Computers and Operations Research 91 (2018) 178–189

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Benders' decomposition for curriculum-based course timetabling

Niels-Christian F. Bagger[a,b,1,2,*], Matias Sørensen[a,b,2], Thomas R. Stidsen[a,2]

[a] mORetime Research Group, Management Science, Department of Management Engineering, Technical University of Denmark, Produktionstorvet, Building 426B, DK-2800 Kgs. Lyngby, Denmark
[b] MaCom A/S, Vesterbrogade 48, 1., DK-1620 København V, Denmark

ARTICLE INFO

Article history:
Received 26 January 2017
Revised 17 October 2017
Accepted 18 October 2017
Available online 20 October 2017

Keywords:
University course timetabling
Integer programming
Benders' decomposition
Maximum flow
Minimum cut

ABSTRACT

In this paper we applied Benders' decomposition to the Curriculum-Based Course Timetabling (CBCT) problem. The objective of the CBCT problem is to assign a set of lectures to time slots and rooms. Our approach was based on segmenting the problem into time scheduling and room allocation problems. The Benders' algorithm was then employed to generate cuts that connected the time schedule and room allocation. We generated only feasibility cuts, meaning that most of the solutions we obtained from a mixed integer programming solver were infeasible, therefore, we also provided a heuristic in order to regain feasibility.

We compared our algorithm with other approaches from the literature for a total of 32 data instances. We obtained a lower bound on 23 of the instances, which were at least as good as the lower bounds obtained by the state-of-the-art, and on eight of these, our lower bounds were higher. On two of the instances, our lower bound was an improvement of the currently best-known. Lastly, we compared our decomposition to the model without the decomposition on an additional six instances, which are much larger than the other 32. To our knowledge, this was the first time that lower bounds were calculated for these six instances.

1. Curriculum-based course timetabling

In this work we considered the Curriculum-Based Course Timetabling Problem (CBCT) introduced in Track 3 of the Second International Timetabling Competition (ITC2007) as described by Di Gaspero et al. (2007), McCollum et al. (2010) and Bonutti et al. (2012). Most of the work on CBCT focused on the discovery of high-quality solutions using heuristics. The drawback of these heuristics is that they do not provide any proof of quality (e.g., how far from optimality the solutions actually are). We need bounding and exact methods to be able to validate the quality of the heuristics and not much work has been put into the development of these methods. In this article we applied Benders' decomposition to a Mixed Integer Programming (MIP) model that we presented in Bagger et al. (2016). We submitted the technical report (Bagger et al., 2016) to the Annals of Operations Research (ANOR). ANOR is, as yet, unaware of our work in this article as we had not fully developed the method. We provide the model from the report in Section 2.1. The proof of the correctness of the model is in the technical report which is essential, as the application of the decomposition in this paper relies on that model.

The CBCT problem entails that we must schedule weekly lectures for multiple courses into time periods and assign the lectures to rooms. We are given a set of days, each divided into a set of time slots. We refer to a day and time slot combination as a period. The basic entities of the problem are the *courses* to schedule, the *periods*, and the *rooms* that are available. The problem originates from a real world application and has thus received significant attention since the competition. Each course contains a number of lectures that must all be scheduled within a period, and assigned a room. Furthermore, all of the lectures are to be scheduled within distinct periods. This requirement is referred to as the Lectures (**L**) constraint. For a course, some of the periods can be specified as *unavailable*, i.e., periods where it is not allowed to schedule the course. This requirement is referred to as the Availability (**A**) constraint. There are no constraints on assigning the courses to rooms, i.e., any course can be assigned to any room. Aside from the courses, the periods and the rooms, the problem also contains *lecturers* and *curricula*; hence the name *Curriculum-Based Course Timetabling*. Each course is taught by a lecturer, and a curriculum is a set of courses that may be followed by the same students. If two

* Corresponding author at: mORetime Research Group, Management Science, Department of Management Engineering, Technical University of Denmark, Produktionstorvet, Building 426B, DK-2800 Kgs. Lyngby, Denmark.

E-mail addresses: nbag@dtu.dk (N.F. Bagger), ms@macom.dk (M. Sørensen), thst@dtu.dk (T.R. Stidsen).

[1] ORCiD: 0000-0003-4665-6761

[2] URL: http://www.moretime.man.dtu.dk

https://doi.org/10.1016/j.cor.2017.10.009
0305-0548/

courses are taught by the same lecturer, or belong to the same curriculum they cannot have lectures scheduled within the same periods. This requirement is referred to as the Conflicts (**C**) constraint. For each room, one lecture, at most, can be assigned in any period, which is referred to as the Room Occupancy (**RO**) constraint. The objective of the CBCT is to develop a timetable that fulfills all of the latter mentioned requirements, **L, A, C** and **RO**, while minimizing a weighted sum of the violation of four soft constraints; Room Capacity (**RC**), Room Stability (**RStab**), Minimum Working Days (**MWD**) and Isolated Lectures (**IL**). When a course is assigned to a room where the number of seats is smaller than the number of students that are attending the course, then the constraint **RC** is violated by one for each student above the capacity of the room. It is desirable to assign lectures from the same course to as few distinct rooms as possible. A course violates the constraint **RStab** by the total number of distinct rooms that it is assigned to minus one. The constraint **MWD** is the desire to spread the lectures across a given number of days. We say that a day is a working day for a course if at least one lecture from the course is scheduled in a time slot on that day. For each course, a number of minimum working days is provided and if the number of working days is below this number then the violation of the constraint is the difference. The final constraint **IL** is associated with the curricula. For each curriculum it is desired to have as few *isolated* lectures as possible. Each *isolated* lecture counts as one violation. A curriculum has an *isolated* lecture in a period if any of the courses belonging to the curriculum has a lecture scheduled in the period, and none of the courses have lectures scheduled in the *adjacent* periods. We say that two periods are adjacent if they belong to the same day and are in consecutive time slots.

In the following Section 1.1 we provide an overview of other approaches that were applied to CBCT. In Section 2 we initially provide a brief introduction to Benders' decomposition, and then describe how we applied it to CBCT. In Section 3 we describe a heuristic to repair partially infeasible solutions, where by partially infeasible we refer to solutions wherein the time schedule is feasible, but not the room assignment. In Section 4 we describe the computational results. Lastly, in Section 5 we state our conclusions on this work.

1.1. Related research

As we considered a MIP model for the problem, we focused primarily on other MIP-based approaches in the literature. For a thorough overview of the problem and different approaches for CBCT we refer to Bettinelli et al. (2015).

Burke et al. (2008; 2010) introduced a compact MIP formulation that was exact, in the sense that the optimal solution can be found by a generic MIP solver given enough computational resources. However, many instances of the CBCT could not be solved for this formulation within a reasonable time using a MIP solver; hence Burke et al. (2010) proposed methods to derive lower and upper bounds. They obtained lower bounds by aggregating the rooms into *multi-rooms*. For each *multi-room* the number of lectures that can be scheduled in it, for any period, is equal to the number of rooms that were aggregated. This problem provides a lower bound for CBCT. To obtain an upper bound they fixed the periods (or portions thereof) according to the solution from the lower bounding mechanism, and subsequently assigned rooms to the lectures. Burke et al. (2012) proposed an exact branch-and-cut algorithm which they also based on the compact formulation; however, some of the objective costs were left out and instead added as cuts during the solution process. This can be seen as a Benders' decomposition; however, rather than generating the cuts dynamically, they were generated a priori and then added as required.

Lach and Lübbecke (2008; 2012) proposed a method that divided the CBCT into two stages. The began by grouping the rooms together such that if two rooms had the same capacity, then they were in the same group. Then, in the first stage, they scheduled the courses into periods and assigned them to these capacities. This method is a Benders' Decomposition (Lübbecke, 2015). In the second stage, they assigned the courses to the rooms, where the solution from the first stage was employed to fix the courses for the determined periods and the selected room capacities.

Hao and Benlic (2011) divided the MIP model that Lach and Lübbecke (2012) used in the first-stage into smaller components by relaxing or removing some of the constraints. These relaxations made it possible to decompose the model into a set of sub-problems, where they calculated a lower bound for each subproblem. The sum of all these lower bounds was then a lower bound for CBCT.

Cacchiani et al. (2013) presented multiple extended MIP formulation, i.e., models with an exponential number of variables. The approach that provided the best results divided the problem into two parts; one that focused on the time scheduling-related soft constraints, while the other focused on the room-related soft constraints. They calculated a lower bound for each part and the sum of these lower bounds was then a lower bound for CBCT.

In Bagger et al. (2016) two MIP models were presented that were inspired by Lach and Lübbecke (2008; 2012) and Burke et al. (2008; 2010). The division of the problem described by Lach and Lübbecke (2012) was applied (excluding the notion of distinct capacities), where it was shown that the two stages could be connected using two underlying flow network formulations. The first formulation was based on a minimum cost flow network, and performed as the best of the two. The second formulation was based on a multi-commodity flow problem which was the formulation where we applied Benders' decomposition (Benders, 1962) in this article. The rationale for using the last formulation, although it did not perform as well as the first, is that the underlying network is a feasibility problem. Therefore, we did not need to generate any optimality cuts, only feasibility cuts.

2. Benders' decomposition

In this section we give an introduction to Benders' decomposition followed by our application of the technique. Our introduction is a crude overview, and we refer to (Benders, 1962) and (Martin, 1999, chapter 10) for a detailed description. We describe the method based on a model that contains two types of variables, x and y. The x variables are non-negative continuous variables, and we do not have any assumptions on the y variables, i.e., $x \geq 0$ and $y \in Y$ where Y can be any domain (e.g., the set of integers). Consider the MIP model (1).

$$\begin{aligned} \min \quad & c^\top x + f(y) \\ \text{s.t.} \quad & Ax + B(y) \geq b \\ & y \in Y \\ & x \geq 0 \end{aligned} \tag{1}$$

In the model (1) $c \in \mathbb{R}^n$ is the cost vector of the x variables, $A \in \mathbb{R}^{n \times m}$ is the constraint matrix of the x variables and $b \in \mathbb{R}^m$ is the right-hand-side vector of the constraints. $f : Y \to \mathbb{R}$ is some function to evaluate the cost of the y variables and B is a vector function that evaluates the contribution of the y variables for the constraints. If we fix the y variables to some value in the domain Y then what remains is a linear programme (LP). This assumption can be extended as described by Geoffrion (1972), but we stick to the (LP) case in this context. Model (1) can be rewrit-