



Development of a linear programming model for the University Course Timetabling Problem

Desenvolvimento de um modelo de programação linear para o Problema da Construção de Grades Horárias em Universidades

Guilherme Brandelli Bucco¹
Camilo José Bornia-Poulsen¹
Denise Lindstrom Bandeira¹

Abstract: Creating timetables for courses is a problem that universities face at the beginning of every semester. This activity represents an important administrative task because it consumes significant amount of resources. In terms of computational complexity, this is classified as NP-hard, as it demands a huge amount of processing capacity. Timetabling is modeled in a number of different ways, aiming to fit the country's educational context, meet specific rules of institutions of higher education or specific goals of managers, among others. In this paper, we propose a mathematical model to solve the University Course Timetabling Problem and optimize classroom utilization. To solve the proposed model, an algorithm that divides the problem was developed, solving it with mixed integer linear programming tools. Computational experiments applied to a real database of a Brazilian public university confirmed the good performance of the proposed approach, which greatly reduces the amount of assigned classrooms.

Keywords: University Timetabling Problem; Integer programming; Mathematical programming; Decomposition; Timetable.

Resumo: A construção de grades horárias dos cursos de uma universidade é um problema que deve ser enfrentado no início de todos os semestres e, por mobilizar quantidades significativas de recursos, se constitui numa importante tarefa administrativa. É classificado, em termos de complexidade computacional, como NP-hard, o que implica grande exigência de capacidade de processamento. É modelado de maneiras muito diversas, no intuito de se obter adequação quanto ao contexto educacional do país, às regras específicas da instituição ou aos objetivos específicos dos gestores, entre outros. Neste artigo, propõe-se um modelo matemático para construir grades de horários, otimizando a utilização de salas de aula. Para resolver o modelo proposto, desenvolveu-se um algoritmo que divide o problema para viabilizar o uso de programação linear inteira mista. Experimentos computacionais aplicados a uma base de dados real de uma universidade pública brasileira confirmaram o bom desempenho da abordagem proposta, reduzindo consideravelmente a quantidade de salas de aulas alocadas.

Palavras-chave: University Timetabling Problem; Programação inteira; Programação matemática; Decomposição; Grades horárias.

1 Introduction

Creating timetables for their courses is a problem that universities must face at the beginning of every semester and, as it consumes significant amounts of resources, it represents an important administrative task. This problem consists in assigning certain resources, in a manner that some material and pedagogical constraints are respected. The most common constraints are physical (classrooms, laboratories, auditoriums, gymnasiums), people related (teachers, students), intangibles (days, periods, semesters, courses, lectures), among others.

Real cases could contain hundreds, or thousands, of these types of resources. Moreover, any feasible solutions must respect equally large amounts of constraints, and must fulfill, in the best way possible, certain demands (regarding teachers, educational institution, cost reductions, among others). Thus, building a good timetable is a very difficult task.

Known in the literature as University Course Timetable Problem (UCTP), according to Schaerf (1999), this problem concerns the scheduling of a course's lecture set in a limited quantity of classrooms

¹ Programa de Pós-graduação em Administração, Universidade Federal do Rio Grande do Sul – UFRGS, Rua Washington Luiz, 855, CEP 90010-460, Porto Alegre, RS, Brazil, e-mail: guilherme.bucco@ufrgs.br; camilo.bornia@ufrgs.br; dlbandeira@ea.ufrgs.br

Received Mar. 14, 2015 - Accepted June 30, 2015

Financial support: This research was partially supported by the Programa de Excelência Acadêmica (PROEX) of the Coordenação de Aperfeiçoamento de Pessoal de Ensino Superior (CAPES).

and periods. The author also notes that the main difference to the school timetabling problem is that, in universities, different courses can have students in common.

Solutions for this problem are usually obtained through the involvement of a lot of people for several days. Besides, some of those solutions could not be satisfactory in some aspects, for instance, a student who wishes to enroll in two courses that are scheduled in the same period (Schaerf, 1999).

Previous studies present a variety of models and approaches. However, even if there are common objectives, some concepts and definitions do not apply to all educational institutions and systems, which are closely related to the country on which they're applied.

Some of these models were implemented in real cases, such as: Virginia Tech (USA), by Sarin et al. (2010); Chile University, by Miranda (2010); Spanish University System, in particular University of Valencia, by Alvarez-Valdes et al. (2002); Purdue University (USA), by Rudová et al. (2011); Università degli studi di Udine (Italy), developed by Burke et al. (2010); Greek educational system, by Daskalaki et al. (2004); and Sydney University (Australia), by Beyrouthy et al. (2007).

There are also solutions applied to Brazilian instances, such as a heuristic formulation to the Exact and Geosciences Institute of Passo Fundo University, developed by Kripka et al. (2011), and a model, to the Production Engineering course of Paraná University, solved by mathematical programming, presented by Andrade et al. (2012).

Nevertheless, some authors claim that few papers applying models and methods using instances of real-world data (Carter & Laporte, 1998; McCollum, 2007; Murray et al., 2007) can be found. These authors note that many articles concentrate on developing solutions to simplified problems, or that they use artificial data sets, and rarely extend their solutions to large scale formulations.

A large number of techniques and approaches to the solution are in the literature (Gunawan et al., 2012). For the authors, exact algorithms can solve small data instances but hardly find the ideal solution for large data sets, due to its computational complexity. According to McCollum (2007), Operations Research techniques to model real data instances can support the decisions of Higher Education Institution managers.

Regarding teaching space use, Beyrouthy et al. (2007) split these studies in two types: "management" and "planning", engaged with short and long term, respectively. In this sense, both management and planning of teaching spaces can have important contributions from Operations Research. The model solution indicates the teaching space needs, supporting

decisions concerning construction, renting or selling such places, and better management of available ones, designating people and other resources to teaching spaces.

We consulted a manager of a large Brazilian public university concerning the use of teaching locations and rental costs. In 2012, this university had spent 1.6 million dollars to rent 29 spaces. According to him, these spaces were probably underutilized.

Thus, this study aimed to develop a decision support model to help university managers construct timetables, and, more specifically, optimize teaching space use, respecting the constraints established by the educational institution.

The paper is organized as follows. Section 2 briefly discusses a literature review on timetable construction problem. Section 3 describes the problem and its mathematical formulation in details. Section 4 provides computational results with the use of real-world data. Section 5 presents conclusions and perspectives for future research.

2 Timetable construction in universities

Burke et al. (2003) proposed a definition to the timetable construction problem. According to these authors, it is composed of four parameters: T , a finite set of time intervals (days, weeks); R , a finite set of resources (classrooms, teachers); M , a finite set of meetings (combination of time intervals and resources); and C , a finite set of constraints. The problem consists of assigning time intervals and resources to meetings, in such a way that constraints are satisfied in the best manner.

Timetabling problems have been studied since the 1950's (Tripathy, 1984), by utilizing a variety of methods, according to De Werra (1997), due to its combinatorial nature, this problem still draws attention, mainly from Operations Research community, because a great number of combinations must be checked in order to find at least one feasible solution.

School (STP) and University Courses (UCTP) Timetabling Problems have received lot of attention in the Educational Timetabling Problems (ETP). This division is due to important differences between the problems. Carter & Laporte (1998) note that in schools, classes are predefined, there are few courses, few options for students, and teachers have high workload, while in universities, students have more freedom of choice, and teachers have lighter workload. Besides, in UCTP, classes share students, something that does not occur in STP.

Carter & Laporte (1998) divide the problem in two types: master timetabling and demand-driven systems. In the first case, courses, teachers, classrooms, other

material, and pedagogical resources are combined, and, after that, students choose which courses to enroll, according to a previously built timetable. In the second, students choose which courses to enroll and then, teachers, classrooms, and courses are combined to construct the timetables. From this division, the institution can adopt the better strategy to create their timetables.

Generally, UCTP formulation involves a great number of constraints, and preferences of students, and teachers. There are two kinds of constraints: hard and soft. While the hard constraints must be met, the soft constraints are characterized by penalizing the objective function, and therefore only contribute to evaluate the solution's quality. Pongcharoen et al. (2008) classify the problem as NP-hard, which implies a great demand for computational resources. According to Cormen et al. (2002), problems of this class cannot be solved by any algorithm in polynomial time, and big instances demand approximate algorithms to solve them.

UCTP models have diverse objectives: maximization of student and teacher preferences, minimization of teachers and students idle time, better distribution of classrooms' use, among others. Besides that, Alvarez-Valdes et al. (2002) point out that timetable structure and specificities vary according to countries, mainly due to educational systems and university particularities, which explain why the problem is studied so much. This article models the problem considering physical space optimization, particularly classrooms. Data was collected from a Brazilian federal university, which initially showed a potential for better usage of space and, saving money on rents, cleaning, maintenance, and security.

3 Mathematical modeling

In this section, we describe the problem, entity sets, parameters, mathematical formulation, and the approach used for searching solutions.

3.1 Problem details

According to the discussion in Section 2, our problem is classified as master timetabling. Therefore, the model does not have student sets. We assumed that each class already has a teacher assigned to it, and so, the model considers the set of teachers only to avoid time conflict between classes with the same teacher. Regarding the physical spaces, the model considers only the classrooms, excluding special locations such as laboratories, gymnasiums, auditoriums, among others. This decision was based on the university's case studied in this research. They needed to rent classrooms to meet their demand for space, but there's no need for special locations.

In addition, the classrooms are managed centrally, while the course coordinators attribute the special locations to events.

To model this problem, it is important to identify the hard and soft constraints. Hard constraints are those that must be satisfied, so that any violation implies in infeasible solutions. Soft constraints are requirements that are not essential to produce feasible solutions, but they are desirable (Kahar & Kendall, 2010). Therefore, we present the hard (HC) and soft (SC) constraints:

- HC01 Exact number of meetings in each subject must be assigned
- HC02 Classes can only be assigned to periods allowed by the educational institution
- HC03 Minimum and maximum daily meetings in each subject must be respected
- HC04 Classes cannot have meetings in consecutive days
- HC05 Meetings of the same class, in the same day, must be consecutive
- HC06 Mandatory subjects of the same semester of curriculum cannot have schedule conflicts, if they have only a single class offered
- HC07 Teachers can be assigned to meetings in periods designed by the educational institution
- HC08 Teachers cannot teach more than one meeting at the same time period
- HC09 Classrooms can host at most one event in a given time period
- HC10 Classrooms can host only events permitted by the educational institution
- HC11 Classroom capacities must be respected
- HC12 Class meetings in the same day must be assigned to the same classroom
- HC13 Teachers can only teach in one academic unit (e.g. building), in a given shift
- SC01 Minimize academic unit utilization costs

3.2 Mathematical formulation

Initial tests show the impossibility of optimally solving the real data sets. Therefore, we propose a viable approach to solve it. We divide the main problem in two subproblems:

- (1) Creation of weekly timetables, aiming to uniformly distribute meetings among all time periods, and
- (2) Assignment of classrooms to events, aiming to minimize the SC01 soft constraint.

The aim of this strategy is to separate the resources that will be assigned to events: days and periods in the first subproblem, and classrooms on the second subproblem. Furthermore, we identified an important characteristic: in the previous academic school years, all the meetings were assigned in the same shift for a given class, which allowed us to solve the problem in each shift separately. From these two division strategies (of the problem, and of data sets), it was possible to find solutions.

We present the parameters and data sets for the first subproblem. Data sets represent modeled entities, and parameters represent the conditions which must be respected.

Sets:

C : courses, $c \in C = \{0, 1, \dots, |C| - 1\}$

D : days, $d \in D = \{0, 1, 2, 3, 4\}$

E : events, $e \in E = \{0, 1, \dots, |E| - 1\}$

S : subjects, $s \in S = \{0, 1, \dots, |S| - 1\}$

T : teachers, $t \in T = \{0, 1, \dots, |T| - 1\}$

E_s : subject's events $s \in S$, $E_s \subseteq E$

E_t : teacher's events $t \in T$, $E_t \subseteq E$

G : semesters, $g \in G = \{0, 1, \dots, |G| - 1\}$

P : time periods, $p \in P = \{0, 1, \dots, |P| - 1\}$

S_{cg} : mandatory subjects that have only one event,

$S_{cg} \subseteq S$, $\forall c \in C, g \in G$

Parameters:

$M_{|s|} \in \mathbb{N}$: vector indicating the number of weekly theoretical meetings of a subject, as required by the curriculum, where M_s indicates the number of weekly theoretical meetings of subject $s \in S$

$DMmin_{|s|} \in \mathbb{N}$ vector indicating the minimum number of daily meetings acceptable for subject $s \in S$, where $DMmin_s$ indicates the minimum number of daily meetings of subject $s \in S$

$DMmax_{|s|} \in \mathbb{N}$: vector indicating maximum number of daily meetings acceptable for subject $s \in S$, where $DMmax_s$ indicates the maximum number of daily meetings of subject $s \in S$

$EA_{|E| \times |D| \times |P|} \in \{0, 1\}$: event timetable availability matrix, where $EA_{edp} = 1$, if event $e \in E$ might have a meeting at day $d \in D$ and time period $p \in P$; $EA_{edp} = 0$, otherwise

$TA_{|T| \times |D| \times |P|} \in \{0, 1\}$: teacher timetable availability matrix, where $TA_{tdp} = 1$ if teacher $t \in T$ might work at day $d \in D$ and time period $p \in P$; $TA_{tdp} = 0$, otherwise

Our STP model is a mixed linear programming problem, with $(0, 1)$ decision variables mapping events of students, teachers, and locations into a matrix of time slots within the week. Thus, we define the decision variables as:

$$x_{edp} = \begin{cases} 1, & \text{if event } e \in E \text{ is assigned to day } d \in D \\ & \text{and time period } p \in P \end{cases}$$

otherwise

Constraints demand the following auxiliary variables:

$$b_{ed} = \begin{cases} 1, & \text{if event } e \in E \text{ have a meeting on day } d \in D \end{cases}$$

otherwise

$e_{ed} \in \mathbb{N}$ = first time period assigned to event $e \in E$ on day $d \in D$

$f_{ed} \in \mathbb{N}$ = last time period assigned to event $e \in E$ on day $d \in D$

$w_{dp} \in \mathbb{N}$ = number of meetings scheduled on day $d \in D$ and in period $p \in P$

$y \in \mathbb{N}$ = value of the biggest element in variable set w_{dp}

To uniformly distribute the meetings in the timetable, it is necessary and sufficient to compute the time slot with the highest amount of meetings. The objective of the model is to minimize the number of assigned meetings to the slot with the highest amount of meetings:

$$\text{Minimize } y \quad (1)$$

In order to follow hard constraints HC01 to HC08, and to compute y , which is used in soft constraint SC01, the Objective Function 1 is subject to the following equations:

$$\sum_{d \in D} \sum_{p \in P} x_{edp} = M_s \quad \forall e \in E, s \in S \quad (2)$$

$$x_{edp} \leq EA_{edp} \quad \forall e \in E, d \in D, p \in P \quad (3)$$

$$\sum_{e \in E_i} x_{edp} \leq TA_{tdp} \quad \forall t \in T, d \in D, p \in P \quad (4)$$

$$DMmin_s b_{ed} \leq \sum_{p \in P} x_{edp} \leq DMmax_s b_{ed} \quad \forall e \in E, s \in S, d \in D \quad (5)$$

$$b_{ed} \geq x_{edp} \quad \forall e \in E, d \in D, p \in P \quad (6)$$

$$b_{ed} + b_{e,d+1} \leq 1 \quad \forall e \in E, d \in \{0, 1, \dots, |D|-2\} \quad (7)$$

$$e_{ed} \leq (|P|+1) - (|P|+1-p)x_{edp} \quad \forall e \in E, d \in D, p \in P \quad (8)$$

$$f_{ed} \geq p x_{edp} \quad \forall e \in E, d \in D, p \in P \quad (9)$$

$$f_{ed} - e_{ed} + b_{ed} - \sum_{p \in P} x_{edp} \leq 0 \quad \forall e \in E, d \in D \quad (10)$$

$$\sum_{e \in E_s, s \in S_g} x_{edp} \leq 1 \quad \forall c \in C, g \in G, d \in D, p \in P \quad (11)$$

$$w_{dp} = \sum_{e \in E} x_{edp} \quad \forall d \in D, p \in P \quad (12)$$

$$y \geq \sum_{e \in E} x_{edp} \quad \forall d \in D, p \in P \quad (13)$$

Constraint Set 2 satisfies HC01, ensuring that the required number of meetings is assigned. Equations 3 and 4 ensure the allocation of the event in allowed timetables for their own event (HC02), and their teacher (HC07). Equation 4 also satisfies HC08 constraint, which prevents allocating a teacher more than once on the same time slot (d, p). The model respects the minimum and maximum daily limit of meetings of one subject (HC03) through the Constraint 5, which uses the auxiliary variable b_{ed} calculated by Equation 6 through an implication constraint. Equation 7 ensures that there will be no assignments in consecutive days (HC04), an important pedagogical constraint. Equations 8 and 9 compute the first and last meeting of a given event per day, respectively, as proposed by Santos et al. (2012), and adapted by Dorneles et al. (2012). Event meetings in a given day must be consecutive (HC05), according to Equation 10, which prevents idle times between meetings of an event. Equation 11 models the constraint HC06, which demands that subjects of a given semester of curriculum do not overlap, as long as they are mandatory and have only one event. This constraint is essential to not impair the students who wish to enroll in all subjects of the current semester of their curriculum. Finally, Equation 12 computes the value of the auxiliary variable w_{dp} , which contains the number of classes of each time slot (d, p). Equation 13 calculates the variable y , which equals the highest amount of auxiliary variables w_{dp} .

The second subproblem aims to assign classrooms to each meeting. It needs the definition of the following sets and parameters:

Sets:

U : academic units, $u \in U = \{0, 1, \dots, |U|-1\}$

R : classrooms, $r \in R = \{0, 1, \dots, |R|-1\}$

R_u : classrooms from academic unit $u \in U$, $R_u \subseteq U$

H : shifts, $h \in H$, $H = \{0, 1, 2\}$

Parameters:

$X_{|E| \times |D| \times |P|} \in \{0, 1\}$: event timetable matrix generated by first subproblem, where $X_{edp} = 1$ if event $e \in E$ is assigned to day $d \in D$ and time period $p \in P$; and $X_{edp} = 0$, otherwise

$Cap_{|R|} \in \mathbb{N}$: classroom capacity vector, where Cap_r indicates the capacity of students of a classroom $r \in R$,

$NS_{|E|} \in \mathbb{N}$: event enrollment vector, where NS_e indicates the forecast of students enrolled in event $e \in E$

$RA_{|R| \times |D| \times |P|} \in \{0, 1\}$ classroom availability matrix, where $RA_{rdp} = 1$ if classroom $r \in R$ might be assigned to events at day $d \in D$ and time period $p \in P$; $RA_{rdp} = 0$, otherwise

$\alpha_{|U|} \in \mathbb{N}$: utilization costs vector for academic unit $u \in U$

Second subproblem model must result in the decision variable values:

$$k_{redp} = \begin{cases} 1, & \text{if classroom } r \in R \text{ is assigned to event} \\ & e \in E \text{ at day } d \in D \text{ and in period } p \in P \\ 0, & \end{cases}$$

otherwise

$$m_u = \begin{cases} 1, & \text{if academic unit } u \in U \text{ is assigned to at} \\ & \text{least one meeting} \\ 0, & \end{cases}$$

otherwise

Model constraints need the following auxiliary variable sets:

$$j_{red} = \begin{cases} 1, & \text{if classroom } r \in R \text{ is assigned to event} \\ & e \in E \text{ at day } d \in D \\ 0, & \end{cases}$$

otherwise

$$q_{tudh} = \begin{cases} 1, & \text{if teacher } t \in T \text{ is assigned to teach in the} \\ & \text{academic unit } u \in U \text{ at day } d \in D \text{ and} \\ & \text{on shift } h \in H \\ 0, & \end{cases}$$

otherwise

$$v_{eu} = \begin{cases} 1, & \text{if event } e \in E \text{ has at least one meeting at} \\ & \text{academic unit } u \in U \\ 0, & \end{cases}$$

otherwise

The second subproblem aims to assign classrooms to events, according to SC01 constraint, in order to

reduce academic unit use. Equation 14 presents its objective function. Each academic unit has a penalty associated to its use, as expressed in $\alpha_{|U|}$ vector.

$$\text{Minimize } \sum_{u \in U} \alpha_u \times m_u \quad (14)$$

Hard constraints HC09 to HC13 and soft constraint SC01 are expressed as follows:

$$\sum_{r \in R} k_{redp} = X_{edp} \quad \forall e \in E, d \in D, p \in P \quad (15)$$

$$\sum_{e \in E} k_{redp} \leq RA_{rdp} \quad \forall r \in R, d \in D, p \in P \quad (16)$$

$$j_{red} \geq k_{redp} \quad \forall r \in R, e \in E, d \in D, p \in P \quad (17)$$

$$NS_e \cdot j_{red} \leq Cap_r \quad \forall e \in E, r \in R, d \in D \quad (18)$$

$$\sum_{r \in R} j_{red} \leq 1 \quad \forall e \in E, d \in D \quad (19)$$

$$q_{tadh} \geq \sum_{r \in R_u} j_{red} \quad \forall e \in E, t \in T, u \in U, d \in D, h \in H \quad (20)$$

$$\sum_{u \in U} q_{tadh} \leq 1 \quad \forall t \in T, d \in D, h \in H \quad (21)$$

$$v_{eu} \geq \sum_{r \in R} j_{red} \quad \forall e \in E, u \in U, d \in D \quad (22)$$

$$m_u \geq v_{eu} \quad \forall u \in U, e \in E \quad (23)$$

Equation 15 loads the first subproblem results to the second one, because its set of parameters expresses the decision variables of the first, guaranteeing their relationship. Equation 16 assures that a classroom receives only one meeting per slot (HC09), and also complies with the availability of classrooms (HC10). The implication Constraint 17 computes auxiliary variable j_{red} , fundamental to compute other restrictions. Equations 18 and 19 ensure compliance with the constraints HC11 (classroom capacity) and HC12 (events with meetings in the same classroom on the day), respectively. Equation 20 forces the variable q_{tadh} to take value '1' if an event from teacher $t \in T$ is scheduled in day $d \in D$, and in the shift $h \in H$. The Equation 21 forces teachers to teach in one academic unit $u \in U$ on the same day

$d \in D$ and shift $h \in H$ (HC13). It uses the variable q_{tadh} calculated from the implication Equation 20. Equation 22 calculates the auxiliary variable v_{eu} used to compute variable m_u in the Equation 23. Variable m_u indicates if an academic unit $u \in U$ has at least one classroom with a meeting assigned, thus allowing the evaluation of constraint SC01 on the Objective Function 14.

4 Computational results

The proposed model was implemented in C++ language using Microsoft Visual Studio 2010 Professional IDE, and solved using IBM ILOG CPLEX Optimization Studio v12.5.1 with default configurations. Experiments were executed using an Intel Core i7-3520M CPU 2.9 GHz, 6GB RAM and Windows 8.1 Pro 64 bits operational system.

Data collected from the first semester of 2013 of a Brazilian federal university were used to test the model, and the timetables used in 2013 were compared with those produced by the model. The university has 120 curricula, 1,688 subjects, 2,674 events, 1,083 teachers, 258 classrooms, and 48 academic units. Meetings were offered in five time periods in the morning from Monday to Saturday, six in the afternoon and four in the evening from Monday to Friday. In order to test the model's robustness, time periods on Saturday were ignored. Therefore, timetables were generated using five days, with five periods in the mornings, six in the afternoons, and four in the evenings. Table 1 summarizes sets and variable sizes from the tested instance.

4.1 First subproblem results

The first subproblem model aimed to uniformly distribute meetings across the timetable, so that we could minimize classrooms demand and, consequently, academic units. The timetable used in the first semester of 2013 had an unbalanced meeting distribution, as shown in Figure 1a. The greater meeting concentration occurred between Tuesday and Thursday mornings, and in the early afternoon periods of Monday to Thursday. Tuesday's second morning period was the one with the largest number of meetings: 234. Therefore, the university demanded

Table 1. Set and variable sizes.

Shift	D	P	C	S	E	T	R	U	x_{edp}	k_{redp}	m_u
Morning	5	5	120	686	1,003	565	258	48	25,075	6,469,350	48
Afternoon	5	6	120	746	1,011	616	258	48	30,330	7,825,140	48
Evening	5	4	120	538	654	392	258	48	13,080	3,374,640	48
Total	5	15	120	1,688	2,688	1,083	258	48	201,600	52,012,800	48

the same amount of classrooms, in order to assign all meetings. Figure 1b presents results obtained from the proposed model, which shows a more balanced distribution of meetings along the timetable.

The model was executed for each of the three shifts. Table 2 shows running time and objective function value for those shifts. The object function value indicates the greatest meeting concentration of all the time slots of shift.

We need to partially relax the HC03 constraint for the evening shift's processing ($h = 2$), eliminating the minimum number of daily meetings. Despite the characterization as hard, this constraint is, practically speaking, only desirable. As most of the events demand a minimum of two daily meetings, and considering that evening shifts have only four periods, it is very hard to comply with this constraint, and also it does not allow overlaps

Table 2. First subproblem results.

Shifts	Running time (minutes)	Objective function value (Z_1)
Morning ($h = 0$)	4.68	101
Afternoon ($h = 1$)	16.14	88
Evening ($h = 2$)	3.31	95

Table 3. Second subproblem results.

Shifts	Running time (minutes)	Objective function value (Z_2)	Assigned units
Morning ($h = 0$)	2.13	730	0,1,2,3,4
Afternoon ($h = 1$)	3.82	490	0,1,2,3
Evening ($h = 2$)	1.27	536	0,1,2,4

between mandatory single event subjects of the same semester of curriculum. Therefore, processing time for this particular instance was short.

These results indicate that the university could use 101 classrooms in the worst case. It represents a 57% reduction in relation to the timetables used in the first semester of 2013 (234 classrooms in the most concentrated time slot).

4.2 Second subproblem results

This second phase was responsible for assigning rooms to each scheduled meeting generated by subproblem 1. Similarly, the second subproblem was executed in each of the three shifts. Table 3 shows running time, objective function values, and assigned academic units for each shift. Table 4 shows the available academic units, amount of classrooms, and related costs. These costs were assigned in decreasing order of preference, so that lower cost units were selected first. The academic unit costs have been assigned so that a combination of units was not equal to any other. Thereby, the possibility that the model produces different academic unit sets for the different shifts is reduced. In the first semester of 2013 this university rented buildings to use for teaching. So, the use of its own academic units was preferred in relation to rented ones that would only be used if necessary.

Timetables produced by the second subproblem resulted in five assigned academic units that, together, have 107 classrooms (sufficient amount to accommodate the 101 simultaneous meetings in the most concentrated time slot). Table 4 shows those academic units marked with an asterisk (*), together they have a combined cost of 730.

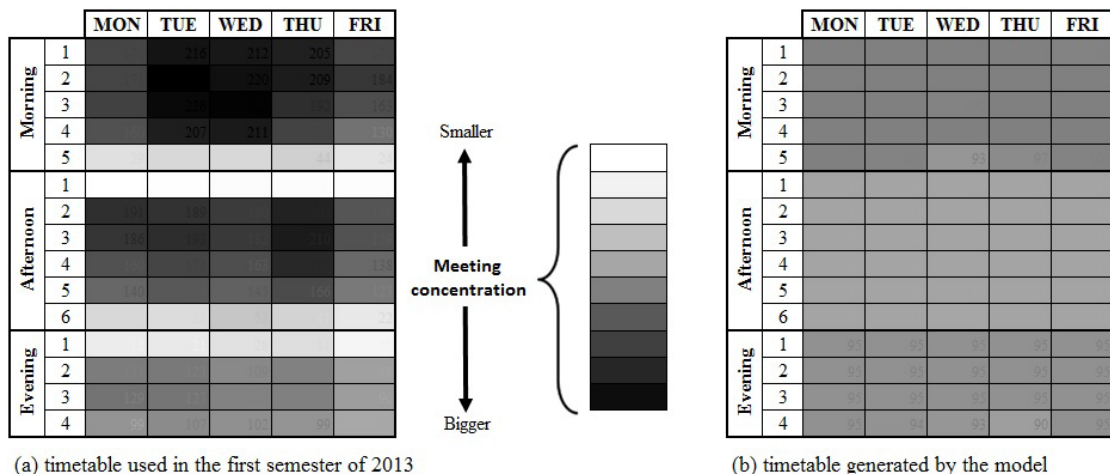


Figure 1. Meeting distribution: (a) used by the educational institution, and (b) generated by the model.

Table 4. Academic unit data.

Unit	Cost	# of classrooms	Unit	Cost	# of classrooms	Unit	Cost	# of classrooms
*0	50	43	16	714	4	32	5,490	8
*1	99	24	17	747	3	33	5,585	6
*2	147	11	18	779	3	34	5,679	5
*3	194	11	19	810	3	35	5,772	5
*4	240	18	20	840	3	36	5,864	3
5	285	12	21	869	2	37	5,955	2
6	329	8	22	897	2	38	6,045	2
7	372	7	23	924	2	39	6,134	1
8	414	7	24	950	2	40	6,222	1
9	455	6	25	975	2	41	6,309	1
10	495	6	26	999	2	42	6,395	1
11	534	5	27	5,000	2	43	6,480	1
12	572	4	28	5,100	1	44	6,564	1
13	609	4	29	5,199	1	45	6,647	1
14	645	4	30	5,297	1	46	6,729	1
15	680	4	31	5,394	11	47	6,810	1

*Necessary and sufficient units to allocate the meetings, according to the model.

5 Conclusions

This research proposed a model for the University Course Timetabling Problem (UCTP), aiming to support managers in creating timetables, and to reduce costs associated with physical space utilization. The model uses linear programming techniques.

Given the size of the problem, and consequently the impossibility to solve it in reasonable computational time with an optimization package, it was divided in two subproblems. In the first, weekly timetables were built disregarding classrooms, aiming to uniformly distribute meetings among the entire timetable. In the second, classrooms were assigned to the meetings of every event, aiming to reduce costs associated with academic units. Additionally, we processed each subproblem separately for the three shifts.

This work shows that it is possible to create feasible timetables that comply with pedagogical requirements, and reduce the classroom demands, decreasing costs. Besides creating feasible timetables, these results can support planning actions for utilizing available space in a university to help better respond to future demands that the educational institution may require.

We noticed that a more balanced distribution of meetings along the weekly timetable leads to classroom demand reduction. In other words, less concentrated time slots contribute to lower classroom demand. Therefore, we reached the main objective of this research, which is to create feasible timetables, and

optimize the use of physical spaces. This reduction in classroom demand allowed the managers to avoid rented buildings.

Another important contribution of this study is developing a division strategy for the problem in two phases: timetable construction in the first, and classroom assignment in the second. As shown, timetabling problems have large amounts of variables and constraints, which, depending on the data instance size, make the solution searches rather difficult. Thereby, resource assignment in separate phases – time in the first, and space in the second – is a promising strategy to solve big data instances to be used in other methods.

It was not possible to obtain information on the teachers schedule restrictions, so this study assumed that each teacher was available on all periods of the shift for all their events. Similarly we could not obtain information referring to the usage of physical resources for postgraduation courses which obviously require classrooms for their activities. Thus one could suppose that if the model was executed with all the data it would take longer processing time, and the final results would demand a higher amount of classrooms and units. However, taking into account the efficiency of classroom distribution was 57%, there is a margin so large that, even if these constraints were considered, the gains would remain high.

For future research, we consider the possibility of including the preferences of both teachers and course managers for units, classrooms, weekdays, and

periods in the model. We would also suggest using heuristics methods for resolution of the full model (before subproblem division), and of the complete instance (before shift division), in a manner that better solutions could be obtained for the instance presented in this paper as well as for other possibly bigger instances.

References

- Alvarez-Valdes, R., Crespo, E., & Tamarit, J. M. (2002). Design and implementation of a course scheduling system using tabu search. *European Journal of Operational Research*, 137(3), 512-523. [http://dx.doi.org/10.1016/S0377-2217\(01\)00091-1](http://dx.doi.org/10.1016/S0377-2217(01)00091-1).
- Andrade, P. R. L., Scarpin, C. T., & Steiner, M. T. A. (2012). Geração da grade horária do curso de engenharia de produção da UFPR através de programação linear binária. In *Anais do 44º Simpósio Brasileiro de Pesquisa Operacional* (pp. 1052-1063). Rio de Janeiro: SOBRAPO.
- Beyrouthy, C., Burke, E. K., Landa-Silva, D., McCollum, B., McMullan, P., & Parkes, A. J. (2007). Towards improving the utilization of university teaching space. *The Journal of the Operational Research Society*, 60(1), 1-14.
- Burke, E. K., Mareček, J., Parkes, A. J., & Rudová, H. (2010). Decomposition, reformulation, and diving in university course timetabling. *Computers & Operations Research*, 37(3), 582-597. <http://dx.doi.org/10.1016/j.cor.2009.02.023>.
- Burke, E., Werra, D., & Kingston, J. (2003). Applications to timetabling. In J. L. Gross & J. Yellen. *Graph Theory* (chap. 5, pp. 445-474). Boca Raton: CRC Press, 2003.
- Carter, M. W., & Laporte, G. (1998). Recent developments in practical course timetabling. In E. Burke & M. Carter. *Practice and theory of automated timetabling* (Vol. 1408, pp. 3-19). Toronto: Springer-Verlag Berlin.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2002). *Algoritmos: Teoria e prática* (2. Ed.). Rio de Janeiro: Elsevier.
- 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. [http://dx.doi.org/10.1016/S0377-2217\(03\)00103-6](http://dx.doi.org/10.1016/S0377-2217(03)00103-6).
- Dorneles, A. P., Araújo, O. C. B., & Buriol, L. S. (2012). The impact of compactness requirements on the resolution of high school timetabling problem. In *Simpósio Brasileiro de Pesquisa Operacional* (pp. 3336-3347). Rio de Janeiro: SOBRAPO.
- Gunawan, A., Ng, K. M., & Poh, K. L. (2012). A hybridized lagrangian relaxation and simulated annealing method for the course timetabling problem. *Computers & Operations Research*, 39(12), 3074-3088. <http://dx.doi.org/10.1016/j.cor.2012.03.011>.
- Kahar, M. N. M., & Kendall, G. (2010). The examination timetabling problem at Universiti Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution. *European Journal of Operational Research*, 207(2), 557-565. <http://dx.doi.org/10.1016/j.ejor.2010.04.011>.
- Kripka, R. M. L., Kripka, M., & Silva, M. C. (2011). Formulação para o problema de alocação de salas de aula com minimização de deslocamentos. In *Anais do 43º Simpósio Brasileiro de Pesquisa Operacional* (pp. 1941-1951). Ubatuba: SOBRAPO.
- Mccollum, B. (2007). A perspective on bridging the gap between theory and practice in university timetabling. In: E. K. Burke & H. Rudová. In *Practice and Theory of automated timetabling VI* (Vol. 3867, pp. 3-23). Berlin: Springer-Verlag.
- Miranda, J. (2010). eClasSkeduler: a course scheduling system for the executive education unit at the Universidad de Chile. *Interfaces*, 40(3), 196-207. <http://dx.doi.org/10.1287/inte.1090.0485>.
- Murray, K., Müller, T., & Rudová, H. (2007). Modelling and solution of a complex university course timetabling problem. In E. K. Burke & H. Rudová. *Lecture notes in computer science* (Vol. 3867, pp. 189-209). Berlin: Springer-Verlag.
- Pongcharoen, P., Promtet, W., Yenradee, P., & Hicks, C. (2008). Stochastic optimisation tool for university course scheduling. *International Journal of Production Economics*, 112(2), 903-918. <http://dx.doi.org/10.1016/j.ijpe.2007.07.009>.
- Rudová, H., Müller, T., & Murray, K. (2011). Complex university course timetabling. *Journal of Scheduling*, 14(2), 187-207. <http://dx.doi.org/10.1007/s10951-010-0171-3>.
- Santos, H. G., Uchoa, E., Ochi, L. S., & Maculan, N. (2012). Strong bounds with cut and column generation for class-teacher timetabling. *Annals of Operations Research*, 194(1), 399-412. <http://dx.doi.org/10.1007/s10479-010-0709-y>.
- Sarin, S. C., Wang, Y., & Varadarajan, A. (2010). A university-timetabling problem and its solution using Benders' partitioning: a case study. *Journal of Scheduling*, 13(2), 131-141. <http://dx.doi.org/10.1007/s10951-009-0157-1>.
- Schaerf, A. (1999). A survey of automated timetabling. *Artificial Intelligence Review*, 13(2), 87-127. <http://dx.doi.org/10.1023/A:1006576209967>.

- Tripathy, A. (1984). School timetabling: a case in large binary integer linear programming. *Management Science*, 30(12), 1473-1489. <http://dx.doi.org/10.1287/mnsc.30.12.1473>.
- Werra, D. (1997). The combinatorics of timetabling. *European Journal of Operational Research*, 96(3), 504-513. [http://dx.doi.org/10.1016/S0377-2217\(96\)00111-7](http://dx.doi.org/10.1016/S0377-2217(96)00111-7).