \leqslant g^{1,d}_{a,t-1}, \quad \forall a,\ (d,t,t_1,j) \in DJ_{STT}, \tag{6.21}$$

$$X^2_{a,t} + Y^4_{a,t_1} + W^2_{a,t+j} \leqslant g^{2,d}_{a,t-1}, \quad \forall a,\ (d,t,t_1,j) \in DJ_{STT}, \tag{6.22}$$

$$X^3_{a,t} + W^3_{a,t+j} \leqslant g^{3,d}_{a,t-1}, \quad \forall a,\ (d,t,t_1,j) \in DJ_{STT}. \tag{6.23}$$

C) Presence at colleges on days in *MW*

Constraints (6.24)–(6.26) below handle the availability-on-campus restrictions for $d \in MW$:

$$X^1_{a,t} + \sum_{c \in C_1} Y^c_{a,t+j} + Z_{a,t+j+24} + W^1_{a,t+j+24} \leqslant g^{1,d}_{a,t-1},$$
$$\forall a,\ t = 14, \ldots, 20,\ (d,j) \in \{(M,0),(W,8)\}, \tag{6.24}$$

$$X^2_{a,t} + Y^4_{a,t+j} + W^2_{a,t+j+24} \leqslant g^{2,d}_{a,t-1},$$
$$\forall a,\ t = 14, \ldots, 20,\ (d,j) \in \{(M,0),(W,8)\}, \tag{6.25}$$

$$X^3_{a,t} + W^3_{a,t+j+24} \leqslant g^{3,d}_{a,t-1},$$
$$\forall a,\ t = 14, \ldots, 20,\ (d,j) \in \{(M,0),(W,8)\}. \tag{6.26}$$

D) One-activity-per-time-slot restrictions

Naturally, at most one activity can be assigned to any assistant during any given time period of the school week. Constraints (6.18)–(6.26) automatically imply this restriction for all time-slots of week except for the first time-slot of each day, which is handled next.

a) Restrictions for the first time-slot of days in D_{STT}

Constraints (6.27)–(6.28) below respectively enforce the single-activity restriction during the first time-slot of days $d \in D_{STT}$.

$$\sum_{k=1}^{3} X^k_{a,1} + Z_{a,1} + \sum_{l=1}^{3} W^l_{a,1} \leqslant 1, \quad \forall a, \tag{6.27}$$

$$\sum_{k=1}^{3} X_{a,1}^{k} + Z_{a,13} + \sum_{l=1}^{3} W_{a,13}^{l} \leqslant 1, \quad \forall a, \tag{6.28}$$

$$\sum_{k=1}^{3} X_{a,1}^{k} + Z_{a,25} + \sum_{l=1}^{3} W_{a,25}^{l} \leqslant 1, \quad \forall a. \tag{6.29}$$

b) Restrictions for the first time-slot of days in D_{MW}

Constraints (6.30)–(6.31) below respectively enforce the single-activity restriction during the first time-slot of days $d \in D_{MW}$.

$$\sum_{k=1}^{3} X_{a,13}^{k} + \sum_{c} Y_{a,13}^{c} + Z_{a,37} + \sum_{l=1}^{3} W_{a,37}^{l} \leqslant 1, \quad \forall a, \tag{6.30}$$

$$\sum_{k=1}^{3} X_{a,13}^{k} + \sum_{c} Y_{a,21}^{c} + Z_{a,45} + \sum_{l=1}^{3} W_{a,45}^{l} \leqslant 1, \quad \forall a. \tag{6.31}$$

6.4. *Time-Spans*

We introduce in this section constraints to represent the daily work time-spans of assistants. The constraints formulated above enforce all the stated departmental rules regarding the generation of the weekly load of assistants. However, the schedule for each assistant can be enhanced by reducing the daily time-span of duties in an equitable manner. This is addressed next in Sections A–C.

A) Earliest time for Sunday

The following constraints provide the earliest activity time of an assistant on Sunday, starting at time zero for the first time-slot, where recall that the total number of time-slots on Sunday is $|T3_S| \equiv 12$:

$$E_a^S \leqslant (t-1)X_{a,t}^{k} + |T3_S|\big(1 - X_{a,t}^{k}\big), \quad \forall a,\ t \in T1_{STT},\ k \in K, \tag{6.32}$$

$$E_a^S \leqslant \big[6 + 1.5(t-1)\big]Y_{a,t}^{c} + |T3_S|\big(1 - Y_{a,t}^{c}\big), \quad \forall a,\ t \in \{1,2,3,4\}, c, \tag{6.33}$$

$$E_a^S \leqslant (t-1)Z_{a,t} + |T3_S|(1 - Z_{a,t}), \quad \forall a,\ t \in T3_S, \tag{6.34}$$

$$E_a^S \leqslant (t-1)W_{a,t}^{l} + |T3_S|\big(1 - W_{a,t}^{l}\big), \quad \forall a,\ t \in T4_S,\ l \in L. \tag{6.35}$$

For any assistant a, constraint (6.32) asserts that the earliest time this assistant teaches a section of Math91 on Sunday is an upper bound on the variable E_a^S. If this assistant does not teach any section of Math91 on Sunday, then all the corresponding X-variables are identically zero, and hence, constraint (6.32) reduces to the redundant constraint $E_a^S \leqslant |T3_S|$. Constraints (6.33)–(6.35) are formulated similarly for tutorials, MathLab sessions, and office-hours, respectively.

B) Latest time for Sunday

Likewise, the following constraints provide the latest activity time for an assistant on Sunday:

$$L_a^S \geqslant tX_{a,t}^k, \quad \forall a,\ t \in T1_{STT},\ k \in K, \tag{6.36}$$

$$L_a^S \geqslant (6 + 1.5t)Y_{a,t}^c, \quad \forall a,\ t \in \{1, 2, 3, 4\}, c, \tag{6.37}$$

$$L_a^S \geqslant tZ_{a,t}, \quad \forall a,\ t \in T3_S, \tag{6.38}$$

$$L_a^S \geqslant tW_{a,t}^l, \quad \forall a,\ t \in T4_S,\ l \in L. \tag{6.39}$$

For an assistant a, minimizing the term $(L_a^S - E_a^S)$ achieves the objective of minimizing the time-span for this assistant on Sunday. The terms for Monday through Thursday can be formulated similarly; let these collectively be denoted by $(L_a^d - E_a^d)$, $\forall d \in D$.

7. A Column Generation Framework

We begin by defining a feasible region, denoted by FR, which characterizes the columns of the coefficient matrix of TAM that are associated with the $x \equiv (x_1, \ldots, x_{|S|})$-variables. This region FR is essentially described in terms of the decision variables (X, Y, Z, W) defined at the end of Section 4, where the components of these variables are delineated below, subject to all the constraints defined above in Sections 5 and 6 along with other auxiliary variables defining these constraints.

$$\begin{aligned} FR = \big\{&(X, Y, Z, W) :\ X = (X_t^k,\ t \in T1,\ k \in K), \\ &Y = (Y_t^c,\ t \in T2,\ c \in C),\ Z = (Z_t,\ t \in T3), \\ &\text{and } W = (Z_t^l,\ t \in T4,\ l \in L),\ \text{subject to (5.1)–(5.19), (6.1)–(6.39)}\big\}, \end{aligned}$$

where all the foregoing constraints are written without the assistant index a. Given any $(X, Y, Z, W) \in FR$, the corresponding values of the δ-, λ-, π-and c-parameters for defining the column of the associated x_s-variable in Model TAM are accordingly given as follows for all relevant index values:

$$\delta_{s,t}^k = X_t^k, \quad \lambda_{s,t}^c = Y_t^c, \quad \pi_{s,t} = Z_t, \quad \text{and} \quad c_s = \sum_{d \in D} \big(L^d - E^d\big), \tag{7.1}$$

where E^d and L^d, are respectively, the earliest and latest activity times obtained when following schedule $s \in S$, $\forall d \in D$.

Next, in Section 7.1 we present a *column generation method* (CGM) to solve the continuous relaxation of Model TAM, denoted by $\overline{\text{TAM}}$, which will then be used in Section 7.2 within a column generation heuristic to derive a solution for Model TAM (see Barnhart *et al.*, 1998; Bazaraa *et al.*, 2010, for a general discussion on column generation).

7.1. *A Column Generation Method (CGM)*

Toward this end, suppose that at some iteration of the revised simplex method as applied to solve $\overline{\text{TAM}}$, we have a basic feasible solution with I^b and I^{nb} respectively representing the index sets for the basic and nonbasic variables. Furthermore, let ζ denote the corresponding complementary dual solution, with components associated with constraints (4.1)–(4.8), respectively, being given as follows:

$$\zeta^1 \equiv \left(\zeta_k^1,\ \forall k \in K\right), \quad \zeta^2 \equiv \left(\zeta_{k,t}^2,\ \forall t \in T1,\ k \in K\right),$$
$$\zeta^3 \equiv \left(\zeta_c^3,\ \forall c \in C\right), \quad \zeta^4 \equiv \left(\zeta_c^4,\ \forall c \in C\right), \quad \zeta^5 \equiv \left(\zeta_{c,t}^5,\ \forall t \in T2,\ c \in C\right),$$
$$\zeta^6 \equiv \left(\zeta_t^6,\ \forall t \in T3\right), \quad \zeta^7 \equiv \left(\zeta_t^7,\ \forall t \in T3\right) \quad \text{and} \quad \zeta^8.$$

At any iteration of the revised simplex method, we first explicitly price the nonbasic slack variables associated with constraints (4.2), (4.5), (4.6), and (4.7) to identify a most enterable candidate nonbasic variable. If any exists, we enter such a variable into the basis and accordingly update the primal and dual solutions and repeat. Hence, suppose that no such variable is enterable. We next implicitly price the x_s-variables to find a candidate nonbasic x_s-variable that has the smallest (most negative) reduced cost to enter the basis by solving the following *Pricing Problem* PP:

PP:

$$\begin{aligned}\text{Minimize} \sum_{d\in D}\left(L^d - E^d\right) - \Bigg[&\sum_{k\in K}\Bigg(\sum_{t\in T1} X_t^k\Bigg)\zeta_k^1 + \sum_{t\in T1}\sum_{k\in K} X_t^k \zeta_{k,t}^2 \\ &+ \sum_c\Bigg(\sum_{t\in T2_{STT}} Y_t^c\Bigg)\zeta_c^3 + \sum_c\Bigg(\sum_{t\in T2_{MW}} Y_t^c\Bigg)\zeta_c^4 + \sum_c\sum_{t\in T2} Y_t^c \zeta_{c,t}^5 \\ &+ \sum_{t\in T3} Z_t\zeta_t^6 + \sum_{t\in T3} Z_t\zeta_t^7 + \zeta^8\Bigg],\end{aligned}$$

subject to $(X, Y, W, Z) \in FR$, where $\zeta_{k,t}^2 \leqslant 0$, $\forall t \in T1, k \in K$; $\zeta_{c,t}^5 \leqslant 0$, $\forall t \in T2, c$; $\zeta_t^6 \geqslant 0$, $\forall t \in T3$; and $\zeta_t^7 \leqslant 0$, $\forall t \in T3$, and where the remaining ζ-variables are unrestricted in sign.

Letting $v(\text{P})$ denote the optimal objective function value for any Model P, if $v(\text{PP}) \geqslant 0$, then none of the x_s-variables are enterable into the basis and we have at hand an optimal solution to Problem $\overline{\text{TAM}}$. Otherwise, if $v(\text{PP}) < 0$, then we will have obtained a candidate entering x_s-variable for Model $\overline{\text{TAM}}$ from the optimal solution derived for Model PP. We thus enter this variable into the basis, re-optimize the restricted master program, and repeat.

Remark 1. For the sake of convenience and efficiency, we use a set of initial columns derived from a manually generated schedule to compose a basis along with artificial columns as necessary. This construct enabled us to solve Model $\overline{\text{TAM}}$ via column generation model relatively easily, without having to resort to any dual stabilization techniques (see Bazaraa *et al.*, 2010, for example).

7.2. *A Column Generation Heuristic CGH for Model TAM*

In this section, we discuss a column generation-based sequential variable-fixing heuristic procedure to construct a good quality feasible solution for Model TAM by recursively applying the CGM procedure for deriving valid schedules for assistants as described in the foregoing section. In this process, suppose that we have obtained a solution $\overline{x}$ to Model $\overline{\text{TAM}}$ by applying the CGM procedure. Let I^b be partitioned into $I^{b,i}$ and $I^{b,f}$, which respectively represents the index sets of basic variables that are integer-valued and that are fractional in the solution $\overline{x}$. Note that if $I^{b,f} = \emptyset$ (and if all artificial variables are zero), we have $|A|$ schedules at hand by constraint (4.8), and we can stop with this solution as optimal to Model TAM. Otherwise, we initialize a set $J \equiv I^{b,i}$, where the index set J represents valid schedules for some $\sum_{s \in J} \overline{x}_s$ assistants.

Let $\hat{s} \in \text{arglexmax}_{s \in I^{b,f}} \{\overline{x}_s - \lfloor \overline{x}_s \rfloor, -c_s\}$, where $\lfloor \overline{x}_s \rfloor$ represents the greatest integer less than or equal to $\overline{x}_s$, and augment the set $J \leftarrow J \cup \{\hat{s}\}$. Consider a modified version of TAM denoted by TAM_J, which is stated as follows:

TAM$_J$:

$$\text{Minimize} \left\{ \sum_{s \in S} c_s x_s : \ (7.1)\text{–}(7.8),\ x_s = \lceil \overline{x}_s \rceil,\ \forall s \in J;\ x_s \text{ integer},\ \forall s \in S \right\},$$

where $\lceil\ \rceil$ denotes the rounding-up operation. Note that we have fixed the selection of schedules for $\sum_{s \in S} \lceil \overline{x}_s \rceil$ assistants in Model TAM, which correspondingly modifies the right-hand sides of the constraints in this model. However, we still retain all the variables x_s, $s \in S$, in the model since some columns corresponding to indices in J might be re-generated as new variable columns (for additional assistants). We can now solve the LP relaxation $\overline{\text{TAM}_J}$ of Model TAM_J using the foregoing CGM procedure, where the pricing operations for column generation are performed as before. The overall proposed column generation heuristic (CGH) for the assistants then proceeds as follows:

Heuristic CGH
Initialization

- Set $J = \emptyset$.

Main Step

- Solve Model $\overline{\text{TAM}_J}$ using CGM, and let $\overline{x}$ denote the resulting solution.
- Determine the index sets $I^{b,i}$ and $I^{b,f}$ as defined above, and let $J \leftarrow J \cup I^{b,i}$.
- If $I^{b,f} = \emptyset$, then stop; the required schedules for assistants are collectively described by the set J. Otherwise, proceed to the next step.

- Let $\hat{s} \in \text{arglexmax}_{s \in I^{b,f}} \{\overline{x}_s - \lfloor \overline{x}_s \rfloor, -c_s\}$, and update $J \leftarrow J \cup \{\hat{s}\}$.
- Repeat the Main Step.

REMARK 2. Observe that we are using a "diving" heuristic here with no backtracking because of the structural complexity of the problem. At each iteration of the procedure, the set J is augmented by at least one element, and accordingly, certain load-activities are assigned to specific teaching assistants. Because the cardinality of J cannot exceed $|A|$, the algorithm terminates whenever $|J| = |A|$, thus yielding the desired teaching assistant schedules. In our computation experimentation, the pricing problem PP was feasible for all test cases. However, one issue that might cause infeasibility during the solution procedure is the lower-bounding restriction of constraint (6.5), whereby at some iteration, because of the sequential fixing process, it is possible that the pricing problem PP is infeasible, whence no entering column would be generated, and so, the overall problem has no feasible completion. In such a case, we could relax these lower bounding restrictions. Also, at each iteration of this heuristic, we can add artificial variables that represent supplementary teaching assistants (i.e. assistants hired on a term-basis as needed) within Model TAM in order to ensure its feasibility. If some of the artificial variables remain positive at termination of the heuristic, we can manually adjust the generated assistants' schedules and combine artificial variables as needed to derive schedules for appropriate supplementary assistants.

8. Computational Results

In this section, we present computational results related to solving Model TAM via the proposed column generation approach. For test purposes, we consider six practical problem instances related to the Mathematics Department at KU. Detailed information related to these test problems as well as time-slot specifications and the schedules generated via the proposed modelling approaches are posted at www.al-yakoob.com. We also comment here that a mixed-integer programming (MIP) model that was formulated similar to the description in Sections 5 and 6 to directly solve Problem TAP was found to be intractable and yielded no feasible solution.

Next, we define some additional notation that will be used in this section. Let $v_{\text{TAM}}^{\text{CGM}}$ be the objective function value of the linear relaxation $\overline{\text{TAM}}$ of Model TAM that is obtained via Procedure CGM, and let $v_{\text{TAM}}^{\text{CGH}}$ be the objective function value of Model TAM that is obtained via Heuristic CGH. Let v^{Manual} be the objective function value associated with the schedules obtained manually using the currently implemented procedure. We also define the percentage gap corresponding to any given objective function value v_1 with respect to v_2 by $\text{PG}(v_1, v_2) = \left[\frac{v_2 - v_1}{v_2}\right]100\%$. We used the software package CPLEX-MIP (version 12.0) with its default settings to solve the different subproblems within Heuristic CGH. Finally, we let RT denote the run-time in seconds (sec), where all runs have been made on a Pentium IV computer having CPU 3.00 GHz and 4GB of RAM, with coding in JAVA.

Table 2
Computational results related to solving Models $\overline{\text{TAM}}$ and TAM.

Test problem	$v_{\overline{\text{TAM}}}^{\text{CGM}}$	$v_{\text{TAM}}^{\text{CGH}}$	No. of iterations for CGH	Total RT for CGH (s)	PG($\overline{\text{TAM}}$, TAM)
	456.0	473.0	13	361.75	3.59%
	592.0	615.0	17	395.07	3.73%
	904.0	934.0	24	451.20	3.21%
	1057.0	1079.0	30	459.12	2.03%
	1435.0	1460.0	35	505.18	1.71%
	1528.0	1542.0	40	512.21	0.90%
Average	**995.33**	**1017.16**	**26.5**	**447.42**	**2.53%**

8.1. *Computational Experience for Solving Models* $\overline{\text{TAM}}$ *and TAM*

Table 2 presents computational results related to solving Model $\overline{\text{TAM}}$ using Procedure CGM and deriving a solution for Model TAM via the proposed column generation heuristic CGH. Feasible solutions to Models $\overline{\text{TAM}}$ and TAM for all the six test problems were obtained in 417.49 and 447.42 CPU seconds on average, respectively. In this solution process, we used a given manually generated solution for each test case to initialize CGM. As a result, no artificial variables were needed to ensure the feasibility of Model $\overline{\text{TAM}}$. The resulting optimality gaps for all the six test cases were relatively small, ranging from 0.90% to 3.73%, with an average of 2.53%. In particular, we obtained the smallest two optimality gaps for test problems P_5 and P_6, even though these two test cases involve more teaching assistant and load activities. Based on this performance, we did not find it necessary to explore any further dual stabilization techniques in order to enhance the solvability of Model $\overline{\text{TAM}}$ (see, for example, Bazaraa *et al.*, 2010, pp. 340–391, for a general discussion). Moreover, the average percentage improvement of the objective function values obtained via the modelling approach (CGH) over that obtained via the manual approach was 27.58%. For the sake of documentation, actual schedules for Test Problems P_1 and P_2 are posted at www.al-yakoob.com.

It is worth noting here that the schedules generated for the test problems $P_1, \dots, P_6$ attain many desirable features, particularly those that enhance the satisfaction of teaching assistants by virtue of reducing the commuting times between campuses as well as the daily work time-spans, which could not be otherwise achieved using manual approaches due to the highly combinatorial nature of the problem. As seen in Table 4, the average over the six test cases of the mean number of commuting trips between campuses obtained via the manual approach and the modelling approach (CGH), are respectively, given by 0.807 and 0.288, while the average standard deviations of the number of commuting trips between campuses obtained via the manual approach and the modelling approach (CGH) are given by 1.085 and 0.548, respectively. The average means of the weekly time spans obtained via the manual approach and the modelling approach (CGH) are given by 48.45 and 42.78, respectively, while the average standard deviations of the weekly time spans obtained via the manual approach and the modelling approach (CGH) are given by 10.32 and 1.828, respectively. This substantiates the superiority of the modelling approach over

Table 3
Comparison between the manual approach and procedure CGH.

Test problem	v^{Manual}	$v_{\text{TAM}}^{\text{CGH}}$	$\left[\frac{v^{\text{Manual}}-v_{\text{TAM}}^{\text{CGH}}}{v^{\text{Manual}}}\right] * 100$
	684	473.0	30.85
	895	615.0	31.28
	1257	934.0	25.70
	1432	1079.0	24.65
	1958	1460.0	25.43
	2128	1542.0	27.54
Average	**1392**	**1017**	**27.58**

Table 4
Comparison between the manual approach and procedure CGH.

Test problem	No. of commuting trips between campuses				Weekly time spans			
	Manual approach		Modelling approach using CGH		Manual approach		Modelling approach using CGH	
	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation
	0	0	0	0	38.46	2.47	36.31	2.14
	0	0	0	0	43.23	6.30	42.31	1.97
	1.41	0.62	0.18	0.39	46.50	7.37	45.83	1.20
	1.06	2.01	0.44	0.98	54.67	15.28	44.17	1.76
	1.16	1.95	0.53	0.96	53.68	15.46	43.79	2.27
	1.21	1.93	0.58	0.96	54.16	15.02	44.26	1.63
Average	**0.807**	**1.085**	**0.288**	**0.548**	**48.45**	**10.32**	**42.78**	**1.828**

the manual approach, where the former provides solutions that are more desirable (lower means) as well as more equitable (lower standard deviations).

9. Summary and Conclusions

In this paper, we have proposed novel formulations and solution procedures for a teaching assistant workload assignment problem. Although our modelling approach and computational results pertain to a case study related to the Department of Mathematics at Kuwait University, from the point of view of presenting the proof-of-concept, this work can be readily extended to solve similar problems that are encountered by many academic institutions worldwide.

Because a directly formulated MIP model was found intractable, we designed a column generation based model, which composes feasible sets of schedules for assistants in order to meet the requirements for various workload activities. Due to the exponential number of variables in Model TAM, a column generation method (CGM) was designed to solve its LP relaxation, based on which, a sequential-fixing column generation-based heuristic (CGH) was devised to derive a good quality feasible solution. Despite the size, complexity, and highly combinatorial nature of the problem, computational results related to solving Model TAM using Heuristic CGH indicated that the proposed column generation approach consistently provided good quality solutions with an average optimality gap and average CPU time of 2.53% and 447.42 seconds, respectively. In particular, we obtained the smallest two gaps of 1.71% and 0.90%, respectively, for test problems P_5 and P_6, respectively, even though these two test cases involve more teaching assistant and load activities than the other test problems. Moreover, the proposed modelling approach (CGH) provided better schedules with respect to desirability and related to commuting trips between campuses and weekly time spans.

Note that equity issues related to assistants have been omitted from the formulation of Model TAM in order to maintain a convenient column structure of the model, which facilitates the design of an efficient column generation heuristic. However, equity issues could be accommodated at a later stage by retaining all the columns generated during the process of solving the relaxation to Model TAM as described in Section 7, and then optimizing TAM using these columns in addition to equity-driven objective terms and side-constraints. We propose this consideration for future research.

Acknowledgements. This research work was supported by Kuwait University under Research Grant No. [US 01/11] and by the National Science Foundation under Grant No. [CMMI-0969169]. The authors also gratefully acknowledge the assistance of Ms. Renju Lekshmi in implementing the developed procedures.

References

Al-Yakoob, S. M., Sherali, H.D. (2006). Mathematical programming models and algorithms for a class-faculty assignment problem. *European Journal of Operational Research*, 173(2), 488–507.

Al-Yakoob, S.M., Sherali, H.D. (2007). A mixed-integer programming approach to a class timetabling problem: A case study with gender policies and traffic considerations. *European Journal of Operational Research*, 180(3), 1028–1044.

Al-Yakoob, S.M., Sherali, H.D. (2015). A column generation mathematical programming approach for a class-faculty assignment problem with preferences. *Computational Management Science*, 12(2), 297–318.

Al-Yakoob, S.M., Sherali, H.D., Al-Jazzaf M. (2010). A mixed-integer mathematical modeling approach to exam timetabling. *Computational Management Science*, 7(1), 19–46.

Avella, P., Vasil'Ev, I. (2005). A computational study of a cutting plane algorithm for university course timetabling. *Journal of Scheduling*, 8(6), 497–514.

Baker, M.A., Aksop, C. (2008). A 0–1 integer programming approach to a university timetabling problem. *Hacettepe Journal of Mathematics and Statistics*, 37(1), 41–55.

Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W., Vance, P.H. (1998). Branch-and-price: column generation for solving huge integer programs. *Operation Research*, 46(3), 316–329.

Bazaraa, M.S., Jarvis, J.J., Sherali, H.D. (2010). *Linear Programming and Network Flows*. 4th ed., John Wiley and Sons, Hoboken.

Birbas, T., Daskalaki, S., Housos, E. (1997). Timetabling for Greek high schools. *Journal of the Operational Research Society*, 48(2), 1191–1200.

Boland, N., Hughes, B.D., Merlot, L.T., Stuckey, P.J. (2008). New integer programming approaches for course timetabling. *Computers and Operations Research*, 35(7), 2209–2233.

Burke, E.K., Petrovic, S. (2002). Recent research directions in automated timetabling. *European Journal of Operational Research*, 140(2), 266–280.

Burke, E.K., Gendreau, M. (2008). Practice and theory of automated timetabling (PATAT). In: *Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling*, August 19–22, Montreal, Canada.

Burke, E.K., Marecek, J., Parkes, A., Rudova, H. (2010). Decomposition, re-formulation, and diving in university course timetabling. *Computers and Operations Research*, 37(3), 582–597.

Daskalaki, S., Birbas, T., Housos, E. (2004). An integer programming formulation for a case study in university timetabling. *European Journal of Operational Research*, 153(1), 117–135.

de Werra, D., Asratian, A.S., Durand, S. (2002). Complexity of some special types of timetabling problems. *Journal of Scheduling* 5(2), 171–183.

Dimopoulou, P., Miliotis, P. (2001). Implementation of a university course and examination timetabling system. *European Journal of Operational Research*, 153(1), 202–213.

Eikelder, H.M. Willemen, R.J. (2001). Some complexity aspects of secondary school timetabling problems. *Computer Science Practice and Theory of Automated Timetabling III, Lecture Notes in Computer Science*, Vol. 2079, pp. 18–27.

Even, S., Itai, A., Shamir, A. (1976). On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing*, 5(4), 691–703.

Ismayilova, N.A., Sagir, M., Gasimov, R.N. (2007). A multiobjective faculty-course-time slot assignment problem with preferences. *Mathematical and Computer Modeling*, 46(7–8), 1017–1029.

Lewis, R. (2007). A survey of metaheuristic-based techniques for university timetabling problems. *OR Spectrum*, 30(1), 167–190.

McCollum, B. (2007). A perspective on bridging the gap between research and practice in university timetabling. In: Burke, E., Rudova, H. (Eds.), *Practice and Theory of Automated Timetabling VI*, *LNCS*, Vol. 3867. Springer-Verlag, Berlin, pp. 3–23.

McCollum, B., McMullan, P., Paechter, B., Lewis, R., Schaerf, A., Gaspero, L.D., Parkes, A.J., Qu, R., Burke, E. (2010). Setting the research agenda in automated timetabling: the second international timetabling competition. *INFORMS Journal on Computing*, 22(1), 120–130.

MirHassani, S.A. (2006). A computational approach to enhancing course timetabling with integer programming. *Applied Mathematics and Computation*, 175(1), 814–822.

Ozdemir, M.S., Gasimov, R.N. (2004). The analytic hierarchy process and multiobjective 0–1 faculty course assignment problem. *European Journal of Operational Research*, 157(2), 398–408.

Papoutsis, K., Valouxis, C., Housos, E. (2003). A column generation approach for the timetabling problem of Greek high schools. *Journal of the Operational Research Society*, 54(3), 230–238.

Petrovic S., Burke, E.K. (2004). *University Timetabling, Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. CRC Press, Boca Raton (Chapter 45).

Qu R., Burke, E.K., McCollum, B., Merlok, L.T., Lee, S.Y. (2009). A survey of search methodologies and automated approaches for examination timetabling. *Journal of Scheduling*, 12(1), 55–89.

Sandhu, K.S. (2001). *Automating class schedule generation in the context of a university timetabling information system*. PhD dissertation, School of Management, Griffith University, Nathan Campus, Queensland, Australia.

Santos, H. G., Uchoa, E., Ochi, L., Maculan, N. (2008). Strong bounds with cut and column generation for class-teacher timetabling. In: *Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling, PATAT*.

Schaerf, A. (1999). A survey of automated timetabling. *Artificial Intelligence Review*, 13(2), 87–127.

Tripathy, A. (1984). School timetabling – a case in large binary integer linear pro-gramming. *Management Science*, 30(12), 1473–1489.

Valouxis, C. Houso, E. (2003). Constraint programming approach for school timetabling. *Computers and Operations Research*, 30(10), 1555–1572.

Yuqiang, W. (2007). *Models and algorithms for some combinatorial optimization problems: university course timetabling, facility layout and integrated production-distribution scheduling*. PhD Dissertation, Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA.

S.M. Al-Yakoob is an associate professor at the Department of Mathematics at Kuwait University. His research interests include mathematical programming and optimization with applications to real world problems such as location, transportation, scheduling, and timetabling problems.

H.D. Sherali is a university distinguished professor and the W. Thomas Rice endowed chaired professor of engineering, Emeritus in the Industrial and Systems Engineering Department at Virginia Polytechnic Institute and State University. His areas of research interest are in analysing problems and designing algorithms for specially structured linear, nonlinear, and integer programs arising in various applications, global optimization methods for non-convex programming problems, location and transportation theory and applications, economic and energy mathematical modelling and analysis. He has published over 345 refereed articles in various operations research journals, has (co-)authored eight books in this area, and serves on the editorial board of ten journals. He is an elected member of the National Academy of Engineering.