AUTOMATIC TIMETABLE GENERATOR

Prof. Manisha. P. Navale*1, Aniket Marne*2, Omkar Pasalkar*3, Atharva Pawar*4, Shubham Yadav*5

*1,2,3,4,5Department Of Computer Engineering, Sinhgad Institute Of Technology And Science, Pune, India.

ABSTRACT

In today's literate world it is very difficult to create time table manually. Timetables are to be created uniquely for all branches and years respectively. It becomes a very hectic, time consuming, and needs manpower for preparing the timetables manually. In some cases this process becomes complex when any staff is on a leave or needs to be substituted. In our attempt, we have derived an algorithm to create timetable which will save a lot of time and reduce the load and pressure on the person doing the job. Using software to do the job saves a lot of time and can also create timetables for complex situations. It will also avoid any human error like: subject clash, vacant slots.

Keywords: Genetic Algorithm, Automated Timetable, Soft And Hard Constrains.

I. INTRODUCTION

Lecture timetabling is a very important process and one of the common scheduling problems in any educational institution, which can be described as the allocation of resources for tasks under predefined constraints so that it maximizes the possibility of allocation or minimizes the violation of constraints.

A key factor in running an educational center or basically an academic environment is the need for a well-planned, well throughout and clash-free timetable. Back in the days when technology was not in wide use, the timetable was manually created by the academic institution.

Therefore, the need to adopt an electronic system as opposed to the manual process cannot be over-emphasized. Several other administrative sectors of most institutions have been automated, but lecture scheduling is still done manually due to the problems involved.

The project timetable concerns all activities with regard to producing a schedule that must be subjective to different constraints. Timetable concerns activities with regards to producing a schedule that must be subjective to the different constraints.

II. LITERATURE SURVEY

When scheduling is modeled, it must cover both hard and soft constraints, i.e. a set of various variables and values such as halls, subjects, staff, and students must be assigned in order to satisfy them. Thereby we can conclude the objective of the above method is to minimize the violation of soft constraints and satisfy the hard constraints.

In the paper[1],The proposed algorithm can be further adapted to suit as per the requirements of different institutes and universities. It considers a wide range of constraints divided into soft and hard, to generate faculty, classroom/laboratory and student timetables. The complexity of the proposed algorithm is n3.

In [2], The paper presents a categorisation of the methodologies conducted in recent years based on chronology, category and application.

[3]We propose two pattern-based formulations and a solution algorithm that simultaneously exploits column generation and a team of metaheuristics to build and improve solutions.

[4]Graph Coloring Algorithm (GCA) takes 11 percent of the time, Genetic and Graph Coloring (GCA) takes 25 percent, Heuristic and Iterated Local Search Algorithms (ILSA) take 44 percent of time, and Heuristic Algorithm takes 20 percent. [5]This means developers can use imperative programming for business logic and declarative programming for user interface (UI) development.

[6]The fitness score relates to the quantity of crashes the timetable has experienced.

The above survey of various researchers of different algorithms of Automated Time-Table Generator trying to develop a software which helps to generate Timetable for an institution automatically. When we look at the existing system, we understand that the schedule generation is done manually. Manual adjustment of the schedule during the absence of any of the faculty, and this is a big challenge for the Automatic schedule.
generator, which manages the schedule automatically in case of absence of any of the faculty. As mentioned earlier, generating the schedule should take into account the maximum and minimum workload that the college has. Moreover, it is a time-consuming process.

1. Automated Timetabling System for University Course

The proposed algorithm can be further adapted to suit as per the requirements of different institutes and universities. It considers a wide range of constraints divided into soft and hard, to generate faculty, classroom/laboratory and student timetables. The complexity of the proposed algorithm is $n^3$.

An algorithm has been proved effective for replacing an oldfashioned, time-consuming timetabling system with an enhanced, flexible and automated automated system. The proposed algorithm saves hours of your precious time which can be utilized in other areas of your life.

2. A survey of the state-of-the-art of optimisation methodologies in school timetabling problems

The paper presents a categorization of methodologies performed in recent years based on chronology, category and application.

The optimization model captures the behavior of the system in all possible scenarios.

3. Pattern-based models and a cooperative parallel metaheuristic for high school timetabling problems

We propose two pattern-based formulations and a solution algorithm that simultaneously uses column generation and a metaheuristics team to generate and improve solutions. More modelling power and More readily implementable answers.


In this method, timetable problem is dealt as graph problem and then they order the events using domain specific heuristics and then assign the events sequentially into valid time slots, so no constraints are violated for each time slot. Graph Coloring Algorithm (GCA) takes 11% of the time.

5. Development of an Efficient Timetable System using AngularJs and Bootstrap3

This means developers can use imperative programming for business logic and declarative programming for user interface (UI) development. A user friendly GUI is provided with the help of Angular material and Bootstrap. Single page webpage is generated with the help of AngularJS.

III. METHODOLOGY

The project provide a Timetable without any kind of clashes.

There are three types of user in this system depending on the usage of the system. The user can either view the timetable or edit the timetable(admin).

1. Admin : - A admin has all the rights to modify the details of the other two entities as well as to generate the new timetable. A admin will add, update and delete the student, subject, classroom no or lab no, etc. The admin can also view the generated timetable faculty wise or section wise.

2. Student : - A student can view the timetable and get the updates quickly by receiving the message from the admin side system. It becomes easier for the student to follow the timetable in this manner as minimum confusion and efficient timetable is generated.

3. Faculty : - A faculty can view the timetable section wise and get the updates quickly by receiving the message from the admin side system. It becomes easier for the faculty to follow the timetable in this manner as minimum confusion and efficient timetable is generated.

USER INTERFACES

1. Login Screen

- Login Screen will enable the admin to login to the admin account to edit or generate timetable or student and faculty details. Also, enables the student and faculty to view the timetable.

2. Administrator Page

- Admin can add or delete faculty, subjects and classrooms also have to allot subjects to teachers and allot classroom by satisfying all the constraints. Admin can view the timetable according to section. -Option screen will have two options:
IV. SYSTEM ARCHITECTURE

The block diagram in Fig. 2.1 gives an overview of the approach towards building a basic version of the intended features for Timetable.

The workflow for timetable generation works in following manner –

1. Admin will modify all the details of the student, faculty and subjects.
2. Admin will generate the timetable by providing the input as subject, faculty, type, etc
3. Admin will update the timetable and notify the student and faculty.
4. The timetable without any clashes and satisfying all the constraints will be generated.
5. Appropriate lab or class will also be allotted for the session.
6. Student and faculty will get the updated timetable message on their registered mobile number.
7. Student and faculty can login their account using their personal login.
8. The timetable can be viewed to the student as per their details provided.

Fig 1: System Design Of Timetable
V. CONCLUSION

This report addresses the procedure of the timetable generation which will be consistently easier with Automated System for Timetable Generation than using spreadsheet manually which might lead to constraints problem that are arduous to establish when timetable is generated manually. The Automated System for Timetable Generation is provided with necessary details of courses, classrooms and faculties which are stored in database (SQL server). The Automated System for Timetable Generation generates timetable based on these details in database with minimum time and satisfies all the constraints.

VI. REFERENCES

[1] Sara Ceschia... Andrea Schaerf, Educational timetabling: Problems, bench-marks, and state-of-the-art results (Survey), Submitted in 2021 , accepted in 2022