

Improved Web Application For Automatic Timetable Generator

Karpagavalli.K¹, Deepika.S.P², Jeeva.E³, Mohammed Mushraf.A⁴

¹Assistant Professor, Dept of CSE

^{2, 3, 4}Dept of CSE

^{1, 2, 3, 4} Sri Ramakrishna Institute of Technology, Coimbatore, India.

Abstract- Most colleges have different types of courses and each course has different subjects. In an institution, there are some staff like teaching, non-teaching, and lab assistants, each faculty teaching more than one subject. So, the timetable is scheduled for the faculty at provided times such that the timings of the staff do not overlap and the timetable we would schedule that makes the best use of all faculty and the subject's demand. For these things, the manual creation of a timetable takes more time. So, we use a genetic algorithm for this purpose. In our algorithm of Timetable Generation, we propose to utilize a timetable object. The number of timetable crashes involving alternate calendars for certain classes affects fitness scores. Classroom-object consists of weak objects. Week objects consist of Days, Days also consist of Time-slots. The system will get more inputs like staff, subjects, and the number of subjects that a faculty can teach, depending upon these inputs the system will generate possible timetables, making the optimal use of all the resources which is that best suit the constraints. And the best timetable is then chosen from the optimal solutions generated.

Keywords- Constrains, Genetic Algorithm, Time tabling, Scheduling, Timetable Generator

I. INTRODUCTION

Manual programming of generating timetables requires a lot of time and effort. In our project, the timetable will be automatically scheduled. The system aids in solving the timetabling problem while giving importance to teacher availability. It will help to manage all the periods automatically and also will be helpful for faculty to get a timetable on their phone by using the web application. Timetable scheduling has been in human requirements since all thoughts of managing time effectively. The timetable is very important and it is widely used in schools, and other fields of teaching and working like crash courses, coaching centers, training programs, etc. It helps the management to avoid collapse in classes taken for the students. In the early days, timetable scheduling was done manually with one single person or some group involved in the task of scheduling it with their hands, which takes a lot of effort and time. The timetable is used to do a job at an appropriate time and place.

A key factor in running an educational center or basically academic environment is the need for a well-planned and clash-free timetable. In previous days, when technology was not mostly used. So, everyone is faced with the tedious task of drawing up timetables. They need to schedule their course to meet the need of the current duration and facilities that are available to them. However, their schedule should meet the requirement of the new course additions and newly enrolled students to fresh batches. This may result in rescheduling the entire table once again for its entire batches and being scheduled in the shortest possible time before the batch courses start. Another problem that occurs when scheduling time table for exams, is when multiple batches have their exams on the same day, they need to be scheduled effectively taking into account all problems related to the facilities that are available to conduct the exam simultaneously. Our aim here is to develop a simple, easy, understandable, efficient, and portable application that could automatically generate good-quality timetables within time.

1.1 BACKGROUND HISTORY

Assuming that you've scanned the web for 'timetabling programming', 'automatic timetable generator', 'school timetable generation', and so on you most likely seen that there are none of the 'enormous names', like Apple, Microsoft, Google, and so on. in the list items. Moreover, school planning programming can't be found as a feature of any bigger programming bundle for schools. Neither does it exist in different schedulers, schedules, calculation sheets, and so forth. It is muddled to cause a machine to dispense examples that specific speakers are to educate to specific gatherings of understudies in specific rooms and certain time-frames... what's more, that all works out well eventually. It is muddled to the point that, by and large, there is no arrangement. It is muddled to the point that main fans working in the field of hereditary calculations have been attempting to track down an answer for this issue. In 1995, the first Global Gathering for the Training and Hypothesis of Robotized Timetabling was held at Napier College, Edinburgh, UK. It was the establishing meeting with another one being held each third year and later every other year. Ways there have been eight gatherings and the 10th one will be held at Child,

Norway, from 28th to 31st August 2012. To more readily comprehend the intricacy of the issue of 'PC helped schedule age', you might investigate the authority site of the 10th gathering. In any case, not so much as a progression of meetings introducing the most elevated scholastic names from everywhere in the world was sufficient to adapt to such a troublesome issue.

1.2 PROBLEM STATEMENT

Normally timetable generation is done manually. As we know all institutions or organizations have their own timetable and managing and maintaining these will not be difficult. Considering the workload with this scheduling will make it more complex. To avoid the complexity of setting and managing the Timetable manually we make an application to generate the timetable automatically. Initially, we will be setting the maximum workloads for a faculty in a day, week, and month. The main challenge is to manage the timetable when any Faculty is absent. In these cases, timetable generation will become more complex. Also, it is a time-consuming process. In this system we intend to display the Timetable as per the faculty is been allotted for the respective subject in accordance with that rooms will be available. The most common issues faced during time-table scheduling can be represented as a constraint-satisfying problem with multiple constraints. As mentioned, the generation of a timetable automatically is a little complex because it should obey all the hard and soft constraints. The goal of portraying these constraints in an organized fashion is attained by replicating the format by the algorithm. Multiple tasks can be achieved simultaneously by allowing these constraints to be paired in a number of different ways. It is desirable for timetables to satisfy both hard and soft constraints. However, it is classically not easy for all these constraints to be satisfied. Soft constraints can be abandoned in order to find a convenient solution. The hard constraints though, are a must to be met at all times. When all these constraints are clear the automatic timetable generation is work as a web application. Faculty can receive their period information on their phone itself.

1.3 APPLICATION

The applications of automatic timetable generators are more. It is very useful generate time table manually with less consumption of time and manpower.

1. **Save time and effort:** When the timetable is generated automatically it saves a lot of time and effort for a staff who is making the timetable manually.

2. **Secure and user-friendly:** The automatic timetable generator is a user-friendly application and it is more secure.
3. **Easy calculation:** When we give the details of staff and subjects it generates the timetable so it is very easy to generate the timetable.
4. **Paperless environment:** This automatic timetable generator is a web-based application so it does not need any paper or pen for a generation.
5. **Accurate information:** Because it is a web-based application it gives a perfect and accurate timetable.
6. **Reduces error:** It is a web application so it does not have more errors than it is done manually.
7. **Easy access:** It is a web-based application so it can be accessed easily and can be viewed at any time and at any place.

1.4 SCOPE OF THE PROJECT

Automatic Timetable generator is a web-based software used to generate timetables automatically. It can be used by educational institutes or colleges to view their timetable in the most efficient and easy manner. This system will help to generate it automatically and also helps save time. It will help to manage all the periods automatically and also will be helpful for faculty to get timetables by using their web application. It is a comprehensive timetable management solution for colleges that helps to overcome the challenges in the current system.

1.5 EXISTING SYSTEM

Previously Time Table was made manually by the head of the departments which involves a lot of manual work and time-consuming processes. So, it takes a lot of time, a lot of manpower energy, and paper wastage, and also there will be also clashes between the classes if there is a new teacher added. So, to overcome all these problems this system has been developed to generate automatic timetable.

1.5.1 Limitations of manually made timetables:

- The existing system is a manual system.
- Needs to be converted into an automated system.
- Risk of mismanagement of data.
- Less Security.
- Not user-friendly.
- No proper coordination between different Applications and Users.
- Accuracy is not guaranteed. Not in reach of distant users.

To avoid all these limitations and make the work more accurate the system needs to be computerized.

1.5.2 Advantages of Existing System:

- Can be made better through collaboration with the different entities involved as it is subjective.
- Changes may be done faster on time only for specific conditions and not for all.

1.5.3 Disadvantages of Existing System:

Very confusing and time-consuming method for manual creation, as well as a hectic job for lecturers as a lot of effort, is put in by them for it.

1.6 PROPOSED SYSTEM

The proposed system is made to be more effective than the current manual system. It invokes all base tasks that are now carried out manually, such as the forms of transactions and reports which is added advantage. Most colleges have a number of different courses and each course has an 'n' number of subjects. There are currently fewer faculties, and each faculty may teach many subjects. So now the timetable needed to schedule all the faculty at provided time slots in such a way that their timings do not overlap and the timetable schedule will make the best use of all subjects demanded by the faculty. So, we use a customized type of algorithm for this purpose. In our Timetable Generation algorithm, we initiated the utilization of a timetable object. This object includes classroom objects, a schedule for them, and a fitness rating for the schedule. The fitness score relates to the number of crashes the timetable has regarding alternate calendars for all the different classes. Classroom objects have weak objects. Week objects have Days, and Days have Time slots. The time slot has an address in which a subject, student gathering going to that particular address, and educator showing to the subject it is related will be shown. Also further on discussing, we have utilized composite configuration (design), which makes it well extendable for numerous obligations. In every obligation class, the condition as determined in our inquiry is now checked between both the timetable objects. On the off chance that condition is fulfilled, there is a crash is available then the score is augmented by one.

1.6.1 Advantages of Proposed System:

- Unlike the existing system, it Saves time and effort It also reduces confusion and error.

- It simplifies the process and gives easy customization, smooth integration, and easy calculation.
- It provides a paperless environment.
- 60-80% optimum timetable gets generated.

1.6.2 Disadvantages of Proposed System:

- The proposed system can generate timetables based on only a few course constraints.
- The proposed system can only generate timetables for a particular class at a time.
- In the system, most of the genetic algorithm principles are implemented but not all of them are covered totally.

II. SYSTEM SPECIFICATIONS

2.1 Constraints:

- Avoid scheduling a teacher's classes at the same time so they cannot be in two places at a time.
- There should be at least a one-hour break between two courses of a teacher.
- Some teachers prefer to teach at particular hours or on specified days.
- The timetable must be customizable.
- Lab hours staff should not get clashes.
- If a staff is handling two subjects the class for those two subjects should not clash.
- More than one staff should not attend the same class.
- Class for teachers should not be continuous.
- Lab hours and class hours should not clash.
- Allocations of two or more classrooms are not made for the same course for a given student group.
- All the necessary resources such as the staff, rooms etc. should be available for each time slot.
- All the available courses must be entered in the timetable with flexibility for multi-period sessions.

2.2 Genetic algorithm:

A heuristic search technique called a genetic algorithm is employed to solve optimization issues. The Darwinian theory of evolution serves as its foundation. In this method, each chromosome stands in for a possible solution to the timetabling problem, and the fittest (best schedule) is ultimately chosen from among the population of possible answers. A fitness function based on hard and soft restrictions assesses the optimal (perfection) of a chromosome. In order to choose the parents (timetables) for the following generation,

which is expected to produce better timetables through crossovers and mutations, genetic algorithms first create a random population of timetables. They then evaluate this population of timetables according to defined criteria. Up till a satisfying conclusion is found, the process is repeated. Using a genetic algorithm. A search heuristic called a genetic algorithm was influenced by Charles Darwin's notion of natural evolution. The fittest people are chosen for reproduction in order to give rise to the next generation's children, which is how natural selection works.

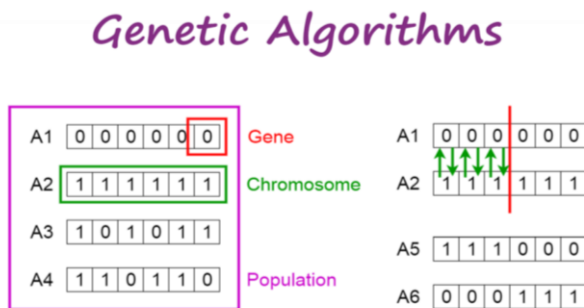
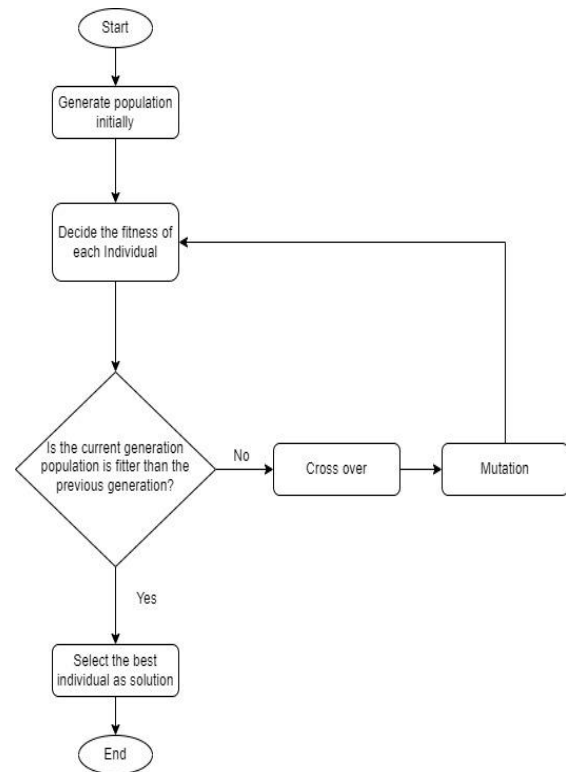


Figure 2.2: Genetic algorithm



2.2.1 Phases of Genetic algorithm:

- Initial population
- Fitness function
- Selection
- Crossover
- Mutation

2.2.2 Working of Genetic algorithm:

- The process of creating a starting population of chromosomes.
- Assessing each chromosome (individual) that makes up the population for appropriateness.
- Using the findings from the above, choose the chromosomes for mating.
- By mating (crossing over) the chosen chromosomes, one can conceive offsprings.
- Random gene mutation.
- Until a new population is produced, repeat steps 3-5.
- When the best answer obtained after the current number of generations has not changed, the algorithm is terminated.

III. CONCLUSION

Trying to develop software that helps to generate a Timetable for an Institution automatically. By viewing the existing timetable system, we can understand that the timetable is generated manually. Manually adjusting the timetable when any of the faculty is absent then is a big challenge for the Automatic Timetable Generator that manages the timetable automatically when any of the faculty is absent. As we know all institutions or organizations have their own timetable and managing and maintaining these will not be difficult. Considering the workload with this scheduling will make it more complex. In those cases, generating a timetable will become more complex. And also it is a very time-consuming process.

REFERENCES

[1] Anuja Chowdhary “Timetable Generation System”. Vol.3 Issue.2, February- 2014, pg.
 [2] Antariksha Bhaduri University Time Table Scheduling using Genetic Artificial Immune Network 2009 International Conference on Advances in Recent Technologies in Communication and Computing
 [3] Saritha M, Pranav Kiran Vaze, Pradeep, Mahesh NR, “Automatic time table generator”, International Journal of Advanced Research in Computer Science and Software Engineering, May 2017.

- [4] Sadaf N. Jat, Shengxiang Yang A mimetic algorithm for university course timetabling problem 2008 20th IEEE International conference on tools with artificial intelligence.
- [5] Dipti Shrinivasan automated time table generation using multiple context reasoning for university modules Published in: evolutionary computation, 2002. cec '02. proceedings of the 2002 congress on (volume:2)
- [6] Shraddha Ambhore, Pooja Walke, Rohit Ghundrudkar, Akshay Alone, Anushree Khedkar. "Automatic Timetable Generator." International Journal of Research in Engineering, Science and Management Volume-3, Issue-3, March-2020.
- [7] Milind Deshkar¹, Mayur Kale², Mashkur Deshmukh³, Mandar Aney⁴, Arpita Ghom⁵. "Timetable At a Click". International Research Journal of Engineering and Technology. Volume: 03 Issue: 03 | Mar-2016.
- [8] Mrs.G.Maneesha¹, T.Deepika², S.BhanuSri³, N.RaviKumar⁴, P.SivaNagamani⁵. "Automatic Timetable Generator using Genetic Algorithm". JETIR July 2021, Volume 8, Issue 7.
- [9] Parkavi A. "A Study on Automatic Timetable Generator" International Journal of Innovative Research & Growth · May 2018.