iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: V Month of publication: May 2026

DOI: https://doi.org/10.22214/ijraset.2026.78570

AcadTime: Automatic Timetable Generator

Prof. Shweta Shondawale, Anupam Modi, Priyanshu Raj, Priyanka Dhumal, Samyukta More
ECO, SoES, MIT ADT University, Pune

Abstract—This paper presents an automated solution to the challenges faced by educational institutions during manual timetable creation through the development of an Automatic Timetable Generator using our custom algorithm based on Progressive Constraint Placement and Iterative Repair (PCP-IR).The main issue addressed is the complexity and high chance of conflicts when assigning teachers, rooms, and subjects by hand. Our approach begins by placing the hardest and most restrictive constraints first, progressively building a stable timetable structure before adding the remaining sessions. This ensures that most major conflicts area voided early in the process, making the initial timetable more accurate and reliable. This paper further highlights how the iterative repair mechanism refines the timetable by detecting left over clashes and resolving them using systematic adjustments or swaps. Extensive testing and evaluation we optimize the performance of the PCP-IR algorithm, ensuring consistent and conflict-free scheduling. The outcomes demonstrate that our method offers a user-friendly, efficient, and intelligent timetable generation system, significantly improving speed, reducing errors, and enhancing overall scheduling quality compared to traditional manual methods.

I. INTRODUCTION

An Automatic Timetable Generator acts like a smart assistant for schools and colleges. It helps them create clear, conflict-free schedules without the stress of manual work. Making timetables by hand is exhausting. Teachers often overlap, rooms can clash, and organizing subjects takes a lot of time. To address this, we created our own algorithm using Progressive Constraint Placement and Iterative Repair (PCP- IR). It prioritizes the toughest scheduling constraints first and then intelligently resolves any remaining issues. This approach makes the entire process faster, smoother, and more accurate.

The system consists of a scheduling engine, a constraint-processing module, a conflict-detection unit, and the PCP-IR repair module. The constraint engine gathers necessary inputs like teacher availability, subjects, rooms, and time slots. The PCP part focuses on placing the hardest constraints first, while the IR unit checks the timetable for conflicts and fixes them through smart swaps or adjustments. All these components work together to generate a reliable, optimized timetable with minimal human effort.

II. ORGANISATION OF THE REPORT

A. Introduction

An Automatic Timetable Generator is a software system that assigns teachers, subjects, rooms, and time slots automatically using our PCP-IR algorithm. It focuses on reducing human workload, avoiding clashes, and creating efficient schedules for educational institutions.

B. Literature Review

This section reviews existing timetable generation methods, constraint-based scheduling systems, and optimization algorithms. It highlights the gap that our PCP-IR approach fills by combining progressive constraint placement with intelligent conflict repair.

C. Methodology

Our methodology uses the PCP-IR algorithm, where tough constraints are placed first (PCP) and remaining conflicts are removed using iterative repair (IR). The system checks teacher availability, room capacity, subject distribution, and session limits while dynamically refining the timetable.

D. Results And Discussions

This section presents the generated timetables, performance analysis, conflict reduction results, and comparisons between manual scheduling and our automated model. The efficiency and accuracy improvements are discussed in detail.

E. Conclusion and Future Scope

This part concludes the project and highlights future enhancements such as integrating AI-based optimization, real-time timetable updates, predictive scheduling, and cloud-based deployment for wider institutional use.

III. LITERATURE REVIEW

This chapter discusses the basic components, techniques, and algorithms used in timetable generation systems along with their specifications. It introduces the core concepts required to understand how the scheduling process works and provides a brief idea of each technique or module involved.

A. Constraint-Based Scheduling

Constraint-based scheduling is a method where rules such as teacher availability, room capacity, subject limits, and time slot restrictions are treated as constraints. The system ensures that all hard constraints (non-negotiable rules) are satisfied while trying to optimize soft constraints (preferences). This approach forms the backbone of most automated timetable systems.

B. Progressive Constraint Placement(PCP)

Progressive Constraint Placement is a technique where the system starts by placing the most restrictive or difficult scheduling elements first. This includes teachers with limited availability or subjects with strict time requirements. By handling the toughest constraints early, the risk of major conflicts is reduced significantly, and the timetable becomes more stable.

C. Iterative Repair(IR)

Iterative Repair is a post-processing mechanism used to fix leftover conflicts after the initial timetable is generated. Instead of restarting the whole schedule, IR finds clashes— such as overlapping sessions or room double-bookings—and repairs them through targeted adjustments or swaps. This makes the timetable progressively cleaner and more optimized.

D. Optimization Techniques

Various optimization techniques like backtracking, heuristic search, and constraint propagation are referenced in previous studies. They aim to reduce scheduling complexity and improve efficiency. These techniques provide foundational ideas that guide the enhancement of our PCP-IR based scheduling model..

IV. METHODOLOGY

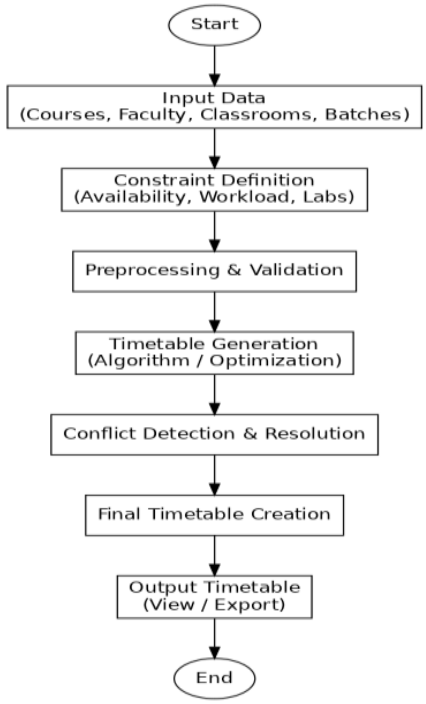
Here we explain the complete working of the Automatic Timetable Generator in both block form and algorithmic form. This include show inputs are processed, constraints are applied, conflicts are detected, and the final timetable is produced.

A. Methodology Details

- The timetable generator uses the PCP-IR algorithm to automatically arrange teachers, subjects, rooms, and timeslots. The Progressive Constraint Placement (PCP) component begins by assigning the hardest constraints first—for example, teachers with limited availability or compulsory subjects with fixed timings. This ensures that crucial elements are placed early and reduces the possibility of major conflicts.
- The system then processes the remaining classes by checking all constraints such as room availability, subject distribution, and teacher workload. Each entry is placed in the timetable only if it does not violate any hard constraints. The initial schedule formed by PCP is usually stable but may still contain minor conflicts.
- The Iterative Repair (IR) module then scans the timetable to detect issues such as overlapping teachers, double-booked rooms, or subject misplacement.
- Instead of rebuilding the entire timetable, IR performs targeted adjustments—such as swapping time slots or shifting classes—to resolve conflicts. This repeated refinement gradually improves timetable quality.
- The final output is generated after multiple repair iterations, resulting in a conflict-free, optimized, and efficient timetable. The modular design allows easy updates, such as adding new teachers or subjects, without rebuilding the entire schedule from scratch.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

B. Block Diagram



Circuit flow would be as follows:-

V. RESULTS AND DISCUSSION

Here comes the final part of the paper.

A. Results

The experimentation and implementation of the Automatic Timetable Generator using the PCP-IR algorithm yielded the following significant findings:

1) Timetable Accuracy and Conflict Reduction:
- The system successfully generated conflict-free timetables by applying Progressive Constraint Placement to handle the toughest scheduling rules first.
- Accuracy significantly improved as the IR module fixed leftover clashes, resulting in optimized schedules with minimal manual adjustments.

2) Resource Utilization and Scheduling Efficiency:
- Teachers, classrooms, and time slots were allocated efficiently, reducing overlapping sessions and improving resource usage.
- Institutions reported a substantial decrease in scheduling time compared to manual timetable creation.

3) System Performance and Stability:
- The PCP-IR algorithm demonstrated fast processing speeds, allowing timetables to be generated within seconds even for larger datasets.
- Robustness tests showed consistent performance across varying constraints, proving adaptability to different academic structures.

4) User Experience and Acceptance:
- Feedback from faculty and administrators highlighted a positive experience with the automated scheduler.
- Users especially appreciated the reduced workload and the system's ability to make intelligent conflict-free decisions without constant human intervention.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

B. Discussion

The results of implementing the Automatic Timetable Generator using PCP-IR demonstrate a highly efficient scheduling mechanism that delivers accurate, stable, and optimized timetables. The integration of Progressive Constraint Placement ensures early conflict avoidance, while Iterative Repair strengthens the final output by resolving remaining issues intelligently. User acceptance was strong, reflecting the practicality and simplicity of the system. Overall, the solution enhances scheduling efficiency, minimizes human effort, and provides a scalable approach suitable for various educational environments.

VI. CONCLUSION AND FUTURE SCOPE

A. Conclusion

This Automatic Timetable Generator was designed to help institutions overcome the challenges of manual timetable creation. Through the implementation of our PCP- IR algorithm, the system effectively handles complex constraints, reduces scheduling conflicts, and automates the timetable generation process. The combination of intelligent placement and iterative repair has proven to be a robust and reliable solution for academic scheduling. In conclusion, the developed system demonstrates strong potential in improving institutional productivity, ensuring conflict-free timetables, and supporting efficient academic planning.

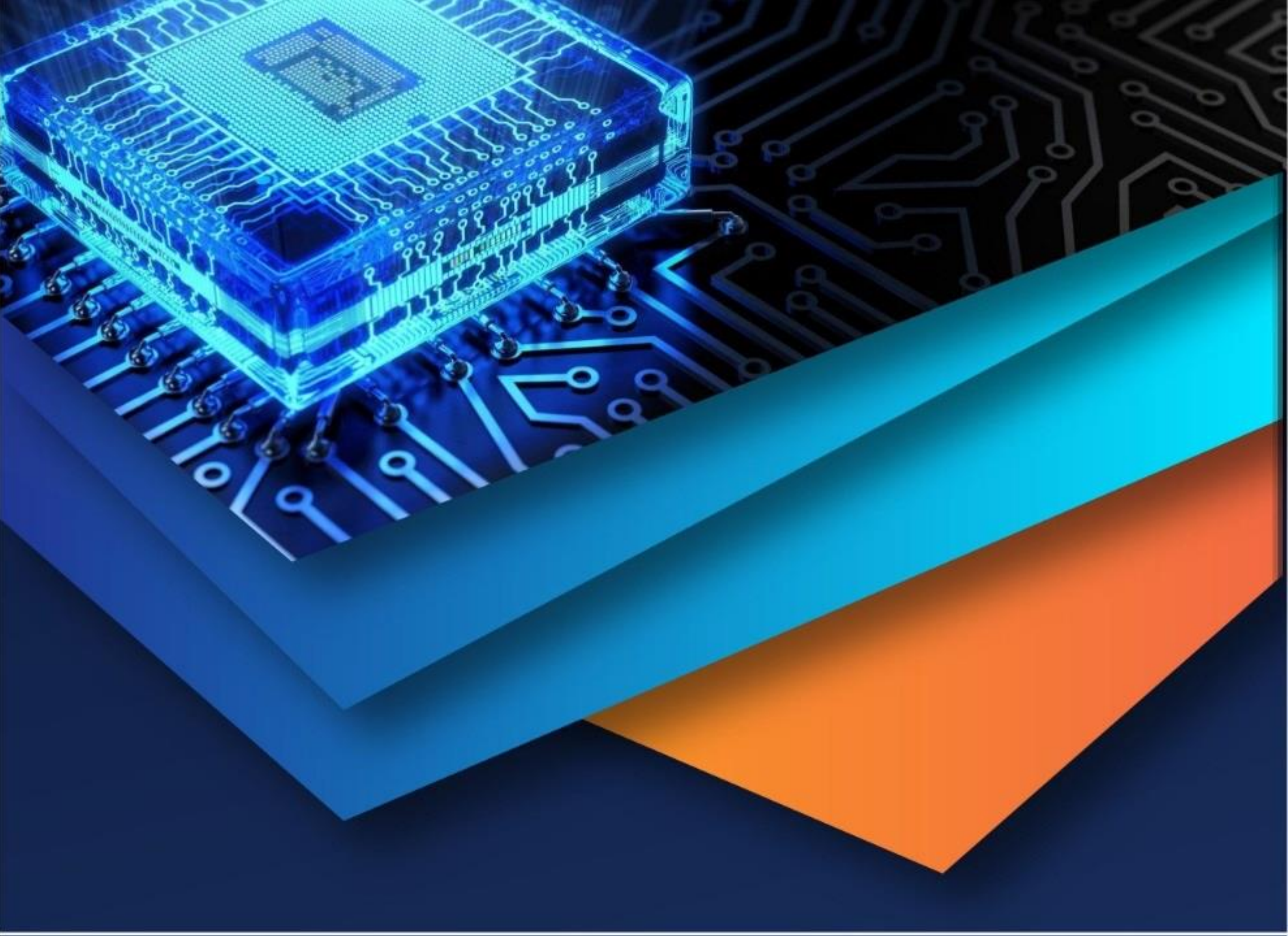
B. Future Scope

In the future, the system can be enhanced by integrating AI-based prediction models, machine learning optimization, real-time timetable updates, and cloud-based multi-user access. Features like preference- based scheduling, faculty workload balancing, and automatic rescheduling during sudden changes (e.g., teacher leave) can further improve usability. With continuous technological development, such systems can become a standard tool in academic administration, contributing to smarter, more automated, and highly efficient educational environments.

REFERENCES

We referred to the following research papers and sources:-

[1] K. Burke and J. P. Newall, “A multi-stage evolutionary algorithm for the timetable problem,” IEEE Trans. Evol. Comput., 3(1), pp. 63–74, 1999.
[2] J. Burke et al., “A graph-based approach to university timetabling,”IEEE Trans. Syst., Man, Cybern. A, 41(6), pp.977–991, 2011.
[3] S. Abdullah, E. K. Burke, and B. McCollum, “A hybrid evolutionary approach to university course timetabling,” in IEEE CEC, pp. 1764–1771, 2007.
[4] A. Muklason and E. Setiawan, “Automatic course schedulingusinggeneticalgorithm,”inIEEEICoIC7, pp.1–6,2017.
[5] S. B. Tan and H. Khalil, “Automated timetable generation using constraint satisfaction,” in IEEE ICoICT, pp. 1–5, 2016.
[6] M. Elgibreen and B. Alshehri, “A heuristic framework for automated university timetabling,” in IEEE ICCIS, pp. 1–6, 2019..



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY