International Journal of Research in Engineering, Science and Management
Volume-2, Issue-12, December-2019
www.ijresm.com | ISSN (Online): 2581-5792

Automatic Time Table Generator

Savita Appaso Bandgar[1], Rutuja Balaso Moglade[2], Anuja Ajit Chougule[3], Deepika Avinash Bhosale[4]
[1,2,3]*Student, Dept. of Computer Science & Engg., DKTE's Yashwantrao Chavan Polytechnic, Ichalkaranji, India*
[4]*Professor, Dept. of Computer Science & Engg., DKTE's Yashwantrao Chavan Polytechnic, Ichalkaranji, India*

***Abstract*: Time Table Generation is a daunting task for those who work on time and man power. Providing an automatic time table generator will help to automatically generate a time table. Our project's proposed system will help create it, and will also automatically save time. This avoids the complexity of manually scheduling and managing schedules. In our project we will be using algorithms like genetic, heuristic, resource scheduling to reduce these scheduling constraints. This algorithm incorporates a set of strategies aimed at improving the efficiency of the search operation. The system will take different types of topics, number of teachers, workload, semester, and subject's preference. Depending on this input, possible timetables for the week's work will be created for the teacher's teaching. This will better restrict all resources in a manner that is limited.**

***Keywords*: Scheduling constraints, Time table generation.**

1. Introduction

Although the work of most college institutions has been mechanized, preparation of lecture schedules is often done by hand due to its inherent difficulties. Physical lecture-scheduling demands considerable time and effort. Manual lecture-scheduling is a limit fulfillment problem in which we find a result that meets the given limit.

In the past, there are many ways to get in trouble with scheduling for colleges and schools. The problem of timetabling can be solved by various methods obtained by operation studies such as graph color, spatial search solutions based boundary handling. In our project, scheduling problems are formulated as a problem fulfillment and we have proposed a realistic scheduling algorithm that is able to take care of both hard and soft problems. This is a perfect timetable for colleges that helps overcome the challenges that are created manually on a timetable.

2. Literature Survey

M. Nandhini, and S. Kanmani, "Implementation of Class Timetabling Using Multi Agents", [1] The time tabling problem means that assigning activities to time slots such as various constraints and constraints is satisfied. In artificial intelligence domain, graph coloring, hill climbing, tabu search, simulated neiling, variable neighborhood search, etc. has been used to speed up discovery using heuristic. In this paper, the implementation of a class time table is proposed by the Stapest Ascent Hill climbing algorithm with most agents. These agents use a combination generator that creates the maximum possible pairings for the inputting time table and the minifinder to achieve the minimum evaluation value for further testing. An improved heuristics are applied when creating a time table for the optimal solution from a random initial solution to reaching the goal as quickly as possible. Also, the interaction between architecture, operations and multi-agents is discussed when implementing a class time table.

S. Abdullah, E. K. Burke and B. McCollum, "A Hybrid Evolutionary Approach to the University Course Timetabling Problem", [2] The combination of evolutionary approaches, including local discovery, has yielded very good results for a variety of scheduling issues. This paper describes the development of such an algorithm in a university course schedule. This problem is related to the assignment of lectures to specific timeslots and rooms. In order for a settlement to be viable, several difficult issues must be met. The quality of the solution is calculated on the basis of the fine value, which represents the degree to which various soft limits are satisfied. This hybrid evolutionary approach is based on established datasets and is compared to the most advanced technology in the literature. The results obtained prove that the approach is capable of generating perspectives on the problem of fixation that exhibits the lowest penalty values in the literature on these benchmark issues. It is, therefore, concluded that the hybrid evolutionary approach represents a particularly effective method for solving the problem of high quality in a university course schedule.

Anirudha Nanda, Manisha P. Pai, and Abhijeet Gole, "An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach", [3] This paper proposes a general solution to the problem of school schedules. Most of the heuristics previously proposed ignore the problem from the student's point of view. This solution works from the teacher's point of view, i.e. the availability of teachers for a given period. While all difficult issues (eg. teacher availability, etc.) are strictly resolved, the timeline presented in this paper is an adaptive one, whose primary purpose is to address teachers' relevant lectures and subject conflicts.

3. System architecture

The final system should be able to generate a time table completely automatically, which will save a lot of time and effort of the organization administration. To make the time table system generic so that i can work equally well for different

schools, colleges and universities. There are user defined limits for handling. Ease of use for the system user so he / she can create an automatic time table. Focus on the optimization of resources i.e. teachers, labs and rooms. Provide scheduling facilities for everyone. Generate many useful views from the time table.

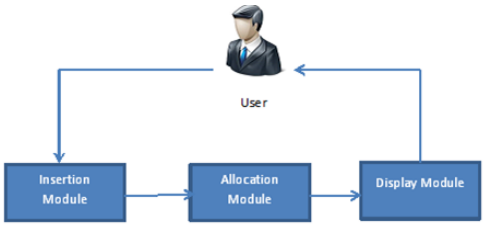

Fig. 1. System architecture

4. Implementation

It has developed into three sections

1. Insertion Module
2. Allocation Module
3. Display Module

1) Insertion Module

In this module we provide various user inputs to our system that work on raw data to create a final time table

a) *Faculty Details*-In this sub module we include various details of the professor such as the professor's name, email and contact number. And we also provide a unique faculty ID that helps reference our entire software, and it also works with login credentials.

b) *Subject Details*-In this sub module we include the names of such topics and topics in our curriculum. We strive to store theory topics and laboratory topics separately in our database so that they are easy for us to use in the future.

c) *Mapping*-In this sub module we take user input such as which theory subjects to take in which discipline and in which particular laboratory, and where do we store it in our database.

2) Allocation Module

In this module, the user can randomly select any semester to begin the process. It starts filling slots from Monday, selecting specific topics, and then mapping with that subject is assigned to that day's slot. The slot is filled so that each time various soft and hard problems are tested. It will not be blocked if any issues are not resolved. We are examining all these constraints by writing queries and using database data in the archive process.

3) Display module

In this section we will see how to create a time table for each class. We have also provided the feature to view class time tables and faculty wise time tables.

a) *Class Wise Time Table*-In this you can see the class wise generated time table by selecting the particular semester you want to view. The subject of the time table will be the professor who handles the subject.

b) *Faculty Wise Time Table*- In this we can see the names of all the topics handled by a particular subject.

5. Conclusion

This paper presented the implementation of online time table generator.

References

[1] M. Nandhini, and S. Kanmani, "Implementation of Class Timetabling Using Multi Agents", 2009.

[2] S. Abdullah, E. K. Burke and B. McCollum, "A Hybrid Evolutionary Approach to the University Course Timetabling Problem", Proceedings of the IEEE Congress Evolutionary Computation, Singapore, 2007.

[3] Anirudha Nanda, Manisha P. Pai, and Abhijeet Gole, "An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach", 2012.