Research Paper MATHEMATICS

A Mathematical Modeling of University Examination System

PUSHPENDRA KUMAR VASHISHTHA	ASSISTANT PROFESSOR, KAMLA NEHRU COLLEGE
ROHIT GOEL	ASSISTANT PROFESSOR, DESHBANDHU COLLEGE, UNIVERSITY OF DELHI
PRIYANKA SAHNI	ASSISTANT PROFESSOR, KAMLA NEHRU COLLEGE
ASHWINI KUMAR	ASSISTANT PROFESSOR, SWAMI SHRADDHANAND COLLEGE

ABSTRACT

This paper attempts to place an emphasis on the fundamental aspects of the Examination Timetabling problem i.e. to develop the mathematical background for the problem. We will be defining examination timetabling problem and introducing various hard and soft constraints associated with the problem. Then the mathematical formulation will be given to make the plate form for further implementation. Here we will be adopting the strategy: Largest degree first. Events that have a large number of conflicts with other events (i.e., a large degree) are scheduled early. The rationale is that the events with a large number of conflicts are more difficult to schedule and so should be tackled first.

KEYWORDS

What is Timetabling?

Timetabling problems are a specific type of scheduling problem and are mainly concerned with the assignment of events to timeslots subject to constraints with the resultant solution constituting a timetable. Wren [18] defined timetabling in the following way:

"Timetabling is the allocation, subject to constraints, of given resources to objects being placed in space time, in such a way as to satisfy as nearly as possible a set of desirable objectives."

Based on the definition given by Wren[18], we need to know whether there are sufficient resources available for the given event to take place at its specified time as well as which resources are allocated. The goal is to optimize some objective function depending on the application domain at hand. For example, in examination timetabling environments, the function to optimize is usually the gap between two examinations that a student has to sit in i.e. try to spread the examinations throughout the examinations periods of time. The basic terminology used in timetabling problems is summarized in Table 2.1.

Table 2.1 Terminology used

Terminology	Definition
Event	An activity to be scheduled. Examples include examinations and courses.
Timeslot (period)	An interval of time in which events can be scheduled
Resource	Resources required by events. Examples include rooms and equipment (i.e. projectors).
Constraint	A restriction to schedule the events. Examples include room capacity and specific timeslot.
Individual	A person who has to attend the events.
Conflict	Two events are clashing with each other if they have at least a common individual and are scheduled in the same timeslot.

The constraints in timetabling can be divided into two categories:

- Hard constraints,
- Soft constraints.

Hard constraints cannot be violated. Soft constraints are not essential but their satisfaction is highly desirable in order to produce a good quality timetable.

A general timetabling problem consists of assigning a number of events like courses, examinations, lectures, lab sessions etc. into a limited number of timeslots and rooms while minimizing the violations of a set of constraints. In general, the timetable problem has a set of events E, a set of timeslots T, and a set of hard constraints C. The intention is to arrange all of the events E, into the timeslots, T, in such a way that no hard constraint C is violated in order to produce a feasible timetable.

THE EXAMINATION TIMETABLING PROBLEM

The examination timetabling problem represents a major administrative activity for academic institutions. It is often a difficult and demanding process and it affects a significant number of people. Carter [21] reports that there are three broad categories of people that are affected by its outcome: administrators, academic staff and students. Many universities are seeing an increasing number of student enrolments into a wider variety of courses and an increasing number of combined degree courses. This is contributing to the growing challenge of developing examination timetabling software to cater for the broad spectrum of constraints and demands that are required by educational institutions across the world and therefore the quality of a timetable should be evaluated from several points of view.

WHY DO WE NEED AUTOMATED TIME TABLE?

University timetabling represents a difficult optimisation problem and finding a high quality timetable is a challenging task. With a large number of events involved and various hard constraints to be fulfilled, finding an optimal timetable is complicated and time consuming. Solving a real world university timetabling problem manually often requires a large amount

of time and expensive resources. Also, it is not flexible, as if we want to modify any data item in the data base, we will have to modify the entire data base manually. Either this leads to inconsistency or will take a lot of time. As the data base increases, then it will be much more difficult to make it manually. To overcome these problems, we need automated time table. Also, this is much faster to response and to modify in comparison to manual modification. It is user friendly as the user can access it from anywhere. It is more efficient than manual system. Any changes in the data base can be made any time; actually we can make changes in the data base just before few minutes of examination.

CONSTRAINTS

The responses from a survey on examination timetabling problem among 56 universities conducted by Burke et al.[17] found that the constraints vary widely from institution to institution. Some examples of hard constraints for examination timetabling are:

- Certain examinations must be consecutive or must take place in a specific order (before/after each other).
- Examinations with the largest number of students should be scheduled earlier in the timetable to allow more time for marking.
- Examinations given by the same instructor, if scheduled in the same timeslots, have to be assigned to nearby classrooms.
- There should be no more than x conflicting examinations in y (>x) consecutive timeslots.
- Certain examinations must take place in a specific room.
- There must be enough seating capacity in the room for the number of students scheduled in it.
- No student should be required to sit two examinations simultaneously.

The generally accepted hard constraints for the examination timetabling problem are (i) there must be enough seating capacity and (ii) no student should be required to sit two examinations at the same time. Solutions that satisfy all the hard constraints are called feasible. On the other hand, there might be some requirements that are not essential. These are referred to as soft constraints. Common constraints, as reported in Burke et al. [15] are:

- Students should not be scheduled to sit more than one examination in a day.
- Students should not be scheduled to sit examinations in two consecutive timeslots.
- Each student's examinations should be spread as evenly as possible over the schedule.
- Some examinations may only be assigned within a particular set of timeslots.
- Examinations of the same length may be scheduled in the same room.
- Examinations must be scheduled to the rooms which are near to the relevant department.
- Examinations with questions in common must be scheduled in the same timeslot.

In a real world situation, it is, of course, usually impossible to satisfy all the soft constraints, but minimizing these violations will increase the quality of the solution by calculating the penalty function to the extent to which a timetable has violated its soft constraints.

MATHEMATICAL FORMULATION

In this section, we present the formal model of the examination timetabling problem:

Indices:

i, j : $1...N$

r, p : $1...R$

s : $1...S$

t : $1...T$

Parameters:

- N: The number of examinations
- R: The number of examination rooms;
- S: The number of students;
- T: The number of available timeslots;
- S_i: The number of registered students in exam i;
- R_t: The number of examination rooms available at timeslot t;
- D: the number of days in which the exams are to be conducted;
- F_r: The total capacity for room r;
- C_{ij}: The conflict matrix where each element $(C_{ij} : i, j \in (1 \ldots N))$ is the number of students that have to take both exam i and j. The conflict matrix is a symmetrical matrix of size N, where diagonal elements $C_{ii} = S_i$;
- d_{tk}: specifies the assigned day for examination k $(k \in \{1 \ldots N\})$ that is scheduled in timeslot t $(t \in \{1 \ldots T\})$ and lies between 1 to D.

We have to find the optimal number of class rooms, time slots and days which can hold all the courses offered by the departments, taking into account all the constraints of the examination scheduling problem.

Decision Variables:

$t_i = x_{it}$ and

$$x_{it} = \begin{cases} 1 & \text{if exam } i \text{ is assigned to time slot } t, \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ir} = \begin{cases} 1 & \text{if exam } i \text{ is assigned to room } r, \\ 0 & \text{otherwise.} \end{cases}$$

$$z_{rt} = \begin{cases} 1 & \text{if room } r \text{ is assignment to timeslot } t, \\ 0 & \text{otherwise.} \end{cases}$$

The objective is to spread out examinations over the exam period (timeslots) for each student, minimize the conflict occurrence in exams.

In this problem, an objective function is used to space out students' examinations throughout the examination period. The examination timetabling problem can be formulated as the minimization of:

$Min\, F(x)$;

$$F = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} c_{ij} * proximity(t_i, t_j)}{2 * S}; \quad \ldots (1)$$

Or it can be written as:

$$Min \frac{\sum_{i=1}^{N} F_1(i)}{2 * S} \quad \ldots (2)$$

where:

$$F_2(i) = \sum_{j=1}^{N} c_{ij} * \text{proximity}(t_i, t_j) \quad \text{... (3)}$$

is a cost for examination i.

$$\text{proximity}(t_i, t_j) = \begin{cases} \frac{32}{2^{|t_i - t_j|}} & \text{if } 1 \le |t_i - t_j| \le 5 \\ 0 & \text{otherwise} \end{cases} \quad \text{... (4)}$$

Where t_i and t_j specify the assigned timeslot for examination i and j where $i, j \in \{1,2,...,N\}$. $F_1(i)$ represents the cost for an exam i that is given by the proximity value multiplied by the number of students in conflict. Proximity values of 16, 8, 4, 2 and 1 are used here. For example, if a student has two consecutive examinations then a proximity value of 16 is assigned. If a student has two examinations, with a free timeslot in between, then a value of 8 is assigned. Two empty periods correspond to a penalty of 4 and so on. . These values are summed up and divided by 2S, to give an average penalty per student since all clashes would otherwise be counted twice (e.g. c_{23} and c_{32}).

Constraints:

1. No student can sit two exams concurrently (clash-free requirement). If examination i and j are scheduled in timeslot t, the number of students sitting both examination i and j must be equal to zero, i.e. = 0. This can be expressed as:

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{N} x_{it} * x_{jt} * c_{ij} = 0; \quad \text{... (5)}$$

This represents a clash-free requirement so that no student is asked to sit two exams at the same time. The clash-free requirement is considered as a hard constraint.

2. All exams must be scheduled and each exam must be scheduled only once in available timeslots, T.

This can be expressed as

$$\sum_{i=1}^{T} x_{it} = 1 \quad \text{... (6)}$$

3. For each timeslot t the number of rooms assigned to a particular timeslot must not exceed the maximum number of rooms available in a timeslot . This can be expressed as:

$$\sum_{r=1}^{R} Z_{rt} \le R_t; \quad \forall t \in \{1...T\} \quad \text{... (7)}$$

4. The total number of students assigned to a particular exam room(s) must be less than the total room capacity. This can be expressed as:

$$S_i \le \sum_{r=1}^{R} y_{ir} * F_r \quad \text{... (8)}$$

i.e.

$$S_t \le Seats \ \ for\ t = 1,...,T$$

Where St is the number of students taking examinations in timeslot t and Seats is the number of Q

5. Only one examination paper is scheduled to a particular room in a particular timeslot. There is no sharing of rooms with other exam papers.

COCLUSION AND FUTURE WORK

$$\sum_{i=1}^{N} x_{it} * y_{ir} = Z_{rt} \quad \text{... (9)}$$

Thus we have been successful to give the mathematical format to our problem. A number of hard and soft constraints have been included. The mathematical formulation, we have given, is an ideal formulation as at has a lot of hard and soft constraints. In real life it is not always possible to satisfy all the constraints, sometimes we have to compromise. Nevertheless we will be doing our best to satisfy as much constraints as possible to make the problem more realistic.

In future, we will be developing the step-wise algorithms to solve the problem in a very easy way. Along with each algorithm one real life examination timetabling example will be given to show it manually. As stated earlier our goal is to schedule all the examinations in minimum number of time slots without any clash. Therefore we will be trying to find the feasible solution that satisfies all the hard constraints and will be finding the final penalty cost for each exam and the average penalty cost for all the exams.

REFERENCES

[1] Azin Setayesh, Examination Scheduling with Days-off Constraints, Ph.D Thesis, Dalhousie University Halifax, Nova Scotia, 2012. | [2] Hamza Turabieh , Salwani Abdullah, "An integrated hybrid approach to the examination timetabling problem", Omega 39, 598–607, 2011. | [3] M.N.M. Kahar, G. Kendall, "The examination timetabling problem at Universiti Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution", European Journal of Operational Research 207, 557–565, 2010. | [4] Weiwei Chen, Leyuan Shi, "A Variant of Examination Timetabling Problem", 4th IEEE Conference on Automation Science and Engineering, August 23-26, 2008. | [5] S. Abdullah, S. Ahmadi, E.K. Burke and M. Dror, Applying Ahuja-Orlin's large neighbourhood for constructing examination timetabling solutions. Extended Abstract, The Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT V), Pittsburgh, USA, pp 413-419, 2004. | [6]. G.M. White and B.S. Xie, Examination timetables and tabu search with longer-term memory. In: E.K. Burke and W. Erben Practice and Theory of Automated Timetabling, Selected Papers from the 3rd International Conference. Lecture Notes in Computer Science, vol. 2079, pp. 85-103, 2001. | [7]. L. Di Gaspero and A. Schaerf Tabu, search techniques for examination timetabling, In: E.K. Burke and W. Erben Practice and Theory of Automated Timetabling: Selected Papers from the 3rd International Conference. Lecture Notes in Computer Science, vol. 2079, pp. 104-117, 2001. | [8] S.C. Brailsford, C.N. Potts and B.M. Smith, Constraint satisfaction problems: Algorithms and applications, European Journal of Operational Research, vol 119, pp. 557-581, 1999. | [9] A. SCHAERF, "A Survey of Automated Timetabling", Artificial Intelligence Review 13: 87–127, 1999. | [10]. E. K. Burke and J. P. Newall. A multistage evolutionary algorithm for the timetable problem. Evolutionary Computation, IEEE Transactions on 3(1), pp. 63-74. 1999. | [11]. H.J. Goltz and D. Matzke . University timetabling using constraint logic programming. In: Practical Aspects of Declarative Languages, Lecture Notes in Computer Science, vol. 1551, 320-334. 1999. | [12]. J. P. Newall. Hybrid Methods for Automated Timetabling, Ph.D Thesis, School of Computer Science and Information Technology, University of Nottingham, UK, 1999. | [13] E.K. Burke and M. Carter. (editors), The Practice and Theory of Automated Timetabling II: Selected Papers from 2nd International Conference on the Practice and Theory of Automated Timetabling (PATAT II), Toronto, Canada, Lecture Notes in Computer Science 1408, Springer-Verlag, 1998. | [14]. P. David, A constraint-based approach for examination timetabling using local repair techniques, In Practice and Theory of Automated Timetabling: Selected Papers from the 2nd International Conference.Lecture Notes in Computer Science, vol. 1408, pp. 169-186, 1998. | [15] E. Burke, J. Newall and R. Weare. A memetic algorithm for university exam timetabling. Selected Papers from the First International Conference on Practice and Theory of Automated Timetabling 1153pp. 241-250. 1996. | [16].M. W. Carter, G. Laporte and S. Y. Lee. Examination timetabling: Algorithmic strategies and applications, European Journal of Operational Research Society 47(3), pp. 373-38, 1996. | [17]. V.A. Bardadym, Practice and Theory of Automated Timetabling: Selected Papers from the 1st International Conference, Lecture Notes in Computer Science, vol. 1153, pp. 22-4, 1996. | [18] Wren, A., Scheduling, Timetabling and Rostering – A Special Relationship? In Edmund Burke and Peter Ross, editors, The Practice and Theory of Automated Time-tabling. Lecture Notes in Computer Science 1153, 46-75. Springer-Verlag, Berlin, 1995. | [19].M. W. Carter, G. Laporte and J.W.Chinneck , A General examination scheduling system, interfaces 24, pp.109-120,1994. | [20]. E. Burke, D. Elliman & R. Weare, Extensions to a University Exam Timetabling System, In IJCAI93 Workshop on KnowledgeBased Production, Planning, Scheduling and Control, pp. 42–48, Chambery, France, 1993. | [21] Carter, M.W., A Survey of Practical Applications of Examination Timetabling. Operations Research 34, 193-202, 1986. |