SN Operations Research Forum (2020) 1: 11
https://doi.org/10.1007/s43069-020-0013-x

CASE REPORT

Curriculum-Based Course Timetabling with Student Flow, Soft Constraints, and Smoothing Objectives: an Application to a Real Case Study

Massimiliano Caramia[1] · Stefano Giordani[1]

Received: 13 September 2019 / Accepted: 1 April 2020 / Published online: 29 April 2020

Abstract
This paper deals with curriculum-based course timetabling. In particular, we describe the results of a real application at the University of Rome "Tor Vergata." In this regard, we developed a multi-objective mixed-integer model which attempts to optimize (i) the flow produced by the students enrolled in the lectures, (ii) soft conflicts produced by the possible overlap among compulsory and non-compulsory courses, and (iii) the number of lecture hours per curriculum within the weekdays. The model has been implemented and solved by means of a commercial solver and experiments show that the model is able to provide satisfactory solutions as compared with the real scenario under consideration.

Keywords Timetabling · MIP model · Multi-objective optimization · Case study

1 Introduction

The course timetabling problem (CTP) refers to the allocation of courses to a given set of resources being placed in space and in time, so as to satisfy a set of objectives as much as possible [34]. Course timetabling specifically involves assigning courses to classrooms, as well as teachers and time slots. In general, classrooms are always scarce and courses are affected by incompatibility relations based on the curricula they refer to, which complicate the assignment.

Course timetabling has many variants. We can start citing, among the most important ones, the school timetabling problem (STP), also known as class/teacher model.

✉ Massimiliano Caramia
caramia@dii.uniroma2.it

Stefano Giordani
stefano.giordani@uniroma2.it

[1] Dipartimento di Ingegneria dell'Impresa, Università di Roma "Tor Vergata", Via del Politecnico, 1 - 00133, Rome, Italy

Given a set of classes, a set of teachers, a number of periods, and a non-negative integer matrix (called requirements matrix) whose generic entry is the number of lectures (lessons) given by a teacher in a class, the problem consists in assigning lectures to periods in such a way that no teacher or class is involved in more than one lecture at a time. A pioneer in this field was de Werra [13] who proposed a basic reduced formulation for this problem which was not real but its minimal set of constraints made it a hard problem. The problem was solved by reduction to edge coloring. Successively, Junginger [21] introduced additional constraints defined by the unavailabilities of teachers and classes. These formulations were given as feasibility problems, even though Junginger [21] proposed also to add an objective based on the desirability of an assignment teacher/course. Colorni et al. [12] proposed objectives based on, e.g., spreading the lessons over the whole week or imposing a day-off for each teacher. Beyond several heuristic algorithms, there have been also proposals based on constraints programming (see, e.g., [9]).

Similar to STP but with some specific differences is the University Course Timetabling (UCT) problem which is the problem considered in this paper. It consists in scheduling a set of lectures for each course within a given number of rooms and time periods. The main difference from STP is that university courses can have common students, whereas school classes are disjoint sets of students. If two courses have common students then they conflict, and they cannot be scheduled at the same period. In addition, in UCT, availability of rooms (and their size) plays an important role, whereas in the school problem they are often neglected because, in most cases, we can assume that each class has its own room [29]. The basic problem firstly studied by, e.g., [33] was the following: given are a set of courses each one consisting of a set of lectures. There is a set of curricula which are groups of courses that have common students and such that all the courses in a curriculum must be scheduled at different times. There is a number of periods and a maximum number of lectures that can be scheduled in a given period due to room availability. The basic problem is to find a feasible solution for this problem. Objective functions associated with this set of constraints in the literature typically comprise desirability of the assignments, and the minimization of soft and hard constraints associated with the conflicts among courses (see, e.g., [28]).

In all institutions, UCT has to be solved at least once each academic year, and the administrative personnel spend much time and effort in solving this problem. This is why, in the literature, this problem has been largely studied and has attracted the attention of many researchers. Consequently, there have been several methods and algorithmic approaches to this problem. In this paper, we will concentrate our attention on that part of the literature in which exact methods are proposed based on formulating and solving mathematical programming models. Even if for large problems, it could be difficult to find an optimal solution within reduced computing times since the problem is NP-Hard [11], exact approaches in the literature have been proved to be effective to some extent and have provided optimal solutions to some specific cases. One of the applications of exact methods includes [18] in which it is shown that timetabling problems with data sizes comparable with that of some scenarios have been solved by means of a basic model. Tripathy [32] presented an integer linear program to determine room and time slot allocations for graduate courses by means of a Lagrangian relaxation procedure gathered with subgradient optimization. A two-stage binary multi-objective course scheduling model is proposed in [2]. Birbas et al. [6] formulated a binary integer program to determine the weekly timetable for Greek high schools. Dimopoulou and Miliotis [15] and Papoutsis et al. [27] employ a column-generation approach to solve a timetabling problem for Greek schools. Mushi [26] shows the existence of various models for timetabling problem by using integer programming and experimented their proposal on a case study related to the University of Dar es Salaam. Al-Yakoob and Sherali [1] adopted a mixed-integer programming approach experimented on the Kuwait University. Hinkin and Thompson [20] proposed Schedule Expert that is a system that uses a 0–1 mathematical model to schedule courses at the Cornell University School of Hotel Administration. Since the UCT structures are not standard as they vary from institution to institution, most of the presented research address specific case studies (see, also, [10, 30]).

In 2007, the Second International Timetabling Competition (ITC-2007) firstly posed the attention on the curriculum-based course timetabling (CB-CTT) problem. CB-CTT consists of finding the best weekly assignment of university course lectures to rooms and time periods. A feasible schedule must satisfy a set of hard constraints and must also take into account a set of soft constraints. From ITC-2007, many researchers have developed advanced models and methods to solve CB-CTT (see, e.g., [3, 4, 8, 16, 17, 19, 22–24, 31]). The survey by [5] is devoted to review the main works on the topic, with focus on mathematical models, lower bounds, and exact and heuristic algorithms. As pointed out by the comments of Lübbecke (2015) referring to the competition induced by the ITC challenges, “the latter certainly stimulated a broad and deep development in timetabling but the theoretical problem definition may be quite far from timetabling reality in the age of complex interdependencies of many interdisciplinary curricula at large universities.” With this paper, we do not aim at competing with ITC-2007 and successive papers on the topic but want to give our experience with CB-CTT in our university. Therefore, the presented research is focused on a particular case study, and deliberately lacks generality. This, in our opinion, tends to go in the direction of the suggestions contained in [25] where the author says that there should be more emphasis on the “dirty” aspects of CB-CTT than only exploring general models which are not completely able to capture real problems. Hence, in this paper, we describe a real-world CB-CTT application referred to the Faculty of Engineering at the University of Rome “Tor Vergata,” for which we proposed a mathematical program with many specific constraints and besides them we concentrated our attention on three objectives: (i) minimizing (weighted) flows produced by the students enrolled in the lectures, weighting flows according to traffic congestion conditions, (ii) minimizing soft conflicts generated by the possible overlap among compulsory and non-compulsory courses of the same curriculum, and (iii) smoothing the number of lesson hours per curriculum within the weekdays. This proposal was shared with people working on this problem in the administrative side of our university and a specific graphical interface was built to let all the components dialog correctly and the results be used easily. The proposed mixed-integer program has been implemented and solved by means of a commercial solver. Experimental findings showed that our modeling proposal is effective in producing satisfactory timetables for the considered applications.

The reminder of the paper is organized as follows: in Section 2, we describe the timetabling problem we are dealing with; in Section 3, the mathematical program is presented and discussed. Section 4 presents the description of the case study and Section 5 reports experimental results. Finally, in Section 6, we draw some conclusions and future research directions.

2 The Timetable Problem Definition

We consider a set of curricula (or degree programs), each one lasting a given number of years, divided in two semesters, and whose courses are taught in a number of lectures (lessons) delivered during given semesters of given years. A course is taught by one or more teachers and belongs to a given year of one or more curricula. The courses of a curriculum are grouped into compulsory and non-compulsory courses: the lectures of two compulsory courses for the same curriculum and of the same year cannot be delivered simultaneously, i.e., during the same time slot (hour) of the same weekday of the same semester (hard time-constraints), while overlapping between the lessons of a compulsory course and of a non-compulsory course is allowed albeit it is preferable to avoid this situation (soft time-constraints). Moreover, also the lessons of pairs of courses sharing a teacher cannot be delivered simultaneously.

A set of rooms of distinct capacities is used for delivering courses. Rooms are assumed to be available during each time slot of a day and cannot be assigned to more than one course at a time. Course lessons are delivered on a week basis, that is, during a semester, they are delivered in the same hours and weekdays as well as in the same room. Therefore, it is assumed that the timetable is organized in week days (from Monday to Saturday), where each day is divided into a set of lesson hours (time slots), including a break of one hour (lunch time) between the morning and the afternoon lessons. Saturday can be used for delivering lessons only for a subset of courses.

Each course requires a given number of weekly lesson hours that have to be delivered according to some requirements: the duration (number of consecutive hours) of a course lesson belongs to a given range, and during each day the total number of lesson hours of the same course cannot be greater than a maximum number. Course lessons cannot be delivered during the break hour, unless this is explicitly requested and motivated by a teacher: in this case, if a course lesson is delivered during the break hour then the lesson must last at least two hours and cannot be delivered during both the hours before and after the break. Courses with a very large number of students (as compared with room capacities) are split into what we call “channels”; for instance, if a course has 400 students and room capacities are not larger that 100 seats, then this course may be split into 4 identical courses (the channels of the course) of 100 students each. The lessons of all the channels of a course require to be delivered simultaneously in distinct rooms and each lesson hour of each course channel requires exactly one room whose capacity is not less than the number of the students enrolled in the course channel; moreover, if the lesson lasts more than one hour then the same room should be assigned to the (channel) course lesson for its entire duration.

Three criteria are considered to define the timetable: *Student flows*, *Soft conflicts*, and *Smoothing*. The first criterion is achieved by minimizing the total (weighted) incoming and leaving student flow produced by the students attending lectures. In particular, student flows are weighted according to the estimated time-dependent traffic congestion level of the road network nearby the campus. The second criterion is obtained by minimizing the number of soft incompatibilities produced by the possible overlap among the lessons of compulsory and non-compulsory courses. Finally, the third criterion is achieved by balancing the number of lecture hours per curriculum within the weekdays. Next, we provide the mathematical formulation of the problem.

3 The Mathematical Model

In this section, we provide a mixed-integer programming formulation of our CB-CTT problem. Tables 1, 2, 3, and 4 list the notations and definitions of sets, parameters, binary variables, and continuous variables, respectively. In particular, parameters $b_{cc'}$ represent the type of conflict between any couple (c, c') of distinct courses. We assume $b_{cc'} = 1$ (hard time-conflict) if there exists a channel $h \in \{1, \dots, ch_c\}$ of course $c \in C$ and a channel $h' \in \{1, \dots, ch_{c'}\}$ of course $c' \in C$ sharing a teacher, i.e., $Q_{ch} \cap Q_{c'h'} \neq \emptyset$. In addition, we also assume $b_{cc'} = 1$ if both c and c' are courses of the same year for a curriculum p and are both compulsory for the latter: that is, there exist a curriculum $p \in P$ such that $year_{cp} = year_{c'p} \neq 0$ and $obl_{cp} = obl_{c'p} = 1$. We assume $b_{cc'} = 2$ (soft time-conflict) when there is no hard time-conflict between courses c and c' and there exists a curriculum p for which $year_{cp} = year_{c'p} \neq 0$ and exactly one of the two courses is compulsory for p. In all the other cases, courses c and c' are considered non-conflicting (in time) and it is assumed $b_{cc'} = 0$. Each couple (c, c') of courses that are hard time-conflicting ($b_{cc'} = 1$) cannot be delivered

Table 1 Notations and definitions of sets

Sets	Definitions
P	Set of curricula
R	Set of rooms
D	Set of lecture week days, $D = \{\texttt{Monday}, \dots, \texttt{Saturday}\}$
T	Set of day time slots (lesson hours), $T = \{1, \dots, pause, \dots, \lvert T\rvert\}$; $pause$ is the lunch time hour, with $1 < pause < \lvert T\rvert$
Q	Set of teachers
Q_{ch}	Subset of teachers of channel h of course c ($Q_{ch} \subseteq Q$)
C	Set of all courses
C^0	Subset of courses taught during both the first and the second semester ($C^0 \subseteq C$)
C^s	Subset of courses taught only during semester $s \in \{1, 2\}$ ($C^s \subseteq C$)
$\widetilde{C}^s$	Subset of courses taught during semester $s \in \{1, 2\}$ (i.e., $\widetilde{C}^s = C^0 \cup C^s \subseteq C$)

Table 2 Notations and definitions of parameters

Parameters	Definitions
fw_{dt}	Student flow weight related to day time slot $t \in T \cup \{\|T\| + 1\}$ of day $d \in D$
u_r	Capacity of room $r \in R$
ny_p	Number of years of curriculum $p \in P$
$q^s_{p\tau}$	Number of students of curriculum $p \in P$, year $\tau \in \{1, \dots, ny_p\}$, and semester $s \in \{1, 2\}$
ch_c	Number of channels of course $c \in C$
d_c	Number of students for each channel of course $c \in C$
wh_c	Number of lesson hours per week of course $c \in C$ $(1 \leq wh_c \leq \|D\|\|T\|)$
dh^{min}_c	Minimum duration (number of consecutive day lesson hours) of a lesson of course $c \in C$ $(1 \leq dh^{min}_c \leq \|T\|)$
dh^{max}_c	Maximum duration of a lesson of course $c \in C$ $(1 \leq dh^{min}_c \leq dh^{max}_c \leq \|T\|)$
dh^{tot}_c	Maximum total number of day lesson hours of course $c \in C$ $(1 \leq dh^{min}_c \leq dh^{max}_c \leq dh^{tot}_c \leq \|T\|)$
sat_c	Equal to 1 if course $c \in C$ may have lesson on Saturday, and 0 otherwise
a_{ct}	Equal to 1 if course $c \in C$ may have lesson in day time slot $t \in T$, and 0 otherwise
$year_{cp}$	The year when course $c \in C$ is taught for curriculum $p \in P$ $(year_{cp} \in \{0\} \cup \{1, \dots, ny_p\})$, $year_{cp} = 0$ means that c is not a course of p
obl_{cp}	Equal to 1 if course $c \in C$ is compulsory for curriculum $p \in P$, and 0 otherwise
$b_{cc'}$	Equal to 1 if the lessons of courses $c, c' \in C$ cannot overlap in time (hard conflict), equal to 2 if time overlapping is not forbidden but should be avoided (soft conflict), and 0 otherwise

Table 3 Notations and definitions of binary variables

Binary variables	Definitions
v_{chrdt}	Equal to 1 if room $r \in R$ is assigned to channel $h \in \{1, \dots, ch_c\}$ of course $c \in C$ in time slot $t \in T$ of day $d \in D$, 0 otherwise
x^s_{cdt}	Equal to 1 if course $c \in \widetilde{C}^s$ is taught in time slot $t \in T$ of day $d \in D$ during semester $s \in \{1, 2\}$, and 0 otherwise
y^s_{cd}	Equal to 1 if course $c \in \widetilde{C}^s$ is taught on day $d \in D$ during semester $s \in \{1, 2\}$, and 0 otherwise
$w^s_{p\tau d}$	Equal to 1 if for curriculum $p \in P$ there is a lecture on day $d \in D$ during semester $s \in \{1, 2\}$ of year $\tau \in \{1, \dots, ny_p\}$, and 0 otherwise
$z^{Ss}_{p\tau dt}$	Equal to 1 if the first lesson of a compulsory course of curriculum $p \in P$, taught on day $d \in D$ during semester $s \in \{1, 2\}$, starts in time slot $t \in T$, 0 otherwise
$z^{Es}_{p\tau dt}$	Equal to 1 if the last lesson of a compulsory course of curriculum $p \in P$, taught on day $d \in D$ during semester $s \in \{1, 2\}$, ends in time slot $t \in T$, 0 otherwise

Table 4 Notations and definitions of continuous variables

Continuous variables	Definitions
α_{cd}^{Fs}	Ranging in $[0, \|T\|]$, being the first time slot when course $c \in \widetilde{C}^s$ is taught on day $d \in D$ (0 if c is not taught on day d) during semester $s \in \{1, 2\}$
α_{cd}^{Ls}	Ranging in $[0, \|T\|]$, being the last time slot when course $c \in \widetilde{C}^s$ is taught on day $d \in D$ (0 if c is not taught on day d) during semester $s \in \{1, 2\}$
$\beta_{p\tau d}^{Fs}$	Ranging in $[0, \|T\|]$, being the first time slot when a lecture of curriculum $p \in P$ is delivered on day $d \in D$ during semester $s \in \{1, 2\}$ of year $\tau \in \{1, \ldots, ny_p\}$ (0 if no lecture for p is delivered on that weekday)
$\beta_{p\tau d}^{Ls}$	Ranging in $[0, \|T\|]$, being the last time slot when a lecture of curriculum $p \in P$ is delivered on day $d \in D$ during semester $s \in \{1, 2\}$ of year $\tau \in \{1, \ldots, ny_p\}$ (0 if no lecture for p is delivered on that weekday)
$\gamma_{p\tau}^{s}$	Ranging in $[0, \|T\|]$, being the maximum number of daily lesson time slots (hours) delivered during semester $s \in \{1, 2\}$ of year $\tau \in \{1, \ldots, ny_p\}$ for program $p \in P$ (0 if no lecture for p is delivered during semester s of year τ)
$\delta_{cc'dt}^{s}$	Ranging in $[0, 1]$, equal to 1 if for two soft conflicting courses $c, c' \in C$ (i.e., with $b_{cc'} = 2$), the lessons are both delivered in time slot $t \in T$ of day $d \in D$ during semester $s \in \{1, 2\}$

simultaneously; this can occur for two soft-conflicting courses (i.e., when $b_{cc'} = 2$) but, as previously said, it could be preferable to avoid it.

The three criteria *Student flows*, *Smoothing*, and *Soft conflicts* are modeled by considering the following three objective functions.

$$of_1 = \sum_{p \in P} \sum_{\tau=1}^{ny_p} \sum_{s \in \{1,2\}} \sum_{d \in D} \sum_{t \in T} q_{p\tau}^{s} (fw_{dt} \cdot z_{p\tau dt}^{Ss} + fw_{d,t+1} \cdot z_{p\tau dt}^{Es}), \qquad (1)$$

$$of_2 = \sum_{p \in P} \sum_{\tau=1}^{ny_p} \sum_{s \in \{1,2\}} \gamma_{p\tau}^{s}, \qquad (2)$$

$$of_3 = \frac{1}{2} \sum_{p \in P} \sum_{\tau=1}^{ny_p} \sum_{s \in \{1,2\}} \sum_{\substack{(c, c') \in \widehat{C}_{p\tau}^{s} \times \widehat{C}_{p\tau}^{s} \,:\\ c \neq c', b_{cc'} = 2}} \sum_{d \in D} \sum_{t \in T} \delta_{cc'dt}^{s}, \qquad (3)$$

where $\widehat{C}_{p\tau}^{s} = \{c \in \widetilde{C}^s : year_{cp} = \tau\}$, with $p \in P$, $\tau \in \{1, \ldots, ny_p\}$, and $s \in \{1, 2\}$, are the sets of courses of curriculum p taught in semester s of year τ.

(1) Objective function of_1 is the sum of the total weighted students incoming flow and the total weighted students leaving flow, calculated over the sets of curricula, years, semesters, weekdays, and day time slots, where flow weight fw_{dt} represents

the level of road traffic congestion nearby the university during time slot t of day d. In particular, incoming and leaving student flows are evaluated on the basis of the number of students enrolled in the first and the last daily lesson hours, respectively, of compulsory courses; it is assumed that students leave the campus during the next time slot (hour) after the end of the last lecture: this is why variable $z^{Es}_{p\tau dt}$ is weighted by $fw_{d,t+1}$. By minimizing of_1 we minimize *Student flows*. Objective function of_2 (2) is the sum of the maximum number of day lesson hours during the week, calculated over the sets of curricula, years, and semesters: minimizing of_2 leads to a balanced timetable with respect to the number of daily lesson hours, that is to *Smoothing*. Finally, by minimizing objective function of_3 (3), being the total number of soft conflicts, we minimize *Soft conflicts*.

The mathematical model contains many constraints and for the sake of clearness they are grouped and then presented according to the following groups:

- *Room assignment constraints*: allow to assign rooms to course channels for each time slot (hour) when course lessons are delivered, by linking variables v_{chrdt} with variables x^s_{cdt}. Room assignment is done while respecting room capacities and assuring the assignment of the same room to a course channel for the whole course lesson duration.
- *Slot preset and lunch time hour constraints*: allow to avoid delivering courses in forbidden time slots (hours). Moreover, they allow to deliver course lesson during the lunch time (break hour), for the courses for which this is allowed, only if the course lesson lasts at least two hours and avoiding to deliver it both before and after the lunch time hour. In addition, these constraints also allow to avoid delivering lessons on Saturday for the courses for which this is forbidden.
- *Course weekly and daily hours assignment constraints*: allow to assign lesson hours (time slots) to courses both on a daily and weekly basis, taking into account the number of weekly course lesson hours to be delivered and the course lesson requirements on a daily basis: for each course, the total number of daily lesson hours cannot exceed a given maximum value and the duration (number of consecutive hours) of each single lesson must be between given minimum and maximum durations.
- *Hard and soft time-conflict constraints*: avoid to deliver simultaneously the lessons of couples of hard time-conflicting courses, and allow the identification of soft time-conflicts among course lessons, necessary for the calculation of *Soft conflicts* objective function of_3 (3).
- *Starting and ending hours of compulsory course constraints*: allow to find for each day when the first compulsory course starts and when the last compulsory course ends (if any), for each semester of each year of each curriculum. This information is necessary for the calculation of *Student flows* objective function of_1 (1).
- *First and last lesson time slots constraints*: are useful to calculate the values of variables α^{Fs}_{cd}, α^{Ls}_{cd}, and the values of variables $\beta^{Fs}_{p\tau d}$, $\beta^{Ls}_{p\tau d}$. In addition, these constraints allow to calculate the values of variables $\gamma^{s}_{p\tau}$, representing the maximum number of daily lesson hours of year τ of curriculum p during semester s.

The latter variables are used for the calculation of *Smoothing* objective function of_2 (2).

- *Daily lecture delivering constraints*: allow to link variables y^s_{cd} with variables x^s_{cdt}, and to combine the former also with variables $w^s_{p\tau d}$.

The *Room assignment constraints* are:

$$\sum_{r\in R} v_{chrdt} = x^s_{cdt}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ h \in \{1,\dots,ch_c\},\ d \in D,\ t \in T, \tag{4}$$

$$\sum_{c\in\widetilde{C}^s}\sum_{h=1}^{ch_c} v_{chrdt} \le 1, \quad \forall s \in \{1,2\},\ r \in R,\ d \in D,\ t \in T, \tag{5}$$

$$x^s_{cdt} + x^s_{cd,t+1} + v_{chrdt} \le 2 + v_{chrd,t+1}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ h \in \{1,\dots,ch_c\},\ r \in R,\ d \in D,\ t \in T\backslash\{|T|\}, \tag{6}$$

$$d_c \cdot v_{chrdt} \le u_r, \quad \forall c \in C,\ h \in \{1,\dots,ch_c\},\ r \in R,\ d \in D,\ t \in T. \tag{7}$$

Constraint (4) imposes the assignment of exactly one (distinct) room to each channel of a course only during each week-daily hour of a semester when the course is taught. Constraint (5) assures that each room can be assigned to at most one course channel during each week-daily hour of each semester. Constraint (6) assures assigning the same room to a course channel during each hour of a course lesson if its duration is greater than one hour. Constraint (7) avoids assigning (small) rooms to (crowded) course channels if the room capacity is less than the number of students enrolled in that course channel.

The *Slot preset and lunch time hour constraints* are:

$$\sum_{d\in D} x^s_{cdt} = 0, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s : a_{ct} = 0,\ t \in T, \tag{8}$$

$$\sum_{t=pause-1}^{pause+1} x^s_{cdt} \le 2, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s : a_{c,\,pause} = 1,\ d \in D, \tag{9}$$

$$x^s_{cd,\,pause} \le x^s_{cd,\,pause-1} + x^s_{cd,\,pause+1}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s : a_{c,\,pause} = 1,\ d \in D, \tag{10}$$

$$\sum_{t\in T} x^s_{c,\texttt{Saturday},t} = 0, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s : sat_c = 0. \tag{11}$$

Constraint (8) avoids delivering courses during courses forbidden time slots (hours). For the courses that may be delivered also during the lunch time (pause) hour, constraints (9) and (10) allow one to deliver course lesson during the lunch time hour only if the course lesson is also delivered during the previous or successive hour, but not in both. Constraint (11) does not allow one to deliver course lessons on Saturday for the courses for which this is forbidden.

The *course weekly and daily hours assignment constraints* are:

$$\sum_{d\in D}\sum_{t\in T} x^s_{cdt} = wh_c, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s, \tag{12}$$

$$\sum_{t\in T} x^s_{cdt} \le dh^{tot}_c, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D, \tag{13}$$

$$\sum_{\theta=t}^{t+dh^{max}_c} x^s_{cd\theta} \le dh^{max}_c, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D,$$
$$t \in \{1,\ldots,|T| - dh^{max}_c\}, \tag{14}$$

$$dh^{min}_c \cdot x^s_{cd1} \le \sum_{\theta=1}^{dh^{min}_c} x^s_{cd\theta}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D, \tag{15}$$

$$dh^{min}_c (x^s_{cdt} - x^s_{cd,t-1}) \le \sum_{\theta=t}^{\min(|T|,t+dh^{min}_c-1)} x^s_{cd\theta},$$
$$\forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D,\ t \in T\backslash\{1\}. \tag{16}$$

Constraint (12) assures that the total lesson hours assigned to each course must be equal to the number of weekly lesson hours required for the course. For each day, constraint (13) imposes that the number of total lesson hours assigned to a course cannot be greater than the maximum daily lesson hours for the course, while constraints (14), (15), and (16) impose upper and lower bounds on the number of hours to each course lesson.

The *Hard and soft time–conflict constraints* are:

$$x^s_{cdt} + x^s_{c'dt} \le 1, \quad \forall s \in \{1,2\},\ (c,c') \in \widetilde{C}^s \times \widetilde{C}^s : c \ne c' \text{ and } b_{cc'} = 1,$$
$$d \in D,\ t \in T, \tag{17}$$

$$\delta^s_{cc'dt} \le x^s_{cdt}, \quad \forall s \in \{1,2\},\ (c,c') \in \widetilde{C}^s \times \widetilde{C}^s : c \ne c' \text{ and } b_{cc'} = 2,$$
$$d \in D,\ t \in T, \tag{18}$$

$$\delta^s_{cc'dt} \le x^s_{c'dt}, \quad \forall s \in \{1,2\},\ (c,c') \in \widetilde{C}^s \times \widetilde{C}^s : c \ne c' \text{ and } b_{cc'} = 2,$$
$$d \in D,\ t \in T, \tag{19}$$

$$\delta^s_{cc'dt} \ge x^s_{cdt} + x^s_{c'dt} - 1, \forall s \in \{1,2\},\ (c,c') \in \widetilde{C}^s \times \widetilde{C}^s : c \ne c' \text{ and } b_{cc'} = 2,$$
$$d \in D,\ t \in T, \tag{20}$$

$$\delta^s_{cc'dt} = \delta^s_{c'cdt}, \quad \forall s \in \{1,2\},\ (c,c') \in \widetilde{C}^s \times \widetilde{C}^s : c \ne c' \text{ and } b_{cc'} = 2,$$
$$d \in D,\ t \in T. \tag{21}$$

Constraint (17) does not allow one to deliver simultaneously the lessons of couples of courses for which this is forbidden (hard time-conflict): that is, for two compulsory courses of the same curriculum that are delivered during the same year and semester. Constraints (18)–(21) allow one to detect soft time-conflict occurrences between a compulsory course and a non-compulsory course, of the same year for a certain curriculum, that are taught during the same semester.

The *Starting and ending hours of compulsory course constraints* are:

$$\sum_{\theta=1}^{t} z^{Ss}_{p\tau d\theta} \geq \sum_{c\in\widehat{C}^s_{p\tau}} obl_{cp}\cdot x^s_{cdt}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D,\ t\in T, \tag{22}$$

$$z^{Ss}_{p\tau dt} \leq \sum_{c\in\widehat{C}^s_{p\tau}} obl_{cp}\cdot x^s_{cdt}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D,\ t\in T, \tag{23}$$

$$\sum_{t\in T} z^{Ss}_{p\tau dt} \leq 1, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D, \tag{24}$$

$$\sum_{\theta=t}^{|T|} z^{Es}_{p\tau d\theta} \geq \sum_{c\in\widehat{C}^s_{p\tau}} obl_{cp}\cdot x^s_{cdt}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D,\ t\in T, \tag{25}$$

$$z^{Es}_{p\tau dt} \leq \sum_{c\in\widehat{C}^s_{p\tau}} obl_{cp}\cdot x^s_{cdt}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D,\ t\in T, \tag{26}$$

$$\sum_{t\in T} z^{Es}_{p\tau dt} \leq 1, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D. \tag{27}$$

Constraints (22)–(24) and constraints (25)–(27) allow one to detect, for each semester and for each weekday, the first and last time slots when the lesson of a compulsory course of year τ for curriculum p is delivered, respectively.

The *first and last lesson time slots constraints* are:

$$t\, x^s_{cdt} + |T|(1-x^s_{cdt}) \geq \alpha^{Fs}_{cd}, \quad \forall s\in\{1,2\},\ c\in\widetilde{C}^s,\ d\in D,\ t\in T, \tag{28}$$

$$t\, x^s_{cdt} \leq \alpha^{Ls}_{cd}, \quad \forall s\in\{1,2\},\ c\in\widetilde{C}^s,\ d\in D,\ t\in T, \tag{29}$$

$$\alpha^{Fs}_{cd} \leq \alpha^{Ls}_{cd}, \quad \forall s\in\{1,2\},\ c\in\widetilde{C}^s,\ d\in D, \tag{30}$$

$$\alpha^{Ls}_{cd} - \alpha^{Ss}_{cd} \leq |T|-1, \quad \forall s\in\{1,2\},\ c\in\widetilde{C}^s,\ d\in D, \tag{31}$$

$$\alpha^{Ls}_{cd} \leq |T|\cdot y^s_{cd}, \quad \forall s\in\{1,2\},\ c\in\widetilde{C}^s,\ d\in D, \tag{32}$$

$$\beta^{Fs}_{p\tau d} \leq \alpha^{Fs}_{cd} + |T|(1-y^s_{cd}), \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ c\in\widehat{C}^s_{p\tau},\ d\in D, \tag{33}$$

$$\beta^{Ls}_{p\tau d} \geq \alpha^{Ls}_{cd} + |T|(1-y^s_{cd}), \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ c\in\widehat{C}^s_{p\tau},\ d\in D, \tag{34}$$

$$\beta^{Fs}_{p\tau d} \leq \beta^{Ls}_{p\tau d}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D, \tag{35}$$

$$\beta^{Ls}_{p\tau d} - \beta^{Fs}_{p\tau d} \leq |T|-1, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D, \tag{36}$$

$$\beta^{Ls}_{p\tau d} \leq |T|\cdot w^s_{p\tau d}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D, \tag{37}$$

$$\beta^{Ls}_{p\tau d} - \beta^{Fs}_{p\tau d} + w^s_{p\tau d} \leq \gamma^s_{p\tau}, \quad \forall s\in\{1,2\},\ p\in P,\ \tau\in\{1,\dots,ny_p\},\ d\in D. \tag{38}$$

Constraints (28)–(32) allow one to calculate, for each weekday of a semester and for each course taught during the semester, when the first lesson of the course starts and when the last lesson of the same course ends, according to the assignment of time slots to the course; in particular, constraint (32), together with (31), forces the last and first time slots to be null (the relative variables are set to 0) if the course is not taught on that day during the semester. Similarly, constraints (33)–(37) allow one to calculate, for each weekday of a semester and for each year of each curriculum, when the first lesson starts and when the last lesson ends if there is at least one lesson for the given year and semester of the given curriculum. The calculations are respectively done by taking the minimum among the starting hours and the maximum among the ending hours of the course lessons of the specific year of the curriculum delivered on the specific day of the semester; also in this case, constraint (37), together with (36), forces the last and first time slots to be null (the relative variables are set to 0) if no lesson is delivered on that day for the semester of the specific year of the curriculum. Finally, constraint (38) allows the calculation of the maximum number of daily lesson hours over the weekdays, for each given year of each curriculum and during each semester.

The *daily lecture delivering constraints* are:

$$\sum_{t \in T} x^s_{cdt} \leq |T| \cdot y^s_{cd}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D, \tag{39}$$

$$\sum_{t \in T} x^s_{cdt} \geq y^s_{cd}, \quad \forall s \in \{1,2\},\ c \in \widetilde{C}^s,\ d \in D, \tag{40}$$

$$\sum_{c \in \widehat{C}^s_{p\tau}} y^s_{cd} \leq |\widehat{C}^s_{p\tau}| \cdot w^s_{p\tau d}, \quad \forall s \in \{1,2\},\ p \in P,\ \tau \in \{1,\ldots,ny_p\},\ d \in D, \tag{41}$$

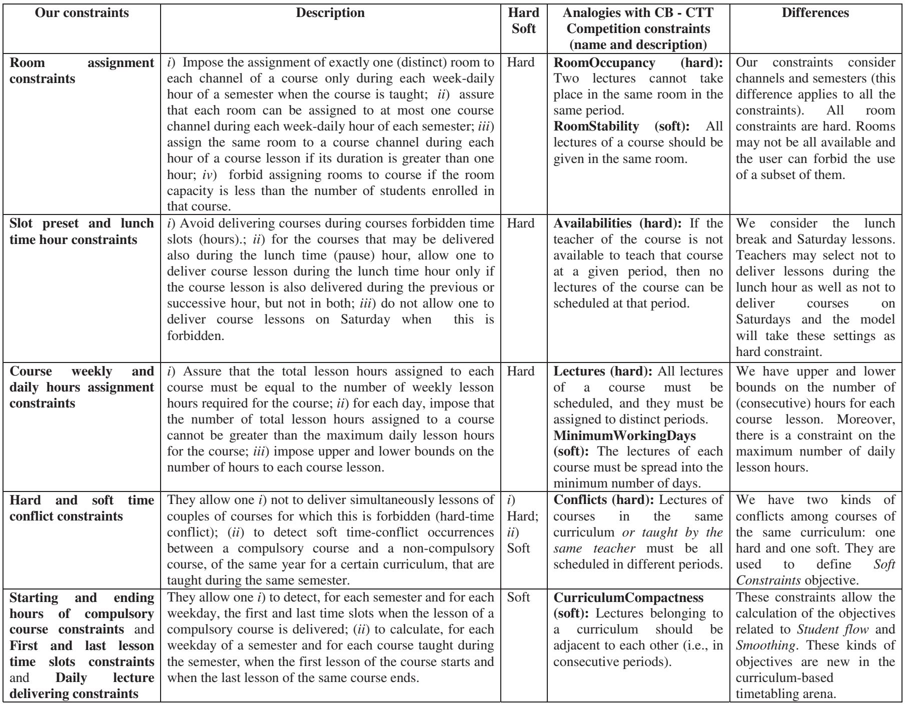
$$\sum_{c \in \widehat{C}^s_{p\tau}} y^s_{cd} \geq w^s_{p\tau d}, \quad \forall s \in \{1,2\},\ p \in P,\ \tau \in \{1,\ldots,ny_p\},\ d \in D. \tag{42}$$

Constraints (39)–(42) combine together decision variables x^s_{cdt}, y^s_{cd}, and $w^s_{p\tau d}$, in order to derive from the former, representing the assignment of lesson hours (time slot) to courses (x^s_{cdt}), the decisions of delivering respectively lessons of distinct courses (y^s_{cd}) and lessons of specific years of distinct curricula ($w^s_{p\tau d}$), during a certain weekday of a semester.

Therefore, the whole mathematical model of our CB-CTT problem is the following multi-objective mixed-integer problem:

$$\begin{aligned} \min \quad & of_i, \quad \forall i \in \{1,\ldots,3\}, \\ s.t. \quad & (4)\text{–}(42), \\ & \text{binary variables of Table 3 in \{0,1\}}, \\ & \text{non-negative variables of Table 4 in the specified range.} \end{aligned}$$

Before concluding this section, we note that in the CB-CTT literature (see, e.g., [7, 14], and [23]), there are similar constraints; in order to give an immediate idea of which constraints are new and which are the same as in the literature, in Fig. 1, we



Our constraints	Description	Hard Soft	Analogies with CB - CTT Competition constraints (name and description)	Differences
Room assignment constraints	*i)* Impose the assignment of exactly one (distinct) room to each channel of a course only during each week-daily hour of a semester when the course is taught; *ii)* assure that each room can be assigned to at most one course channel during each week-daily hour of each semester; *iii)* assign the same room to a course channel during each hour of a course lesson if its duration is greater than one hour; *iv)* forbid assigning rooms to course if the room capacity is less than the number of students enrolled in that course.	Hard	**RoomOccupancy (hard):** Two lectures cannot take place in the same room in the same period. **RoomStability (soft):** All lectures of a course should be given in the same room.	Our constraints consider channels and semesters (this difference applies to all the constraints). All room constraints are hard. Rooms may not be all available and the user can forbid the use of a subset of them.
Slot preset and lunch time hour constraints	*i)* Avoid delivering courses during courses forbidden time slots (hours).; *ii)* for the courses that may be delivered also during the lunch time (pause) hour, allow one to deliver course lesson during the lunch time hour only if the course lesson is also delivered during the previous or successive hour, but not in both; *iii)* do not allow one to deliver course lessons on Saturday when this is forbidden.	Hard	**Availabilities (hard):** If the teacher of the course is not available to teach that course at a given period, then no lectures of the course can be scheduled at that period.	We consider the lunch break and Saturday lessons. Teachers may select not to deliver lessons during the lunch hour as well as not to deliver courses on Saturdays and the model will take these settings as hard constraint.
Course weekly and daily hours assignment constraints	*i)* Assure that the total lesson hours assigned to each course must be equal to the number of weekly lesson hours required for the course; *ii)* for each day, impose that the number of total lesson hours assigned to a course cannot be greater than the maximum daily lesson hours for the course; *iii)* impose upper and lower bounds on the number of hours to each course lesson.	Hard	**Lectures (hard):** All lectures of a course must be scheduled, and they must be assigned to distinct periods. **MinimumWorkingDays (soft):** The lectures of each course must be spread into the minimum number of days.	We have upper and lower bounds on the number of (consecutive) hours for each course lesson. Moreover, there is a constraint on the maximum number of daily lesson hours.
Hard and soft time conflict constraints	They allow one *i)* not to deliver simultaneously lessons of couples of courses for which this is forbidden (hard-time conflict); (*ii*) to detect soft time-conflict occurrences between a compulsory course and a non-compulsory course, of the same year for a certain curriculum, that are taught during the same semester.	*i)* Hard; *ii)* Soft	**Conflicts (hard):** Lectures of courses in the same curriculum *or taught by the same teacher* must be all scheduled in different periods.	We have two kinds of conflicts among courses of the same curriculum: one hard and one soft. They are used to define *Soft Constraints* objective.
Starting and ending hours of compulsory course constraints and **First and last lesson time slots constraints** and **Daily lecture delivering constraints**	They allow one *i)* to detect, for each semester and for each weekday, the first and last time slots when the lesson of a compulsory course is delivered; (*ii*) to calculate, for each weekday of a semester and for each course taught during the semester, when the first lesson of the course starts and when the last lesson of the same course ends.	Soft	**CurriculumCompactness (soft):** Lectures belonging to a curriculum should be adjacent to each other (i.e., in consecutive periods).	These constraints allow the calculation of the objectives related to *Student flow* and *Smoothing*. These kinds of objectives are new in the curriculum-based timetabling arena.

Fig. 1 Comparison between the constraints in our model and those in the literature

report a comparison between the (hard and soft) constraints reported in this paper and those from the literature (in particular in the curriculum-based course timetabling problem as treated in the international competition).

4 The Case Study

We applied our timetable model to timetabling a subset of the courses of a two semesters academic year of our faculty (the Faculty of Engineering at the University of Rome “Tor Vergata,” Rome, Italy). The case study is related to a project of our University in which the goal was to consider both traffic congestion due to student flow and quality of the timetable. The case study considers 5 bachelor degree curricula in Management Engineering, each one organized in three years and two semesters per year, a set of 55 courses, and the availability of a set of 34 rooms of different size (capacity). Table 5 lists rooms availability with related capacities.

Both curricula and courses have different amount of student enrolled. In particular, we consider a curriculum with many students enrolled (large size), two medium size curricula and other two of small size. Table 6 lists the number of students enrolled in each curriculum, for each semester and for each year.

In general, a course is taught for more than one curriculum, and the courses with a large number of students (greater than 200) are subdivided into a set of channels,

Table 5 Number of rooms of distinct capacities

Capacity	Number of rooms
234	4
165	6
135	5
123	2
83	2
70	1
62	4
54	9
45	1
Total	34

each one followed by at most 200 students (typically such courses serve also other curricula besides those considered in our case study): two courses are subdivided into six channels, one course into three channels and one into two channels, while all the other courses have only one channel. Recall that all the channels of a course require their lessons to be delivered simultaneously on distinct rooms. Table 7 lists the number of courses with respect to the students enrolled.

Courses also differ in compulsory and non-compulsory, delivering period, and according to lesson hours per week. Table 8 lists the number of courses with respect to the curricula and Table 9 shows the number of courses based on the other characteristics. Typically, the lessons of courses with 8 or 6 lesson hours per week have durations of two or three (consecutive) hours, while lessons of the courses with 4 hours per week last exactly two hours; only two (laboratory) courses of 4 hours per week are in exception with lessons' duration of four hours (i.e., for these two courses, there is only one lesson of 4 consecutive hours per week). For each course, the maximum number of lesson hours per day is equal to the largest possible duration of its lessons, and this (along with the minimum durations of course lessons) implies that there will be at most one lesson per day for each course.

Finally, congestion flow weights are evaluated for each time slot of a weekday, according to the estimations of different travel times of a typical trip traversing portion of the road network nearby the location of the campus, in relation to different

Table 6 Number of students enrolled in the given curricula

Periods		Curricula				
Year	Semester	1	2	3	4	5
1	1	180	30	20	10	10
	2	150	25	18	8	8
2	1	120	20	15	7	7
	2	100	18	12	6	6
3	1	90	16	10	5	5
	2	80	14	8	4	4

Table 7 Number of courses per courses per enrolled student range

Student range	Number of courses
[0, 40]	19
[41, 70]	11
[71, 135]	9
[136, 165]	16*
Total	55

*With 2 in 6 channels, 1 in 3 channels, and 1 in 2 channels

traffic congestion conditions during each hour of a weekday. Congestion flow weights are scaled in a range between 0 and 10 (the higher is the congestion level and the larger is the flow weight). Lessons can be delivered during the first ten time slots of a weekday. The last (number 11) time slot are considered to weight the outflow of students enrolled in the very last lectures of a day, assuming that each day the students leave the campus after the end of last followed lecture. Table 10 reports the considered congestion flow weights for each time slot (hour) of a weekday. In particular, the larger values (greater than 7 over 10) are concentrated in the first two time slot of each day (with the exception of Saturday), where typically we register the traffic congestion peak hour, while we have the smallest values during the two time slots (hours) before lunch time (time slot 6). Another traffic congestion peak period is located after 5:00pm.

We note that the considered case study can be decomposed for the two semesters, but the model can handle the more general case of annual courses.

5 Experimental Results

In this section, we show the findings of our experimental campaign. Tests were conducted on a PC with processor Inter Core i5 AMD Atlon with 2.4GHz and 2GB VRAM. The mathematical model described in the previous section has been implemented in the AMPL language and solved by means of the CPLEX solver version 12.3.0.1.

We solved the mathematical model both in a single-objective version using proper weights for the objectives, i.e., 1 for the target objective (e.g., *Student Flow*) and 0 otherwise, and in a multi-objective version to evaluate the capability of the model to reconstruct the Pareto front.

Table 8 Number of courses per curriculum

Curriculum	Number of courses	(Compulsory courses)
1	31	(19)
2	41	(20)
3	38	(20)
4	32	(24)
5	25	(18)

Table 9 Courses characteristics

Characteristic		Number of courses
Semester	1	19
	2	36
Weekly hours	4	29
	6	14
	8	12

We set a time limit for the computation equal to 3 h.

Single-objective analysis In the left part of Fig. 2, we report the values of the three objectives when *Student Flow* is minimized (for ease of presentation, here and through the values of the objective functions are normalized to 1). The best value obtained for the latter objective within the time limit is 0.15 and is the lowest than the other two objective values which are 0.45 and 0.48 for the *Soft Conflicts* and *Smoothing* objectives, respectively. In the right part of Fig. 2, we show the trend of the *Student Flow* objective over time. We note that the solver is able to improve the solution value from 0.71 to 0.15 in about 2 h while in the remaining hour it was not able to provide further best solutions. The optimality gap at the time limit was 2.4%.

In Fig. 3, we report a chart related to the best values of the three objectives when, in turn, they are minimized.

Bi-objective analysis In the following, we report our analysis on bi-objective scenarios. Among the three pairs of objectives, in this analysis, we took into account *Student Flow* in association with one of the remaining two objectives. Under this hypothesis, in Fig. 4, we report the Pareto curves obtained by considering *Soft Conflicts* vs *Student Flow* (left chart) and *Smoothing* vs *Student Flow* objectives (right chart), respectively. The values of the objectives reported in the charts have

Table 10 Congestion flow weights

Time slots	Times	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
1	08:30–09:30	9.48	10.00	9.87	9.74	8.83	0.65
2	09:30–10:30	7.58	8.00	7.90	7.79	7.06	0.52
3	10:30–11:30	5.69	6.00	5.92	5.84	5.30	0.39
4	11:30–12:30	0.95	1.00	0.99	0.97	0.88	0.06
5	12:30–12:30	0.95	1.00	0.99	0.97	0.88	0.06
6	13:30–14:00	4.75	5.30	5.45	5.61	5.38	0.78
7	14:00–15:00	3.17	3.53	3.64	3.74	3.58	0.52
8	15:00–16:00	1.58	1.77	1.82	1.87	1.79	0.26
9	16:00–17:00	4.75	5.30	5.45	5.61	5.38	0.78
10	17:00–18:00	6.34	7.06	7.27	7.48	7.17	1.04
11	18:00–19:00	7.92	8.83	9.09	9.35	8.96	1.30

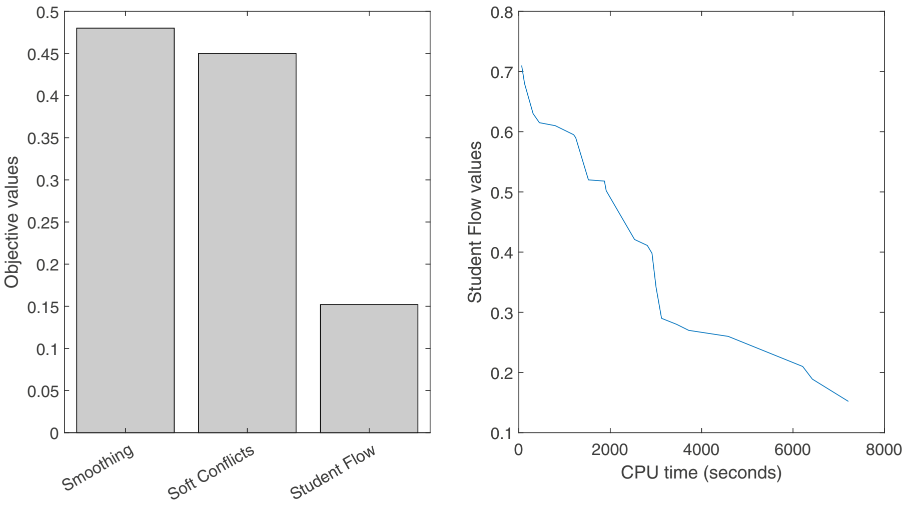


Fig. 2 Objective function values when student flow is minimized

been obtained as follows. Each bi-objective problem has been solved 100 times by specific instantiations of the convex combination between the two target (normalized) objectives. In particular, we firstly set to 0 and to 1, respectively, the weights of the two objectives and then iteratively increased the former and decreased the latter by an amount of 0.01, respectively. For each test, we stored the values of the two objectives and drawn a chart in which the $\mathbb{R}^2$ vectors so obtained have been represented in the Cartesian plane.

By the charts, one can note that there is a conflicting behavior of the two pairs of objectives. In the left chart, we see that the *Student Flow* values range from 0.15 to 0.23 while the *Soft Conflicts* values range from 0.16 to 0.45. The ideal point

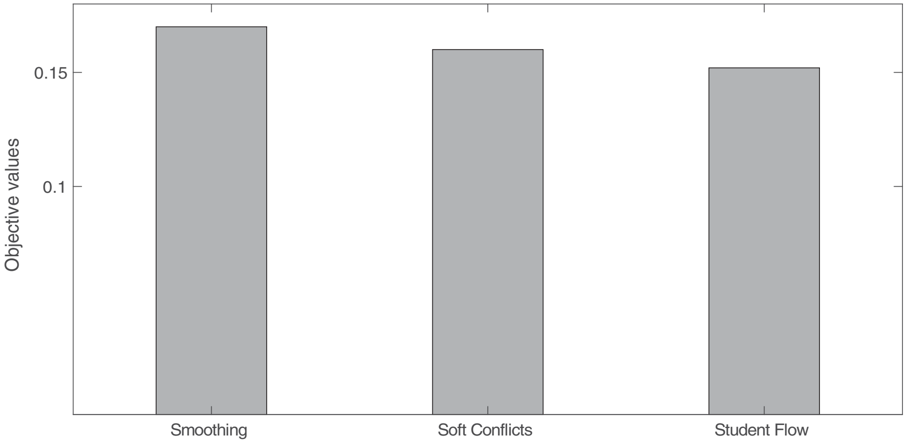


Fig. 3 Best objective function values when minimizing them individually

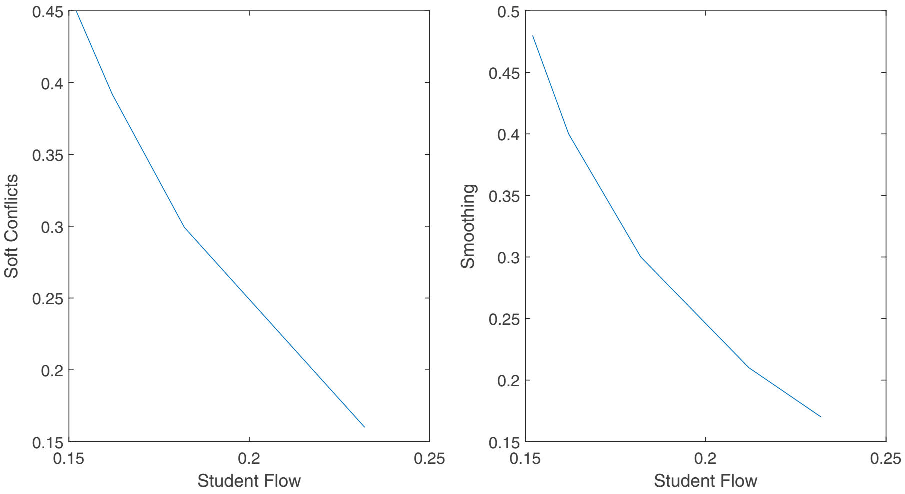


Fig. 4 Pareto curves obtained by considering *Soft Conflicts* vs *Student Flow* (left chart) and *Smoothing* vs *Student Flow* objectives (right chart), respectively

with coordinates [0.15, 0.16] has an Euclidean distance from the curve equal to 0.13. The rationale behind the obtained results is that trying to diminish the effect of the soft time-conflicts—which means trying to obtain a schedule where the set of non-compulsory courses is not in conflict with the set of compulsory courses as much as possible—tends to spread the flow of students in quite a large number of different daily time slots (hours) and with a consequent increase of the (congested weighted) *Student Flow* values. In fact, spreading the student flows on a larger set of time periods implies the arrival of part of the students at the university during more congested time periods, i.e., with larger flow weights, and similarly for their return home from the university. Conversely, as soon as one reduces the weight of the former objective function, the model tends to produce a timetable that allows the students to arrive and leave the campus during less congested time slots, therefore decreasing the value of *Student Flow*.

As for the right chart, the *Smoothing* values range from 0.17 to 0.48. The ideal point with coordinates [0.15, 0.17] has an Euclidean distance from the curve equal to 0.18. We can borrow from the previous chart the same analysis: indeed, for increasing weights of *Smoothing*, the model tends to generate a timetable with more or less the same number of lesson hours during all the weekdays with a consequent increase of the student flow on more congested time slots in place of a schedule which tries to minimize the (congested weighted) *Student Flow*. The latter situation happens when one reduces the weight associated with *Smoothing* and, consequently, increases that of *Student Flow* objective function.

5.1 Comparison with the Real Practical Timetable

The proposed model is able to provide satisfactorily solutions as compared with the real scenario under consideration. In fact, its affectiveness can be appreciated calculating the objective values of the solution currently adopted at our University; this can be done by running the model after having properly fixed decision variables v_{chrdt} and x^{s}_{cdt}. The results say that the current timetable, i.e., before the introduction of such a tool, produces a *Student Flow* value equal to 0.40, a *Soft Constraints* value equal to 0.48, and a *Smoothing* value equal to 0.50. These values, compared with the results obtained in the previous section, show the benefits of the introduction of our model.

6 Conclusions and Further Research

In this paper, we tackled a curriculum-based university course timetabling problem. A real-world application associated with the scenario of the Faculty of Engineering at the University of Rome "Tor Vergata" has stimulated the definition of a mathematical program with many specific constraints and a three-objective function aiming at (i) minimizing the (weighted) flow produced by the students enrolled in the lectures, (ii) minimizing soft conflicts generated by the possible overlap among compulsory and non-compulsory courses of the same curriculum, and (iii) smoothing the number of lecture hours per curriculum within the weekdays. The proposed mixed-integer program has been implemented and solved by means of a commercial solver and the experimental findings showed that our modeling proposal is effective in producing satisfactorily timetables for the considered application. Future research will be devoted to enlarge the pool of objectives taking into account the desiderata of students and teachers by distributing questionnaires to the latter and producing an AHP analysis from which one can derive proper weights for the objectives and experiment with the associated scalarized objective function. Also, we plan to formalize the analysis on the transportation aspects related to the local flow of students generated by the course timetable considering the impact on the traffic jam in a more comprehensive area comprising also trajectories of vehicles directed to the university and coming from further places.

Funding Information This study has been supported by the University of Rome "Tor Vergata" under grant TRAN.S.S.UM.: Transportation Sustainability and Students' Urban Mobility: A Multicriteria Approach Through Class Timetabling Optimization.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Al-Yakoob SM, Sherali HD (2007) A mixed-integer programming approach to a class timetabling problem: a case study with gender policies and traffic considerations. Eur J Oper Res 180(3):1028–1044
2. Badri MA (1996) A two-stage multi-objective scheduling model for faculty-course time assignments. Eur J Oper Res 94:16–28

3. Bellio R, Ceschia S, Di Gaspero L, Schaerf A, Urli T (2013) A simulated annealing approach to the curriculum-based course timetabling problem. In: Proceedings of the 6th multidisciplinary international conference on scheduling: theory and applications, MISTA 2013, Belgium, pp 314–317
4. Bellio R, Ceschia S, Di Gaspero L, Schaerf A, Urli T (2014) Feature-based tuning of simulated annealing applied to the curriculum-based course timetabling problem. arXiv:1409.7186
5. Bettinelli A, Cacchiani V, Roberti R, Toth P (2015) An overview of curriculum-based course timetabling. TOP 23:313–349
6. Birbas T, Daskalaki S, Housos E (1997) Timetabling for Greek high schools. J Oper Res Soc 48:1191–1200
7. Bonutti A, De Cesco F, Di Gaspero L, Schaerf A (2012) Benchmarking curriculum-based course timetabling: formulations, data formats, instances, validation, visualization, and results. Ann Oper Res 194(1):59–70
8. Cacchiani V, Caprara A, Roberti R, Toth P (2013) A new lower bound for curriculum-based course timetabling. Comput Oper Res 40(10):2466–2477
9. Cambazard H, Demazeau F, Jussien N, David P (2005) Interactively solving school timetabling problems using extensions of constraint programming. In: Burke E, Trick M (eds) LNCS, vol 3616, pp 190–207
10. Carter MW (2001) A comprehensive course timetabling and student scheduling system at the University of Waterloo. In: Burke E, Erben W (eds) LNCS, vol 2079. Springer, Berlin, pp 64–82
11. Costa D (1994) A tabu search algorithm for computing an operational timetable. Eur J Oper Res 76:98–110
12. Colorni A, Dorigo M, Maniezzo V (1992) A genetic algorithm to solve the timetable problem, Technical Report 90-060 revised, Politecnico di Milano Italy
13. de Werra D (1985) An introduction to timetabling. Eur J Oper Res 19:151–162
14. Di Gaspero L, McCollum B, Schaerf A (2007) The second international timetabling competition (ITC-2007): curriculum-based course timetabling (track 3). Technical report, School of Electronics, Electrical Engineering and Computer Science, Queens University, Belfast (UK), (site: http://www.cs.qub.ac.uk/itc2007/)
15. Dimopoulou M, Miliotis P (2001) Implementation of a university course and examination timetabling system. Eur J Oper Res 130:202–213
16. Geiger MJ (2009) Multi-criteria curriculum-based course timetabling—a comparison of a weighted sum and a reference point based approach. In: Ehrgott M, Fonseca CM, Gandibleux X, Hao JK, Sevaux M (eds) Proceedings of the 5th International Conference on Evolutionary Multi-criterion Optimization, EMO 2009, LNCS, vol 5467. Springer, pp 290–304
17. Geiger MJ (2012) Applying the threshold accepting metaheuristic to curriculum based course timetabling. Ann Oper Res 194(1):189–202
18. Gunawan A, Ng KM, Poh KL (2006) A mathematical programming model for a timetabling problem. In: Proceedings of the International Conference on Scientific Computing, Navade, USA
19. Hao JK, Benlic U (2011) Lower bounds for the ITC-2007 curriculum-based course timetabling problem. Eur J Oper Res 212(3):464–472
20. Hinkin R, Thompson MG (2002) Schedulexpert: scheduling courses in the Cornell University School of Hotel Administration. Interfaces 32(6):45–57
21. Junginger W (1986) Timetabling in Germany: a survey. Interfaces 16:66–74
22. Kiefer A, Hartl R, Schnell A (2014) Adaptive large neighborhood search for the curriculum-based course timetabling problem. Technical report UNIVIE-PLIS-2014-001 University of Vienna
23. Lach G, Lübbecke M (2012) Curriculum based course timetabling: new solutions to Udine benchmark instances. Ann Oper Res 194(1):255–272
24. Lü Z, Hao JK, Glover F (2011) Neighborhood analysis: a case study on curriculum-based course timetabling. J Heuristics 17(2):97–118
25. Lübbecke M (2015) Comments on: an overview of curriculum-based course timetabling. TOP 23(2):359–361
26. Mushi AR (2004) Mathematical programming formulations for the examinations timetable problem, the case of the university of Dar es salaam. African Journal Science and Technology, Science and Engineering Series 5:34–40
27. Papoutsis K, Valouxis C, Housos E (2003) A column generation approach for the timetabling problem of Greek high schools. J Oper Res Soc 54:230–238

28. Rudová H, Murray K (2003) University course timetabling with soft constraints. In: Burke E, De Causmaecker P (eds) LNCS, vol 2740. Springer, Berlin, pp 310–328
29. Schaerf A (1999) A survey of automated timetabling. Artif Intell Rev 13(2):87–127
30. Schimmelpfeng K, Helber S (2007) Application of a real-world university-course timetabling model solved by integer programming. OR Spectr 29(4):783–803
31. Tarawneh HY, Ayob M, Ahmad Z (2013) A hybrid simulated annealing with solutions memory for curriculum-based course timetabling problem. J Appl Sci 13:262–269
32. Tripathy A (1984) School timetabling, a case in large binary integer linear programming. Manag Sci 30:1473–1489
33. Tripathy A (1992) Computerised decision aid for timetabling - a case analysis. Discret Appl Math 35(3):313–323
34. Wren A (1996) Scheduling, timetabling and rostering, a special relationship? LNCS 1153:46–75. Springer, Berlin

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.