International Conference of Sciences, Engineering & Environmental Technology (ICONSEET), 5(15): 114 – 122, 2020
ISSN 0794-9650

www.repcomseet.com

Design and Implementation of Lectures Time Table Scheduling System (A Case Study of Federal Polytechnic Ede)

[1]Oyetunji, Olumayowa O., [2]Adegoke, Moses A., [3]Olatunji, Funsho A.
[1,2, 3]Department of Computer Science, Federal Polytechnic, Ede
Author's Corresponding Email: oludmayor1@gmail.com, adegokemoses88@gmail.com

Abstract: ***Lecture time-tabling system is a system that schedules and monitors the lecture time and available resources in order to maximize such resources. It is one of the most important yet time consuming task done periodically in any academic institution of learning. In this paper, the course sets, halls and time allocations are represented by a multi-dimensional array on which a local search is performed and a combination of the direct representation of the timetable with heuristic crossover is made to ensure that fundamental constraints are not violated. Analytical Hierarchy algorithm was applied in the development of a viable timetable system which was tested to demonstrate the variety of possible timetables that can be generated based on user specified constraints and requirements. Hypertext Preprocessor (PHP) and Structural Query Language (MySQL) was used for designing the application. The main requirement of the application is to provide the details about the lecturers, subjects, and total number of class. Then the system generates the time table according to the need.***
Keywords: Time Table, Scheduling, Monitoring, Analytical Hierarchy algorithm

1.0 INTRODUCTION

Lecture time-tabling system is a system that schedules and monitors the lecture time and available resources in order to maximize such resources. It is one of the most important yet time consuming task done periodically in any academic institution of learning. Therefore, lecture time table should be carefully assigned into appropriate time slot for students, lecturers and lecture halls based on constraints. (Burke E. K. and Petrovic S, 2002)
Timetabling is a sub-class of scheduling for which the events take place at educational institutions (Willemen, 2014). Finding a feasible lecture/tutorial timetable in a large university department is a challenging problem faced continually in educational establishments. This paper presents an evolutionary algorithm (EA) based approach to solve a heavily constrained university timetabling problem. (Dipti, 2017). Scheduling is one of the important tasks encountered in real life situations. (Antariksha, 2016).
In scheduling the lecture time table, the following constraints must be taken into consideration; No student can attend more than one lecture at a time. No lecturer can teach more than one course per time and lecture hall cannot be assigned to more than one particular course at a time. The tedious task of data introduction and revision of usually incomplete solutions are the bottleneck in these cases. Most educational institutions have resorted to the manual generation of their timetables which according to statistics takes over a month to get completed and optimal. Even at the optimal stage of the manually generated timetable, there are still a few clashes and it is the lecturer that takes a clashing course that works out the logistics of the course so as to avoid the clash (Luisa et al, 2011).
In the most general terms, scheduling can be described as the constrained allocation, of resources to objects, being placed in space-time in such a way as to minimize the total cost of a set of the resources used. (Wren, 2013). In practical terms the timetabling problem can be described as scheduling a sequence of lectures between teachers and students in a prefixed time period (typically a week), satisfying a set of varying constraints(Schaerf, 2017).

1.1 Research Motivation

The traditional manual generations of timetables encounters a lot of problems which may include the following:

a) Repeated time allocations may be made for a particular course thereby leading to data redundancy.
b) A lot of administrative error may occur as a result of confusing time requirements.
c) Timetable generation by center staff may have a slow turnaround.
d) Final generated timetable may not be near optimal as a result of clashing course requirements and allocations.

e) It generates a lot of paperwork and is very tasking.
f) It is not flexible as changes may not be easily made.

Hence is the need for electronic lecture time scheduler/monitoring software was generated to overcome the shortcoming of the manual time-tabling system.

1.2 Design Approach

Having derived the requirements for the new system to be designed through the detailed analysis of the old manual system, the next step to be taken is the construction of the detailed designed plan. The aim of this is to divide the overall problems into smaller tasks, and manageable problems that can be easily handled by separate program modules or subunits. The separate program modules will later be integrated into one to form the entire new system. This designed methodology is called "Top-down design"
That is:

a.) To optimize the algorithm used in today's timetable systems to generate the best of timetabling data with fewer or no clashes.
b.) To enable generation of course schedules while demonstrating the possibility of building the schedules automatically through the use of computers.
c.) To expand the scope of timetable automation systems by making it generic thereby bringing about uniformity in the creation of timetables.

2.0 Literature Review

As a result of time-table system generation, growing number of authors who have done research in time-table system, are being discussed:

Fernandes, 2002 classified the constraints of class-teacher timetabling problem in constraints strong and weak. Violations to strong constraints (such as schedule a teacher in two classes at the same time) result in an invalid timetable. Violations to weak constraints result in valid timetable, but affect the quality of the solution (for example, the preference of teachers for certain hours). The proposed algorithm, evolutionary, has been tested in a university comprising 109 teachers, 37 rooms, 1131 a time interval of one hour each and 472 classes. The algorithm proposed in resolving the scheduling without violating the strong constraints in 30% of executions.

Another author proposed a repair process during neighborhood search process in the Simulated Annealing Algorithm to the course scheduling problem that consider lecturer preference to certain time slots. The aim of this experiment is to set some Simulated Annealing parameters that have better performance for their problems that include initial temperature, final temperature, neighborhood search operators and cooling schemes (Chainate, 2014). A scheduling activity usually produces a kind of table called a timetable so that some researchers referred the scheduling activity as timetabling activity. (Duong, 2015)

The university course timetabling problem (UCTP) is a combinatorial optimization problem, in which a set of events has to be scheduled into time slots and located into suitable rooms. The design of course time- tables for academic institutions is a very difficult task because it is an NP-hard problem. This paper investigates genetic algorithms (GAs) with a guided search strategy and local search (LS) techniques for the UCTP. The guided search strategy which is used here is to create offspring into the population based on a data structure that stores information extracted from good individuals of previous generations. The LS techniques use their exploitive search ability to improve the search efficiency of the proposed GAs and the quality of individuals. The proposed GAs is tested on two sets of benchmark problems in comparison with a set of state- of the - art methods from the literature. The experiment al results show that the proposed GAs is able to produce promising results for the UCTP. (Shengxiang, 2014).

There are different categories to solve the timetabling problem. They are Cluster method, Sequential method, Meta-Heuristics and Constraint Based method. Meta Heuristics is a higher level procedure which is used to provide good enough solutions for optimization problems. On some class of problems, they do not guarantee a globally optimum solution. This method is used when the classical methods are too slow or fail to give a solution. This is achieved at the cost of optimality and precision for speed. (Carter, 2015).

In a different research, it says even though there are number of researches found in timetabling few of them only developed as software. Their paper provides an overview of methodologies such as Bee algorithm, Constraint programming, Cyclic transfers, Evolutionary algorithms, Integer programming, Neural networks, Simulated annealing and so on. There is no comparative study on the success of different methodologies on timetabling

problems. The complexity of the timetabling is another issue. Manual scheduling generally takes number of weeks to generate timetables. (Nelishia, 2014)

Preparation of timetable for specific university is a very complex task. Therefore, they could introduce multi objective Evolutionary Algorithm based class timetable optimizer to reduce time. Some of the key challenges of timetabling by an automatic software engineering process as task based conceptual graph. There hard and soft constraints can be easily inserted or removed while the specifications are maintainable. However, there were some drawbacks as necessity of generalized methodology, specialist's skills while the problems are varying by concerning type of institute. (Dilip, 2015).

3.0 Research Methodology

3.1 Data Collection

The following method of data collection was used in writing this research:

i. Interview
ii. Observation
iii. Review of procedure or existing system or procedural manual
iv. Evaluation of forms

3.2 Design and Framework

The System Design describes the design and framework of the project shows the framework of the whole system. This explains all the processes involve in this system in form of diagram. The outputs from this system are timetable management. As shown in the figure below, both outputs are stored in database server. While admin assign the classroom, subject and course, Analytical Hierarchy Process will be implementing.

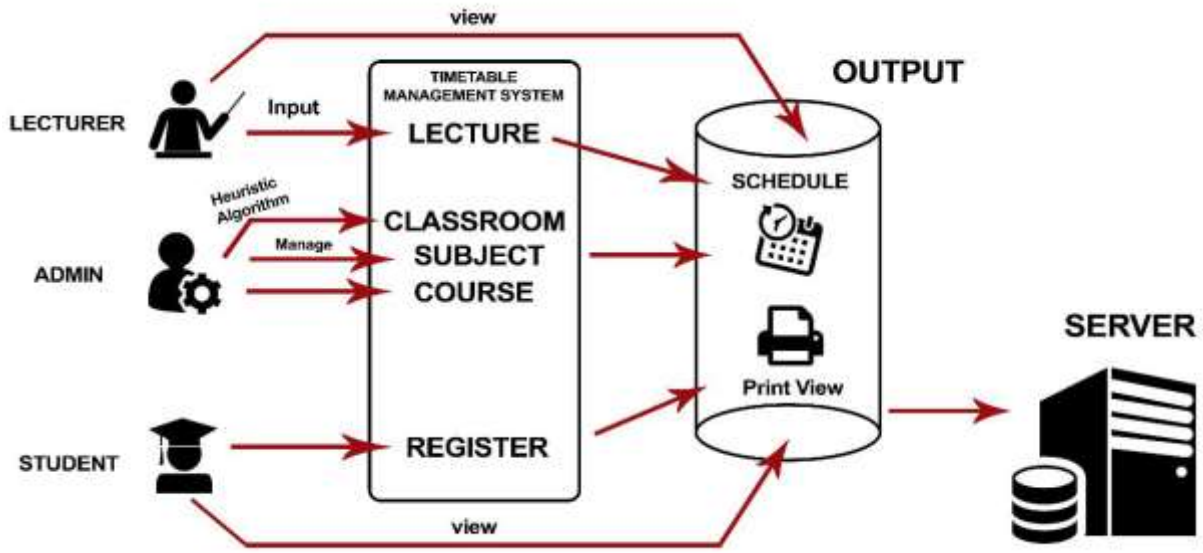

Fig 1: Design and Framework (Source; Authors)

3.3 Class Diagram

A class diagram is an organization of related objects. It gives an overview of a system by showing its classes and the relationships among them. Class diagrams only display what interacts but not what happens during the interaction hence they are static diagrams. Classes includes

- Lecturers
- Buildings
- Halls
- Programs
- Courses
- Levels
- Allocations
- Front End Staff
- Generator Module
- File Writer

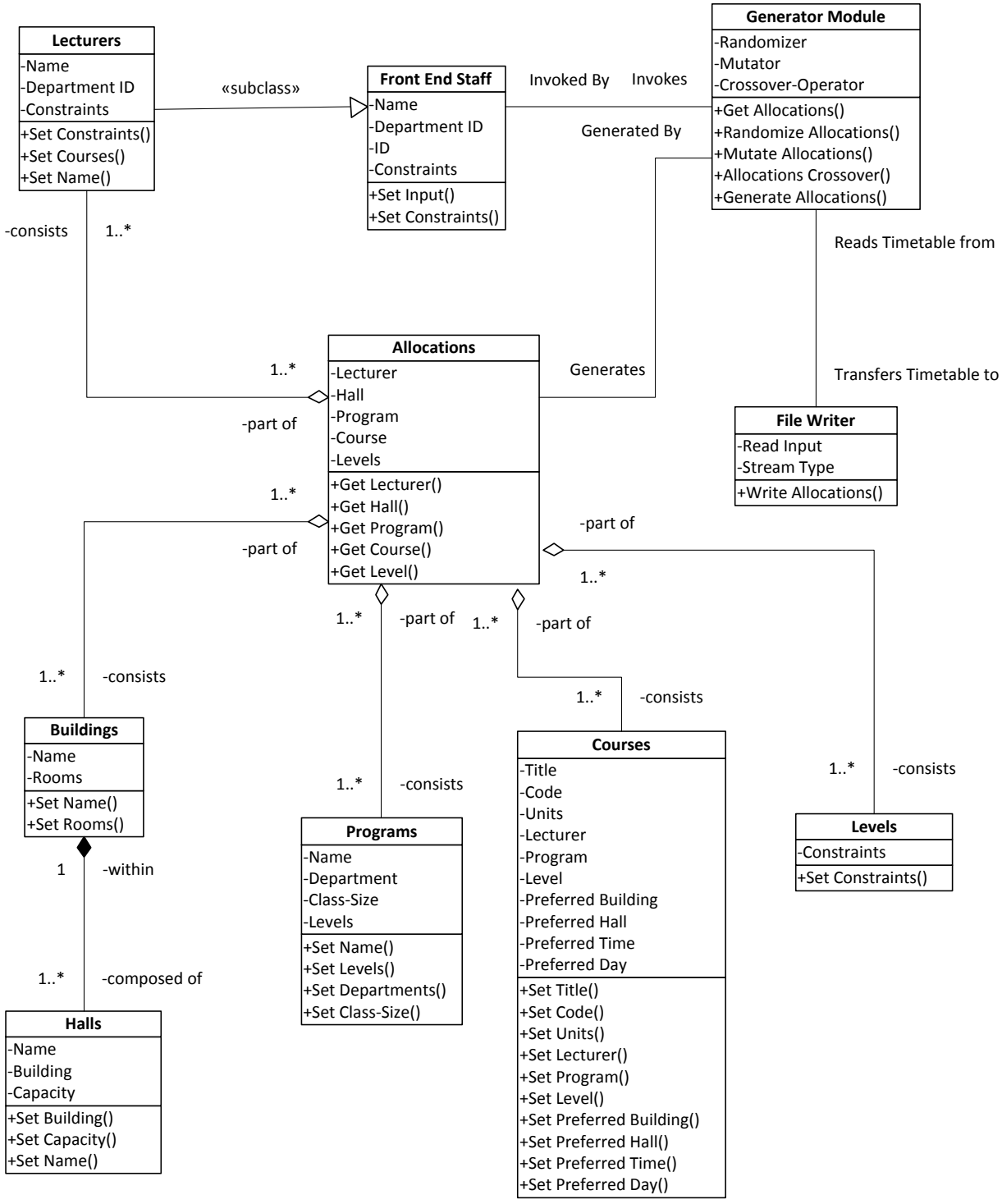

Figure 2.: Class Diagram to show the relationships between the different classes associated with the system

3.4 Context diagram

Diagram below shows the context diagram that displays all the function of the systems. It also shows the context diagram that includes all the three main entities and their main process in this system. The lecturer will input the select information based on the subject he teach in current semester. Lecturer also will be able to view, update, and delete the profile information. Besides, lecturer also will be able to view the schedule to know when he needs to invigilate the class. When lecturer has inserted the data information, admin will create the time table schedule based on the information. Admin also will assign the suitable classroom by using Analytical Hierarchy

algorithm, allocate students, and generate their time table class. Last but not least, students will be able to view and download the time table schedule.

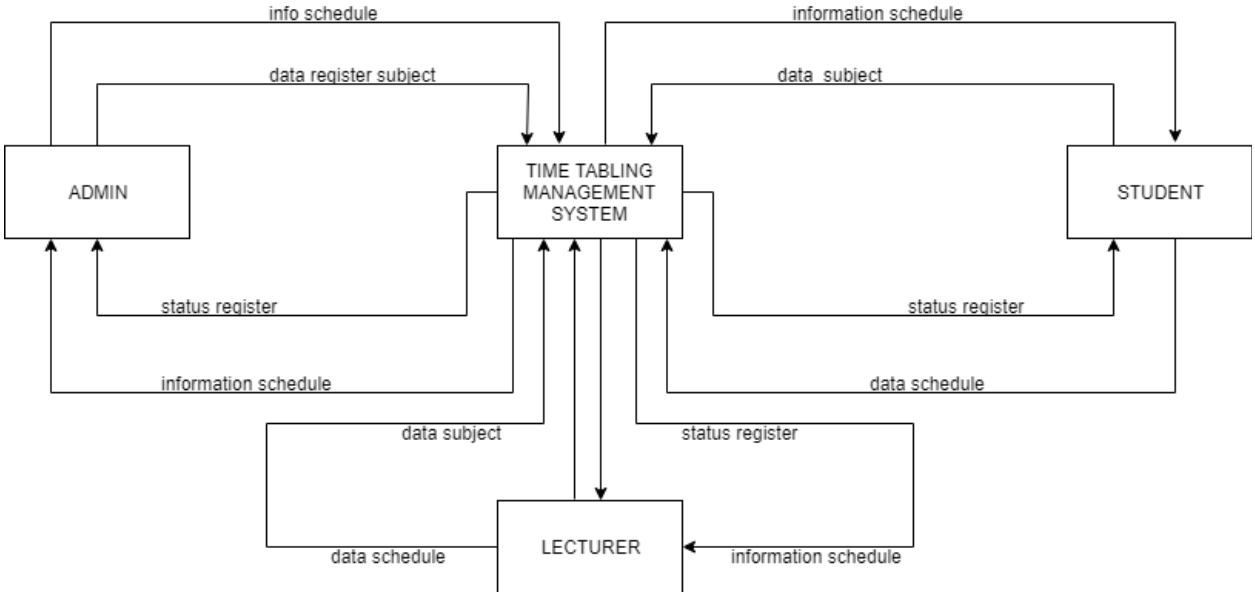

Fig.3.2.1. Context Diagram

4.0 Results and Discussion

User interface design is the design of computers, appliances, software applications and websites with the focus on the user's experience and interaction. The goal in designing user interface is to make a great interaction between user and the system in term of efficiency, user-friendly, compatible of the system with target users. The interfaces should be understandable, easier to use with a proper arrangement of system flow.

4.1 Log in

This shows the login page of the system. At this page, the user has to enter their username and password. The system will verify login either it is admin, lecturer or student based on their username and password.

Figure 3: Login (Source: Authors)

4.2 Admin Register Student

In this interface, admin will be able to register new student for them to log in to the system. Admin need to filling all columns otherwise the system cannot add the data into database.

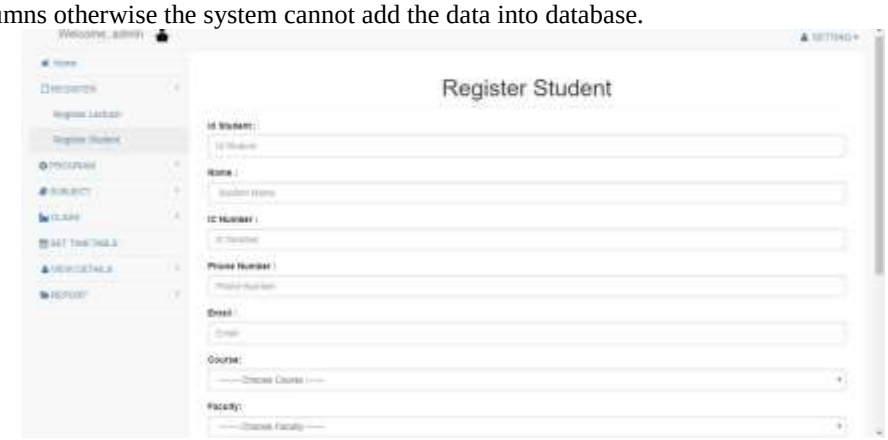

Figure 4: Register student (Source: Authors)

4.3 Admin Register Lecturer

In this interface, admin will be able to register new lecturer for them to log in to the system. Admin need to filling all columns otherwise the system cannot add the data into database.

Figure 5: Admin Register Lecturer (Source; Authors)

4.4 Admin Register Class

In this interface, admin will be able to register new class for quantity of student. Admin need to filling all columns otherwise the system cannot add the data into database.

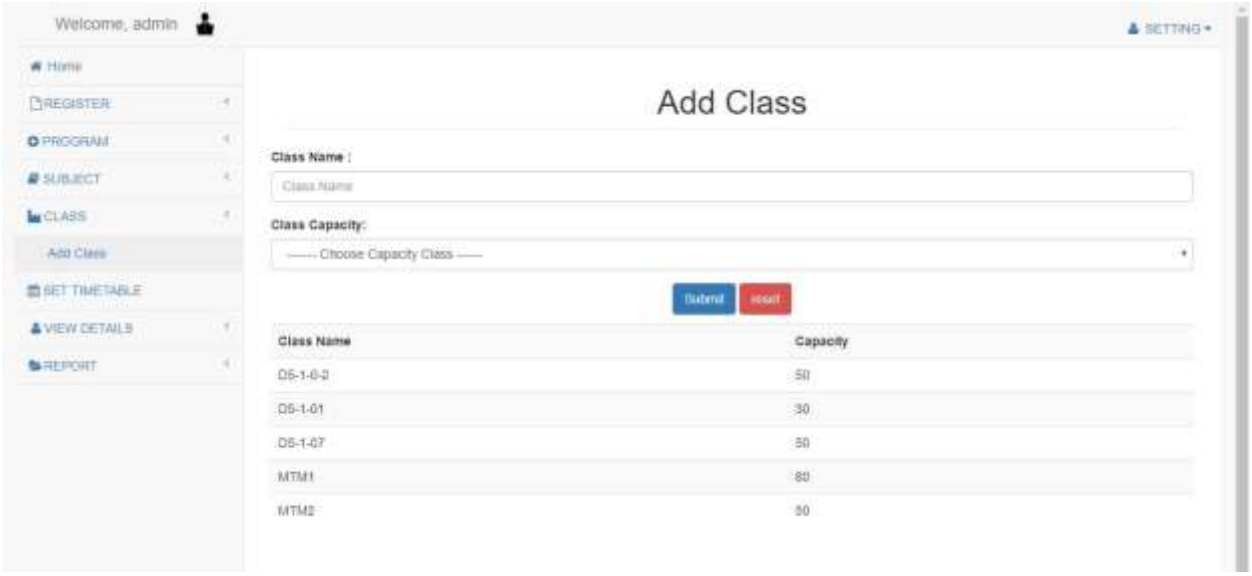

Figure 6: Admin Register Class (Source: Authors)

5.0 Recommendation & Conclusion

5.1 Conclusion

Having x-rayed the previous manual system with regards to my case study and with the hope that the proposed system will be implemented. I hereby conclude that the management with the assistance of the new system will facilitate fastness, sometimes in processing and publishing of computerized lectures and practical timetable.

5.2 Recommendation

With respect to the new system developed and data analysis, it was discovered that the data application in processing lecture and practical timetable would improve efficiently. It is therefore recommended that the management of higher institution or head of department of computer science should use this program.

References

A. Cornelissen, M. J. Sprengers and B.Mader (2010). "OPUS-College Timetable Module Design Document" Journal of Computer Science

Abramson D. & Abela J. (1992). "A parallel genetic algorithm for solving the school timetabling problem." In Proceedings of the 15th Australian Computer Science Conference, Hobart,.

Adam Marczyk (2004). "Genetic Algorithms and Evolutionary Computation ".

Al-Attar A. (1994). White Paper: "A hybrid GA-heuristic search strategy." AI Expert, USA.

Alberto Colorni, Marco Dorigo, Vittorio Manniezzo (1992). "A Genetic Algorithm to Solve the Timetable Problem" Journal of Computational Optimization and Applications.

Bufe M., Fischer T., Gubbels H., Hacker C., Hasprich O., Scheibel C., Weicker K., Weicker N., Wenig M., & Wolfangel C. (2001). Automated solution of a highly constrained school timetabling problem - preliminary results. EvoWorkshops, Como-Italy.

Burke E, Elliman D and Weare R (1994)."A genetic algorithm for university timetabling system." Presented at the East-West Conference on Computer Technologies in Education, Crimea, Ukraine.

Carrasco M.P. & Pato M.V. (2001). "A multi objective genetic algorithm for the class/teacher timetabling problem." In Proceedings of the Practice and Theory of Automated Timetabling (PATAT'00), Lecture Notes in Computer Science, Springer.

Chan H. W. (1997). "School Timetabling Using Genetic Search." 2th International Conference on the Practice and Theory of Automated Timetabling, PATAT'97.

Coello Carlos (2000). "An updated survey of GA-based multi objective optimization techniques." ACM Computing Surveys.

Costa D. (1994). "A tabu search algorithm for computing an operational timetable." European Journal of Operational Research.

Datta D., Deb K., & Fonseca, C.M. (2006). Multi-objective evolutionary algorithm for university class timetabling problem, In Evolutionary Scheduling, Springer-Verlag Press.

David A Coley (1999). An Introduction to Genetic Algorithms for Scientists and Engineers, 1st ed. World Scientific Publishing Co. Pte. Ltd.

Dawkins Richard (1996). The Blind Watchmaker: Why the Evidence of Evolution Reveals a Universe Without Design. W.W. Norton.