

International Journal of Scientific Research in Computer Science, Engineering and Information Technology © 2020 IJSRCSEIT | Volume 6 | Issue 2 | ISSN : 2456-3307 DOI : https://doi.org/10.32628/CSEIT206229

# A Review - Automated Timetable Generation

# Prajakta Tanksali, Ila Dhond, Shivani Pednekar, Varshanda Singbal, Shruti Sivaraman

Department of Information Technology, Shree Rayeshwar Institute of Engineering and Information

Technology, , Goa University, Goa

# ABSTRACT

Timetable generation is a complex job for instructors in terms of time and man power. Proposed system will help to generate it automatically along with 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, priority of subjects. Depending on these inputs, it will generate possible timetables for working days of the week for teaching staff. This will integrate by making optimal use of all the resources in a way that will best suit the constraints.

Keywords : Timetable Generation, Fitness Value, Genetic Algorithm, Chromosomes, Mutation

# 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 fulfillment 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. SCOPE OF PROJECT

Automatic Timetable Generation System generates timetable for each class and lecturers, following the constraint, availability of lecturers, availability and capacity of physical resources (such as classrooms, laboratories and computers) and rules applicable in different classes, semesters, professors and subjects level. Additionally, this Timetable Generation System immensely improves resource utilization and optimization.

#### A. PROPOSED SYSTEM

The fig1 represents timetable generator consisting of input module, and output module where timetable will be generated.

A. Input Data

The input data module can be depicted by a type of data given, the data contains:

- ✓ Person: which depicts the name of lecturers.
- ✓ Subject: which depicts the name of the subjects belonging to desired year and semesters.
- $\checkmark$  Room: which depicts the name of the class
- ✓ Time interval: it is a time slot with a starting time and duration.

#### B. Constraints

Constraints can be divided into 3 parts:

- ✓ Validity violation constraints
- ✓ Hard constraints
- ✓ Soft constraints

1) Validity violation constraints: These are the constraints which are needed to be followed :

✓ There are certain lecturers that may be assigned at the same time in more than one class.

- ✓ The most insignificant violation constraint is that a teacher must not clash in two different tables of a time table.
- ✓ Fixed slots.

2) Hard constraints: Hard constraints are the ones which need to be compulsorily fulfilled.

- ✓ Class room must not be booked twice.
- ✓ Every class must be scheduled exactly once.

3) Soft constraints: These are the constraints that are not compulsory but still demanding. They are not to be really satisfied but the solutions are generally considered healthy if large numbers of them are satisfied.

- ✓ No consecutive lectures of the same professor in the class.
- ✓ Minimize continuous lectures of the same subject in a day.
- ✓ Same lecturer must not have consecutive lectures unless specified.
- ✓ Assigning fixed slots for certain subjects.

#### III. LITERATURESURVEY

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.

#### a. Genetic Algorithm (GA)

Genetic Algorithms (GA) was invented by John Holland and has described this concept 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 1 : 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.

2) 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.

#### **IV. IMPLEMENTATION**

This system is implemented using the minimum hardware requirements such as RAM 512MB and above, hard disk used is 20GB or above, processor used is 2.4GHz or above, display is standard output display and data input is keyboard/mouse.

Software requirements need essential software resource necessities and fundamentals that need to be installed on a computer to provide correct functioning of an application. These requirements or fundamentals are generally not included within the software installation package and wish to be installed separately before the software is installed. Operating system is Windows 7/8/10.

#### V. RESULT AND ANALYSIS

The final system must be able to generate completely automated timetables which will save a lot 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.

#### VI. 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:

- $\checkmark$  Smooth user experience
- ✓ Faster and generation of potential timetable
- ✓ Efficient in time and manpower.

#### VII. REFERENCES

- [1]. A STUDY ON AUTOMATIC TIMETABLE GENERATOR Akshayputtaswamy, H M Arshad Ali Khan, Chandan S.V, Parkavi.A Department of Computer Science and Engineering, M S Ramaiah Institute of Technology,Bangalore ://www.researchgate.net/publication/326265336 \_A\_STUDY\_ON\_AUTOMATIC\_TIMETABLE\_ GENERATOR
- [2]. International Journal of Advanced Research in Computer Science and Software Engineering online Research Available Paper 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 https://www.researchgate.net/publication/32626 5336\_A\_STUDY\_ON\_AUTOMATIC\_TIMETAB LE\_GENERATOR
- [3]. 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

https://www.ijser.org/researchpaper/Automated-college-timetable-generator.pdf

[4]. 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.
http://www.jetir.org/papers/JETIR1504029.pdf

# Cite this article as :

Prajakta Tanksali, Ila Dhond, Shivani Pednekar, Varshanda Singbal, Shruti Sivaraman, "A Review : Automated Timetable Generation", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 2, pp. 96-99, March-April 2020. Available at doi : https://doi.org/10.32628/CSEIT206229 Journal URL : http://ijsrcseit.com/CSEIT206229