Enhancing Timetable Generation Through Machine Learning

Prof.Rupali Kaldoke
Department of Computer Engineering
Nutan Maharashtra Institute of Engineering and Technology.
Pune, Maharashtra

Gagan Matkar
Department of Computer Engineering
Nutan Maharashtra Institute of Engineering and Technology
Pune, Maharashtra

Gaurav Bhalerao
Department of Computer Engineering
Nutan Maharashtra Institute of Engineering and Technology.
Pune, Maharashtra

Prasad Adhav
Department of Computer Engineering
Nutan Maharashtra Institute of Engineering and Technology.
Pune, Maharashtra

Abstract**—** The process of manually creating timetables in colleges with a large number of students is very time-consuming and often results in scheduling conflicts, with classes clashing in timing or room, or teachers having more than one class simultaneously. This manual workflow leads to many system issues and restrictions. The organization cannot meet its needs in a timely manner and the outcomes may also lack accuracy, mainly due to common human errors that are difficult to prevent in such processes. To resolve these issues, we propose developing an automated system. The Automatic Timetable Generator system would take various inputs such as faculty, student, subject details, and based on these inputs, generate a feasible timetable that optimally utilizes all resources to best fit the specified constraints or college policies. The Adaptive Timetable Generator system is an automated system that produces timetables according to the data provided by the user. The main requirements of the application are to collect details about the branch, semester, subjects, labs and total number of periods. The list of subjects may include electives and core subjects, with students needing to choose their electives. The application then generates the timetable according to specifications.

Keywords**—**Automated, time-table, constraints, college, clashes.

I. Introduction

Generating college timetables manually becomes highly time-intensive for institutions with large enrollments, often culminating in problems like class timing clashes or same room bookings. The main goal of an adaptive timetable generator system is to create a timetable based on user input. To mitigate these scheduling conflicts, we recommend developing a computerized system that can automatically create timetables. Timetables are an important part of college management systems, as they help maintain discipline on college premises. Timetabling involves all activities related to generating a schedule subject to various constraints. The objective of an adaptive automatic timetable generator is to save time and effort, reduce errors, and create a paperless system. An important aspect is developing a user-friendly interface that enables smooth integration.

II. Objective

The automatic timetable generator system's primary goal is to generate a schedule using user input. An automatic timetable generator aims to eliminate mistakes, save time and effort, and foster a paperless atmosphere. creating an intuitive user interface that facilitates seamless integration.

III. Literature Survey

A comprehensive review of the automated timetabling systems, which include a variety of problem representations, solution methodologies and real world applications, is provided in this classic work. Based on different constraints and objectives, such as allocation of resources, room assignment, and teacher preferences, it classifies timetabling problems. A broad range of techniques for solving problems, including constraint programming, evolutionary algorithms and mathematical optimization approaches are examined in the paper[1]. This study offers an in-depth investigation of mechanized timetabling procedures, centering on constraint-based, chart coloring, and developmental calculations. It looks at the qualities and confinements of each approach, highlighting their appropriateness to diverse sorts of timetabling issues. The paper moreover talks about the integration of machine learning and optimization procedures in timetabling frameworks, emphasizing the require for versatile and brilliantly planning arrangements. Besides, it addresses viable contemplations such as information representation, arrangement quality evaluation, and client interface plan in timetabling computer program [2]. This review article provides an overview of recent advances in planning algorithms and software tools.It discusses emerging trends such as the hybrid metaheuristic approach, which combines multiple optimization techniques to improve the quality and efficiency of solutions .The paper also explores the integration of artificial intelligence techniques, including machine learning and constraint satisfaction, into timetabling systems. rephrase additionally, it emphasizes the importance of benchmarking and standardization in evaluating the performance of planning algorithms and encourages collaboration between researchers and practitioners in their development. Innovative planning solutions[3].

Centering particularly on college timetabling, this overview recognizes common imperatives and targets in scholastic planning issues. It analyzes issues such as course scheduling, room task, and workforce inclinations, emphasizing the complexity of college timetabling compared to other planning spaces. The paper audits different arrangement techniques, counting scientific programming, metaheuristic optimization, and constraint-based thinking, and examines their pertinence to distinctive viewpoints of college timetabling. Moreover, it addresses down to earth challenges such as taking care of large-scale datasets, pleasing last-minute changes, and fulfilling assorted partner prerequisites in college planning [4]. This study centers on the application of half breed metaheuristic calculations to college course timetabling issues. It surveys diverse hybridization methods, such as joining hereditary calculations with neighborhood look, reenacted strengthening with tabu look, and insect colony optimization with imperative engendering. The paper examines the focal points of cross breed approaches in adjusting investigation and misuse amid the look handle, driving to moved forward arrangement quality and meeting speed. Moreover, it highlights later improvements in crossover metaheuristics, such as versatile and self-adaptive calculations, and their potential for tending to complex college timetabling challenges successfully [5.] This survey addresses the issue of school scheduling and examines how constraints in scheduling optimization have evolved from hard to soft. Preference-based optimization techniques, which enable stakeholders to express flexible scheduling preferences and priorities, are discussed as a replacement for traditional constraint-based approaches. Penalty functions, constraint relaxation, and preference aggregation are a few techniques that can be used to introduce soft constraints into the timetabling process. Furthermore, in school scheduling applications, it looks at how soft constraints affect user satisfaction, computational complexity, and solution quality. [6]. This study attempts to concurrently optimize competing goals, such as reducing student conflicts, room usage, and faculty workload, by focusing on multi-objective optimization techniques used to university timetabling problems. The paper examines and evaluates several evolutionary algorithms that are capable of handling multiple objectives in an efficient manner. These algorithms include differential evolution, particle swarm optimization, and genetic algorithms. Taking into account variables like dominance relationships, diversity, and convergence, the paper also looks at methods for assessing and contrasting Pareto-optimal strategies. [7]. This survey addresses nurse rostering issues, which are similar to academic timetabling issues in that they involve shift scheduling and resource allocation. However, it is not directly related to academic timetabling. It examines several optimization methods, such as integer programming, constraint programming, and metaheuristic algorithms, that are used in nurse rostering. The study explores the unique goals and limitations associated with nurse rostering issues, including staffing needs, skill mix restrictions, and employee preferences. It also addresses the difficulties in creating efficient scheduling programs for healthcare institutions.[8]. Creating recurring schedules for teacher-student conferences while accounting for both required and optional courses is one way to handle scheduling issues in high school. A great deal of research has been done on this issue family since the 1950s, primarily with mixed-integer programming and met heuristic techniques. Efficiently searching for optimal or nearly optimal solutions is still a challenge for numerous practical problems. We examine mixed-integer programming formulations and parallel met heuristic-based algorithms in this study to address compactness and balance criteria in high school scheduling problems. Along with a solution method that builds and refines solutions simultaneously using a set of metaheuristics and column generation, we offer two pattern-based formulations.[9]. The subject of school scheduling optimization has been extensively studied due to its importance in both theory and practice. This involves arranging teacher-class sessions within a specific time period to meet different criteria. The complexity of timetabling makes it challenging to find optimal solutions for medium and large cases, given the combinatorial nature of the problem. Various strategies have been developed in the literature to tackle high school scheduling issues, with metaheuristics and hybrid metaheuristics being the most commonly used approaches for solving complex real-world problems, as more precise methods like mathematical programming techniques are considered impractical [10]. It takes a parcel of work to make a college plan with a few branches, different a long time, and various bunches. It requires a part of time and is fundamental for each six-month semester. This procedure in this manner requires a huge sum of time and labor. The manual approach of making plans is as well difficult in certain circumstances. When a workforce member is on leave, it isn't practical to form transitory plans. In this investigate, we have created an calculation to build plans, which may soothe a incredible bargain of the labor- seriously and time-consuming nature of the manual assignment..[11]. With an accentuation on "standard" definitions and the related benchmark cases, we propose conducting a consider of the inquire about commitments made within the zone of instructive planning. We list six of these definitions and conversation around their characteristics, emphasizing their appropriateness and value. A brief diagram and talk of extra open details and datasets is included. At that point, we display the essential state-of-the-art discoveries on the chosen benchmarks for running times, look methodologies, arrangement quality (upper and lower limits), and other side parameters [12]. This overview gives an in-depth investigation of the current state-of- the-art in timetable administration frameworks. It audits existing program arrangements and assesses their highlights, functionalities, and convenience. The paper talks about the challenges confronted by instructive teach in timetable administration and investigates how advanced advances such as cloud computing, machine learning, and information analytics are being utilized to address these challenges[13]. College timetabling may be a profoundly compelled problem that requires impressive time and exertion within the handle. Partners confront a serious situation when coming up with a timetable manually, as various clashes may emerge. This requires the have to be plan an productive robotized implies of getting freed of the boisterous and overwhelming assignments included in examination timetables and invigilation planning[14].Making an instructive institution's plan is still a troublesome undertaking. This approach proposes an robotized strategy for making timetables that analyzes college specific information. It employments a information category strategy to classify the department-specific time planning as well. This thing incorporates the classroom things, their plan, and a wellness rating for the plan. In arrange to guarantee that staff members' plans don't struggle which the timetable arrange completely utilizes all staff subject requests, a timetable was required. The college's organization, the vital, and the head of office will all show the timetable plans. Ought to an worker be missing, a substitution will be relegated by the admin[15].

IV. CONSTRAINTS

Limitations, guidelines, or requirements that must be followed during the scheduling process are referred to as constraints in the context of a Time Table Management System (TTMS) or any scheduling system. These limitations guarantee that the created schedule satisfies particular specifications and conforms to organizational rules or preferences. Depending on the type of scheduling issue and the requirements of the institution or organization, there may be different constraints [1].

Hard Constraint

- There should be no teacher teaching more than one class at once.
- Classroom capacity should be taken into account.
- There isn't more than one lecture scheduled in the classroom at once.
- Teachers ought to attend no more than two or three lectures each day.
- Collapsing lectures is not appropriate.

Soft Constraints

- Intervals between lectures for breaks.
- Lunchtime is preferred

V. SYSTEM ARCHITECTURE

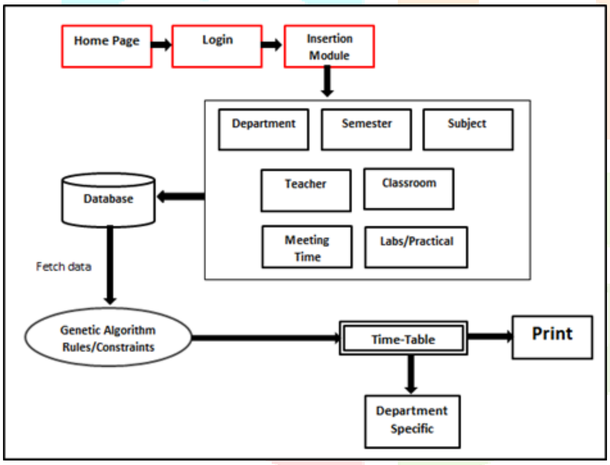

Fig. 1. System Design

User Interface:
The front-end part of the timetable management system that communicates with users is called the user interface. Users can check timetables, submit requests, and interact with the system via a graphical user interface. Features like course selection, room allocation, schedule customization, and timetable display are parts of an intuitively designed user interface.It sends user input and actions to the backend modules so they can process and update them.

Insertion Module:

The timetable data is added and modified by the insertion module in response to user inputs or system requirements.It has features like adding classes, designating teachers, setting up time slots, assigning rooms, and controlling limits (such the largest class size and the favored timeslots).This module takes user preferences and system needs into account while making sure the schedule is realistic and follows rules.

Database:

All pertinent information needed for timetable management, such as details on classes, teachers, rooms, time slots, and restrictions, is kept in the database component. It offers an organized method of storing data that makes data retrieval and manipulation more effective. Depending on the scalability and complexity requirements of the system, the database may employ a NoSQL database like MongoDB or a relational database management system (RDBMS) like MySQL, PostgreSQL, or SQLite.

Genetic Algorithm:

One optimization method used to create and enhance timetables is the genetic algorithm (GA).It uses genetic operators including selection, crossover, and mutation to create a population of possible timetable solutions over a series of generations. The GA chooses the best scheduling solutions for replication and additional optimization by assessing each one's fitness in relation to predetermined goals (such as avoiding conflicts, maximizing room use, and fulfilling preferences). This module retrieves timetable data via a database interface, assesses fitness functions, and iteratively adjusts the timetable population until acceptable solutions are found.

VI. ALGORITHM

I. Start
II. Login into the project as Admin/Faculty Member or a User/Student. if client login as an Admin/Faculty Member go to step III.
III. Input data for Department, teachers, semester, time, subjects.
IV. Store the data into database.
V. Fetch the data from Database.
VI. Apply genetic algorithm on fetched data.
VII. Wait for optimal solution.
VIII. Time table is generated for classroom , Department.
IX. View Time-Table or Print Time-Table
X. Stop.

VII. MATHEMATICAL MODEL

Compute Fitness:
The best solution was determined using the formula of fitness function i.e. [7]

$$f(x) = \frac{\sum_{i=1}^{n} xi\ G\ W}{T}$$

Where:

W= Number of constraints
T= Overall fitness value
x= Timetable under evaluation

VIII. METHODOLOGY

Crossover Evolution:
This process involves starting with an older population and working your way up to a new one. Two chromosomes are used in the basic crossover evolution, which allows for the creation of X new chromosomes. It involves dividing the two chromosomes into separate pieces and recombining the components to form new chromosomes.

Mutation:
The algorithm is triggered by mutation. It involves arbitrarily altering a gene's values to get a novel, unexpected answer. These answers provide the fitness function from a fresh angle. The mutation doesn't influence other solutions; it just modifies the chromosome. New Population: A new population of unique solutions can be produced thanks to crossover and mutation.

Population evaluation:
A solution's fitness is an estimate of its quality based on a flexible set of restrictions. The solution is valid within this range. The Genetic Algorithm's fundamental component is the population assessment. In this stage, a fitness function is used to determine which solution is superior to the others. Fitness may be expressed as a range between 0 and 1, with 1 representing the population's best solution and other individuals being used to range them. In this instance, there are always going to be two possible solutions: one where the fitness is 1 and the other where the fitness is 0.

Initial population:
This is GA's initial phase. It is comprised of employing strict limitations to generate a multitude of randomly selected individuals. The user's needs determine the population choice. Due to evolution, a tiny population will eventually go extinct and wipe out the whole population. Large populations, on the other hand, will provide greater outcomes but will use more resources and move more slowly. One way to depict the population is as a set.

IX. Applications

Applications for a timetable generator system can be found in many different sectors where time management and scheduling are essential. The following are a few typical uses:

Educational Institutions:
To make timetables for lessons, tests, and other events, timetable generators are widely used in schools, colleges, and universities. These systems can distribute classes equitably among teachers and students, maximize resource use, and prevent conflicts.

Employee Scheduling:
A timetable generator can automate the process of scheduling personnel in industries like retail, healthcare, and hospitality where staffing levels and shifts need to be controlled effectively. It can consider things like labor regulations, skill sets, and staff availability.

Transportation Management:
Timetable generators are used by airlines, railroads, and public transportation systems to plan and optimize routes and allocate resources (such vehicles).

Conference and Event Management:
Timetable generators are used by conference, seminar, and event planners to arrange speakers, workshops, and other events. Complexities like concurrent sessions and participant preferences are handled by these systems.

Healthcare Facilities:
To plan appointments, surgeries, and other medical operations, hospitals and clinics employ timetable generators. These systems can increase efficiency and patient care by allocating resources like doctors, nurses, and operating rooms as efficiently as possible.

X. Future Scope

Future improvements that could result from this project include scheduling exam rooms, managing leaves of absence, creating exam schedules, and solving time-constraint issues. This improvement can be attained by making additional changes while adhering to the methodology and methods employed for this project. By employing a prediction model, we can analyze the performance of both faculties and students[8].
Providing pupils with individualized schedules that take into account their unique interests, learning preferences, and extracurricular activities. This could entail incorporating comments and data from the students to create timetables that best meet their requirements and improve their education as a whole.
Using historical data, enrollment trends, and other pertinent information to forecast future scheduling demands through the use of data analytics and predictive modeling. This could assist organizations in anticipating possible scheduling issues and making plans for the allocation of resources in the future.
In general, timetable generation systems have a lot of room to grow in the future by utilizing cutting-edge technologies to produce more effective, customized, and flexible scheduling solutions that satisfy the various demands of stakeholders and educational institutions[15].

XI. Conclusion

The system's simplicity and the addition of a changeable algorithm's performance and purpose decreased the requirement for as many constraints because solutions are generated dynamically. This makes it simple for people to utilize and play with the application until they discover the one that works best for their needs. Finding a correct evaluation for the application through a huge number of combinations of testing has shown to be an impossibility. But given the amount of limitations placed on it, it can be said that the system was able to provide outcomes from the models that were given that, despite their imperfections, are still legitimate and acceptable. The algorithm's goal of automatically inducing a schedule is accomplished. The algorithm combines several techniques designed to improve the search operation's efficacy. Additionally, it prevents conflicts between professors' availability. The number of limitations has decreased due to the system's simplicity and the addition of adjustable algorithmic powers and goals, as solutions are generated on the fly. This makes it simple for customers to utilize the application and try different things until they figure out what works best for their particular situation. It turned out that the several combinations we tried in an attempt to determine an accurate rating for our application were not feasible. On the other hand, based on the models supplied, we can say that the system produced outcomes.

Acknowledgment

We would like to express our profound gratitude to the Prof. Rupali Kaldoke and institutions for their vital contributions that have helped shape the Time Table Generation System We are grateful to our excellent faculty members for their insightful comments and thoughts into the limitations and needs of the scheduling system We would like to sincerely thank everyone who has contributed to our project.

REFERENCES

[1] Edmund K. Burke, Patrick De Causmaecker, and Sanja Petrović, "Automated Timetabling: The State of the Art", Journal of the Operational Research Society, Volume 47, Issue 5, May 1996.

[2] Ian P. Gent, Barry H. Smith, and Paul H. T. Thorburn, "A Survey of Automated Timetabling", Artificial Intelligence Review, Volume 13, Issue 2, April 1999.

[3] Edmund K. Burke and James Newall, "Recent Developments in Timetabling", European Journal of Operational Research, Volume 176, Issue 1, January 2007.

[4] Armin R. Mikler and Wenbin Luo, "A Survey of University Timetabling Problems", Annals of Operations Research, Volume 218, Issue 1, March 2014.

[5] Xiaoyang Kang, Meng Sun, and Juncheng Wei, "A Survey on School Timetabling Problem: From Hard Constraints to Soft Constraints", Applied Soft Computing, Volume 67, November 2018.

[6] Gokcen Cimen, Edward Burke, and Kemal Kilic, "Multi- objective Evolutionary Algorithms for University Timetabling: A Survey", Soft Computing, Volume 22, Issue 17, September 2018.

[7] Adedayo Sobowale, Adebimpe Esan, Samuel Ajayi, "Development Of An Automated Timetable Generator Using Genetic Algorithm", Lautech Journal of Computing and Informatics (LAUJCI) – ISSN: 2714-4194 Volume 3 Issue 1, June 2023.

[8] V. Abhinaya, K. Sahithi, K. Akaanksha, "Online Application of Automatic Timetable generator", International Research Journal of Engineering and Technology, February 2019.

[9] Landir Saviniec, Maristela O. Santos, Alysson M.Costa, Lana M.R. dos Santos "Pattern-based models and a cooperative parallel metaheuristic for high school timetabling problems"1 Feb 2020.

[10] Joo Siang Tan, Say Leng Goh, Graham kendall, Nasser R.Sabard "A survey of the state-of-the-art of optimisation methodologies in school timetabling problems" 1 March 2021.

[11] Mrunmayee V. Rane, Vikram M. Apte, Vishakha N. Nerkar Mani Roja Edinburgh, K. Y. Rajput "Automated Timetabling System for University Course" 5-7 March 2021.

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

[13] "State-of-the-Art in Timetable Management Systems: A Comprehensive Survey" by A. K. Sharma and P. Gupta (Publication Year: 2018).

[14] Abdulaziz Aminu , abubakar Sani , Mansur Said "Design and Implementation of An Automatic Examination Timetable Generation and Invigilation Scheduling System Using Genetic Algorithm" IEEE 2019.

[15] M. K. Mohamed Faizal, Thamarai Selvi Balu, Vanmathi Karunanithi, Nithiya Senthilkumar, 2022, College Timetable using Time Scheduling Algorithm, international journal of engineering research and technology (ijert) etedm – 2022