International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 3 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT206393

Automated Timetable Generation

Prajakta Tanksali[1], Ila Dhond[2], Shivani Pednekar[2], Varshanda Singbal[2], Shruti Sivaraman[2]
[1]Information Technology Department, Goa University, Goa, India
[2]Shree Rayeshwar Institute of Engineering and Information Technology, Karai Shiroda, Goa, India

ABSTRACT

Timetable generation is a complex job for the administration in terms of time and man power. The system proposed will help to generate it automatically thus saving time. It prevents the complexity of setting and managing Timetable manually. Different algorithms like genetic, heuristic, resource scheduling are used to minimise difficulties faced during generation of timetable. These algorithms consist of a numeral of strategy, aimed to improve the cooperativeness of the search operation. The system will take various inputs like number of subjects, professors, workload of a teacher, semester. Depending on these inputs, timetables for working days of the week for teaching staff will be generated. This will integrate by making optimal use of all the resources in a way that will best suit the constraints.

Keywords : Fitness function, Genetic Algorithm, Heuristic Algorithm, Score

I. INTRODUCTION

Although majority college organization work has been mechanized, the timetable preparation is still commonly done manually due to its expected difficulties. The manual timetable generation demands significant time and efforts.

Timetable Scheduler algorithm targets to design software for institutions in order to handle the "Timetable Formation" for the staff. The head of every department has problem in assigning work to their subordinates and response for the work position. It resolves the problem by permitting the lectures to see their assigned subject and timetable. This software helps to handle the particulars of the timetable of staff.

Most institutions have multiple streams and each stream has multiple subjects. Usually, there are limited faculties, each faculty teaching more than one subject. Therefore, the timetable needed to schedule the faculty at provided time slots in such a way that their timings do not overlap and the time table schedule makes best use of all faculty subject demands.

These objects comprise of classroom lectures and a fitness value for the timetable. Fitness value is related to the quantity of crashes the timetable has, regarding alternate calendars for different classes. Classroom object comprises of week lectures. Week objects comprise of days, comprise of timeslots.

There have been numeric approaches made earlier in the difficulty of setting timetables for institutions and universities. Timetabling problems may be solved by diverse methods inherited from operation study such as graph colouring, local search measures such as tabu search, simulated annealing, genetic algorithms or from backtracking based on constraint fulfilment handling. This problem is seen

as a constraint satisfaction problem and we discuss the various approaches that are capable of handling both hard and soft constraints. It is a complete timetable solution for institutions which will help to overcome the challenges in constructing the timetable manually.

II. LITERATURE REVIEW

Carter and Laporte (1998) considered diverse categories to solve the timetabling setback. They are – Cluster method, Sequential method, and Meta-Heuristics and Constraint Based method.

Meta Heuristics is a higher level procedure which is used to present comparatively improved solutions for optimization problems. On some class of problems, they do not guarantee a globally optimal solution. This method is used when the classical methods are too long or fail to provide a solution. This is achieved at the cost of optimality and precision for speed. Here, we consider the following Meta Heuristic methods.

III. PROPOSED PLAN

Timetabling Algorithm "Genetic Algorithm" is main component of our project which produces Netbeans based timetable even / odd semester sheet as the output. Our project takes various inputs from the user such as Lecturer List, Course List, Semester List, Room List, Day List, Meeting Time List and Timeslot as well as various constraints, facts. Constraints using Netbeans forms, which are stored in java based Database. This knowledge base serves as input to our Timetable Generator Algorithm residing on server machine. After the representation of Frontend is standardized, we designed the timetabling algorithm.

The designing of timetabling algorithm consumed a majority of time period. During designing of algorithm, the encountered problems were- from where to begin? Is it really going to work? But after all due to our superior design of knowledgebase, flowcharts and enough thinking on timetabling data structure representation helped us to really build our fine working algorithm.

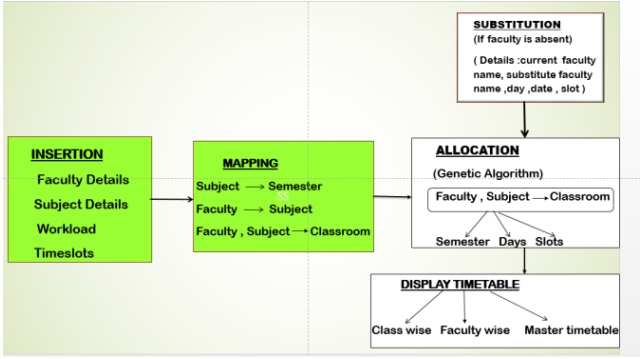

Fig 1: Block Diagram

Insertion Module:
In this module faculty details, subject details, workload per faculty and time slots of lecture is being inserted by the admin

Mapping Module:
In this module admin maps the subject to semester, faculty to subject and then faculty-subject data to classroom.

Substitution Module:
This module functions only if any faculty is absent. The admin provides the absent faculty id, substitute faulty id, day and date on which the particular faculty is absent to the allocation module.

Allocation Module:
In this module genetic algorithm is used to allocate the mapped faculty-subject and classroom data to time slots and generate the timetable.In case if any faculty is absent then it takes the details of the absent faculty and the substitute faculty, day and date of leave of the absent faculty from the substitution module and performs the changes and generates the timetable

Display Module:

This module displays the timetable which has been automatically generated.

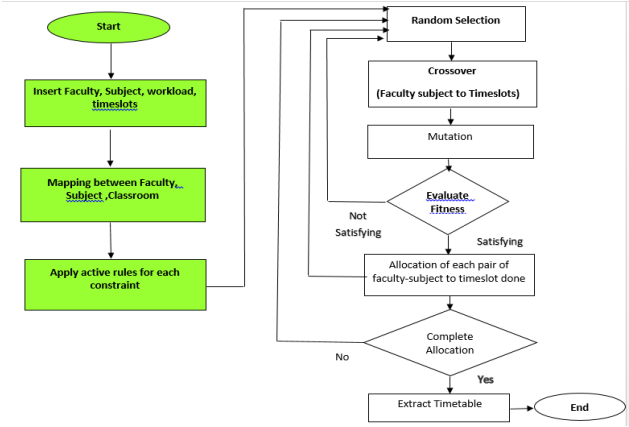

Fig 2: Flowchart of Automated Timetable working

Flowchart is a type of diagram that represents the workflow. Flowchart is the sequence of actions involved in the system. Flowchart begins with the start symbol. Admin inserts the faculty details, subject details, workload of the faculty and the lecture time slots. After successful insertion of all the details the admin maps subject to semester, faculty to subject and finally maps the faculty-subject data to classrooms. After mapping Admin applies constraints for the data that must be satisfied that is :.

- A classroom must not be assigned to more than one faculty at a time.
- A faculty cannot teach more than one class at the same time.
- Every subject hours and practical sessions should be adjusted for a week.
- Faculties should get equal hours of lecture in a week
- Lunch break should be allocated properly
- Faculties should not be assigned for same time slots.
- Then the genetic algorithm starts. It performs the following steps:

1) Random Selection Genetic algorithm randomly selects the mapped faculty-subject-classroom data and the time slots. Faculty-subject-classroom and time slots are like parents. So it does random selection of parents from large amount of data.

2) Crossover In this step it does the crossover of the selected faculty-subject-classroom and time slot data.

3) Mutation does random changes. This step is used only when the faculty is absent and needs to be replaced by the substitute faculty. So it does changes by replacing the absent faculty with the substitute faculty.

4)Evaluate Fitness In this step the fitness is evaluated, constraints are checked for the data which is assigned with the time slots. If the data which is assigned to the time slots does not satisfy the constraints then it is diverted to the random selection state, so that it can randomly selects new set of data then do the crossover and check for the fitness. If the data assigned to the time slots satisfies all the constraints then it is moved to the next state. So now the allocation of faculty-subject-classroom data to time slots is done after satisfying all the constraint. Complete Allocation In this step it is checked if all the pair of faculty-subject-classroom is assigned to all specified time slots. If all the faculty-subject-classroom is not assigned to all the time slots then it move to random selection state and randomly selects new set of data. If all pair of faculty-subject-classroom data is assigned to all the timeslots then it moves to the next state. Extract Timetable In this step it generates the classwise, facultywise, master timetable. This is the last stage that is the end state of the flowchart.

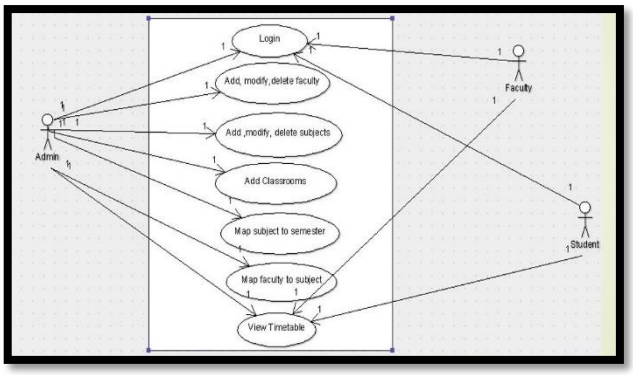

Fig 3 : Class Diagram for Automated Timetable Generation

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. It is also known as a structural diagram.

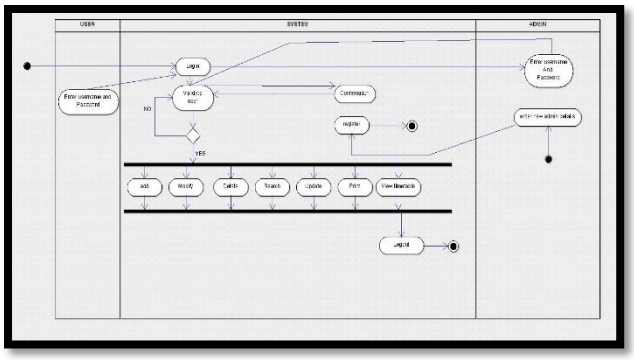

Fig 4 : Activity Diagram for Automated Timetable Generation

Activity diagram is basically a flowchart to represent the flow from oneactivity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

IV. Implementation

A. Frontend

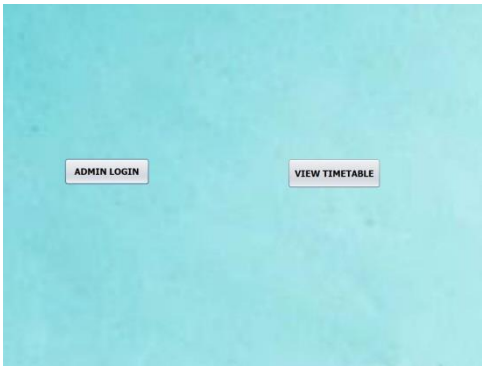

Fig 5 : Main Page of Frontend

The very First page of our GUI is the welcome page where a choice is given between viewing the timetable or to logging in for the admin into the configuration.

If clicked on admin login it redirects to the login page where the admin is required to enter the username and password.

Fig 6 : Admin Login Page

On Logging in the admin can enter the faculty and the subject details and can generate the timetable on one click. The forms are being displayed below.

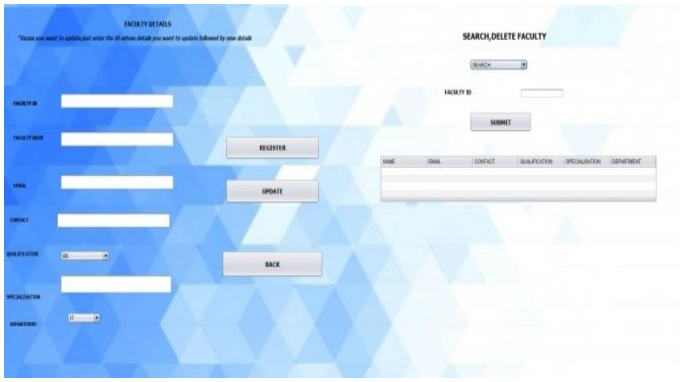

Fig 7 : Mapping Details of subject and lecturer

While entering the faculty details we take in all their basic information and a unique faculty id is assigned to them. We also provide the facility of updating and deleting a particular faculty by using their id. Similarly you can even retrieve the details of a particular student. Same goes with the subject details.

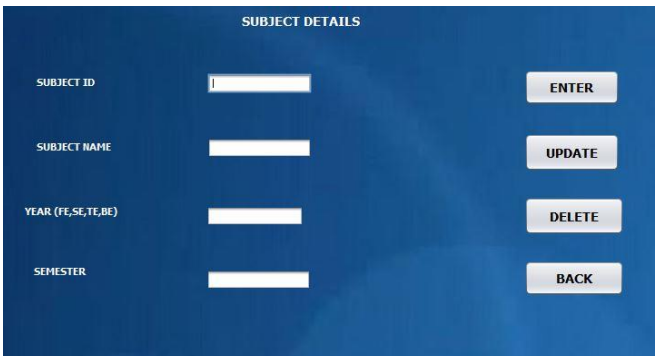

Fig 8 : Subject Details for frontend form

In order to view the generated timetable we need to click on the view timetable option and from there we will be able to view the class wise as well as the generated timetable.

Fig 9 : Viewing of Timetable generated

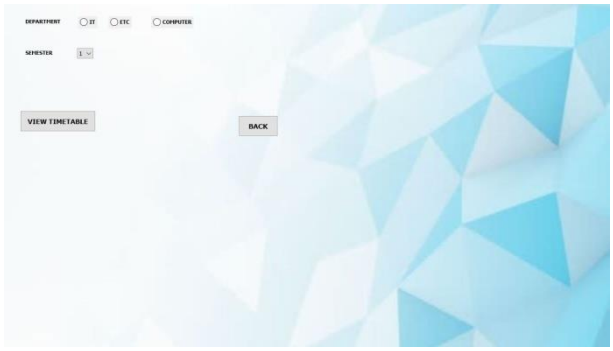

Fig 10 : Academic Details

B. Backend

Genetic Algorithm

Genetic Algorithms (GA) was invented by John Holland and has described thisconcept in his book "Adaptation in natural and artificial systems" within the year 1975. Genetic Algorithms are inspired by Darwin's evolutionary theory. GA comes under the category of Evolutionary algorithms that use the principle of survival to derive a group of solutions towards the optimal solution. It is an enquiry heuristic which generates solutions to optimization problems using techniques inspired by natural evolution like mutation, inheritance, crossover and selection. Here the algorithm is usually started with a group of candidate solutions called the population. Each solution within the initial population features a set of characteristics (its chromosomes or genotypes) which may be altered and mutated. Solutions from one population are taken and wont to make another population, with a hope that the new population are going to be better than the old one.

Solutions are selected for breeding on the idea of their fitness. A timetable is claimed to be healthier if it violates less number of constraints. In the timetable generation problem, the population may be a set of timetables maintained in memory. Each timetable is evaluated by finding the amount of times it violates the constraints. Each

and every timetable has a fair chance to participate in breeding.

Algorithm: Genetic Algorithm (GA)

[Start]Produce a random population of n chromosomes (suitable solutions for the problem)

1] [Fitness] Compute the fitness function f(x) of each chromosome x in the initial set of population.
2) [New Population] Generate a new population by repeating the steps given below until the new population is generated.
(a) [Selection] Select two parent chromosomes from the existing population on the basis of their fitness score. (Chromosomes with highest fitness score are selected.)
(b) [Crossover] Considering a crossover probability, crossover the parents to form a new solution of combination of both parents.
(c) [Mutation] considering a mutation probability, mutate new offspring at each locus (position in chromosome).
(d) [Accepting] Add the new child in the generated new population.
3) Use the newly generated population for further functioning of the algorithm.
4) If the end condition is satisfactory, return the existing best solution from the current population.
5) [Loop] Go to step 2.

The effectiveness of the genetic algorithm is mainly based on the fitness function. Mutation and crossover are the two main aspects of the algorithm. They are called the operators of Genetic Algorithm (GA). The task of crossover is the comprehension of the deterministic search. From the formerly selected pool, two parent solutions are selected for breeding. The new solution is obtained by the method of crossover and mutation and inherits many characteristics of the parent solution.

New set of parents are selected for every new child and this process continues until a population of suitable size is generated. Only the best solutions from the prior pool are selected for breeding, along with a small number of less fit solutions to ensure genetic diversity. The generation of new population can be stopped when a solution which satisfies the minimum criteria can be found. In this case, the process can be stopped when a timetable satisfying all the hard constraints is found.

Advantages: Diverse values of solutions are obtained and reaches global maxima. Mutation is used to induce diversity. Saving best solution is useful.
Disadvantages : It possesses implementation complexity.

The program for the timetable generation consists of 2 packages-cs and domain and 11 classes. The details of them are discussed below:

Package cs contains:
Data Class
Driver Class
Genetic Algorithm Class
Population Class
Schedule Class

Package domain contains:
1) Class Class
2) Course Class
3) Department Class
4) Instructor Class
5) MeetingTime Class
6) Room Class

The Class Class.java sets and returns the basic values and entities.
The class course.java sets and initialises the course details like Name, instructor maximum number of students etc.

Department and Instructor class contains the details of the department and the faculty respectively i.e it sets and initialises the retrieved value.

The Meeting Time class sets and initialises the meeting times i.e the timeslots. The room class contains the room details like number of rooms and seating Capacity.

The data.java class contains the details of Instructor, rooms, meeting time, course etc. This data is stored in an array-list where it is directly retrieved from the database. The population class calculates the population and the fitness and thus helping us to find the most optimal solution. The schedule class creates the schedules and checks for the conflicts.The entire genetic algorithm is being worked in the genetic algorithm class. The Driver Class is the main class where in the entire program gets compiled and run thus resulting in the generation of the timetable.

V. CONCLUSION

The major advantage of this project is to store information at one place and it can be accessed via transaction. Instead of manual paper work, students can view the timetable with a quick turnaround. This system is user friendly and provides faster and better generation of timetable, which in turn saves time and efforts. There are few key points that justify the need of same:

- Smooth user experience
- Faster and generation of potential timetable
- Efficient in time and manpower.

This application will simplify the process of time table generation smoothly which may otherwise needed to be done using spread sheet manually possibly leading to constraints problem that are difficult to determine when time table is generated manually. The intention of the algorithm to generate a time-table schedule automatically is satisfied. The algorithm incorporates a number of techniques, aimed to improve the efficiency of the search operation. It also, addresses the important hard constraint of clashes between the availability of teachers. The non-rigid soft constraints i.e. optimization objectives for the search operation are also effectively handled.

VI. FUTURE SCOPE

Given the generality of the algorithm operation, it can further be adapted to more specific scenarios, e.g. University, examination scheduling and further be enhanced to create railway time tables. Thus, through the process of automation of the time-table problem, many an-hours of creating an effective timetable have been reduced eventually. The most interesting future direction in the development of the algorithm lies in its extension to constraint propagation. When there is a value assigned to a variable, such assignment can be propagated to unassigned variables to prohibit all values which come into conflict with the current assignments. The information about such prohibited values can be propagated as well.

It is complicated task that to handle many Faculty's and allocating subjects for them at a time physically. So our proposed system will help to overcome this disadvantage. Thus we can produce timetable for any number of courses and multiple semesters. This system will help to create dynamic pages so that for implementing such a system we can make use of the different tools are widely applicable and free to use also It is complicated task that to handle many Faculty's and allocating subjects for them at a time physically. So our proposed system will help to overcome this disadvantage. Thus we can produce timetable for any number of courses and multiple semesters. This system will help to create dynamic

pages so that for implementing such a system we can make use of the different tools are widely applicable and free to use also It is complicated task that to handle many Faculty's and allocating subjects for them at a time physically. So our proposed system will help to overcome this disadvantage. Thus we can produce timetable for any number of courses and multiple semesters. This system will help to create dynamic pages so that for implementing such a system we can make use of the different tools are widely applicable and free to use also. It is complicated task that to handle many Faculty's and allocating subjects for them at a time physically. So our proposed system will help to overcome this disadvantage. Thus we can produce timetable for any number of courses and multiple semesters. This system will help to create dynamic pages so that for implementing such a system we can make use of the different tools are widely applicable and free to use also.

VII. RESULT AND ANALYSIS

The final system must be able to generate completely automated timetables which will save a majority of time and efforts of a department administration.

Focus on optimization and utilisation of resources i.e. teachers, classrooms etc. provide an ability for everyone to view the time table. This application is provided with necessary details of teaching staff and subjects which are stored in database and by making use of available data it generates timetable with minimum time as compared to the manual generation of timetable. The system developed by us will be able to generate timetable automatically in a short period of time thus having optimal time and space complexity. Our timetable system will be accessible to the admin, lecturers as well as the students. The subject details and the faculty details alteration will be strictly done by the admin and nobody else. The admin is the main and the primary user here who will have access to all the sensitive information, internal guidelines and rules. Being developed in java it gives usbetter and faster results. Considering the platform independence property it is feasible for us to install our application on any other system. Our system will access the data from the database when it wants to generate the timetable. The data is fed as an input by the admin himself.

VIII. REFERENCES

[1]. ijcsmc.com/docs/papers/February2014,V3I2201474.pdf. Online].

[2]. A Study on Automatic Timetable GeneratorAkshayputtaswamy, H M Arshad Ali Khan, Chandan S.V, Parkavi.ADepartment of Computer Science and Engineering, M S Ramaiah Institute of Technology,Bangalore.

[3]. https://www.youtube.com/channel/UCUvwlMMaeppKPdtAK8PxO8Q?reload=9

[4]. International Journal of Advanced Research in Computer Science and Software Engineering Research Paper Available online at:www.ijarcsse.com Automatic Time Table Generator 1 Saritha M, 2 Pranav Kiran Vaze, 3 Pradeep, 4Mahesh N R 1Assistant Professor, 2, 3, 4 UG Scholar 1, 2, 3, 4 Department of CSE, SDMIT Ujire, Karnataka, India

[5]. International Journal of Scientific & Engineering Research Volume 9, Issue 4, April-2018 31 ISSN 2229-5518 IJSER © 2018 http://www.ijser.org Automated college timetable generator Adithya R Pai, Ashwitha S, Raksha Shetty, Prof.Geethalaxmi

[6]. Automatic Timetable Generation System 1Deeksha C S, 2A Kavya Reddy, 3Nagambika A, 4Akash Castelino, 5K Panimozhi 1,2,3,4UG Student,5Assistant Professor 1Dept. of Computer Science and Engineering, 1BMSCE, Bangalore, India. (PHY) Specification, IEEE Std. 802.11, 1997.

Cite this article as : Prajakta Tanksali, Ila Dhond, Shivani Pednekar, Varshanda Singbal, Shruti Sivaraman, "Automated Timetable Generation", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 3, pp. 389-396, May-June 2020. Available at
doi : https://doi.org/10.32628/CSEIT206393
Journal URL : http://ijsrcseit.com/CSEIT206393