ITETT: An Automatic IT Examination Timetable Tool

Souad Larabi Marie-Sainte *

College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

Abstract

In this paper we describe a BSc graduation project, which deals with the automatic assignment of exams (for first and second mid-terms) to timeslots. This project is a web-based application dedicated to support the Information Technology (IT) department at King Saud University (KSU) to create an examination timetables automatically. The tool will allow the IT administrators to automatically generate the examinations schedules for all levels while taking into consideration a number of constraints. The system will meet all the IT requirements and will also permit the modification of these requirements if needed. Until today, the generation of examination timetable is done manually. So, ITETT will be made in a simple and effective way in order to serve IT department staff and save their time and effort.

Keywords: Scheduling, Timetabling Problem, Tabu search.

1. Introduction

Scheduling and timetabling problems involve many contexts like scheduling of transport employee, scheduling of sport events, and timetables in educational institutions. Educational timetabling in universities is divided into course timetabling and exam timetabling. The most problem in the administration of Information Technology department (IT) of the College of Computer and Information Sciences, King Saud University (KSU) consists in making examinations timetable. The examination timetable problem is considered as a most important administrative work for academic organizations. A growing number of students registered, an extensive diversity of courses and a rising number of combined degree courses contribute to the increasing confront of developing examination timetable. The manual generation of examination timetable takes a lot of time and maybe expensive in terms of money and resources. This paper describes an ongoing BSc graduation project that aims to handle the problem of automatically generating examination timetables for the first and second mid-terms in the IT department. In this department there is a large number of courses associated with different levels, and each course has a number of different sections. The sections associated with one course should have exam in the same time. Only one timeslot is required in IT department, 3-5 pm. Due to the considerable number of students following courses of different levels, a student may have two exams in the same time, which leads to a conflict. The IT administration faces many problems when assigning exams to timeslots because of the difficulty in handling conflicts in schedules. Our purpose is to create an automatic exam to timeslots assignment tool, called ITETT (IT Examination Timetable Tool). This tool will provide a satisfactory schedule for all students, and fulfill the needs of the IT department. The ITETT is a web-based manageable system that saves all IT courses, sections and IT students information in a centralized database. It should provide some useful online features, not only to the administrator but also to the students and instructors. The ITETT is based on a heuristic algorithm that tries to construct a feasible examination timetable for IT department.

In this paper, we briefly describe some existing related scheduling tools and optimization methods. Then, we define the problem and address the context of use of ITETT, its system analysis and architectural design. Later we present the development environment of our system and provide a general description of the algorithm that is used for automatically generating examination timetables. Finally, we finish this work by a brief conclusion.

* Corresponding author. Email: smariesainte@ksu.edu.sa
2. Related Tools

There are a number of existing online tools of examination timetables. Most of these tools are commercial general-purpose software, which offers solutions to a large number of university examination timetabling problems. Our tool is different from those because it is a special-purpose tool designated to meet the requirements and constraints of the IT department in KSU. Among the existing online tools are:

1. Scientia (http://www.scientia.com): multi user resource management software that assists educational institutions to construct optimal timetable for all resources including exams.
4. Uni timetable manager (http://jack.valmadre.net/timetable): is free open-source software for automatically finding an optimal timetable.

According to our research, systems that focus on the creation of examination timetables are infrequent. So, this reason encourages us to provide a system that not only meets the IT department requirements but also being flexible to serve any academic institutions.

3. Existing Methods

Timetabling problem has been treated by a large number of researchers owing to its NP complete nature. There are various types of timetabling problems on which various approaches are proposed. For example educational timetabling, nurse scheduling, sports timetabling, transportation timetabling, etc.

According to A. Wren, 1996 [1], “Timetabling is the allocation, subject to constraints, of given resources to objects being placed in space and time, in such a way to satisfy as nearly as possible a set of desirable objectives”.

Examination Timetabling Problem (ETP) is a specific case of timetabling problems. Carter [2] illustrated Examination Timetabling Problem as “The assigning of examinations to a limited number of available time periods in such a way that there are no conflicts or clashes”.

This topic has obtained special attention of the scientific community in the last ten years. Researchers on Operational Research and Artificial Intelligent have shown an eminent interest on this subject. Qu et al. [3] carried out a thorough survey of examination scheduling solving methods.

Among the various methods that have been applied to ETP include the following. Graph Based Sequential Techniques were explored by [4]. Population Based Algorithms are prominent in solving ETP. For example, Genetic Algorithms [5], Ants algorithms [6] and Particle Swarm Optimization [7]. Local search based techniques represent a large part of the work that has been applied on a variety of ETP. For example, Simulated Annealing was investigated by [8]. Tabu search is one of the oldest local search based technique that has been used since 1991 to solve a variety of timetabling problems [9]. Several Tabu Search methods were proposed for solving educational timetabling problems and particularly Examination Timetabling Problems. For example, Schaerf (1996) [10] provided combinations of Tabu Search with other local search techniques to solve course timetabling problem; Di Gaspero and Schaerf (2001) [11] performed a constructive study on a family of Tabu Search based techniques to solve ETP; White and Xie (2001) [12] proposed a four-stage Tabu Search called OTTABU, where solutions were progressively enhanced by taking into consideration more constraints at each stage, for the ETP at the University of Ottawa; Paquete and Stutzle (2002) [13] presented a Tabu Search method for Exam Timetabling where ordered priorities were provided for the constraints. Moreover, Tabu Search based technique is carried out within the Hyper-heuristic framework which investigates the best permutation of sequential heuristics for solving ETP such as [14] who introduced a Tabu Search hyper-heuristic which considered the selection of low level heuristics as a competition.

4. Problem definition

Examination timetabling deals with a set of a number of exams to a given number of timeslots while satisfying a set of constraints. Constraints are divided on 2 types, hard and soft constraints [7]. Hard constraint must be satisfied to generate a feasible timetable. The universities may have their own specialized hard constraints based on their needs and requirements. Soft constraints are generally more frequent and different from hard constraints. The violation of soft constraints should be minimized. It is the soft constraints, which effectively define how well given feasible solutions, are.

Our project consists in handling exams from level three up to level eight. In the IT department of the College of Computer and Information Sciences at KSU, there are approximately 36 courses. Each course has a number of sections (usually between 1 and 6). Our examination timetable will be set with one timeslot (3-5 p.m.) in three weeks (working days) for each mid-term. Hard constraints are defined as:

(1) Students should not have two exams in the same day and time,
(2) All the IT exams should be scheduled.

Soft constraints are:

(1) No exams on two consecutive days for the student at the same level.
(2) Certain exams may need to be scheduled in a specific timeslot (a specific day).

The objective of the ITETT is to obtain a feasible solution (that satisfies the hard constraints), having the
minimum possible number of soft constraint violations.

5. **Context of Use of ITETT**

The IT administration is responsible for assigning exams to timeslots. To do this, the administrator first asks the instructor (the responsible) of each course to propose an adequate date for his/her exam. After having all the exams dates and times, the administrator shall ensure that there is no more than one exam on the same day (since there is only one timeslot per day) for the same level. As said above the scheduling process is done manually. So, this task is very daunting and time consuming. Moreover, the resulting timetables may be inefficient in terms of not satisfying the hard constraint (conflicts for students). The administrator cannot detect conflicts for students following courses of different levels. If the conflict occurs, only the interested student can detect it and try to solve it with the instructors of the conflicting courses.

To solve this problem, we propose to use a metaheuristic method explained in section 9. In order to take benefit of this proposed strategy, a web-based system is implemented and designed (called ITETT) to meet the department’s shortcomings.

The role of the ITETT is to make easy this task not only for IT administrator but also for instructors and students. The system will provide an easy access to the web-based interface for administrator, instructors and students. Administrator can log in, generate the examination timetables for both mid-terms, update information related to students, courses, sections, levels, and current semester. In addition, the administrator can modify the timeslot of some exams after creating the schedule. On the other hand, instructors and students can view, save and print exam timetable for each mid-term. They also can search for specific exam by selecting the associated course or level.

6. **System Analysis**

The ITETT is intended to the following users:

1. **IT administrator:** is the basic user in our website. He/She is able to:
   a. Login / Logout.
   b. Update his/her personal information.
   c. Manage all the system by adding and/or deleting section, student, course, semester.
   d. Add exams coming from outside the IT department and concern IT students.
   e. Automatically generate examination timetables during the academic year.
   f. Manually modify the examination timetable (by changing only the timeslot in the same day).
   g. Import data related to students, sections, courses and semester to the databases.
   h. Export data related to students, sections, courses and semester from the databases.
   i. View the examination timetables.

2. **Instructor of IT course and/or student:** who are the users of our system. ITETT allows them to:
   a. View examination timetable at any time during the academic year.
   b. Save and/or print examination timetables.
   c. Search for a specific exam by choosing either the corresponding level or the corresponding course.

All the functionalities presented in ITETT for both the administrator and the users are indicated in the use case diagram in Fig. 1 and 2.

![Fig. 1. Use Case Diagram of User](image)

ITETT offers to the administrator a less time to construct the exam timetable. Also, IT students and instructors can obtain the exam timetable at the beginning of the semester without any conflict.

A context diagram that shows the dataflow within the ITETT is illustrated in Fig. 3.

7. **System Design and Architecture**

The ITETT stores all the information related to IT students, IT courses, sections and the current semester in a centralized database. The system architecture is presented in Fig. 4.

This architecture represents an easy solution to the problem in consideration. It decreases the cost of modifications and avoids redundancy of both code and data.

This architecture allows the construction of timetable through the database and the website. The website is mostly utilized for data input and will check the data validity before the data are recorded in the database. Additionally, the website will present a personalized timetable. The database permits data retrieval according to a specific format.

The interface of this website is designed to provide user-friendly interactions for user with varying technical
background. The main page of ITETT System is depicted in Fig. 5.

The user (either student or instructor) can navigate through it to view (and eventually to save and/or print) the exam timetable of the first/second mid-term for the current semester. he/she can also inquire about policy of exam and makeup exam. Moreover, he/she can search for a specific exam through the associated course or level. Finally, the user can find additional links like KSU website and IT department website.

For the administrator, he can not only benefit from all these features but also log in into the system and manage (by adding/deleting/modifying) all information related to sections, courses, students, …etc as shown in Fig. 6.

Recall that the main function of the administrator is construction of the timetable. This function is performed by just clicking on Generate Timetable button in Timetable Tab. This button allows the running of the entire algorithm explained below.

8. Development Environment

To implement this architecture, some technological elements are requested to perform our system.

- HTML & JSP: these languages are used to facilitate the display of data and the conception of forms to insert data.
- XML: permits the data exchange with other systems.
- Java: is the language chosen for coding.
- Database: allows managing data.

9. Algorithm

This section briefly describes the heuristic algorithm, which is the heart of the ITETT.

As mentioned above, all information related to students, courses, sections, current semester and timeslot are stored in the database. Initially, the IT administrator begins by
adding all the exams coming from outside of the IT department, and which concern some of the IT students. These exams are scheduled by other departments and the modification of their times and/or dates are generally not allowed. The algorithm will proceed in two phases.

In the IT department, only one timeslot is required 3-5 pm during three weeks.

Phase1: consists in finding an initial solution (examination timetable). The algorithm will try to assign exams to timeslots with respect of the prerequisite requirement. In fact, each course that pre-requires another course, both are assigned to the timeslot of the same day because students cannot enroll in these both courses in the same semester. For example, Operating System course pre-requires Data Structure course. Thus, students who are enrolled in Data Structure course cannot be enrolled in Operating System course in the same semester and vice versa. Some courses do not necessitate an IT prerequisite course or their prerequisite courses are already scheduled. In this case, the algorithm will assign the course to a specific day (timeslot) after checking that there is no conflict for students in the same day.

Phase2: after having an initial solution, Tabu Search is applied in order to improve the solution (Exam Timetable) and satisfy as well as possible the soft constraints cited above.

This process is repeated for a number of trials (e.g. 10). Each new schedule generated will be compared with the best so far schedule. If the new schedule is better, in terms of satisfying the soft constraints with respect to the hard constraints, it will replace the best so far schedule. The final best schedule will be returned as the output of the algorithm.

9.1. Tabu Search algorithm

Tabu search [15] is a local search technique considered to solve optimization problems. The idea is to explore the
solution space beyond local optimality while taking into account some attributes. Contrary to Simulated Annealing and Great Deluge Algorithms in which a decision is done according to the acceptance of the neighboring solution randomly selected. Tabu Search algorithm assesses all neighboring solutions in order to select the next move. The main attribute of Tabu Search is the use of a short-term memory called Tabu list, which permits to store the note of the most recent move history. Its role is to avoid the search process from cycling in those recently visited solutions by preventing moves recently done. The application of Tabu Search involves the definition of the following parameters:

1. The initial solution;
2. The neighbor relation and the neighbor selection procedure;
3. The objective function to minimize;
4. The length of the Tabu list;
5. The stopping criteria.

The application of Tabu Search to Examination Timetabling Problem has been shown to be successful. Tabu Search algorithm applied in this project focused on the work presented by Di Gaspero and Schaerf [11]. In their technique, a move is defined as reassigning a new timeslot to a randomly chosen exam. For the neighborhood size of the current solution, the examinations for the consideration in the next move are chosen from the set of examinations that have violations of hard and/or soft constraints. Contrary to [11] in which the initial solution is made using the Largest Enrollment Sequential Heuristic, the construction of the initial solution used in this work is based on a proposed heuristic discussed in phase 1 of this section. So, this point is the unique difference between our algorithm and that of [11]. The objective function considered in this work is to minimize the violation of soft constraints. For more details, the authors can refer to [11].

10. Conclusion and Future work

In this article we described ITETT, a web-based system intended to solve the examination-timetabling problem in IT department at KSU, subject to a number of hard and soft constraints. The ITETT is constituted of a heuristic algorithm that utilizes a Tabu optimization approach to improve the resulting solution. The algorithm is straightforward and robust. It can fulfill the requirements and constraints of various educational institutions by easily including some modifications in it. This tool is currently under implementation, providing encouraging results.

When the implementation phase is completed, the system will be evaluated by testing the algorithm with real and artificial data to assess its effectiveness.

After finishing this project, our future research work will be devoted to the investigation of the possibilities of applying meta-heuristic methods for examination timetabling such as genetic algorithms and Particle Swarm Optimization. This last method is not well developed in this context and deserves more attention.

Acknowledgment

This work was supported by the College of Computer and Information Sciences of King Saud University. Many thanks to Haya Al-Hurishay, Khawllah Abo-Hashem, May Naif Al-Markan, Morooj Abd-Alallh Al-Suwaiebeh, Sultanah Saud Al-Mefgaet for their effort in designing and implementing the ITETT.

References