# Final Exam Scheduling Timetable a Case Study

**Conference Paper** · May 2010

CITATIONS
2

READS
9,264

**1 author:**

Mohammed Moreb
Smart University College for Modern Education
**53** PUBLICATIONS **157** CITATIONS

# Final Exam Scheduling Timetable a Case Study

My professional knowledge and research in two fields first: develop and study the Scheduling problem or related problem, second research in e learning management system such as research in mobile learning, stander for e learning course, consistency for e learning course, e learning with web 2 impact.

Scheduling problem has many application such as co-production and consistency, I worked about four years in studying Final Exam Scheduling Timetable in my BS.C degree and in master level for two university, and I can work to make the algorithm to solve Final Exam Scheduling Timetable or relater Scheduling problem as general for most universities depending in input date and some parameters to customize this algorithms for that university.


## Abstract

Scheduling problem is defined in many areas and it's hard to solve, because it includes many constraints that should be solved. Final exam scheduling is one type of Scheduling problem, and it's hard to be solved because each university may include hard and soft constraints that are different for each case. So in this research we will solve the final exam scheduling problem for alquds university, and we will develop the system, make experiment for the problem to generate final exam scheduling timetable and compare it with other systems if existed.

In the following section we will describe the Background and the literature review of the problem, then define the n/p complete problem, hard constraints and soft constraints, then describe the problem, and finally our solution for the problem and the experiment result.

**Keywords**

Final Exam scheduling, timetable, soft and hard constraints, N/P complete problem

**Previous work**

The scheduling of exams is a common problem faced by educational institutions. Even in the simplest forms of this problem, determining if a feasible schedule exists is NP-Complete. Many others have previously studied this problem, and how to solve it such with hyper approach that contains three phases to solve the problem. It achieved good result by comparing the result with other approaches and the result shows that the hybrid approach is the best for large dataset [(1)]. Hybrid approach solved the timetable problem in three phases: phase one applying hard constraints, phase two applying soft constraint, phase three improve the table generated from phase one, two. Hybrid approach introduces the problem and use reference paper for the experiments to reduce work, but most of this approach is applied for timetable problem and for large data set and not developed for final exam scheduling problem.

Colored approach for exam scheduling described in Stochastic Search Algorithms for Exam Scheduling [(2)] defined scheduling problem and specified the final exam scheduling and show how to solve the scheduling problem using clusters to reduce iterations for conflict, shown two algorithms. Used by coloring and how to reduce the colored in scheduling problem, experiments for Stochastic Search Algorithms for Exam Scheduling for two alg. For real data and previous algorithm and the experiment is well and good; Graph Hybrid [(7)] takes the colored approach and Graph

search for new algorithm to reduce the complexity of iteration of problem by categorizing the related constraints to use it for solving and generating timetable.

Agents technology applied for scheduling problem and New methods is introduced, to solve time table problem using agent by applying hard constraints and soft constraints by agent, three agents in system first for CHC, 1-M HC, 1-m SC [3], The agent works in three models passive, active, parallel; time table using agent to solve and generate timetable and need to develop in such away.

New algorithm is introduced recently to solve the examination and introduce benchmark to use it for definition and experiments result and to compare with other works or algorithm named 13 Carter benchmark problems [4], algorithm is described in 5 steps, to produce exam time table.

EA algorithm [5] uses new hard/soft constraints (time slot with day) used to generate timetable, experiments result which are compared with 2 algorithms to show that the EA algorithm is the best.

Software engendering is defined and described which affects Exam scheduling timetable process and stages of building a substantial [6], carefully specified, fully tested and fully operational university and school timetabling system.

Some developers develop tools for FESP and have already taken an interest in the XML standard; SchoolTool http://www.schooltool.org, Tablix http://www.tablix.org, Final Exam Scheduler (FES) http://cse.yeditepe.edu.tr/~eozcan/research/TTML[8]., PATAT http://www.idsia.ch/Files/ttcomp2002/[9].

Most of the research solve the problem in general approach, for this it's hard to satisfy all hard constraints and most of soft constraints for the problem. Most of

research study the recent algorithm for timetable problem and aren't applied for FESP, thus the research will study and define the scheduling problem, final exam scheduling timetable at Alquds University as a case study and define the hard and soft constraints that we will use, by using hyper approach that describes time table and it's not applied for Exam scheduling problem, so we will take the main steps and apply to final exam scheduling problem FESP. So our research will be to develop this model to be compatible with our case study and to be efficient for FESP and make experiments for the problem with real data.

**The examination timetabling problem**

Scheduling problem is defined as n/p complete problem which means that there is no perfect solution for it, but there is a nearest solution to perfect solution; because there are many constraints that should be satisfied for solving the problem and for this many methods can solve the problem but there is one solution better than others because it satisfies the hard constraints and another solution is better than the first because it satisfies all hard constraint and some of soft constraint and so on.

Hard constraints are those constraints that must be met by the timetable in order for the Exam timetable to be feasible. For example, there should not be any clashes, i.e. students should not be required to write two exams at the same time.

Soft constraints are constraints that we would like the timetable to satisfy, but can be broken if necessary. For example, students' examinations must be well spaced over the timetable; examinations with a large number of students must be scheduled early in the examination timetable. It is highly unlikely that all the soft constraints will be met by a timetable as those are usually contradictory. Timetabling systems attempt to minimize the number of soft constraints violated.

Hard and soft constraints differ considerably from one examination timetabling problem to the next and are institution dependant. For example, some versions of the problem may require the number of examination sessions used to be minimised; some examinations may have to be scheduled simultaneously or after other exams; special requirements of students, such as religious requests, may need to be catered for.

An examination timetabling problem may be capacitated or uncapacitated [(5)]. In the capacitated version of the problem venue (lecture room) allocation is taken into

consideration and the number of students writing examinations in a particular venue and session must not exceed the capacity of the venue. This is treated as a hard constraint.

**Research Problem and Motivation**

We have N/P complete scheduling problem and it's a hot research area in the word, this proposal will study the scheduling problem (final exam timetable) at Alquds University as a case study, then apply research algorithm after selecting the problem and will develop and make experiment result for the real problem and compare the result for exam scheduling with current table and use some benchmark for experiment result.

Our scheduling problem has many motivations to achieve. Some of them that it's real problem and many students in the university face the final exam problem, example, some students take more than 2 exams at the same time, others take all exams in the first day of examination period and such problem motivated me to solve this problem.

**Problem**

1) **Final exam scheduling at Alquds University**

The student exam scheduling problem is a standard event scheduling problem. Given sets of exams and students, each student has a schedule of exams he/she must attend. Furthermore, these exams must be scheduled into a set of timeslots such that no student has multiple simultaneous exams.

We are given:

- A set of exams

- A set of students
- A set of rooms
- A number of timeslots

With each of these sets comes additional information interrelating them.

- Each student has a schedule of exams he/she must attend.
- Each room has multiple features, such room size.
- Each exam has a set of required features including the number of students attending it,
- Rooms can only accommodate exams if they have enough seats to fit all the attending students, and have at least all the features required for the exam.

For our problem we will consider 3 timeslots for one day. A feasible schedule would be the assignment of events to rooms within the limited number of timeslots such that: no students have to write multiple simultaneous exams and every room can accommodate each exam assigned to it. We will call the constraints required for feasible schedule hard constraints.

In addition to finding a feasible schedule, there are other factors we might find desirable when finding a schedule. For example, if we can find multiple feasible schedules, we may want to decide when one is better than another.

There are many factors one could consider when evaluating which schedule is the best. For our specific problem we look at a set of soft constraints that we will try to avoid violating, but which violation will not result in infeasibility. From these soft constraints we can derive a cost function that computes a penalty for any feasible

schedule based on the number of soft constraints it violates. In our case the soft constraints are as follows:

(1) Avoid students having a block with three consecutive exams.

(2) Avoid students having a single exam in one day.

(3) Avoid students having an exam in the last timeslot of the day.

Each occurrence of any of these situations is considered equally bad, so the value of the cost function associated with a feasible schedule gives the number of times each soft constraint is violated. For example a feasible schedule with a cost function value of zero would be an optimal solution for any problem.

The specific instances we set out to solve were of medium size. They were large enough that brute force style methods would be utterly useless, yet small enough that one might run across many larger instances in real world settings. The problem instances given by alquds university had:

• 10000–11000 students

• 1000–1100 exams

• 90–100 rooms

• 39 timeslots split evenly within five days

For the problem, speed will be important, and contestants were to solve many different instances of this size by running their program on a single processor machine for a limited period of time. The time allowed for each machine was computed by a benchmarking program that took many factors into account. On a Pentium 4 system we were limited to roughly 10 minutes.

## 2) Short description of problem

Study the scheduling problem and how it will improve the cost and time and how it affects students and administration office. To solve the problem we will introduce the current state for examination timetable and the hard and soft constraints which we will use.

Given that E exams are to be taken by students over D days, where P exam periods can be done per day, the exam scheduling problem consists of assigning A exams to Π (=D*P) exam periods, within specified classrooms.

The objective is to minimize the conflict and the unfairness factors, which are:

i. The number of students having simultaneous exams,

ii. The number of students having consecutive exams, and

iii. The number of students having two or more exams on the same day.

In our problem, we assume the following conditions and constraints:

i. The user should be provided with the flexibility of assigning weights to the three conflicts and unfairness factors.

ii. A limited number of exam periods, Π, is predefined.

iii. A limited predefined number of classrooms, R, are available for exams.

iv. Room capacities (Ψ1, Ψ2, …, ΨR) are taken into consideration in assigning exams to rooms . Also, more than one exam can be assigned to the same room at the same time if they fit.

**Methodology**

Many algorithms and methods describe how to solve the scheduling problem such Simulated annealing, Constrains logic programming, Linear programming, graph coloring heuristic, Genetic algorithm, Memetic algorithm, Heuristic algorithm, Heuristic ordering method, and Hybrid algorithm, we will study these algorithms and determine which are suitable for our case then determine hard and soft constraints for Alquds University timetable, after this we will design and implement the system to make experiment result in the data to generate timetable.

**Summary**

Final exam scheduling is n/p complete problem and its hot research area, we will study this problem for Alquds University as case then and determine the hard and soft constraints for the problem and install the system in real environment to make experiment result.

**References**


1. Salwani A and A.R Hamdan, **a Hybrid Approach for University Course Timetabling**, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.8, August 2008.
2. N Mansour and M Timany, **Stochastic Search Algorithms for Exam Scheduling**, International Journal of Computational Intelligence Research, Vol.3, No.4 (2007), pp. 353–361.
3. Y Yang, R Paranjape and L Benedicenti, **an Agent Based General Solution Model For the Course Timetabling Problem**, ACM May 8-12, 2006.
4. N Pillay, and W Banzhaf, **a study of heuristic combinations for hyper-heuristic systems for the uncapacitated examination timetabling problem**, European Journal of Operational Research, 2008.
5. Grigorios N. Beligiannisa, Charalampos N. Moschopoulosa, Georgios P. Kaperonisa, and Spiridon D. Likothanassisa, **Applying evolutionary computation to the school timetabling problem: The Greek case**, Computers & Operations Research 35 (2008) 1265 – 1280.
6. A Abbas, and E Tsang, **Software engineering aspects of constraint-based timetabling—a case study**, Information and Software Technology 46 (2004) 359–372.
7. E BURKE, B MCCOLLUM, A MEISELS,S PETROVIC, and R QU, **a Graph-Based Hyper-Heuristic for Educational Timetabling Problems**, European Journal of Operational Research, 2006.
8. **E Özcan, E Ersoy , Final Exam Scheduler – FES** , 2006
9. E CHENG, R KLEINBERG, S KRUK, W LINDSEY, AND D STEFFY, **A STRICTLY COMBINATORIAL APPROACH TO A UNIVERSITY**

**EXAM SCHEDULING PROBLEM,** Practice and Theory of Automated Timetabling (PATAT) http://www.asap.cs.nott.ac.uk/ASAP/ttg/patat-index.html, 2003

**Standards in eLearning course on Palestine University**

Introduction:

Most of Palestine University established the e- learning unit for developing, and implement course, to improve the quality of learning in university and most of the developed course applied as e-learning course (on line course or blended course), depending on some criteria on that university.

Keywords: eLearning, Quality Assurance, Courseware Design and Production, Support of eLearning, Course Delivery.

Reasons and purposes for undertaking this research

This programme of study continues personal research and professional practice in the field of Information Science, particularly within the area Quality assurance of E learning course.

Within Higher Education, there is evidence of constant innovation and changing approaches to provision of online services; however, the wide ranging and long term issue of user accessibility has clearly become a secondary consideration. My PhD research should underpin improved provision of defines model for standard in eLearning course for Palestine University.

Questions for research may include the following:

Most e-learning unite in Palestine university established to improve the quality of learning, so their many

question need to answer from developer, course create, course administer or that university likes:

- What criteria should be taken to take this course as e-learning course
- What's the minimum activity should include on E learning course?
- What this activity type should be in the course.
- After develop the course is applied this course will improve the quality of learning for learners and unit.

So my research area will be introduce very important topic for most of Palestine university, and Ministry of

high Education, by found the infrastructure of standardize, and quality assurers of e-learning course in Palestine university.

Methodology:

This research will study the current satiation of all university in e-learning, then will generate the level zero

of common of policy that admin the e-learning course in each university, then study the standers of E learning in Global university and how we can produce Model in Palestine university.

Our study will focus in costly Model for e-learning and how to improve quality assurance by Appling standardize e-learning course; and will focus in stander for e-learning course of tow type blended learning and on line course

Summary:

In summary e-learning is a hot topic in Palestine University, and each university make proposed play it to higher ministry educative to earn funds to develop the e-learning course, but there is no Model at this time to help the teacher or administer on what's the level of developing course will be.

So we need to define model to standardize for developing e-learning course to help course developer to improve the quality of e-learning course