Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 6 –Issue 2, August 2023
Paper:81

University Time Tablining and Automated Class Schedule Generation for Information System

[1]Ms.J.S. Lourdeena,[2]J. Lekha Kamaleshwari,[3]V. Kaviya,[4]R. Kalki,[5]N.R. Susritha
[1]Assistant Professor, Department of Computer Science and Engineering, Jeppiaar Institute of Technology, Chennai, India
[2,3,4,5]UG Scholar, Department of Computer Science and Engineering, Jeppiaar Institute of Technology, Chennai, India
[1]Lourdeena88@gmail.com [2]lekhajayachandran42@gmail.com,
[3]kaviyavaidiyanathan@gmail.com, [4]kalkir.2002@gmail.com, [5]susritharaja@gmail.com

ABSTRACT

The suggested system in this study presents a novel approach to tackling the scheduling of school and college timetables by utilising complex algorithms to give an efficient model for scheduling under demanding constraint considerations. The study's objective is to create a genetic algorithm model that can handle difficult combinatorial optimisation problems. Despite the fact that this industry has been the focus of extensive research, many of the results are still in their infancy. It has been claimed that setting timetables is an extremely challenging, NP-hard task. The main difficulties of the hard and soft constraints necessary for scheduling are completely addressed by the scheduling technique presented in this article.

Keywords: Combinatorial optimisation, scheduling methods, genetic, soft, and hard restrictions, rule-based agents, and active rules.

I.INTRODUCTION

Timetable planning is one of the most difficult and error-prone applications. There are still important challenges that develop during scheduling and recur repeatedly, such as the creation of expensive timetables. The need for a programme that distributes the route evenly and without collisions is therefore great. The objective is to develop an application that can instantly produce a top-notch timetable and is quick, simple to use, efficient, and portable. A genetic algorithm is the primary algorithm used to create timetables. Genetic techniques are used to design the optimal timetable by regulating all the rules.

II.LITERATURE SURVEY

University Time Tabling and Automated Class Schedule Generation for Information System is the title of the essay. The method for automated timetable creation is suggested in this article. The thesis looks at the difficulty of making university schedules. University Scheduling Based on Strict Restrictions The paper's title is Using Genetic Algorithm. The recommended system provides a novel approach for using genetic algorithms to resolve the NP-hard scheduling problem in higher education. a survey of the research on genetic scheduling algorithms. It has been claimed that setting timetables is an extremely challenging, NP-hard task.

III.PROPOSED SYSTEM

A population pool of chromosomes known as strings is maintained in genetic algorithms. They are also referred to as the genotype (the solution's coding), as opposed to the phenotype, which refers to the actual solution. It is necessary to evaluate these chromosomes' fitness. Bad solutions are not taken into account. The remaining responses are slightly modified, and "natural selection" is then let to continue. This promotes the gene pool to adapt in order to provide more accurate results. They have described how the operation of genetic algorithms (GA) is comparable to that of natural selection.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 6 –Issue 2, August 2023
Paper:81

The population pool of chromosomes that is retained is known as a "string." Symbols or numbers make up the chromosomes. They are also referred to as the genotype (the solution's coding), as opposed to the phenotype, which refers to the actual solution. It is necessary to evaluate these chromosomes' fitness. Bad solutions are not taken into account. The remaining responses are slightly modified, and "natural selection" is then let to continue. This promotes the gene pool to adapt in order to provide more accurate results. Setting limitations and rules to create a typical, effective schedule: In this study, we have proposed a more effective way to automate timetable creation. There are two distinct sorts of constraints: hard and soft. To create timetables that are valid, hard constraints must be taken into account. Clear but not as challenging to achieve restrictions are referred to as soft restrictions. Solutions are considered to be superior if these can be incorporated.

Hard Constraints: 1. The system generates a large number of final reports, including weekly timetables, teacher schedules, room-by-room schedules, student schedules, department-level schedules, etc., in addition to many intermediate level reports. 2. The customer can choose the number of days between lectures and can alter it at any time. 3. The time tabling algorithm aims to adjust courses to slots that have been customised by the user in accordance with the time permitted. 4. If conflicts cannot be resolved or modified, it displays the course and the number of lectures that cannot be changed. It also displays the advancement of course alterations at the intermediate report level. 5. A display of available, free slots is available. 6. Teachers should abstain from competing on two different schedules. 7. Particular lectures that might or might not be delivered to several classes at once. 8. Classroom double bookings are not allowed. 9. There can only be one appointment per class. 10. There can never be two reservations made at the same time for a class of students. 11. A classroom needs to be large enough to hold all of the scheduled classes. 12. It is not permitted to book lecturers twice. 13. Each lecturer's weekly working hours have a minimum and a maximum.

Soft Constraints: 1. It makes the importance of time slots modifiable. If the lecture cannot be changed, it might be moved to a time slot of higher priority until it can. 2. The user can modify every parameter. 3. Some classes call for a space. For instance, laboratories may be used for experiments. 4. Some classes require specific classroom amenities, such as audio-visual equipment. 5. Some classes need to follow one another. Examine a six-hour real-world experiment. 6. How the system schedules tasks: The following system architecture serves as an illustration of how the system functions:

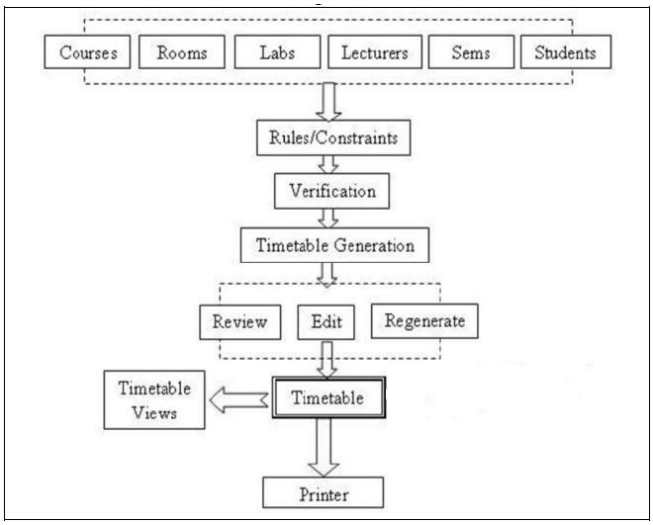

Fig.1 System Architecture

The article provides an example of how to create a schedule by module. It includes:

User Module.

Database Module.

Constraints.

Timetable Generation.

User Module: User Module contains the User GUI. The GUI consists of user input and Authorized User verification. Only the authorised user should be able to access the system. Much of the sensitive information, such as the specifics of the workload of the faculty, should be known to and accessible to the authorised person. The

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 6 –Issue 2, August 2023
Paper:81

produced schedule may be approved or rejected by the authorised user. A schedule is made based on the input. The user's input is used to build a database that contains all of the available information. The following inputs from users are required for the system: There are no detailed entries for any faculties. academic burden. details pertaining to classrooms. Details about the laboratory. Detailed information about the curriculum semester by semester and year by year. Work hours are included in the timetable.

Constraints and Rules: The restrictions will be applied in accordance with the definitions of the regulations provided above. The hard constraints are the restrictions that must be included without exception if proper timetables are to be constructed. Restrictions that are easy to meet yet are nonetheless clear-cut are known as soft Constraints. If these can be included, solutions are thought to be more beneficial. The system will confirm the restrictions.

System Module: The system module consists of:

Database Handling.

Rules Verification.

Timetable generation.

Display generated schedules.

Review/ Regeneration.

Database Handling: Includes data that the user has entered. A database is created using the information entries, and access is made simpler by connectivity between the system and the database. The database includes student databases, lecture databases, faculty databases, and databases for rooms and labs.

Rules Verification: A useful timetable that helps manage the workload is created using the stated parameters and restrictions. An effective and user-friendly timetable must follow rules. As a result, confirming these requirements is necessary to determine the hard and soft limitations for creating the timeline.

Timetable generation: Through the process of timetable development, an efficient, best-fit, optimised, valued, and constraint-satisfying timetable is produced. In this work, three combinatorial methods are employed. The algorithms will be used to create a phase-wise timetable. Three of the algorithms that will be used are genetic algorithms. Algorithms should be applied sequentially for the best results. The algorithm being used will have an impact on how quickly the best match is chosen. Development of a complex algorithm-based timetable that is the most effective, advantageous, and estimable is the main goal of these studies. Timetables were produced using a single-way technique that featured low-module or simulated annealing in an earlier study.

The basic building blocks of a genetic algorithm are as follows: Initialise the pool at random for each generation, then pick advantageous solutions to create a fresh population. create innovative methods from parents Examine novel fitness strategies The original pool that was selected at random is probably not very good, therefore replace the existing population with a new one. However, following generations get better for a number of reasons.

Selection: Each subsequent generation selects a portion of the present population to breed a new generation. The likelihood of choosing a particular solution depends on its fitness, with fitter options (as measured by a fitness function) frequently having a larger probability. Mutation aids the algorithm in avoiding local minima and slows or even stops evolution by preventing the population of chromosomes from becoming extremely identical to one another.

Crossover: It combines the genetic elements of the parents' DNA during breeding to produce kids. Because only the top candidates are selected for breeding during the selection process, the crossover operator combines the genetic material to create children with even greater fitness.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 6 –Issue 2, August 2023
Paper:81

Review/ Re-edit/ Regeneration: The optimised timetable will be evaluated if the authorised user is dissatisfied with the limits specified, the inputs given, the database alteration, etc. Once more, a fresh schedule will be made that is effective and perfectly suits the requirements of the user.

Automated timetable generator structure:

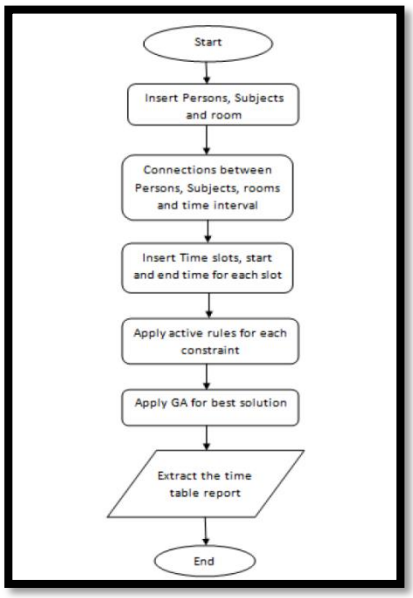

Fig.2 Flow Chart of Automated Timetable Generator

Advantages of Proposed System

1. Unlike the manual timetabling system, the system offers flexibility.
2. It consumes the least processing and computing resources.
3. It significantly reduces the amount of time needed to create the most error-free schedules.
4. It provides a straightforward approach for data entry and updating and has an intuitive UI.
5. It increases productivity.
6. The percentage of timetables with the best solutions ranges from 60% to 80%.
7. It almost completely eliminates paperwork.
8. It simplifies the timetabling process.

Disadvantages of Proposed System

1. That demands a lot of memory.
2. We can only utilise on mobile devices.
3. It necessitates internet use.

IV.RESULT ANALYSIS

Correctness relates to the validity of the result obtained, or the outcomes of the experiments conducted using the recommended strategy on the same hardware and software platform, when comparing the suggested approach to the original or base technique. When comparing the number of requirements met for the number of lectures, subject-wise, laboratory, and faculty-wise allocations, the algorithm's ability to deliver the highest possible score for the timetable is used to assess the validity of the algorithm.

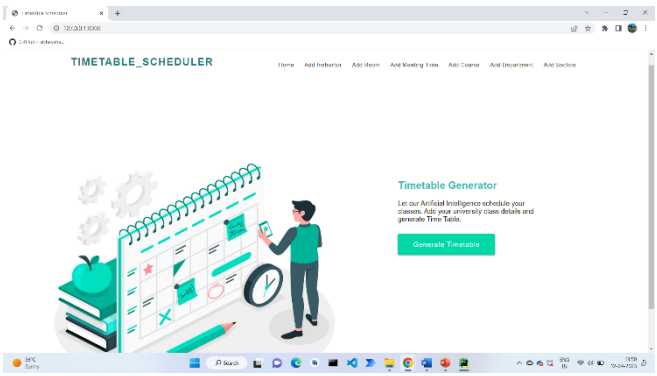

Fig.1 Home Page

K18PV (Computer Science)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
0	C1 Java	37 808	T7 Goutam Mishra	M2 Monday 10:30 - 11:30
1	C1 Java	13 101	T6 Nandidni	T3 Tuesday 12:30 - 1:30
2	C1 Java	13 101	T6 Nandidni	T1 Tuesday 10:30 - 11:30
3	C1 Java	13 101	T7 Goutam Mishra	T4 Tuesday 2:30 - 3:30
4	C1 Java	34 504	T6 Nandidni	M1 Monday 9:30 - 10:30
5	C2 OS	13 101	T3 Dr Isha	Th2 Thursday 2:30 - 3:30
6	C2 OS	37 808	T1 Abhik	T1 Tuesday 10:30 - 11:30
7	C2 OS	57 705	T3 Dr Isha	T5 Tuesday 3:30 - 4:30
8	C2 OS	38 815	T3 Dr Isha	T1 Tuesday 10:30 - 11:30
9	C2 OS	14 503	T1 Abhik	M1 Monday 9:30 - 10:30

Fig.4 Department Wise Timetable (Computer Science)

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 6 –Issue 2, August 2023
Paper:81

M1841 (Mechanical)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
10	C5 aircraft	37 808	T8 Prity chadda	M2 Monday 10:30 - 11:30
11	C5 aircraft	13 101	T7 Goutam Mishra	M2 Monday 10:30 - 11:30
12	C5 aircraft	37 808	T8 Prity chadda	M1 Monday 9:30 - 10:30
13	C5 aircraft	37 808	T8 Prity chadda	T2 Tuesday 11:30 - 12:30
14	C5 aircraft	13 101	T7 Goutam Mishra	Th1 Thursday 9:30 - 10:30
15	C6 Math	14 503	T1 Abhik	T3 Tuesday 12:30 - 1:30
16	C6 Math	14 503	T5 Cherry Khosla	M4 Monday 12:30 - 1:30
17	C6 Math	14 503	T5 Cherry Khosla	M2 Monday 10:30 - 11:30
18	C6 Math	34 504	T3 Dr Isha	T5 Tuesday 3:30 - 4:30
19	C6 Math	13 101	T4 Max Bhatiya	T1 Tuesday 10:30 - 11:30

Fig.5 (Mechanical)

X18wm (ECE)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
30	C3 Networking	34 504	T1 Abhik	T3 Tuesday 12:30 - 1:30
31	C3 Networking	14 503	T3 Dr Isha	T5 Tuesday 3:30 - 4:30
32	C4 Swift	14 503	T5 Cherry Khosla	M1 Monday 9:30 - 10:30
33	C4 Swift	34 504	T4 Max Bhatiya	T1 Tuesday 10:30 - 11:30
34	C5 aircraft	37 808	T8 Prity chadda	T5 Tuesday 3:30 - 4:30
35	C5 aircraft	14 503	T7 Goutam Mishra	Th2 Thursday 2:30 - 3:30

Fig.6 (ECE)

X23wc (EEE)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
36	C10 Circuit	13 101	T4 Max Bhatiya	T2 Tuesday 11:30 - 12:30
37	C10 Circuit	14 503	T4 Max Bhatiya	M4 Monday 12:30 - 1:30
38	C10 Circuit	13 101	T8 Prity chadda	T4 Tuesday 2:30 - 3:30
39	C10 Circuit	14 503	T8 Prity chadda	T3 Tuesday 12:30 - 1:30
40	C9 Wirirng	37 808	T5 Cherry Khosla	Th1 Thursday 9:30 - 10:30
41	C9 Wirirng	13 101	T5 Cherry Khosla	M2 Monday 10:30 - 11:30
42	C9 Wirirng	37 808	T5 Cherry Khosla	Th2 Thursday 2:30 - 3:30
43	C9 Wirirng	14 503	T5 Cherry Khosla	T1 Tuesday 10:30 - 11:30

Fig.6 (EEE)

CP21r (Civil Engineering)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
54	C11 construction	37 808	T7 Goutam Mishra	T2 Tuesday 11:30 - 12:30
55	C11 construction	34 504	T3 Dr Isha	T1 Tuesday 10:30 - 11:30
56	C11 construction	37 808	T3 Dr Isha	M2 Monday 10:30 - 11:30
57	C11 construction	14 503	T3 Dr Isha	T5 Tuesday 3:30 - 4:30
58	C12 E Drawing	34 504	T5 Cherry Khosla	M3 Monday 11:30 - 12:30
59	C12 E Drawing	37 808	T3 Dr Isha	Th1 Thursday 9:30 - 10:30
60	C12 E Drawing	14 503	T5 Cherry Khosla	Th1 Thursday 9:30 - 10:30
61	C12 E Drawing	37 808	T3 Dr Isha	T3 Tuesday 12:30 - 1:30

Fig.6 (Civil)

V.CONCLUSION

A method for automating the construction of schedules utilising a cutting-edge technique like a genetic algorithm has been presented in the study. The survey gets off to a successful start by using genetic technology to identify the most suitable solution as opposed to those using earlier methods. There is nothing that stops this design from being used as a component of an advanced and effective event/action language. This study discussed how a genetic algorithm may be used to express the knowledge of intelligent beings through a collection of active rules and how a genetic algorithm can be used to dynamically prioritise rules in the face of dynamically changing surroundings.

VI.FUTURE SCOPE

The work creates automatically created timetables using complex algorithms. The writers are still striving to implement genetic algorithms to get the most effective and ideal results. Future work will focus on creating performance-enhancing, evolvable algorithms. Another potential is to create a dynamic schedule for the institute using interactions with the staff, head, and students. Another staff person might step in at the last minute to cover for faculty absences. by building the infrastructure for mobile connectivity (android apps), which would enable teachers to share notes with students. by making updates to the timings of lectures and tests on the school website and informing students of any changes.

REFERENCES

[1] Prof. Mamta Patil, Shraddha Thakare, Tejal Nikam, “Automated Timetable Generation using Genetic Algorithm”. International Journal of Engineering Research & Technology (IJERT), 2020.

[2] Shraddha Ambhore, Pooja Walke, Rohit Ghundgrudkar, Akshay Alone, Anushree Khedkar, “Automatic Timetable Generator”.

International Journal of Research in Engineering, Science and Management, 2020.

[3] J. J. Grefenstette, editor. Proceedings of the First International Conference on Genetic Algorithms and their Applications. Practice and Theory of Automated Timetabling VI Proceedings of the 6th International Conference on the Practice and Theory of Automata.

[4] N. R. Jennings.Coordination Techniques for Distributed Artificial Intelligence. University of London Mile End Rd. London E1 4NS UK, 1995.

[5] Om Prakash Shukla, Amit Bahekar, Jaya Vijayvergiya," Effective Fault Diagnosis and Maintenance Optimization by Genetic Algorithm".

[6] Ender Özcan, Alpay Alkan, "A Memetic Algorithm for Solving a Timetabling Problem: An Incremental Strategy", MISTA 2007.

[7] Luca Di Gaspero, Andrea Schaerf, "Tabu search techniques for examination timetabling". In E. Burke and W. Erben, editors, Practice and Theory of Automated Timetabling, Vol. 2079, Lecture Notes in Computer Science, pages 104-117. Springer Verlag, Berlin-Heidelberg, Germany, 2001.

[8] Jean-François Cordeau, Brigitte Jaumard, Rodrigo Morales, "Efficient Timetabling Solution with Tabu Search", International Timetable Competition, 2003.

[9] S.Abdennadher and H. Schlenker. Nursescheduling using constraint logic programming. In B. McKay, X. Yao, C. S. Newton, J.-H. Kim, and T. Furuhashi, editors, Proceedings of the Sixteenth National Conference on Artifcial Intelligence and the Eleventh Innovative Applications of Artifcial Intelligence Conference,99, Orlando, Florida, July 1999, pages 838-843. AAAI Press/MIT Press, 1999.

[10] U. Aickelin and K. A. Dowsland. An indirect genetic algorithm for a nurse scheduling problem. Computers and Operational Research, forthcoming.

[11] N. Brauner, R. Echahed, G. Finke, F. Prost, and W. Serwe. Integration des methods de reecriture et de recherche operationnelle pour la modelisation et la resolution de contraintes: application a la plani_cation de personnel medical. In GISEH 003, Lyon, January 2003.

[12] E. K. Burke, P. D. Causmaecker, And G. V. Berghe. A Hybrid Tabu Search Algorithm for The Nurse Rostering Problem. in B. Mckay, X. Yao, C. S. Newton, J.-H. Kim, and T. Furuhashi, Editors, Simulated Evolution and Learning, Second Asia-Paci_C Conference on Simulated Evolution and Learning, Seal `98, Canberra, Australia, November 24-27, 1998, Selected Papers, Volume

[13] E. Psarra and D. Apostolou, "Timetable scheduling using a hybrid particle swarm optimization with local search approach," in 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA), 2019.

[14] A. Aminu et al., "Design and implementation of an automatic examination timetable generation and invigilation scheduling system using genetic algorithm," in 2nd International Conference on Applied Engineering (ICAE), 2019.

[15] T. W. Ekanayake, P. Subasinghe, S. Ragel, A. Gamage, and S. Attanayaka, "Intelligent timetable scheduler: A comparison of genetic, graph coloring, heuristic and iterated local search algorithms," in 2019 International Conference on Advancements in Computing (ICAC), 2019.

[16] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," Soft Comput., vol. 22, no. 2, pp. 387–408, 2018.

[17] K. Y. Junn, J. H. Obit, and R. Alfred, "The study of genetic algorithm approach to solving university course timetabling problem," in Lecture Notes in Electrical Engineering, Singapore: Springer Singapore, 2018, pp. 454–463.

[18] T. Elsaka, "Autonomous generation of conflict-free examination timetable using

constraint satisfaction modelling," in 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), 2017.

[19] R. E. Febrita and W. F. Mahmudy, "Modified genetic algorithm for high school timetable scheduling with fuzzy time window," in 2017 International Conference on Sustainable Information Engineering and Technology (SIET), 2017.

[20] J. Soyemi, J. Akinode, and S. Oloruntoba, "Electronic lecture time-table scheduler using genetic algorithm," in 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology (DASC/PiCom/DataCom/CyberSciTech), 2017.

[21] A. Azlan and N. M. Hussin, "Implementing graph coloring heuristic in construction phase of curriculum-based course timetabling problem," in 2013 IEEE Symposium on Computers & Informatics (ISCI), 2013.

[22] K. Nguyen, D. Nguyen, K. Trieu, and N. Tran, "Automating a real-world university timetabling problem with Tabu search algorithm," in 2010 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010.

[23] F. D. Wihartiko, H. Wijayanti, and F. Virgantari, "Performance comparison of genetic algorithms and particle swarm optimization for model integer programming bus timetabling problem," IOP Conf. Ser. Mater. Sci. Eng., vol. 332, p. 012020, 2018.

[24] Burke, E.K. and Petrovic, S., 2002. Recent research directions in automated timetabling European Journal of Operational Research, 140(2), pp.266Y280.

[25] Chowdhary A., Kande, P., Dhone, S., Ingle, S., Rushiya, R. And Gawande, D.,2014 Timetable Generation system. International Journal of Computer Science and Mobile Computing, 3(2).