Ann Oper Res
DOI 10.1007/s10479-011-1012-2

XHSTT: an XML archive for high school timetabling problems in different countries

Gerhard Post · Jeffrey H. Kingston · Samad Ahmadi · Sophia Daskalaki · Christos Gogos · Jari Kyngas · Cimmo Nurmi · Nysret Musliu · Nelishia Pillay · Haroldo Santos · Andrea Schaerf

G. Post (✉)
University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: g.f.post@math.utwente.nl

G. Post
ORTEC, Groningenweg 6k, 2803 PV Gouda, The Netherlands

J.H. Kingston
School of Information Technologies, The University of Sydney, Sydney, Australia

S. Ahmadi
Dept. of Informatics, De Montfort University, The Gateway, Leicester LE1 9BH, UK

S. Daskalaki
Engineering Sciences Department, University of Patras, 26500 Rio Patras, Greece

C. Gogos
Department of Finance and Auditing, Technological Educational Institute of Epirus, Psathaki, 48100 Preveza, Greece

J. Kyngas · C. Nurmi
Satakunta University of Applied Sciences, Tiedepuisto 3, 28600 Pori, Finland

N. Musliu
Vienna University of Technology, Favoritenstraße 9, 1040 Vienna, Austria

N. Pillay
School of Computer Science, University of KwaZulu-Natal, Private Bag X01, Scottsville, 3209 Pietermaritzburg, South Africa

H. Santos
Computing Department, Universidade Federal de Ouro Preto, Ouro Preto, Brazil

A. Schaerf
DIEGM, University of Udine, via delle Scienze 206, 33100 Udine, Italy

Published online: 10 November 2011

Abstract We present the progress on the benchmarking project for high school timetabling that was introduced at PATAT 2008. In particular, we announce the High School Timetabling Archive XHSTT-2011 with 21 instances from 8 countries and an evaluator capable of checking the syntax of instances and evaluating the solutions.

Keywords Timetabling · High school · Benchmark · XML · Scheduling

1 Introduction

"It is surprising that no standard format for exchanging datasets in the field of high school timetabling has emerged until now." This sentence was the motivation for a group of researchers to define a format capable of expressing high school timetabling instances from all over the world, see Post et al. (2011).

The high school organization is different around the world, consequently the problems in high school timetabling that arise from real cases in various countries differ as well. As a specific example, one of the main differences that has emerged from our research is related to allowing idle times for students during school hours versus the cases where this is not allowed. In the first case, teachers are usually not preassigned to the lessons, as this may lead to infeasibilities. In the second case, teachers are mostly preassigned, leading to the problem of eliminating idle times for students and minimizing them for teachers.

Another important difference is related to the granularity of the scheduling process: sometimes it is performed at the level of an entire class, whereas in other cases of a single student. In the latter case, the problem usually becomes harder, since the schedule of each individual student has to be evaluated during the solving process, thus making the process computationally more expensive.

Similar standardization processes are going on also for the two other mainstream in educational timetabling, namely examination timetabling and course timetabling. For these problems, however, the process has followed a less guided and thus more complex pathway.

For the examination timetabling problem, the formulation proposed by Carter et al. (1996), along with the corresponding benchmark set proposed by Carter himself, has become a sort of standard *de facto*. Such a formulation is a limited one, supplied by means of a plain text-only file format. In order to capture real-world problems, the afore-mentioned formulation has been subsequently extended by several authors. At present, the most complex (and real-world) formulation available is the one described in McCollum et al. (2007) and used for the second international timetabling competition (ITC 2007); it is still based on text-only files McCollum et al. (2010). The definition of a general and comprehensive formulation and XML file format is still to come.

Regarding the course timetabling problem, unfortunately, there is still no general consensus about the standard formulation and data format. Nevertheless, thanks also to the ITC 2007, two formulations have emerged as the most developed and investigated. They are the so-called *Post-Enrollment Course Timetabling (PE-CTT)* and the *Curriculum-based Course Timetabling (CB-CTT)*, for which many instances and results are available in the literature. For the latter problem, the instance files are available also in XML format, along with the DTD file, see Bonutti et al. (2010). The XML format, though much less general than the one described here, includes the data for various variants of the CB-CTT problem.

The purpose of this paper is to report on the progress of this collaborative research in high school timetabling and reflect on the current situation; we will give a short overview and motivation of the XML format in Sect. 2, discuss the current archive (XHSTT-2011) in Sect. 3, the evaluator (HSEval) in Sect. 4, and give an outlook to the future in Sect. 5.

```
<Instance Id="Example">
  <Times>
    <TimeGroups>
      <Day Id="Day1"/> <Name>Monday</Name> </Day>
      ...
      <Day Id="Day5"/> <Name>Friday</Name> </Day>
      <TimeGroup Id="AllTimes"/> <Name>AllTimes</Name> </TimeGroup>
      ...
    </TimeGroups>
    <Time Id="Day1_1">  <Name>Monday 1</Name>
      <TimeGroups>
        <Day Reference="Day1"/>
        <TimeGroup Reference="AllTimes"/>
      </TimeGroups>
    </Time>
    ...
  </Times>
  <Resources>
     ...
  </Resources>
  <Events>
     ...
  </Events>
  <Constraints>
     ...
  </Constraints>
</Instance>
```

Fig. 1 A problem instance in the XML format

2 The format

Differences in the organization of high schools in different countries imply that the definition of a unified format for high school timetabling is not a trivial task. The current format has emerged after many iterations; indeed the format discussed in Post et al. (2011) differs considerably from the original version presented at PATAT 2008.

The format of our benchmark is mapped out by an XML schema which defines the compulsory and optional elements that need to be present in the XML files holding the instances and solutions. The basic elements of an instance are the *times*, *resources*, and *events*, complemented by the *constraints*, which are imposed on them. We believe that the structure of the format as it is now will essentially remain the same over time. The reason lies in the principal choice to embed the "business logic" in the constraints, and not in the basic elements. At present, the format contains (only) 15 constraint types, including "obvious" ones, like *AssignTimeConstraint* (assign a start time to selected events) and *AvoidClashesConstraint* (a resource may be involved with at most one event at a time). The modular nature of the schema assures that new constraints can be added without having to change its structure. Indeed, we believe that the set of constraints will probably be extended further to incorporate new specialized constraints to deal with unforeseen problems in other countries.

A fragment of an instance file is shown in Fig. 1. All objects have the attribute *Id* for referencing, and the child *Name* for displaying. *Times*, *Resources* and *Events* can be grouped in *TimeGroups*, *ResourceGroups* and *EventGroups*, respectively. For demonstration purposes the *Times* section is expanded with some more detail.

One of the main discussions during the design of this format was about the "domain specific" structure and the "solver needed" structure. The "domain specific" structure reflects

how a timetabler at a school has structured the data; for example, a timetabler will distinguish days of the week, will think in terms of students, classes or teachers (certainly not of general resources), and will consider courses and subjects as principal objects for scheduling. The solver instead requires a structure organized in terms of variables which represent units of lessons or resources, while conceptual entities are not important.

In the format, although some of the domain specific structural elements (for example "days" and "courses") are supported and their use is recommended, it is not obligatory to use them. We mention the two most important ones. A *Day* is introduced as a time group, that almost certainly will be needed, because in high school timetabling there are many constraints at a daily level. For example, constraints about "working hours", "idle times", and "number of days present". In addition, by introducing days we are able to display daily schedules for the resources. Another element is the *Course*, which is introduced as an *EventGroup* for a subject and a student group combination. This is important in order to control the spreading of the individual lessons (events) of a course over the week days. *Courses* also allow control on events with similar properties; if certain events are identical, they can be clustered to one event and allow the lessons to be sub-events. Moreover, constraints like the *SplitEventsConstraint* can prescribe how such an event should be split into sub-events.

Our view on inclusion of new constraints has changed during the past three years. Originally we tried to include all the constraints that we encountered in the literature, or that we could imagine to be useful. On the contrary, the current set of constraints in the format reflect only the constraints needed by the contributors, and no more. The reason for this is that complicated constraints usually need to be clarified by an expert.

3 The archive XHSTT-2011

At the website Post et al. (2008) the archive XHSTT-2011 with 21 instances from 8 different countries is available. Most instances have appeared previously in the literature, but were not available for download. Apart from the instances, solutions to most instances are also available. We keep record of the best found solutions so that researchers are able to validate their solvers. In Table 1 we present the instances that have been contributed. In the columns are given

- the country (Country);
- name of the instance (Name);
- total duration of all events (EvD);
- the number of teachers (T);
- the number of classes (C);
- the number of students (St);
- the number of rooms (Ro).

Note that the instances vary significantly in size. Most instances are described at the level of classes, which might split further to form sub groups, see for example the High school instance of Finland. The Dutch instances, however, carry information at the level of individual students as well. For the lower grades the groups of students (classes) are fixed and all students of a group attend most lessons together. Conversely, for the higher grades the timetable of each student is mostly personal, since the compulsory lessons constitute only one third of the lessons. For the Australian instances, the teachers have to be assigned as well in the timetabling process and in such case split assignments should be avoided.

Table 1 The instances in the archive XHSTT-2011

Country	Name	EvD	T	C	St	Ro
Australia	BGHS98	1564	56	30		45
	SAHS6	1876	43	20		36
	TES99	806	37	13		26
Brazil	Instance 1	75	8	3		
	Instance 4	300	23	12		
	Instance 5	325	31	13		
	Instance 6	350	30	14		
	Instance 7	500	33	20		
England	St Paul	1227	68	67		67
Finland	Artificial	200	22	13		12
	College	854	46	31		33
	High school	319	18	102		13
	Secondary school	306	25	14		25
Greece	High school 1	372	29	66		
	Patras 3^{rd} HS	340	29	57		
	Preveza 3^{rd} HS	340	29	53		
Italy	Instance 1	133	13	3		
Netherlands	GEPRO	2675	132	44	846	80
	Kottenpark 2003	1203	75	18	453	41
	Kottenpark 2005	1272	78	26	498	42
South Africa	Lewitt 2009	838	19	16		2

4 The evaluator

Apart from the instances, the format also models solutions. A solution is presented by describing the duration of all events (as mentioned previously it is possible to define a "course" event of duration 3, which can be split into, for example, three sub-events of duration 1), the times assigned to each event, and (in some cases) the resources assigned to events. Once a solution is provided we can then evaluate it. The evaluation leads to two integers: the infeasibility value (i.e. the total cost of the hard constraints violations) and the objective value (the total cost of the soft constraints). The total cost is generated from two different constraint types: if the constraint is "hard", then the cost is added to the infeasibility value, otherwise to the objective value. Depending on the type of the constraint, the cost is attributed to: an event (for example: "is there a time assigned to the event?"), to an event group (for example: "is the course well-spread?") or to a resource (for example: "are the idle times within the given limits?"). The cost of the schedule is the sum of all separate costs. The cost value $V(C, R)$ for resource R and constraint C with weight λ_C and cost function f_C is calculated from the deviation $D_{(C,R)}$ of constraint C for resource R by the formula:

$$V(C, R) = \lambda_C \cdot f_C(D_{(C,R)})$$

The cost functions supported currently are: step function, linear function, and quadratic function. Quadratic functions are useful for spreading the inevitable violations: it prefers twice a deviation of 1 above a deviation of 0 (i.e. no deviation) and a deviation of 2.

The format supports multiple instances and multiple groups of solutions to these instances:

```
<HighSchoolTimetableArchive>
   <Instances>
      <Instance Id="Instance1">
         ...
      </Instance>
      <Instance Id="Instance2">
         ...
      </Instance>
      ...
   </Instances>
   <SolutionGroups>
      <SolutionGroup>
         <Solution Reference="Instance1">
            ...
         </Solution>
         <Solution Reference="Instance1">
            ...
         </Solution>
         ...
      </SolutionGroup>
   </SolutionGroups>
</HighSchoolTimetableArchive>
```

In benchmarking an indisputable interpretation of the data and constraints is essential. For this, an evaluator and documentation were developed and made accessible at Kingston (2009). The task of the evaluator is three-fold: first it checks if the provided instances and solutions satisfy the syntax rules. This includes checking consistency of the used Ids and whether a solution respects the *preassignments*. The second task is to provide the infeasibility value and the objective value of the solution and, if indicated, a full report on the deviations for all constraints. Finally, if several solutions of the same instance are included, a comparison table is presented. The first two parts are very useful for the implementation as they provide the user with checks on the generated format and implementation of the constraints. In this sense the evaluator is the ultimate documentation: either the result of the evaluator is accepted, or the behaviour of the evaluator is marked as "bug".

Some cases led to discussions of the interpretation. One example is the constraint *LimitBusyTimesConstraint*, which limits the busy times of a resource on a day between a minimum and a maximum. Initially this constraint generated the cost for the days without work too. In the revised implementation these days are skipped. This is reasonable, since by using another constraint one can describe the number of days a resource should be busy.

5 The future

After the past and the present, let us make some speculations about the future, based on our experiences till now. First of all we have noted that there is a great interest in this format; interest from researchers in high school timetabling, but also from other areas of timetabling. In our opinion this shows that many researchers feel the urge for exchangeable datasets. Our vision in building this data format is our belief that efforts to define very general formats right from the start have a great chance to fail. We believe that the current format keeps a good balance between tangibility and abstraction.

Though several researchers have expressed their interest, converting their formats to the proposed format requires time. In view of this we are proud that we can present an archive with 21 datasets from 8 contributing groups. By active acquisition and support we hope to extend this even further in the near future. Moreover, attention will be paid to the functional properties: the format is well-defined, the evaluation is clear, but the semantics might be different from expected.

Another initiative is the development of a automated repository for High School timetabling. This repository will have facilities to convert data sets to the standard data format, uploading new data sets, download of existing data and use of the evaluator. A work in progress version of this site is available at Ahmadi and Rorije (2010).

If new contributors appear, new constraints or variants of the current constraints may be needed. One type of constraint concerns the sequencing of events. An example of this is when the events take place at different locations: in such cases one would like to minimize the number of location changes, or have breaks or idle times in between. Another example could be a sequence of the subjects like Mathematics, Physics, Chemistry, and Biology for a group of students.

Though new constraints will be needed, one should keep in mind that an instance is usually just an approximation of practice. The timetabler at school will have a clear view of the schedules, but to formalize this in constraints is not always easy (or interesting). In practice hard constraints can turn out to be soft, if necessary, while giving weights to the soft constraints can be difficult. The violation of some important soft constraints can turn out to be unacceptable to the timetabler. The timetabler will rather change the data, and try again to find a good solution. On one hand this might be discouraging for the researchers, but on the other hand it is always a challenge to cope with the reality.

References

Ahmadi, S., & Rorije, B. (2010). High school timetabling problem repository. http://opt-kd.cse.dmu.ac.uk/www/.

Bonutti, A., De Cesco, F., Di Gaspero, L., & Schaerf, A. (2010). Benchmarking curriculum-based course timetabling: formulations, data formats, instances, validation, visualization, and results. *Annals of Operations Research* (to appear). doi:10.1007/s10479-010-0707-0.

Carter, M. W., Laporte, G., & Lee, S. Y. (1996). Examination timetabling: algorithmic strategies and applications. *The Journal of the Operational Research Society*, *74*, 373–383.

Kingston, J. H. (2009). The HSEval high school timetable evaluator. http://www.it.usyd.edu.au/~jeff/hseval.cgi.

McCollum, B., McMullan, P., Burke, E. K., Parkes, A. J., & Qu, R. (2007). In *The second international timetabling competition: examination timetabling track*. Queen's University, QUB/IEEE/Tech/ITC2007/Exam/v4.0/17.

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

Post, G. (2008). Benchmarking project for (high) school timetabling. http://www.utwente.nl/ctit/hstt.

Post, G., Ahmadi, S., Daskalaki, S., Kingston, J. H., Kyngas, J., Nurmi, C., & Ranson, D. (2011). An XML format for benchmarks in high school timetabling. *Annals of Operations Research* (to appear). doi:10.1007/s10479-010-0699-9.