MJRET
Open Access
Multidisciplinary Journal of Research in Engineering and Technology
Journal homepage: www.mjret.in
ISSN:2348 - 6953

TIMETABLE SCHEDULING USING GENETIC ALGORITHM

Prof. R.V. Dagade[1], Sayyad Shaha Hassan[2], Pratik Devhare[3], Srujan Khilari[4], Swapnil Sarda[5]

Computer Department
Marathwada Mitra Mandal's College of Engineering
Pune, India
[1]rahuldagade@mmcoe.edu.in, [2]shahahassansayyad.comp@mmcoe.edu.in,
[3]pratikdevhare.comp@mmcoe.edu.in, [4]srujankhilari.comp@mmcoe.edu.in,
[5]swapnilsarda.comp@mmcoe.edu.in

Abstract: *The main objective of this project is to demonstrate our solution to the Timetable making problem. Timetable generation is a kind of problem in which events (classes, exams, courses, etc.) have to be arranged into a number of time-slots such that conflicts in using a given set of resources are avoided. There are usually a very large number of feasible solutions for it. Manual creation of timetable may lead to imperfections and partiality in the favor of a particular person or a group of people. Using Genetic algorithm, which revolves around the process of natural selection, we optimize this problem by choosing the best solution among the available.*

Keywords: *Genetic algorithm, Timetable, Optimization*

1. INTRODUCTION

The class timetabling problem is a typical scheduling problem that appears to be a tedious job in every academic institute [1]. Earlier the task was performed manually.

The problem with doing it manually is that as the number of variables increases the complexity and difficulty of the task grows exponentially. It becomes very tedious and time consuming as the person has to remember and manage all the constraints.

The other major problem is the human bias. A human may make a biased timetable which favors some and is against others. It is prone to human errors which may come up because of human lack of attention.

Using Genetic Algorithm, a number of trade-off solutions, in terms of multiple objectives of the problem, could be obtained very easily. Moreover, each of the obtained solutions has been found much better than a manually prepared solution which is in use. [2] There are no human errors and the timetable thus generated is neutral towards all.

2. LITERATURE SURVEY

Usually timetable is scheduled manually in schools, colleges and in universities. Scheduling timetable manually is hectic and time consuming. As timetable is created by the human being there may be possibilities of error in the timetable.
Observations about existing system are:
1. Generating Timetable manually is highly prone to human errors.
2. Stress on human to satisfy all the constraints.
3. Problem in satisfying time critical constraints.

Hard constraints involved in existing system:
1. No participant can be in more than two rooms at the same period.
2. No room should be double booked.
3. The room capacity should be large enough to hold each.

3. PROPOSED APPROACH

Genetic algorithm mimics the process of natural selection and can be used as a technique for solving complex optimization problems which have large spaces [3]. Rather than starting from a single point within the search space, GA is initialized to the population of guesses. These are usually random and will be spread throughout the search space [2].

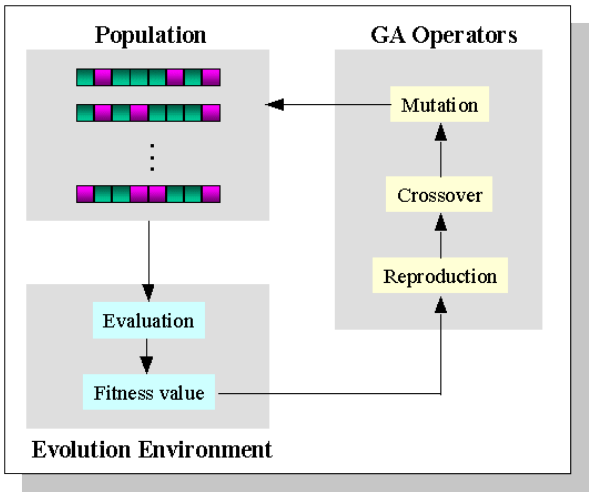

Fig.1: General Genetic Algorithm [4]

3.1 Genetic Algorithm Operators

1. **Chromosome**: A chromosome is the basic data structure used in the genetic algorithm to represent one solution. Each different combination of the bits

represents a different phenotype. A phenotype is the physical representation of the trait or gene. Chromosomes are also referred to as individuals or organisms.

2. **Initial population**: A population is the collection of chromosomes. The initial population is generated by using random strings of 1's and 0's. Thus, they may not represent a valid solution, but they are further worked upon to produce an optimal solution.
3. **Mutation**: Mutation is doing minor random changes in the chromosomes. This is done rarely but this helps in generating a chromosome which may not be possible by simple crossovers.
4. **Crossover**: A crossover is done by mixing more than one chromosomes. Usually two chromosomes are chosen and some of their parts are swapped. Crossover without mutation will eventually lead to the same individuals getting repeated.
5. **Fitness function**: A fitness function evaluates the fitness of an individual. The chromosome with the highest level of fitness is closest to the required solution.

Structure of time table generator consists of input data, relation between the input data, system constraints and application of genetics algorithm.

3.2 Input Data

The input data contains:

1) Professor: Data describes the name of lecturers along with their identification number.
2) Subject: Data describes the name of courses in the current term.
3) Room: Data describes the room number and their capacity.
4) Time intervals: It indicates starting time along with duration of a lecture.

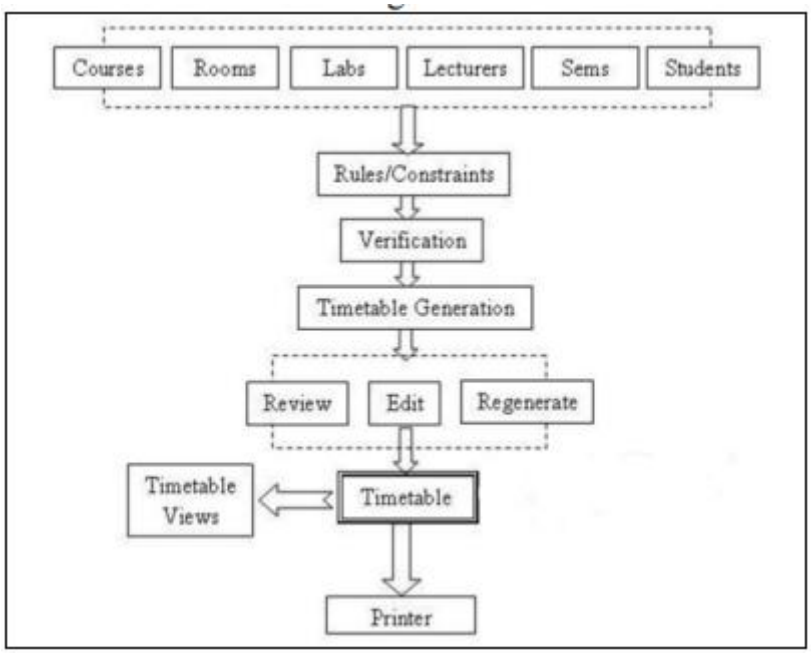

Fig.2: General View of Time Table generator [5]

3.3 System Constraints System constraints are

1) The timetable is subjected to the following four types of hard constraints, which must be satisfied by a solution to be considered as a valid one:

a. A student should have only one class at a Time.

b. A Teacher should have only one class at a time.

c. A room should be booked only for one class at a time.

d. Some classes require classes to have particular equipment. For example, audio visual equipment, projectors etc.

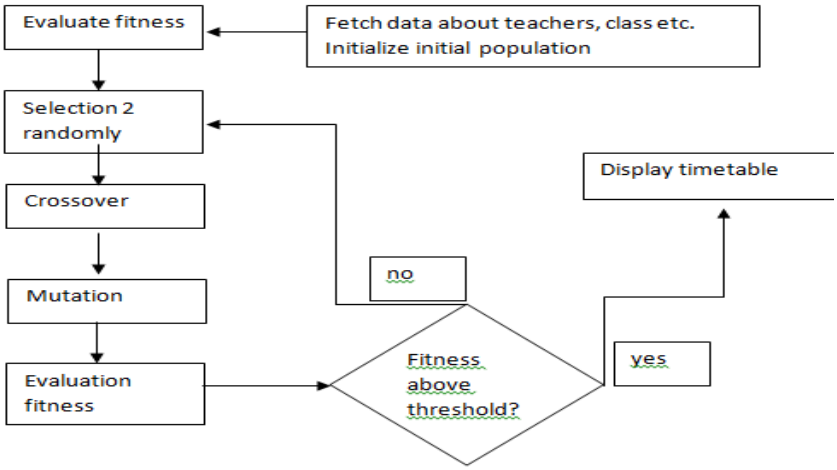

Fig.3: Genetic Algorithm for TTS

4. FUTURE SCOPE

The algorithm can be used to generate timetables in schools and colleges. Soft constraints can also be considered for calculating the fitness which will make the algorithm more flexible, appealing and helpful.

5. CONCLUSION

The timetables generated using the algorithm are much more neutral and efficient than those made by a human. The mechanical approach saves a lot of time and is much more convenient than the conventional methods.

There are fewer errors in such timetables. Though a manually made timetable can be more flexible than the machine generated but this approach saves a lot of time.

Thus, a combination of the algorithm along with manual approach is the best. A timetable generated using the algorithm can be tweaked by a human to make slight changes that may be preferred.

ACKNOWLEDGEMENT

We are thankful to our HOD, Prof. H.K. Khanuja, and project guide Prof. R.V. Dagade for providing us the opportunity and the support for making the project. We are also grateful to Prof. Jagruti Wagh for explaining us in detail about timetable generation.

REFERENCES

[1]. DilipDatta, Kalyanmoy Deb, Carlos M. Fonseca, "Solving Class Timetabling Problem of IIT Kanpur using Multi-Objective Evolutionary Algorithm".KanGAL 2005.

[2]. Dipesh Mittal , Hiral Doshi, Mohammed Sunasra , Renuka Nagpure, " Automatic Timetable Generation using Genetic Algorithm ", IJARCCE, 2015.

[3]. D. Abramson, J.Abela,"A Parallel Genetic Algorithm for Solving the School Timetabling Problem", 15 Australian Computer Science Conference, Hobart, Feb 1992.

[4]. Ying-Hong Liao, and Chuen-Tsai Sun, "An Educational Genetic Algorithms Learning Tool"Member, IEEE,2001

[5]. AnujaChowdhary, PriyankaKakde, ShrutiDhoke, SonaliIngle,RupalRushiya, Dinesh Gawande, 'Time table Generation System", International Journal of Computer Science and Mobile Computing, Vol.3 Issue.2, February- 2014.