Project on University Timetable Problem

(A Report written in Partial fulfilment
of the requirements of Course End – Semester of IME 639A)

Submitted On:

18th May 2021

Submitted by:

Kuldeep Kumar (180369)

Shubhendu Singh (20114271)

Under the Guidance of

Dr. Faiz Hamid

Assistant Professor

Department of Industrial and Management Engineering

Indian Institute of Technology, Kanpur

1. Introduction

Timetabling is a multidimensional assignment problem, which needs to be solved regularly at educational institutions. It is the assignment of courses to faculty members and the assignment of these courses to the classroom and time slots in a way that makes optimal use of the available resources. Such a timetable must satisfy certain constraints such as no single teacher teaches more than one class at the same time, no single room is allocated for more than one class at the same time, and so on. Further, it may try to achieve certain objectives such as maximum utilization of classrooms, assigning teachers to his or her preferred courses, etc. Practically, a typical university timetabling problem may comprise thousands of courses, thousands of students, hundreds of instructors, hundreds of classrooms and other resources.

Moreover, the timetabling problem has been classified as NP-hard optimization problem (i.e., no polynomial time algorithm is known to solve the problem), meaning that if all combinations were to be examined, the time to solution for reasonable problems would rise dramatically. Therefore, in order to find optimal solutions to such problems, it is necessary to consider all possible solutions to choose the best one that satisfies a wide range of constraints, preferences, and participants and it must be solved in reasonable time.

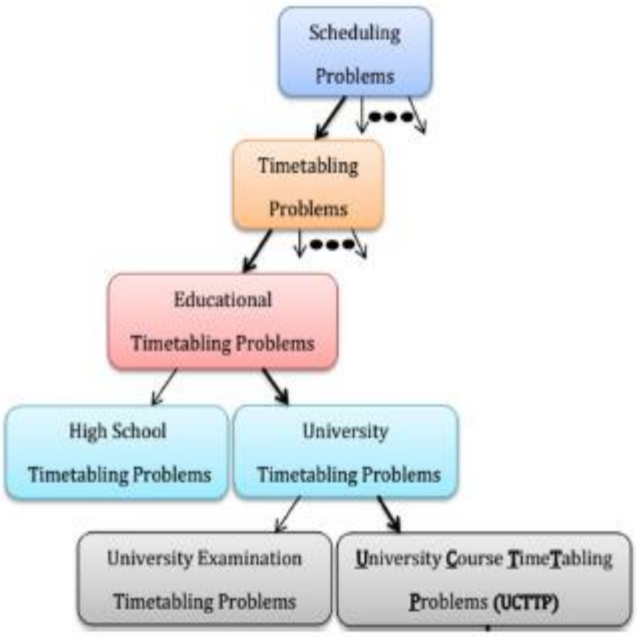
Scheduling Problems
Timetabling Problems
Educational Timetabling Problems
High School Timetabling Problems
University Timetabling Problems
University Examination Timetabling Problems
University Course TimeTabling Problems (UCTTP)

Fig1: Diagram of University Course Timetable Problem

Although the university timetable (UTT) is a major, regular and complex administrative activity in most academic institutions, only a few organizations possess reliable automated timetable solvers, and fewer still possess solvers that require no manual intervention. However, most institutions employ the knowledge and experience of expert personnel with regard to the production of a good timetable that satisfies all given requirements. Categorically, UTT problems can be classified into two main categories: course timetabling (CTT) problems and examination timetabling (ETT) problems, each with its own sets of constraints and requirements. This report focuses on CTT. Therefore, a lot of research has been invested in order to provide automated support for solving a real-world timetabling problem. Contributions come from the fields of operations research (e.g., graph coloring, network flow techniques) and artificial intelligence (e.g., simulated annealing, tabu search, genetic algorithms and constraint satisfaction).

2. Problem Description

In a University Time Table problem (UTTP) number of classes (C), teachers (T), venues(V), periods per day (P) and days in a week (D) are given . Classes and teachers must be assigned in such a manner that there are no clashes. In this problem the number of times a class has to be held by a particular teacher in the given venue is given in the form of a matrix. Number of working days and number of periods are specified using which total number of timeslots can be calculated. The problem has been designed to be totally constrained. That is, each class, teacher, and venue is required for each period. The optimal objective function for each of these problems is to allot class, teacher and venue for each of the time periods and ensure zero clashes among them.

2.1. Data Set

Number of teachers = 4

Number of classes = 4

Number of room available = 4

No. of periods per day = 6

No. of days per week = 5

No. of slots per week = 30

Total number of TimeSlots =120

Matrix for the Problem

$$\begin{array}{cccc|l}
2 & 2 & 1 & 2 & \\
1 & 1 & 1 & 2 & \\
1 & 1 & 1 & 6 & \boxed{\text{Venue 1}} \\
2 & 2 & 3 & 2 & \\
 & & & & \\
2 & 5 & 1 & 2 & \\
0 & 4 & 3 & 2 & \\
1 & 2 & 1 & 0 & \boxed{\text{Venue 2}} \\
2 & 2 & 1 & 2 & \\
 & & & & \\
2 & 1 & 1 & 2 & \\
0 & 0 & 5 & 1 & \\
2 & 1 & 4 & 1 & \boxed{\text{Venue 3}} \\
6 & 1 & 2 & 1 & \\
 & & & & \\
3 & 1 & 2 & 1 & \\
1 & 4 & 1 & 4 & \\
3 & 3 & 2 & 1 & \boxed{\text{Venue 4}} \\
2 & 0 & 1 & 1 &
\end{array}$$

Where, the first four rows of matrix indicate the number of times each class-teacher combination is to meet each other in venue 1 across the P periods. The next four rows indicate the number of times each class-teacher combination is to meet each other in venue 2 across the P periods, etc.

3. Problem Formulation

No. of Classes = C

No. of Teachers = T

No. of Venues = R

No. of periods per day = P

No. of days per week = D

No. of slots per week = K = P * D

Decision Variables

$x_{ijkl} =$ 1, if the ith class meets jth teacher in the kth slot and in the lth room.

0 otherwise

Objective Function:

The objective function for the problem is to minimize the number of clashes. Since the problem is to find a feasible solution we may use other objective functions as well. One such function can be to minimize the summation of all decision variables. This ensures that our IP solver is not biased towards any decision variable. But the best objective function would be just a constant value as it won't take computational time to calculate the objective function value. If CPLEX is able to solve the IP, we get a feasible solution.

Minimize $\sum_i \sum_j \sum_k \sum_l x_{ijkl}$ ---(1)

Or

Minimize 1 ---(2)

Constraints:

1. Requirement Constraint

$$\sum_k x_{ijkl} = M[C * l + i]\,[j] \qquad \forall\, i, j, l$$

Where M = Matrix

2. Constraint for no clashes for classes

$$\sum_j \sum_l x_{ijkl} \leq 1 \quad \forall k, i$$

3. Constraint for no clashes for teachers

$$\sum_i \sum_l x_{ijkl} \leq 1 \quad \forall k, j$$

4. Constraint for no clashes for rooms

$$\sum_i \sum_j x_{ijkl} \leq 1 \quad \forall k, l$$

4. Computational Results

The problem was solved using CPLEX in C++ in Visual Studio 2019. A 4 dimensional matrix was created to store the decision variables.

The time taken by the program to get the feasible arrangement of teachers and classes is shown below.

S. No.	No. of classes	No. of teachers	No. of rooms	Time taken in seconds
1	4	4	4	0.21
2	5	5	5	0.48
3	6	6	6	0.83
4	7	7	7	28.79
5	8	8	8	32.54

So for problem size of 8 classes, 8 teachers and 8 rooms, the program takes nearly 0.5 minutes. If applied to the real life problems with 100-200 of teachers and classes it may take hours to arrive at a proper assignment of classes and teachers to rooms.

5. Conclusion

The university course timetabling problem was solved by the exact method. It was formulated as an Integer Programming problem. The time elapsed to get the feasible distribution of the classes and teachers for a week was 32.54 seconds for problem size of 8 classes, 8 teachers and 8 rooms. Since the dataset has a smaller number of classes and teachers, we were able to solve it by exact method. The IP approach will take much more time for larger datasets. So a more sophisticated algorithm is required to solve real life university timetable problems.

References

- Babaei, H., Karimpour, J., & Hadidi, A. (2015). Computers & Industrial Engineering A survey of approaches for university course timetabling problem. Computers & Industrial Engineering, 86, 43–59. https://doi.org/10.1016/j.cie.2014.11.010
- El-sakka, T. (n.d.). University Course Timetable using Constraint Satisfaction and Optimization. 4(3), 83–95.
- Rudová, H., Müller, T., & Murray, K. (2016). Complex university course timetabling. May 2010, 187–207. https://doi.org/10.1007/s10951-010-0171-3
- Zuzana, M. (2019). University course timetabling and International Timetabling Competition 2019.
- Hav, J., Olsson, A., & Persson, J. (n.d.). Modeling and optimization of university timetabling A case study in integer programming as.
- Sandström, V. (2012). Scheduling of Teaching Resources and Classes using Mixed integer linear programming - A case study in a university of applied sciences.
- Schimmelpfeng, K., & Helber, S. (2007). Application of a real-world university-course timetabling model solved by integer programming. 783–803. https://doi.org/10.1007/s00291-006-0074-z