A Hybrid Multi-Swarm Particle Swarm Optimization and Integer Programming Solution for the University Course Timetabling Problem

Gian Myrl D. Renomeron
CMSC 199.1: Research in Computer Science I
Division of Natural Sciences and Mathematics
University of the Philippines Tacloban College
gdrenomeron@up.edu.ph

A Hybrid Multi-Swarm Particle Swarm Optimization and Integer Programming Solution for the University Course Timetabling Problem

1 Introduction

1.1 Background of the Study

University Course Timetabling Problem (UCTP)

Integer Programming (IP)

Swarm Intelligence

Particle Swarm Optimization (PSO)

Multi-Swarm Particle Swarm Optimization (MSPSO)

Hybrid Optimization Approach

1.2 Related Works

2 Statement of the Problem

The UCTP problem is quite complex and key for any institution of learning because it considers assigning courses to specific rooms and time slots while considering constraints such as room capacities, course prerequisites, and faculty courses. [1] To this date, most universities still consider non-optimal manual or heuristic methods, and the inefficiencies include uneven teaching loads and a mismatch between qualifications and requirements in course teaching. [2] While the complexity of the problem is increased, it happens to be highly infeasible to find the best solutions, especially when there are conflicting constraints that need to be fulfilled.

While in the current literature IP[1] [3] and PSO [2] [4] are applied separately for solving optimization problems, a hybrid approach combining the strengths of both has not been produced, particularly with MSPSO [5] to handle the complexities of the issue under discussion. The absence of such approaches means current solutions may not optimize the timetabling process or institute needs to the fullest extent. This gap requires a more sophisticated and efficient approach to improving the quality of fairness in faculty course assignments.

This study aims to develop a hybrid optimization approach combining MSPSO with IP that effectively solves the UCTP problem. The combination of these methods would enable the research to explore a large solution space more effectively, combined with the proximity of a better solution within a reasonable time frame, to help ensure that many other constraints, such as room capacities, course prerequisites, and scheduling conflicts are well satisfied. The approach would then be estimated using such key metrics as workload balance, course-faculty alignment, and computational efficiency to enhance the proper resource allocation within an institution.

3 Objectives of the Study

3.1 General Objective

To design and develop a hybrid optimization approach that merges Multi-Swarm Particle Swarm Optimization with IP to solve the University Course Timetabling Problem.

3.2 Specific Objectives

1. To analyze constraints and requirements involved in UCTP, such as room availability, course prerequisites, and faculty courses.

2. To design and implement a Multi-Swarm Particle Swarm Optimization (MSPSO) algorithm to effectively explore the solution space for course scheduling.

3. To formulate an Integer Programming (IP) model specifically to enhance MSPSO-evolved solutions to fit well against the hard constraints, maximize resource utilization, as well as schedulability.

4. To develop the hybrid model of MSPSO-IP and validate using real-time data.

5. To compare the hybrid model and existing algorithms in terms of performance.

4 Scope and Limitation

This paper presents a hybrid optimization approach, in which Multi-Swarm Particle Swarm Optimization and Integer Programming are combined to design the solution to the University Course Timetabling Problem. It includes an exhaustive analysis of critical constraints such as room availability, course prerequisites, and faculty course assignments. The proposed hybrid model will improve the effectiveness of schedules by addressing multiple constraints across different institutions. It is validated using real-time data to ensure adaptability in other contexts and environments. In addition, the evaluation of the hybrid model will incorporate a comparison with existing algorithms to fulfill the faculty preferences for an even more comprehensive benchmark.

This study still bears a few limitations despite its strengths. Though the system aims to maximize resource usage and minimize potential resource conflicts, it does not consider more dynamic changes that may occur in scheduling, such as a course change at a stage of scheduling or an immediate availability of rooms at some stage of scheduling. The model will be tested strictly within constraints and data about multiple institutions that may be available but would limit the general applicability of results from this study in different educational scenarios. Therefore, this study focuses on the hard constraints, though subjective elements of scheduling may not be completely reflected in the solution developed considering faculty preferences for benchmarking purposes only.

5 Significance of the Study

In this study, the novelty behind solving the UCTP problem involves developing a hybrid solution that combines Multi-Swarm Particle Swarm Optimization and Integer Programming techniques. This will be important because of its ability to handle weight and room allocation issues and conflict-free timetables, maximizing the use of all entities. Using the hybrid approach, wide exploration of potential solutions using MSPSO would be possible while refining these using IP to ensure the solutions are within the hard constraints. Thus, this optimization technique can lay a foundation for automating course scheduling in many institutions to make the process more efficient and fair.

6 Theoretical and Conceptual Framework

7 Methodology

This section outlines the hybrid methodology combining Multi-Swarm Particle Swarm Optimization (MSPSO) and Integer Programming (IP) to effectively solve the University Course Timetabling Problem (UCTP).

7.1 Problem Decomposition

- Decompose the UCTP into exploratory and refinement components.

- Identify exploratory tasks suitable for MSPSO and refinement tasks for IP.

- Define the structure of the solutions represented by particles in MSPSO, including course, room, and time slot assignments.

7.2 Multi-Swarm Particle Swarm Optimization (MSPSO)

- Initialize multiple swarms to explore various potential timetabling solutions.

- Define the representation of solutions (particle encoding) for the assignment of courses to rooms and time slots.

- Develop a fitness function to evaluate solutions based on:

  - Minimizing scheduling conflicts (e.g., room clashes, instructor availability).

  - Maximizing resource usage (e.g., optimal room utilization and course distribution across time slots).

- Implement the MSPSO algorithm:

  - Update particle positions and velocities based on individual and global best solutions.

- Perform iterations until a convergence criterion is met or a predefined number of iterations is reached.
- Collect the best solutions generated by the swarms for refinement.

7.3 Integer Programming (IP) Refinement

- Formulate an Integer Programming model based on the best candidate timetables from MSPSO.
- Define decision variables, objective function, and constraints:
  - Decision variables: Assignments of courses to specific time slots and rooms.
  - Objective function: Minimize conflicts (e.g., no double-booked rooms) and optimize the balance of room and time slot usage.
  - Hard constraints: No overlaps between courses scheduled in the same room at the same time, no double-booking of instructors, etc.
- Use the IP solver to refine the timetable by ensuring all hard constraints are satisfied while optimizing the objective function.

7.4 Hybrid Iteration

- Optionally, implement an iterative feedback loop between MSPSO and IP:
  - Re-apply MSPSO to the refined solutions from IP to explore new or improved regions of the solution space.
  - Adapt MSPSO parameters based on the results from the IP refinement process to search for better solutions.
- Repeat the hybrid optimization process for several iterations to further enhance solution quality and robustness.

7.5 Post-processing and Evaluation

- Post-process the final timetabling solutions to evaluate against key performance metrics:
  - Conflict minimization: Ensure no room or instructor conflicts in the final schedule.
  - Course-resource alignment: Maximize the alignment of course scheduling with available room resources and time slots.
  - Computational efficiency: Measure the overall performance of the hybrid optimization approach in terms of speed and accuracy.
- Select the best overall timetable based on the combination of fitness and objective evaluations.

8 Schedule of Activities

OBJECTIVES	TARGET ACTIVITIES	TARGET ACCOMPLISHMENTS (quantify, if possible)	YEAR 1											
			1	2	3	4	5	6	7	8	9	10	11	12
Objective 1	To analyze constraints and requirements involved in UCTP, such as room availability, course prerequisites, and faculty courses.	1. Gather data on room availability, course prerequisites, and faculty loads. 2. Identify and categorize hard and soft constraints in UCTP.									███	███	███	███
Objective 2	To design and implement a Multi-Swarm Particle Swarm Optimization (MSPSO) algorithm to explore the solution space for course scheduling.	1. Develop the MSPSO algorithm for course scheduling. 2. Test and refine the algorithm using synthetic data.									███	███	███	███

OBJECTIVES	TARGET ACTIVITIES	TARGET ACCOMPLISHMENTS												
Objective 3	To formulate an Integer Programming (IP) model specifically to enhance MSPSO-evolved solutions to fit hard constraints and maximize resource utilization.	1. Develop the IP model to optimize resource utilization. 2. Combine IP with MSPSO to improve solution accuracy and feasibility.									▓	▓	▓	▓

OBJECTIVES	TARGET ACTIVITIES	TARGET ACCOMPLISHMENTS (quantify, if possible)	YEAR 2											
			1	2	3	4	5	6	7	8	9	10	11	12
Objective 4	To validate the hybrid MSPSO-IP model using real-time data.	1. Collect real-time data for validation. 2. Test the hybrid model in real-time scheduling scenarios.	▓	▓	▓	▓	▓							
Objective 5	To compare the hybrid model and existing algorithms in terms of performance.	1. Compare performance metrics with existing algorithms such as PSO, GA. 2. Conduct detailed benchmarking and performance analysis.	▓	▓	▓	▓	▓							

References

[1] Nancy Maribel Arratia-Martinez, Cristina Maya-Padron, and Paulina A Avila-Torres. "University course timetabling problem with professor assignment". en. In: *Math. Probl. Eng.* 2021 (Jan. 2021), pp. 1–9.

[2] Oswald C. and Anand Deva Durai C. "Novel hybrid PSO algorithms with search optimization strategies for a University Course Timetabling Problem". In: (Dec. 2013).

[3] Monica Torres, Kyla Kiela Villegas, and Maica Krizna Gavina. "Solving faculty-course allocation problem using integer programming model". en. In: *Philipp J. Sci.* 150.4 (June 2021), p. 679.

[4] A I Ali and R Talal. "UCTP based on Hybrid PSO with Tabu Search Algorithm using Mosul university dataset". In: *International Journal of Computer Applications* 975 (2014).

[5] Xuewen Xia, Ling Gui, and Zhi-Hui Zhan. "A multi-swarm particle swarm optimization algorithm based on dynamical topology and purposeful detecting". In: *Applied Soft Computing* 67 (2018), pp. 126–140. ISSN: 1568-4946. DOI: `https://doi.org/10.1016/j.asoc.2018.02.042`. URL: `https://www.sciencedirect.com/science/article/pii/S1568494618301017`.