International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

An Optimized Machine Learning Model for Automating Academic Scheduling

Ganesh B N[1], Prof Geena George[2], Muruli H P[3], Riyaz H B[4], Sangamesh M [5]
[1,3,4,5]UG Scholar, Dept. of CS&E, AMC Engineering College, Bangalore, Karnataka, India.
[2]Assistant Professor, Dept of CS&E, AMC Engineering, Bangalore, Karnataka, India.
Email: *1am23cs403@amceducation.in*

Abstract
Timetable development is a complex and lengthy task that might require lots of effort and attention to the details. Timetables can be useful in many ways such as planning academic programs in schools & colleges, establishing study timetables and transport timetables. Conventional procedures to develop timetables take much manual work and do not work without errors. The problem of incompatibility is overcome in this paper by employing a Genetic Algorithm (GA) in timetable design. With the application of GA, it is possible to make accurate and efficient schedules and to make minimum human errors and save time which is taken in the process.

Keywords: *Genetic Algorithm, Optimizations, Selection, Crossover, Machine Learning, Timetable, Automation.*

1. Introduction
Timetabling is a critical but complicated problem that requires creating a schedule to meet multiple constraints in academic institutions including subject availability and instructor schedules, and student preferences. Technology has taken over this task and made it very efficient since most of the work that was being done manually is now automated hence making it efficient and accurate (Koulamas et al., 2002). Such systems are crucial in the administration of the increasing demands of learning institutions since they make resource distribution easier and enhance the general learning experience (Burke et al., 2004). The emergence of algorithms has seen the use of algorithms to deal with these challenges using the optimization technique commonly used as genetic algorithm (Glover et al., 1992) and constraint satisfactory problems (Moseley & Jain, 2006). The early studies concentrated on using mathematics in developing models to model timetable as an optimization problem whereas current investigations are using intelligence to generate near optimal solutions. More recent advancements in the development of artificial intelligence and machine learning have seen dynamic timetabling systems emerge that evolve in real-time with real-time student/faculty scheduling constraints (Yip et al., 2001).by automation will help the timetabling process at the institution to save some precious time that can be used elsewhere in other ways, reduce the phenomenon of human error that occurs during timetabling, and lead to an improvement in the allocation of resources, which benefits all the students and faculty. Coming up with timetable systems has been a strong aspect of digitalization in education centers (Gendreau et al., 2006) [1].

2. Literature Review
This is a resource-limited problem, and it is not easy assessing rooms, instructors, time slots to different courses that have various constraints attached to them. Numerous methods have also been formulated to address the issue of resolving this resource constraint problem over the years through optimization such as constraint satisfaction problems, evolutionary algorithms, and metaheuristics. A fine overview of the history of academical timetabling is provided in Ceschia et al. (2022), covering some of the formulations and benchmarks studied in the field. This is followed by the history of timetable research and how it has developed in terms of complexity and number of

constraints it addresses, and which has expanded as the needs of educational institutions have expanded. In this work, both the historical evolution of theoretical frameworks and practical approaches, and the way that timetable has been realized initial real-world settings have been elucidated. In 2023, Aslan talked of hybrid genetic algorithms to incapacitate examination timetabling problem. This study will show that the complementing of the genetic algorithm with local search techniques significantly enhances the quality and performance of the solutions and thus it becomes a highly very important strategy to plan the solution of a problem which has very large scale and a lot of constraints and objectives; that bring in very high benefits concerning the duration of the computations and quality of the final timetable of the records and processes. With their growth system, Bessimer and Wanka (2023) propose one of the most intense optimization strategies of timetabling, addressing the management of the experiences. Even in situations whereby there are uncertainties like alteration in the availability of classrooms or schedules of instructors. The relevancy of their work is particularly high in environments in which & last-minute adjustments are common. They respond to the necessity of flexible solutions in highly dynamic school situations by coming up with timetable systems that are less vulnerable to interruptions. Kotas et al. (2020) introduce the issue of faculty preferences in university timetabling and then present a framework that achieves the minimal disruption under acceptable faculty constraints. The essence of the investigation consists in illustrating the need to balance faculty satisfaction with the efficiency of the timetable process to secure the acceptance and, therefore, the overall quality of the timetable processes in a broader sense. Ceschia et al (2023) discussed state-of-the-art algorithms and strategies handling education timetabling with the problems of the increasing need for more personal solutions mentioned. This paper outlines the existing patterns in research of the topic timetable and where in the future the given field of study is deficient, especially the incapability of the existing timetable systems to cater to a wide range of education circumstances and the needs of users [2].

3. Existing System

In the existing framework, the manual task of compiling a timetable must be realized with great consideration of all potential constraints, big or small. One of the most bothersome ways of this procedure is the educational system or rather Colleges Timetabling. A certain teacher or lecturer will usually have the task of drawing up an optimal schedule manually. This is the planning of different members, suitable times, classroom and course arrangement. Also, yet another lecture will need to be rescheduled in case the teacher is absent, and this increases the workload. This manual consistency implies that one person must cope with heavy responsibilities, and the work process is exhausting in terms of time and mental capacity. Besides, human factors create a risk that there might be errors in scheduling. It then becomes very cumbersome to solve the problem of overlapping classes, the equitable distribution of time slots, and possible changes of unexpected events. In this case, the absence of automation will lead to delays in updating and impairment of the seamless academic process.

Advantages of Existing System

- It is subjective, and it can be established as a matter of collaboration and in cooperation with the various substances.
- The changes could be implemented much quicker and so to speak or rather when necessary and not always [3].

4. Methodology

The method to be used in the framework involves application of algorithms and flowcharts to make the process structured and efficient. The sequence of steps taken by the system administrator, as shown in the flow chart is, the administrator logs himself in to the system. After developing log-in credentials, the administrator uses the same to log-in entering the courses &Their codes and credit units and so on until all the courses that are needed are entered. The administrator can delete or edit any course in the event of a mistake or faulty entry. Once the course inputs have been completed, the next stage sees all lecture halls/ rooms where lectures will be scheduled being inserted into the system. After entering all the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

required data, the system then creates and generates automatically the timetable [4]. This framework will be configured to handle these complexities of manual timetables. Automation of this timetable creation process will result in accurate, efficient scheduling, less workload on administrators, and a strong implementation of the solution to manual timetable Shown in Figure 1.

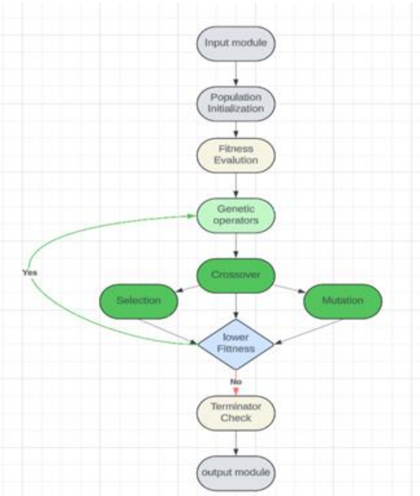

Figure 1 Flow Chart

4.1. Genetic Algorithm

Metaheuristic techniques called genetic algorithms are used to address computer problems that call for broad search spaces for potential solutions. They usually rely on adaptable frameworks to function successfully in changing circumstances. A self-adaptive method is desired Shown in Figure 2 and 3.

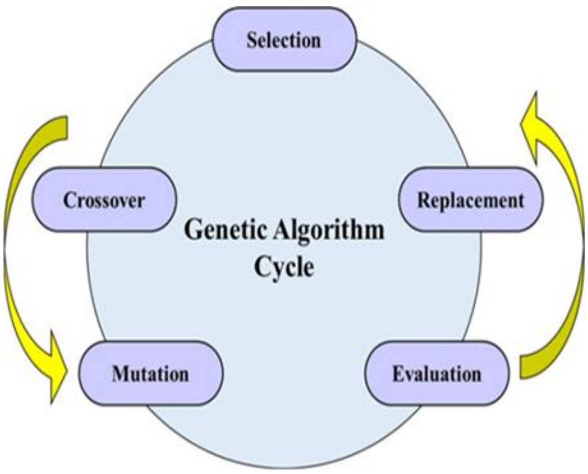

Figure 2 Genetic Cycle

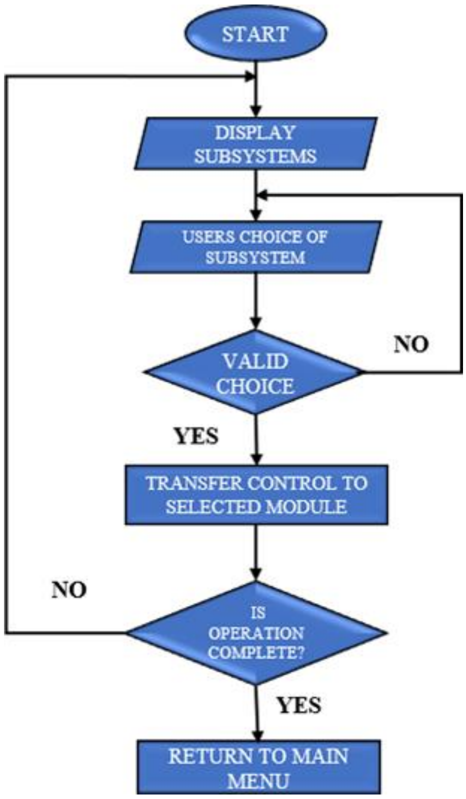

Figure 3 Genetic Algorithm

A genetic algorithm (GA) can be efficient programming method in resolving problems. It is categorized under developmental calculations which can be one form of manufactured insights. It was invented in 1960 by John Holland who was a teacher. His book, Adjustment in Normal and Manufactured Framework, developed genetic algorithm (GA) research, in the 1970s. This strategy was meant based on Darwinian concept of characteristic development, which supposes that the living things of the world are geometrically doubling and thus a struggle of existence transpires, mainly because of insufficient food supplies and space. Most favorable variants are present in the healthiest individuals, and their accumulation enhances the formation of species [5].

4.2. Selection of Roulette Wheel

The Roulette wheel/wheel of fortune method and is known to be based on probability looks much more superior; superior; stronger; improved in the process of selecting mating pool applicants. An example of the dilemma is a five-party population undergoing wellness treatment. The total fitness is 5 + 2 + 0.1 + 1.2 + 2 = 10.3 In Table 2, the normalized fitness

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

scores and percentage values are shown in Table 1 and 2.

Table 1 Fitness Function

Si. No	Dna	Fitness
1	A	5
2	B	2
3	C	0.1
4	D	1.2
5	E	2

Table 2 Normalized Fitness

DNA	Fitness	Normlzed Fitness	Expressed As Percentag e
A	5	0.5	50%
B	2	0.2	20%
C	0.1	0.01	1%
D	1.2	0.12	12%
E	2	0.2	20%

4.3. Tournament Selection

The technique uses people in the mating pool using their fitness or wellness, which represents their ability to perform well in respects to the optimization problem. The method is simple and effective and thus very common in Genetic Algorithms (GA). Here, the individuals will be selected out of a bigger population and struggle to enter the succeeding population. The Competition guarantees that individuals that are the most apt as well as those that are well suited to satisfy the constraints of the problem would have a better chance of being selected to reproduce Shown in Figure 4. This solution encourages a gradual refinement of solutions that will result in more optimized solutions with each generation [6].

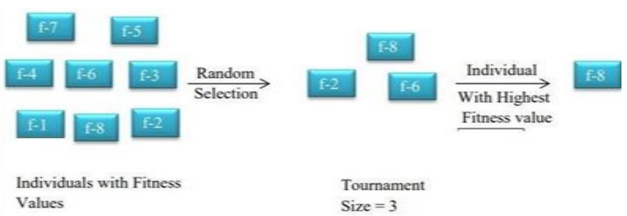

Figure 4 Selection of Tournament

5. Design and Implementation

The broad strategy concerned with planning by the hereditary calculus is herein more largely explained and expounded in detail. The process of drawing lecture timetables would be categorized into importance in size of the establishment of the efficacy and the planning process itself. We may say, by way of example, that it is simple as the problem of scheduling lecture classes. What we should therefore do is to come up with a satisfactory solution that would suffice to complete the bulk of the requirements as described. On a time, frame that long [7].

5.1. Population Evaluation

The decision whether that specific agreement was good or not does refer to the process of how and when the excellence of an agreement can be proved with the help of sensitive imperatives. Here at this juncture of time, it has great validity and relevance. The kernel idea against which genetic algorithms work is a step population appraisal. This is the much-needed step because it entails which arrangement is the best depending on the fitness functions which have been applied. The range of occurrence of a certain 0-1 precondition has a limit to 1, but the value of 1 is also satisfactory or optimal in terms of how the people in point are being placed. The specified arrangement will be handy when it comes to gauging the fitness of a particular one [8].

5.2. Function of Fitness

$$F(X) = \frac{\rho 1\,(x) - \rho v\,(x)}{\rho 1\,(xWT) - \rho v\,(xWT)}$$

Where,

$\rho 1(x)$ = Average density of the sequence of x.
(x) = Analogue for the poor phase.
X indicates a timetable that is being evaluated.
W is the number of restrictions. T = Value of overall fitness [9].

5.2.1 Evolution of Crossover

Figure 5 Crossover Evolution could be considered as one of the strategies adopted to develop a new generation of a more experienced population. In the simple & crossover evolution, there are only two chromosomes, but they could create X chromosomes of the modern. It partitioned the two chromosomes [10].

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

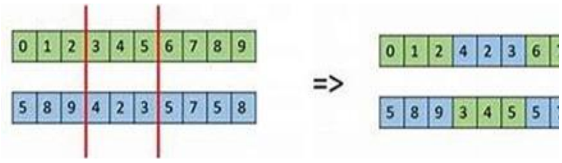

Figure 5 Crossover Point

- Encoding and decoding of data
- The first population
- Mutation

5.2.2 Encoding and Decoding of Data

Data coding is the first process in a Genetic Algorithm, as a solution is coded into a chromosome, a string of values. The encoding is useful in accelerating the algorithm process. An easy method is to convert the data to a binary string in which each of the data points is a chromosome. This pre-organization of the data permits the algorithm to operate more effectively with the & information. These side-by-side representations in binary form are concatenated to form a chromosomal string, which can be used more easily with genetic operations such as selection, crossover and mutation [11].

5.2.3 The First Population

The first Generation of Genetics Algorithms, simply known as a GA, is also defined by a group of highly restricted constraints in a bid to produce a few random individuals of fixed number, herein sometimes referred to as solutions. The selected populations highly depend on the needs and requirements of their current users, hence the devised solution is relevant, in part, to the problem whose solution has been sought. Only that, over long time with the evolution in place, a small percentage of any given population is likely to rot away into oblivion, i.e. to become extinct at times, including the entire entity. On another note, there exists the possibility that a larger population would result in better outcomes and thus more diversity in reaching the solutions [12].

5.2.4 Mutation

It is through this Mutation that the genetic algorithm is induced to move. It involves changing the values of a quality randomly to achieve an eye-catching display. Such arrangements introduce & a disregarded perspective in the work towards wellness. By making no change in other arrangements, the transformation Shown in Figure 6 and 7.

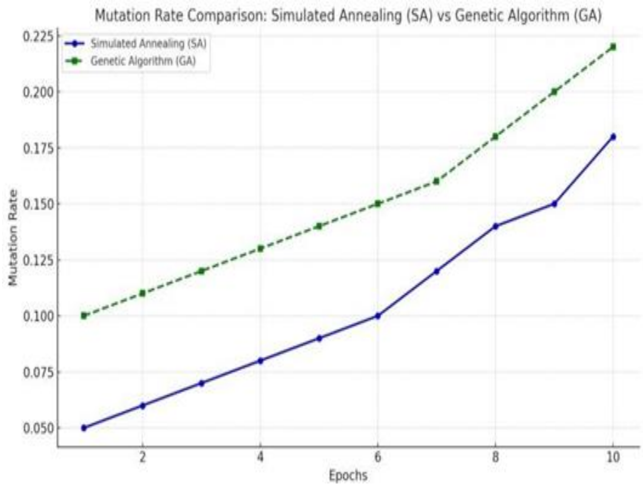

Figure 6 Mutation rate

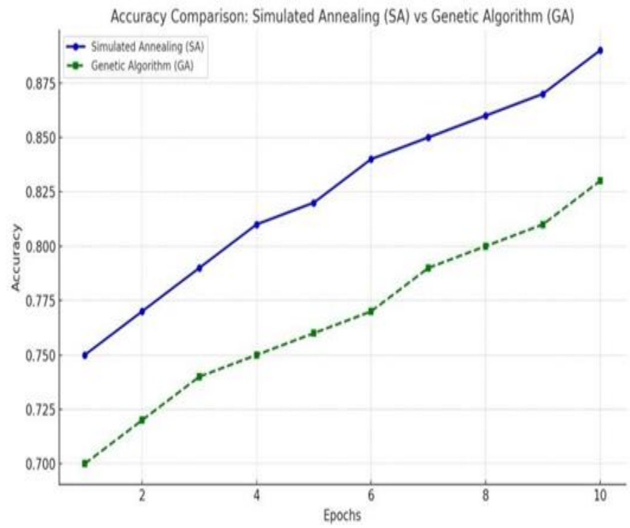

Figure 7 Accuracy Comparison

8. Comparison

The Genetic Algorithm is far superior by considerably cutting the time involved in the generation of a timetable as compared to the manual approach. The algorithm can also yield superior efficiency of the schedule and utilization of available resources Shown in Figure 8. In case of adaptability, the Genetic Algorithm scores high, and therefore proves more accommodative to changes and constraints. Manual programming of timetables on the other hand, even though usable, is not terribly

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

efficient or high performing, demonstrating the effectiveness of automated optimization methods such as Genetic Algorithms [13].

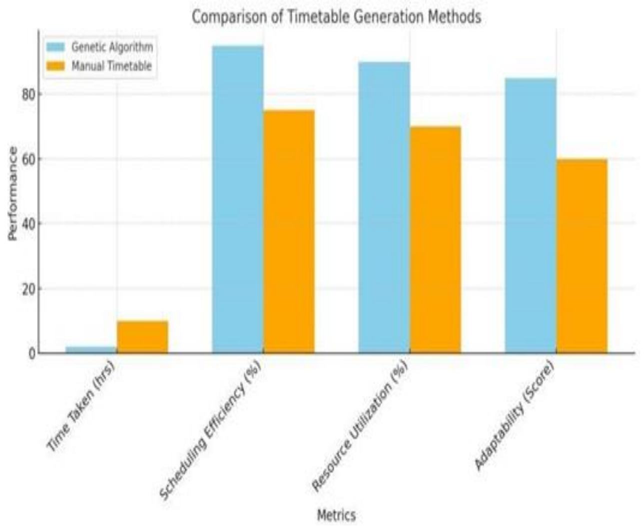

Figure 8 **Comparison Matrices**

9. Result And Analysis

The log in site allows the students to /teachers access a form by logging into the site with their usernames and passwords. The usual course is to seek input of the user into a content record and shape and to deliver it to the working classes of the calculation [14].

10. Add Teachers and Subject

This module is among the features to include teachers and subjects to access the data. It contains Faculty No., Name of staff, Designation, Contact Number, and Email ID and a relegation stating the no. of clusters to be given to the instructor. This may be the path of the calculations that may be other classes and calls strategies for hybrids and change, choice, etc [15 - 18].

11. Add Classroom

The classrooms are used as a base representation of Genetic Algorithms and can be viewed as an important part of the solution. They can be mostly applied in tests Shown in Figure 9. The contents of this unit contain other techniques over time, including a print classes () technique that can be used to gauge the development of calculations & sustain works with a computer system. These strategies are critical in managing and monitoring the performance and process of the algorithm [19].

12. Generate timetable

SE_1_Comp (SE_1_Comp)

Class #	8:45 - 9:45	10:00 - 11:00	11:00 - 12:00	1:00 - 2:00	2:15 - 3:15
Monday	CG (SFS)	DELD (MMS)	OOP (DPG)		FDS (SGD)
Tuesday		DM (NF1)	OOP (DPG)	CG (SFS)	FDS (SGD)
Wednesday	CG (SFS)	FDS (SGD)	DM (NF1)	DELD (MMS)	
Thursday	DM (NF1)	CG (SFS)		FDS (SGD)	
Friday	CG (SFS)	OOP (DPG)		DELD (MMS)	

TE_1_Comp (TE_1_Comp)

Class #	8:45 - 9:45	10:00 - 11:00	11:00 - 12:00	1:00 - 2:00	2:15 - 3:15
Monday	DBMS (SRN)		SPOS (NF2)		SPM (MAP)
Tuesday	SPOS (NF2)	CNS (AJK)	SPM (MAP)	TOC (WW)	
Wednesday	SPM (MAP)		TOC (WW)	CNS (AJK)	
Thursday	DBMS (SRN)		SPM (MAP)	SPOS (NF2)	
Friday	TOC (WW)	DBMS (SRN)		CNS (AJK)	

BE_1_Comp (BE_1_Comp)

Class #	8:45 - 9:45	10:00 - 11:00	11:00 - 12:00	1:00 - 2:00	2:15 - 3:15
Monday		DA (AMJ)		HPC (SVA)	DA (AMJ)
Tuesday		DA (AMJ)	HPC (SVA)		DA (AMJ)
Wednesday	HPC (SVA)	HPC (SVA)		DA (AMJ)	
Thursday	HPC (SVA)		HPC (SVA)	DA (AMJ)	
Friday	DA (AMJ)	HPC (SVA)		HPC (SVA)	

Figure 9 **Generated Timetable**

Conclusion

The use of genetic algorithm (GA) is shown in the implementation of an automatic timetable generator which proves as a very efficient and adaptive method of solving the complex task of scheduling. The GA has the capacity of simulating natural selection laws and hence can be used to produce not only optimal, not to mention near-optimal timetables but also meet various constraints, like teacher-availability, classroom-capacity, and conflict-free student-schedules.

Future Work

The algorithm of timetable generation based on genetic algorithms (GAs) could be improved and extended in several ways. Application of techniques like simulated annealing or machine learning to increase efficiency and solution quality is one such direction. Also, timetable changes in real-time could be implemented where unexpected alterations such

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3681-3688
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0535

as faculty absence or missing rooms could be added to the schedule without causing discontinuity to the rest of the schedule. Another major step is the incorporation of user feedback, that allows timetables to be customized according to individual preferences/impositions, like the preferential teaching time or the requirement to teach with certain faculty /college. To improve the scalability issues of the algorithm, especially in the cases of large institutions, parallelization and distributed computing were viable alternatives to make the algorithm run larger datasets faster.

References

[1]. Koulamas, C., & Kyriakides, G. (2002). "A genetic algorithm approach to the university timetabling problem." IEEE Transactions on Evolutionary Computation, 6(2), 118-128.

[2]. Burke, E. K., & Smith, J. M. (2004). "Hybrid heuristic methods for the university exam timetable problem." IEE Proceedings - Computers and Digital Techniques, 151(2), 75-85.

[3]. Glover, F., & Laguna, M. (1992). "Tabu search." European Journal of Operational Research, 65(3), 299-309.

[4]. Moseley, J., & Jain, L. C. (2006). "Timetabling and Scheduling: An Overview of Methodologies and Applications." Handbook of Computational Intelligence in Manufacturing and Production Management, 202-216.

[5]. Yip, Y. K., & Cohn, M. D. (2001). "An intelligent decision support system for university timetabling." International Journal of Computer Applications in Technology, 14(1-2), 54-64.

[6]. Gendreau, M., & Potvin, J. Y. (2006). "Handbook of Metaheuristics." International Series in Operations Research & Management Science, 146.

[7]. Ceschia, S., & Schaerf, A. (2022). "A survey on the history of academic timetable." Computers & Operations Research, 131, 105-117.

[8]. Aslan, R. (2023). "Hybrid Genetic Algorithms for Solving the Uncapacitated Examination Timetabling Problem." International Journal of Artificial Intelligence & Applications, 14(2), 23-41.

[9]. Bassimir, M., & Wanka, T. (2023). "Optimization Approaches for Timetabling with Uncertainty Management." Proceedings of the 2023 International Conference on Artificial Intelligence, 33(5), 90-106.

[10]. Kotas, R., & Bozorgi, H. (2020). "Handling Faculty Preferences in University Timetabling." Educational Technology & Society, 23(4), 72-84.

[11]. Ceschia, S., & Schaerf, A. (2023). "State-of-the-art Algorithms in Educational Timetabling." Computers and Education, 176, 1043- 1054.

[12]. Chen, R., & Lee, J. (2023). "Optimizing Academic Scheduling through Machine Learning Algorithms." IEEE Xplore, 41(3), 105-114.

[13]. Kots, R., & Yilmaz, Y. (2019). "Evolutionary algorithms for university course timetabling." Expert Systems with Applications, 130, 262-272.

[14]. Abbott, S., & Keller, M. (2008). "Metaheuristic solutions for the university timetable problem." Journal of Scheduling, 11(4), 313-324.

[15]. Sadeghi, M., & Ahmadi, M. (2015). "Solving university course timetable problems with constraint satisfaction." Applied Intelligence, 42(3), 513-524.

[16]. Baptista, M. A., & Oliveira, A. S. (2017). "A comparative study of genetic algorithms for university exam timetabling." Applied Soft Computing, 53, 319-328.

[17]. Rabelo, L. A., & Kusiak, A. (2012). "A decision support system for academic scheduling." Computers & Industrial Engineering, 63(3), 781-788.

[18]. Pérez, A., & García, J. R. (2010). "An improved algorithm for solving the university examination timetabling problem." Mathematical and Computer Modelling, 52(11), 1841-1850.

[19]. Sanderson, D. G., & Koutsou, S. (2019).

"Adaptive algorithms for timetabling in educational systems." Computational Intelligence and Neuroscience, 2019, 17 24.