Noname manuscript No.
(will be inserted by the editor)

Comments on: An overview of Curriculum-Based Course Timetabling

Andrea Schaerf

Received: date / Accepted: date

1 Introduction

I think that the paper provides a very good and well organized overview of the research on the Curriculum-Based Course Timetabling (CB-CTT) problem.

Up to my knowledge, all the relevant literature on the specific problem has been discussed. The literature on educational timetabling in general has not been considered, but I believe that it would broaden the scope of the paper too much; therefore I agree with the selection of references made by the authors.

In Section 2, I discuss a few topics about the problem, upon which I would like to contribute some additional information to the picture provided by the authors. In Section 3, I bring up some minor criticisms to the paper and its organization.

2 Discussion

In this section, I list the aspects of the problem that I believe might benefit from some additional remarks.

2.1 Problem formulation

The real-world course timetabling problem is rather complex and also different from university to university. Therefore, in order to keep the formulation simple enough, a strong selection of features had to be done in the definition of the "standard" problems (McCollum et al, 2010; Bonutti et al, 2012). In some

A. Schaerf
DIEGM, University of Udine
Tel.: +39-0432-558280
E-mail: schaerf@uniud.it

cases, the selection has been done at the expenses of the faithfulness to the real-world model.

For example, the "curriculum compactness" component is not modeled in the most realistic way, but in this case simplicity of evaluation has been privileged. Similarly, the "room stability" component is not particularly significant in practical applications, therefore it could have been neglected, in favor of other components. However, it was included because the organizers believed that it could be a good representative of a group of room-related constraints. Without this component, it would have been possible to resort to techniques that take rooms into consideration only indirectly, for example by using a *matching* procedure as a post-processing phase. Such techniques would have been effective for the problem at hand, but difficult to be extended to the real-world ones.

2.2 Benchmark instances

The most used benchmark dataset are the `comp*` instances, that have been collected from data of University of Udine. These instances have been criticized by Phillips et al (2015) for having peculiar features, that make them not representative enough of normal cases. I would like to reply to these criticisms one by one. Below, I report the list of concerns (in quotes) along with my explanation for the actual data in our university.

- "*Courses in the ITC problems frequently have an extremely high number of events*". Some courses are actually practical sessions in which the students work autonomously, and the teacher supervise their activity. They can easily run for 14 hours per week as in the data. What is missing here is the constraint that they have to be as consecutive as possible.
- "*Studies into the utilisation of teaching space at real universities (see Beyrouthy et al, 2009) suggest that rooms are occupied 50% of the time on average, rather than the 60%-80%, which is typical for the ITC problems*". Unfortunately, the occupation of 80% is absolutely normal at University of Udine, and I believe in many other universities. This high occupancy makes more crucial the use of a software tool implementing an effective optimization technique.
- "*We also find that the scale of ITC problems varies between small to medium size problems, but does not cater to problems faced by the very largest institutions. The largest ITC problem [...] is significantly smaller than the problem faced by the University of Auckland*". This is the typical size of a faculty or a department in Italy, which is normally responsible of its own timetable.
- "*We find that using a soft limit for room capacity, is less realistic than a hard limit*". This choice is related to the uncertainty in the number of students that actually take the course, in comparison with the nominal one. A hard constraint would be too strict in our context.

2.3 Instance generator

The paper discusses about instance generators, and in particular the ones developed by Mareček (see Burke et al, 2010) and by Lopes (see Lopes and Smith-Miles, 2010), which are available on the web.

I would like to mention our experience with the latter one, which has been very positive. In fact, as reported in Bellio et al (2014), a tuning procedure that uses only generated instances and no real ones, has been able to obtain comparable or better results with those tuned (maybe overtuned) on the competition instances.

However, some selection of the instances turned out to be necessary, given that not all generated ones turned out to assume realistic values for the features. In particular, some instances turned out to be either infeasible, or too easy, or unrealistic in terms of the ratio between room capacity and students attending the courses. These behaviors, however, could be easily detected, and the corresponding instances quickly eliminated from the experimental pool.

3 Minor criticisms

3.1 Connection with other problems

In my opinion, the placing of the problem in the area of course timetabling could have been explained in more details.

The paper highlights that the main difference between CB-CTT and PE-CTT (the other "standard" formulation) regards the source of the conflict matrix, which is the set of curricula in one case and the enrolment records in the other. However, this is not the only difference between the two, and there are other remarkable distinctions. An important one is that, in PE-CTT, each course is a single event, whereas in CB-CTT a course consists of a series of lectures. As a result, the set of constraints and objectives is very different. For example, for PE-CTT, the approach of assigning periods and rooms in separate stages mentioned above is more successful than for CB-CTT.

Another important observation about CB-CTT is that it does not involve *student sectioning* (Müller and Murray, 2010, see, e.g.), which is an important step in course timetabling, that has been neglected in CB-CTT. As for other features, the reason is that it would have made the problem too complex.

3.2 Constraint satisfaction techniques

I believe that placing constraint satisfaction techniques (Section 4.3) within the section dedicated to heuristics (Section 4) is not accurate, given that the algorithms normally used for solving constraint satisfaction problems are actually exact ones.

In particular, the contributions hosted in this section should in my opinion be moved to other sections, and thus the section itself can be removed.

In fact, the contribution by Atsuta et al (2008) can be moved in Section 4.2.1 given that their underlying technique is actually Tabu Search. The contribution of Banbara et al (2013) on Answer Set Programming could be accommodated in Section 3 as ASP techniques are exact methods. Finally, the description of the heuristic contribution of Asín Achá and Nieuwenhuis (2014) could be merged with description of its contribution on exact methods, given in Section 3.6.

References

Asín Achá R, Nieuwenhuis R (2014) Curriculum-based course timetabling with SAT and MaxSAT. Annals of Operations Research 218:71–91

Atsuta M, Nonobe K, Ibaraki T (2008) Itc-2007 track2: an approach using general csp solver. Available at `http://www.cs.qub.ac.uk/itc2007/`

Banbara M, Soh T, Tamura N, Inoue K, Schaub T (2013) Answer set programming as a modeling language for course timetabling. Theory and Practice of Logic Programming 13(4-5):783–798

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. CoRR abs/1409.7186, URL `http://arxiv.org/abs/1409.7186`

Beyrouthy C, Burke EK, Landa-Silva D, McCollum B, McMullan P, Parkes AJ (2009) Towards improving the utilization of university teaching space. Journal of the Operational Research Society 60(1):130–143

Bonutti A, De Cesco F, Di Gaspero L, Schaerf A (2012) Benchmarking curriculum-based course timetabling: formulations, data formats, instances, validation, visualization, and results. Annals of Operations Research 194(1):59–70, DOI 10.1007/s10479-010-0707-0

Burke E, Mareček J, Parkes A, Rudová H (2010) A supernodal formulation of vertex colouring with applications in course timetabling. Annals of Operations Research 179(1):105–130

Lopes L, Smith-Miles K (2010) Pitfalls in instance generation for Udine timetabling. In: Learning and Intelligent Optimization (LION4), Springer, pp 299–302

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

Müller T, Murray K (2010) Comprehensive approach to student sectioning. Annals of Operations Research 181(1):249–269

Phillips AE, Waterer H, Ehrgott M, Ryan DM (2015) Integer programming methods for large-scale practical classroom assignment problems. Computers and Operations Research 53:42–53