School Timetabling: from a Great Number of Combination to a Reasonable Number of Appropriate Choices

I.V.Gadolina (Dept. of Mechanical Design)

Key words: Timetable, Optimization, Random choice, Matrix operations.

ABSTRACT: A method for school timetable creation using optimization of teachers' preferences is proposed. There are three stages: 1) Optimization of teachers' preferences; 2) Timetabling with a complete randomized procedure; 3) Optimization of timetables, using the system of costs. A proposed approach to optimization, allowing to restrict the number of the options during optimization, create a great opportunity for a successful timetabling in different situations under any particular restrictions in educational institutions.

Nomenclature

S	number of lessons, which should be timetabled;
TT	timetable
$k=1,2...k_{max}$	period number through all the week;
$i=1,2....i_{max}$	class number;
$d=1,2.... d_{max}$	the numbers of subjects at school;
$j=1,2... j_{max}$	teacher's number;
{A}	the set of appropriate teachers preferences;
{TT*}	the set of 'possible' timetables (feasible solutions);
{TT}**	the set of 'good ' timetables.

1. Foreword

No need to explain the importance of a good timetable for successful school management. Not only a teaching process suffers if the timetable is far from being good. It may cost a lot of money. If for example, we could not keep classrooms and labs fully occupied, soon we can experience the rooms shortage with all the bad consequences.

How to create a good computer program, which could automatically construct timetables for educational institutions? In general, this problem in Operation Research science is still unsolved [(1)]. Despite of the enormous calculation speed of the computers, people possess some heuristics and that allows human to win the race with a computer at that point.

Let **{TT*}** be the set of feasible solutions and the set **{TT**}** be the set of 'good' solutions, which also satisfy a number of requirement, which can vary from school to school, **{TT*}** $\subset$ **{TT**}.** In complicated real situation in educational institutions even any **TT** $\in$ **{TT*}** is not so easy to construct. However our own experience as the teacher (schools are working) gives us an optimistically hope, that the decision **TT** exists. Our disappointments about timetables at the beginnings of the semesters and sometimes successful classes swaps just among the teachers give us a tip that the decisions is not unique.

Here we propose a decision of a timetabling problem using the optimization of teachers' preferences. That approach can be described as follows:

1) Optimization of teachers' preferences. Construct a set of the appropriate teachers' preferences **{A}.** Method: randomization + matrix operations.

2) On the basis of **{A}** construct a set of feasible solutions {**TT***} such $\forall$ TT $\in$ {**TT***} should satisfy the imposed constrains. Method: randomization.

3) Timetable optimization. On the basis of {**TT***} construct a set of 'good' solutions {**TT****}**,** such so $\underset{\forall TT \in \{TT^*\}}{f(TT) < \delta}$, where f(TT) is the function of 'notional costs' of a timetable [(1)]. It is not financial, but just a measure. The better the timetable, the lower the cost.

2. Using matrix to describe the data and solutions

As we will see many of the timetable constrains can be considered easily in terms of matrix equations. Matrixes are also appeared to be helpful in describing the initial data, and the field of solutions.

Let's emphasize the timetabling problem as a construction of a non-conflict binary matrix **TT**= (tt_{ijk}), i=1,2 .... i_{max}; j=1,2 ..... j_{max}; k=1,2 ..... k_{max}, where tt_{ijk}=1 means that the teacher j teaches class i in the period k and tt_{ijk}=0 otherwise.

The main requirement here is **TT**'s adequacy to a teaching plan and its correspondence to the plan for classes to study. The latest is described in binary matrix **CL**:

$$\mathbf{CL}=(cl_{ik}),\ i=1,2 \ldots i_{max};k=1,2 \ldots k_{max}. \qquad (1)$$

Here cl_{ik}=1 means that class i studies in the period k and cl_{ik}=0 otherwise. Sums $\sum_{k=1}^{k_{max}} cl_{ik};\ i = 1,2 \ldots i;$ shows how many lessons class i has during the week.

Using data in **CL** it is possible to calculate the total number of classes, which study in period k :

$$\mathbf{CLS}= \mathbf{CL}^* \mathbf{E}, \qquad (2)$$

where **E** is a column vector of ones, **E**=(e_{i1}), e_{i1}=1; i=1,2 .... i_{max}. **CLS** is the row vector, **CLS**=(cls_{1k}), k=1,2 ... k_{max}. The numbers $cls_{1k} \in$ {**Z**} show how many classes study in period k. ({**Z**} is the set of natural numbers).

Matrix **SPL** describes the school plan:

SPL=(spl_{id}), i=1,2 .... i_{max}; d=1,2.... d_{max},. Numbers $spl_{id} \in$ {**Z**} show how many lessons in subject d class i should have.

Often however before timetabling it is already known which teacher should teach each class. In this case it will be more convenient to use matrix **TPL** instead of **SPL**. **TPL** is the teachers' plan matrix:

TPL=(tpl_{ij}), i=1,2 .... i_{max}; j=1,2 ..... j_{max}, Here also $tpl_{id} \in$ **{Z}.** (3)

Matrix-indicator **ITPL** is a binary-matrix of the same dimension as **TPL**. Here $itpl_{ij}$=1, if $tpl_{ij} \neq 0$; $itpl_{ij}$=0 otherwise.

The sums of elements in **CL**, **SPL** and **TPL** are equal

$$\sum_{i=1}^{i_{max}} \sum_{k=1}^{k_{max}} cl_{ik} = \sum_{i=1}^{i_{max}} \sum_{d=1}^{d_{max}} spl_{id} = \sum_{i=1}^{i_{max}} \sum_{j=1}^{j_{max}} tpl_{ij} = S$$

Here S is a total number of lessons, which should be timetabled. After successful timetabling we should also have

$$\sum_{k=1}^{k_{max}} \sum_{i=1}^{i_{max}} \sum_{j=1}^{j_{max}} tt_{kij} = S$$

To avoid the 'teachers clashes', that means the occasion when the teacher supposed to teach two different classes at the same time, and 'class clashes', that means a situation, when a class has two different lessons timetabled at the same time, **TT** should satisfy requirements:

$$\sum_{j=1}^{j_{max}} tt_{kij} \leq 1, \quad i=1,2 \dots i_{max} \text{ and } k=1,2 \dots k_{max} \text{ and}$$

$$\sum_{i=1}^{i_{max}} tt_{kij} \leq 1, \quad j=1,2 \dots j_{max} \text{ and } k=1,2 \dots k_{max}. \qquad (4)$$

3. Optimization. Choosing appropriate combinations of the teachers' preferences

In educational institutions the points, where the most severe conflicts occur may vary. In situation of the large school with a great number of teachers, and some of them having additional jobs, the most difficult problem is to avoid the undesirable teachers' breaks between classes. Usually before timetabling the manager asks the teachers to fill in the list of the preferences. To have more flexibility for our heuristic search imagine that each teacher fills in 4 lists of preferences (four choices), different at some extend. (For example: the teacher could be interested in a research day either on Monday or on Tuesday, no difference to him). All these data are stored in the binary matrix **BTF,** which is a bank of teachers preferences.

BTF=(bpf_{njk}), n=1,2,3,4; i=1,2 .... i_{max}; k=1,2 ..... k_{max}, where n is a number of individual choice. Here bpf_{njk}=1, if the teacher j does not mind teaching in period k according to his n-th list of the preferences; bpf_{njk}=0 otherwise.

To have more flexibility the lists of teachers preferences should have some reasonable reserve:

$$\sum_{j=1}^{j_{max}} \sum_{k=1}^{k_{max}} bpf_{njk} \geq S \quad \forall n = \{1,2,3,4\}^{*}$$

The field for optimization is a field of random choices from **BTF**. If each teacher gives the manager even a little freedom in his four preferences lists, all teachers' choices taken together provide a good opportunity for optimization. Using data in **BTF** we could construct a great number of possible choices for school. The total number of choices is $B=4^{j_{max}}$. So, if there are 17 teachers in school (j_{max}=17), B>2*10^{10} and for greater j_{max} the number of combinations is can be said is not restricted. We do not need so many. To find an appropriate combination of choices we will use the random choice without replacement.

Let *b* be the order number of random choice among all possible B choices, $1 \leq b \leq B$. Here for each teacher *j*=1,2 ..... j_{max}, we select independent and identically distributed random numbers *n*={1,2,3,4}. The corresponding data are taken from **BTF** and presented in a random matrix, depended on b, namely **PF**(b).

PF(b)=(j $pf(b)_{jk}$), *j*=1,2 ..... j_{max}; *k*=1,2 ..... k_{max}.

Here $pf(b)_{jk}$ =1 means that the teacher j could teach in period *k* according the list of common preferences *b,* $pf(b)_{jk}$ =0 otherwise.

We did not use any heuristics so far and the probability of the single choice *b* to be good is very low **. Again, we are going to have a big number of **PF**(*b*) and this provide as an opportunity to get a number of the appropriate combinations. Using the data in **PF**(*b*) together with **CL** and **ITPL** we can eliminate these bad combinations.

Verification 1 . Checking the total number of the teachers available at school.

Likewise we estimate the number of classes which study in period *k* (2), we can calculate the number of teachers available in the period k:

TA(b)= **E** ***PF**(*b*); where **E** is column vector of ones, **E**=(e_{j1}), e_{j1}=1 ; *j*=1,2 .... j_{max};

TA(*b*) is the row vector, **TA**(*b*) =($ta(b)_{1k}$), *k*=1,2 ... k_{max}.

The numbers $ta(b)_{1k} \in$ ***{Z}*** show how many teachers available in period *k* according to the choice *b.* If $ta(b)_{1k} < cl_{1k}$ for any *k* =1,2 ... k_{max}, that means that **PF**(*b*) is an unreal choice, because there are fewer teachers than the classes. Here we should stop and make different random choice *b* among the rest [1,B].

---

* Some reasonable correction of preferenses should have been done here.

If we want to avoid the stortgage of labs, the preferanses of the teachers j∈ {j*}, where {j*} the set of teachers who teach in one laboratory one after another should be complementary:

$$\sum_{\forall j \in j^*} pf_{jnk} \leq 1,\ k = 1,2....k_{max};\ \forall n = \{1,2,3,4\}.$$

If the teacher teaches a subject, which requares two consecutive slots, he/she should consider the sum in his preferenses list in one day to be even.

** An example of the unreal combination: imagine that in according to the choice *b* all the teachers in school are going to have Monday as a day off. What should students do on Monday?

Verification 2. Checking the average number of the teachers, who can teach classes.

Since we have teachers' plan **TPL,** not each teacher can teach any class. Binary matrix **AVI**=(avi_{ik}), $i=1,2$ .... i_{max}; $k=1,2$ ..... k_{max} shows if any teachers available for class i in accordance with matrix-indicator **ITPL** *b*-choice:

AVI = **ITPL* PF**(b).

If $avi_{ik}=0 \cup cl_{ik}\neq 0$; $i=1,2$ .... i_{max}; $k=1,2$ ..... k_{max} that means the choice **PF**(*b*) is unreal because of absence any particular teacher for class i in period k when the class is supposed to study ($cl_{ik}\neq 0$). Here again we should stop and make different random choice b among the rest [1,B].

The solutions **PF**(b), which passed verifications 1 and 2, are included into a set {**A**}.

4. Randomization. Choosing the appropriate position of all lessons

PF(b) ∈ {**A**} could serve as a proper basis for creating feasible **TT** ∈ {**TT***}**.** Next step is fully randomized timetabling procedure. Let's take one **PF**(b) and imagine each lesson, as a 'labeled brick'. At the beginning all S bricks are in the 'storage' in a matrix **TPL**. The labels are *i* and *j*, i.e. classes and teacher numbers. Let's randomly choose one 'brick' and place it into the also randomly chosen available place in **TT,** which is empty at the beginning. This 'place' should satisfied the main constrains (1), (3),(4) and also the requirements of **PF**(*b*). If there are no a proper place for a brick, we should try again: erase **TT** and repeat the procedure(say, 1000000 times). If even after 100 attempts a feasible decision **TT** is still not achieved, that can mean that the choice **PF**(*b*) is not good and we shall try another one from **{A}.** If we were successfully in placing all the bricks [1,S], that means that feasible decision is achieved, **TT** ∈ {**TT***}**.** After that we can proceed with the next stage: timetable optimization.

5. Timetable optimization

In [(1)] a method for a heuristic search for timetabling was proposed. The author used a system of 'notional costs', which enclose a number of timetabling objectives, which are many. (An example: it is not good if the same subject is the last period on one day and the first period on the next day.) The more important the objective, the higher a subcost. The sum of all the subcosts complies the particular cost of the timetable f(**TT**) . We should not save only the 'best' timetable, we should better construct a set **{TT**}** ∀ **TT**∈ **{TT**},** f (**TT**)<δ . Among these TT people can choose the most appropriate one .

Conclusions

The proposed approach allows using the advantages of computer randomization. It means a possibility to consider an enormous number of combinations in the short time. Describing the timetable problem in a matrix form makes it easily to perform some useful verifications, which deal with the important school constrains.

Since so far there are no any mathematical model for a school timetabling problem and the proposed approach is just an attempt to solve the problem empirically, the true verification of the method could be performed only in a real situation at school.

REFERENCE

(1) Wright, M., 1996, School Timetabling Using Heuristic Search, *Journal of the Operation Research Society*, Vol.47, pp.347-357.