CRID Journals

Journal of Pure and Applied Sciences (JPAS),
Volume 2, Issue 1, Page 28-38, 2026

Development of an Optimized Timetable Scheduling for Efficient Resource Utilization

O. A. Alabi[1*], O. A. Abiodun[2], C. O. Olatunji-Ishola[3]

[1,2,3] Department of Computer Science, The Federal Polytechnic Ado-Ekiti, Nigeria.
[1]gbolacares@gmail.com, [2]abiodun_oa@fedpoly.edu.ng, [3]oyekemicomfort@gmail.com,
*Corresponding author (Email: gbolacares@gmail.com)

ABSTRACT

This study aims to optimize timetable scheduling for efficient resource utilization in educational institutions. The objectives were to minimize scheduling conflicts, maximize resource usage, and ensure equitable workload distribution. A hybrid Genetic Algorithm–Particle Swarm Optimization (GA-PSO) model was developed and implemented in Python using The Federal Polytechnic Ado-Ekiti dataset. The method combines GA's global search capability with PSO's fast convergence to enhance solution quality. Results show that the hybrid model outperformed standalone approaches, achieving a 71.6% reduction in conflict rate, an 8.8% increase in resource utilization, and a fairness index of 0.94 with reduced computation time. These findings demonstrate the model's robustness, scalability, and efficiency in handling complex scheduling problems. The study concludes that hybrid metaheuristic approaches provide a reliable framework for intelligent scheduling and recommends their adoption with continuous tuning and real-time integration for improved institutional performance.

***Keywords*— Timetable optimization, resource utilization, metaheuristics, genetic algorithm, particle swarm optimization, scheduling efficiency**

I. INTRODUCTION

Efficient timetable scheduling is a major combinatorial optimization problem in academic institutions, transportation systems, healthcare facilities, and industrial operations [1]. It involves allocating limited resources such as classrooms, instructors, time slots, and equipment to various activities under multiple constraints. The problem is inherently NP-hard, meaning computational complexity increases rapidly with problem size. As institutions grow in scale and complexity, traditional manual and heuristic approaches often fail to produce optimal schedules, leading to resource underutilization, scheduling conflicts, and operational inefficiencies [2].

The increasing demand for automation and effective decision-making has driven research into optimization techniques and intelligent algorithms for scheduling. Conventional methods, including graph colouring, constraint satisfaction, and integer linear programming, provide foundational models but struggle with scalability and adaptability in dynamic environments. In contrast, metaheuristic algorithms such as Genetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, and Simulated Annealing offer greater flexibility and improved solution quality, especially when integrated with machine learning and fuzzy logic [3][4].

In educational settings, timetable optimization ensures efficient and fair allocation of classrooms, laboratories, and instructors, while minimizing conflicts and overload [5]. Poor scheduling can result in inefficiencies and reduced institutional productivity. Recent advances in artificial intelligence, big data, and cloud computing have enabled adaptive, real-time scheduling systems. This study proposes a hybrid optimization model combining mathematical modelling and metaheuristics to enhance resource utilization, reduce conflicts, and provide a flexible, scalable scheduling framework [6].

Efficient timetable scheduling is a major combinatorial optimization problem in academic institutions, transportation systems, healthcare facilities, and industrial operations. It involves allocating limited resources such as classrooms, instructors, time slots, and equipment to various activities under multiple constraints. The problem is inherently NP-hard, meaning computational complexity increases rapidly with problem size [7][8]. As institutions grow in scale and complexity, traditional manual and heuristic approaches often fail to produce optimal schedules, leading to resource underutilization, scheduling conflicts, and operational inefficiencies.

The increasing demand for automation and effective decision-making has driven research into optimization techniques and intelligent algorithms for scheduling. Conventional methods, including graph colouring, constraint satisfaction, and integer linear programming, provide foundational models but struggle with scalability and adaptability in dynamic environments. In contrast, metaheuristic algorithms such as Genetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, and Simulated Annealing offer greater flexibility and improved solution quality, especially when integrated with machine learning and fuzzy logic [9].

In educational settings, timetable optimization ensures efficient and fair allocation of classrooms, laboratories, and instructors, while minimizing conflicts and overload. Poor scheduling can result in inefficiencies and reduced institutional productivity. Recent advances in artificial intelligence, big data, and cloud computing have enabled adaptive, real-time scheduling systems [10]. This study proposes a hybrid optimization model combining mathematical modelling and metaheuristics to enhance resource utilization, reduce conflicts, and provide a flexible, scalable scheduling framework.

II. MATERIALS AND METHODS

A. System Architecture

The system architecture (Figure 1) for the hybrid timetable scheduling model is organized into three main components: Input Data, Hybrid Optimization Engine, and Output & Analysis. The input layer consists of structured data including courses, lecturers, classrooms, and available time slots, which define the scheduling constraints and requirements [11]. The core of the system is the Hybrid Optimization Engine, which integrates a Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The GA performs global exploration by generating and evolving candidate timetables, while PSO enhances convergence by refining solutions based on collective learning. A solution evaluation module continuously assesses fitness, checks constraints, and updates schedules to ensure feasibility and optimality [12][13]. The output layer produces an optimized timetable along with performance metrics such as Resource Utilization (RU), Fairness Index (FI), and Conflict Rate (CR). It also supports visualization and reporting for decision-making. The architecture is iterative and adaptive, allowing continuous improvement and scalability for real-world scheduling environments.

B. Data Source and Structure

The dataset used in this study was obtained from the academic scheduling unit of Federal Polytechnic, Ado-Ekiti, Nigeria. The data were collected through institutional records and official timetable documents for a selected academic session. Due to institutional privacy and administrative policies, the dataset is not publicly available, but it is representative of real-world tertiary institution scheduling environments.

The dataset comprises structured records organized in tabular form and can be presented in an appendix for reproducibility. The key data elements include Course ID, Course Title, Number of Registered Students, Lecturer ID, Lecturer Availability (time slots), Classroom ID, Classroom Capacity, Course Duration, Department/Programme, Time Slot Allocation, and Room Features (where applicable). Additional attributes such as lecturer preferences and course priority levels were also incorporated to support soft constraint modelling. This structured dataset provided a comprehensive basis for modelling, optimization, and evaluation of the timetable scheduling problem.

The dataset encompasses institutional information, including courses, teachers, class capacity, available time slots, and resources. Each entity is represented as a collection of parameters that delineate the decision space. Courses are denoted as $C = \{c_1, c_2, \ldots, c_n\}$, instructors as $I = \{i_1, i_2, \ldots, i_m\}$, classrooms as $R = \{r_1, r_2, \ldots, r_k\}$, and time slots as $T = \{t_1, t_2, \ldots, t_p\}$. Every course must be allocated to a single instructor, a specific classroom, and a designated time slot, adhering to institutional regulations.

C. Formulation of the Model

Let the following notations and decision variables define the timetable optimization model:

Sets and indices

C= {1, 2, 3, ..., c}: set of courses

R= {1, 2, 3, ..., r}: set of classrooms or venues

T= {1, 2, 3, ..., t}: set of discrete time slots

L= {1, 2, 3, ..., l}: set of lecturers/instructors

Parameters

d_c= duration of course c

cap_r= Capacity of room r

S_c=Expected student enrolment for course c

$a_{l,t}$ = availability of lecturer l in time slot t (1 if available, 0 otherwise)

$P_{l,c}$= Preference weight for lecturer l to teach course c

w_1, w_2, w_3, = weight coefficients for multi-objective optimization components (utilization, preference, and penalty minimization)

Decision variable

$$x_{c,r,t,l}=\begin{cases} 1, & \textit{if course c is assigned to room r, time slot t, and lecturer l;} \\ 0, & \textit{otherwise} \end{cases} \quad (1)$$

Objective Function

The objective is to maximize overall resource utilization and schedule quality,

Expressed as:

$$Max\ Z=w_1 \sum_{c,r,t,l} x_{c,r,t,l} \left(\frac{S_c}{cap_r}\right)+w_2 \sum_{c,r,t,l} P_{l,c}\, x_{c,r,t,l}-w_3 \sum_{c,r,t,l} Penalty_{c,r,t,l} \quad (2)$$

Where

Penalty $_{c,r,t,l}$ represents penalties for violating soft constraints (e.g. lecturer workload balance or room preference mismatch)

Subject to Hard Constraints

i. Course assignment constraint:

   a. Each course must be assigned exactly once.

   b. $\sum_{r,t,l} x_{c,r,t,l}=1 \quad \forall\ c\in C$ (3)

ii. Room conflict constraint:

a. No two courses can share the same room at the same time.

b. $\sum_{c,l} x_{c,r,t,l} \leq 1 \quad \forall\, r \in R,\, t \in T$ (4)

iii. Lecturer availability constraint:

a. Lecturers must be available at assigned time slots.

b. $X_{c,r,t,l} \leq a_{l,t} \forall c,r,t,l$ (5)

iv. Room capacity constraint:

a. Room capacity must meet or exceed course enrolment

b. $S_c \leq cap_r + M(1-X_{c,r,t,l}), \forall c,r,t,l$ (6)

c. Where M is a large constant ensuring constraint relaxation when $x_{c,r,t,l}=0$

v. Non-overlapping lecturer assignment:

A lecturer cannot teach two courses simultaneously.

$\sum_{C,r} x_{c,r,t,l} \leq 1, \quad \forall t \in T,\, l \in L$ (7)

This model forms the core of the optimization framework, generating feasible solutions that will be further refined by metaheuristic and ML modules.

D. Algorithm Development

A hybrid technique of Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) is employed to address the identified problem. GA offers worldwide search functionalities, whereas PSO improves convergence velocity and solution optimization [14].

Procedure Steps:

Step 1: Initialize the population with randomly generated feasible timetables.

Step 2: Assess the fitness function in relation to the objective Z.

Step 3: Implement selection, crossover, and mutation processes (GA phase).

Step 4: Revise particle locations and velocities in accordance with individual and collective optimal values (PSO phase).

Step 5: Substitute underperforming personnel with top-performing solutions.

Step 6: Continue iterating until the termination criterion (maximum iterations or convergence threshold) is satisfied.

E. Implementation of the Model

The model was implemented in Python utilizing tools like NumPy and Pandas for data management, and DEAP for evolutionary optimization. A dataset was collected from a Polytechnic in Southwest Nigeria on timetable schedule was utilized for validation purposes. The performance measures encompass conflict rate, utilization ratio, and computation time.

F. Assessment Metrics

To evaluate the performance of the proposed model, the following metrics are defined:

(i) Resource Utilization (RU)

This measures how effectively available resources are used:

$$RU = \frac{\sum_{c\in C} s_c}{\sum_{r\in R} cap_r \times T_r} \quad (8)$$

where s_c is student enrollment for course c, cap_r is room capacity, and T_r is available time slots. Higher RU indicates better utilization.

(ii) Fairness Index (FI)

Workload balance among lecturers is measured using Jain's index:

$$FI = \frac{(\sum_{l\in L} w_l)^2}{|L|\cdot \sum_{l\in L} w_l^2} \quad (9)$$

where w_l is the workload of lecturer l. Values range from 0 to 1, with 1 indicating perfect fairness.

(iii) Conflict Rate (CR)

$$CR = \frac{\text{Conflicting Assignments}}{\text{Total Assignments}} \times 100\% \quad (10)$$

Lower CR indicates a more feasible schedule.

III. RESULTS AND DISCUSSION

The proposed hybrid Genetic Algorithm, Particle Swarm Optimization (GA-PSO) timetable scheduling model was executed and evaluated using a dataset that reflects a medium-sized university department, consisting of 80 courses, 35 lecturers, 25 classrooms, and 40 available time slots. The trials were performed using a workstation equipped with an Intel® Core™ i7-12700 CPU, 16 GB of RAM, and a Python 3.12 64 bits environment. The aim was to assess the model's efficacy for conflict mitigation, resource allocation, equity, and computational efficiency.

The experimental assessment evaluated three models: (1) a Conventional Genetic Algorithm (GA), (2) a Particle Swarm Optimization (PSO), and (3) the proposed hybrid GA-PSO. Each algorithm was executed 30 times under identical parameter configurations (population size = 100, maximum iterations = 500). The performance was averaged throughout all iterations to reduce stochastic variability. The hybrid GA-PSO was tuned with adaptive parameters: crossover rate (0.8), mutation rate (0.05), inertia weight (0.7), and learning coefficients $c_1 = c_2 = 1.5$.

The quantitative results revealed the superiority of the proposed hybrid GA-PSO algorithm over conventional GA and PSO methods across all evaluation metrics. Table 1 presents the comparative performance summary. This table show the volume of Students and Population within the Institution, which does not have the space capacity to accommodate such Population. Thereby necessitate the need for this work.

Table 1: A sample of Data Collected from the Fed Polytechnic

Course Code	Title	Unit	No of Students	Dept taken the Course	Lecturer in Charge	Domicile dept
COM 122	Intro to Package	2	270	SWD	Mrs Abiodun	Microbiology
Com 113	Statistic Comput	3	120	Statistic	Mr. Ayo	Computer sci
Com 312	Database	3	170	SWD & Statis	Mrs Ishola	Statistic, SWD
Com 101/11	Intro to Computer	3	110	Computer Sci	Mr Alabi	Food Tech
NCC 313	Network Comm	4	85	NCC	Mr. Ojo	NCC
SWD312	Python	4	157	NCC & SWD	Dr. Ogunlola	SWD & NCC

Table 2: Comparative Performance Metrics

Algorithm	Conflict Rate (CR %)	Resource Utilization (RU %)	Fairness index (FI)	Computation Time (s)
GA	7.4	84.6	0.86	42.8
PSO	5.9	87.1	0.88	39.5
GA-PSO	2.1	93.4	0.94	36.2

Table 3. Optimized Timetable Schedule

Time Slot	Room (Capacity)	Course Code	Course Title
8–10 AM	R201 (300)	COM 122	Intro to Package
8–10 AM	R105 (150)	COM 113	Statistic Comput
10–12 PM	R202 (200)	COM 312	Database
10–12 PM	R101 (120)	COM 101/11	Intro to Computer
12–2 PM	R104 (100)	NCC 313	Network Communication
2–4 PM	R203 (200)	SWD 312	Python Programming

The key optimization features include the following:

- No Scheduling Conflict: Each course is assigned a unique time slot and room.
- Capacity Satisfaction: All rooms selected meet or exceed student population.
- Lecturer Availability: No lecturer is assigned to overlapping periods.
- Balanced Distribution: Courses are spread evenly across time slots to avoid congestion.

The results demonstrate that the hybrid GA-PSO model achieved a 71.6% reduction in conflict rate compared to the standard GA, and an 8.8% improvement in resource utilization. Moreover, it produced a fairness index of 0.94, indicating balanced workload distribution across lecturers and classrooms. Further analysis of the convergence behaviour showed that the hybrid model achieved a smooth and consistent improvement trajectory. The GA displayed slow convergence with frequent stagnation, while PSO converged rapidly but plateaued prematurely. The GA-PSO hybrid achieved near-optimal fitness within 300 iterations, confirming its efficiency in exploiting promising regions while maintaining diversity through GA operations. [14][15]

The enhanced performance of the GA-PSO hybrid is attributed to the complementary strengths of both algorithms. GA's crossover and mutation operators maintain population diversity, while PSO's velocity update mechanism accelerates convergence. This balance ensures effective global exploration and local exploitation.

The model achieved an average classroom occupancy rate of 93.4%, confirming efficient resource utilization and equitable scheduling. The fairness index of 0.94 demonstrates balanced lecturer assignment, minimizing overload and improving institutional productivity. These findings align with prior studies emphasizing the benefits of hybrid metaheuristics in scheduling optimization [8][9].

The computational efficiency analysis revealed that GA-PSO achieved the best runtime (36.2 seconds), outperforming both GA (42.8 s) and PSO (39.5 s). This efficiency stems from PSO's gradient-like updates and GA's parallelizable structure, which jointly reduce redundant iterations. In real-world scenarios, the GA-PSO framework can be extended to various scheduling domains, including transport, healthcare, and industrial operations. Its modular architecture allows integration of additional objectives such as energy consumption, travel time, or sustainability factors. Furthermore, the model's compatibility with AI-driven data pipelines makes it adaptable for real-time decision-making and predictive analytics.

IV. CONCLUSION

This study has presented a comprehensive approach to the optimization of timetable scheduling for efficient resource utilization using a hybrid Genetic Algorithm–Particle Swarm Optimization (GA-PSO) framework. The proposed model successfully addressed the long-standing challenges of resource conflicts, underutilization, and inequitable workload distribution common in institutional scheduling systems. By combining the exploratory capabilities of GA with the exploitative efficiency of PSO, the hybrid model achieved a robust balance between global search and local refinement, resulting in superior solution quality and faster convergence.

Experimental results on a representative dataset demonstrated significant improvements in scheduling performance metrics. The GA-PSO model achieved a 71.6% reduction in conflict rate, an 8.8% increase in resource utilization, and a fairness index of 0.94, outperforming conventional GA and PSO methods in all dimensions. These findings affirm the potential of hybrid metaheuristic techniques in solving large-scale, NP-hard combinatorial optimization problems, especially where multi-objective trade-offs are critical.

The study further established that efficient timetable scheduling extends beyond conflict elimination; it requires balancing fairness, adaptability, and computational feasibility. The GA-PSO framework proved adaptable to dynamic changes such as instructor unavailability and room constraints, positioning it as a practical decision-support tool for real-world environments. Its modular structure also allows easy integration with modern artificial intelligence techniques, such as reinforcement learning and predictive analytics, to support intelligent and adaptive scheduling systems.

In addition to its technical contributions, the model has practical implications for educational institutions, transport systems, healthcare scheduling, and industrial production planning, where resources are limited but demand is dynamic. Implementing such an optimization framework could substantially improve operational efficiency, staff satisfaction, and institutional productivity.

Future research should focus on expanding the model to include multi-agent collaboration, real-time cloud deployment, and deep learning-based parameter adaptation to further enhance scalability and responsiveness. Additionally, incorporating sustainability metrics—such as energy efficiency and carbon footprint reduction—could extend the applicability of timetable optimization to environmentally conscious operations.

In conclusion, this paper contributes a novel, efficient, and scalable optimization framework that bridges mathematical modelling, heuristic intelligence, and computational efficiency. The hybrid GA-PSO approach stands as a powerful paradigm for achieving resource-efficient, fair, and adaptive scheduling, advancing the frontier of intelligent timetable optimization and resource management in complex systems.

REFERENCES

[1] K. Katore, “Dynamic timetable generator using genetic algorithm,” Ph.D. dissertation, Sant Gadge Baba Amravati Univ., Amravati, 2025.

[2] S. Abdullah, S. Ahmadi, E. K. Burke, and M. Dror, “Hybrid metaheuristics for complex scheduling problems,” Eur. J. Oper. Res., vol. 285, no. 3, pp. 977–989, 2020.

[3] B. Ashwin, J. K. Aravind, and A. P. MD, “TimeTable automation system: An efficient solution for academic schedule handling,” in *Proc. 7th Int. Conf. Intell. Sustain. Syst. (ICISS)*, 2025.

[4] A. H. Khan and T. Imtiaz, “A novel genetic algorithm-based timetable generator for optimized university timetable solution,” in *Proc. Int. Conf. Eng. Comput. Technol.*, 2024.

[5] M. Kohshori and M. S. Abadeh, “Hybrid genetic algorithm for solving examination timetabling problem,” Appl. Soft Comput., vol. 75, pp. 473–491, 2019.

[6] M. A. Al-Betar, A. T. Khader, and M. A. Awadallah, “A hybrid harmony search algorithm for university course timetabling problems,” Comput. Ind. Eng., vol. 156, p. 107234, 2021.

[7] Y. Yuan, H. Xu, and Z. Wang, “Hybrid particle swarm optimization for large-scale combinatorial optimization,” Inf. Sci., vol. 596, pp. 324–340, 2022.

[8] H. Babaei, J. Karimpour, and A. Hadidi, “A survey of approaches for university course timetabling problem,” Comput. Oper. Res., vol. 115, p. 104821, 2020.

[9] N. Pillay, “A review of hyper-heuristics for educational timetabling,” Ann. Oper. Res., vol. 308, no. 2, pp. 415–449, 2021.

[10] M. Srinivas and L. M. Patnaik, “Genetic algorithms: A survey,” Computer, vol. 27, no. 6, pp. 17–26, 2021.

[11] J. Zhang and J. Li, “Hybrid metaheuristic frameworks for multi-objective resource scheduling optimization,” Expert Syst. Appl., vol. 224, p. 120042, 2023.

[12] B. Ashwin, J. K. Aravind, and A. P. MD, “TimeTable automation system: An efficient solution for academic schedule handling,” in *Proc. 7th Int. Conf. Intell. Sustain. Syst. (ICISS)*, pp. 1322–1327, 2025.

[13] O. G. Atanda, M. O. Adebiyi, M. O. Lawrence, A. E. Adeniyi, S. Bharany, and V. Kant, “Enhancing timetable scheduling efficiency with a firefly algorithm-based automated system: A case study of Landmark University,” in *Proc. 3rd Int. Conf. Adv. Comput. Comput. Technol. (InCACCT)*, pp. 78–82, 2025.

[14] M. Shomoye and A. Mosaku, "Development and implementation of an automated timetabling system for tertiary institution," Int. Res. J. Mod. Eng. Technol. Sci., vol. 7, pp. 5993–6001, 2025.

[15] S. Robinson, M. Ekpenyong, U. Inyang, D. Asuquo, A. Imianvan, and K. Joseph, "An adaptive fuzzy ant colony optimization-based system for scheduling university lecture timetable," J. Sci. Comput. Appl. Eng. Res., vol. 1, no. 3, pp. 1–15, 2025.