FPT-HA NOI UNIVERSITY

-----***-----

FPT Education

FPT UNIVERSITY

GRADUATION THESIS REPORT

- Artificial Intelligence -

Solving The Examination Timetabling Problem: A Nash Equilibrium Approach With Genetic Algorithm And Tabu Search Comparisons

INSTRUCTOR: LUONG TRUNG KIEN

GROUP MEMBER: NGUYEN DUY TUNG

BUI VAN QUYEN

BUI TUAN DAT

MAJOR: ARTIFICIAL INTELLIGENCE

April 26, 2024

ACKNOWLEDGMENTS

We would like to express our heartfelt thanks to our respected teacher, Mr. Luong Trung Kien for his consistent mentorship and assistance during the time we worked on our final project. His knowledge and commitment have been extremely helpful to us throughout the past four months, enabling us to finish our project with confidence.

We are also grateful to FPT SAP Lab, particularly Mrs. Le Phuong Chi, for their cooperation and support. Their input has played a key role in the success of our project, providing us with valuable resources and perspectives that significantly enhanced our work.

Furthermore, we want to sincerely express our appreciation to all the lecturers at FPT University, particularly those within the Information Technology Specialization Department. Their dedication to teaching and their enthusiasm for sharing knowledge have greatly influenced our academic journey. The education and experiences we acquired during our time at the university have established a strong foundation for our future pursuits.

We are truly thankful to all who have contributed, both mentioned and unmentioned. Your support has guided us towards achieving our goals and has been a beacon of light for our growth and success.

FPT University, Hanoi, Spring Semester 2024

The authors of this thesis

ABSTRACT

Scheduling examinations is a complex challenge for educational institutions, as it involves meeting strict requirements and satisfying various parties such as invigilators, students, and academic departments. This research addresses the Examination Timetabling Problem in university settings, aiming to optimize the scheduling of exams by balancing various constraints and stakeholder interests. We employ Nash Equilibrium to evaluate the competing interests of students, faculty, and administrative departments while contrasting and assessing the effectiveness of two meta-heuristic algorithms: Genetic Algorithm and Tabu Search. The experimentation phase utilized real-world data encompassing scheduling needs for 11,509 students, 156 subjects, and 275 invigilators over a span of nine examination days. Results indicate that Tabu Search notably provides superior outcomes by more effectively minimizing conflicts while maximizing stakeholder satisfaction.

Keywords: Nash Equilibrium, Genetic Algorithm, Tabu Search.

TABLE OF CONTENTS

List of Abbreviations

Abbreviations	Description
UTP	University Timetabling Problem
ETP	Examination Timetabling Problem
TAP	Teaching Assignment Problem
GA	Genetic Algorithm
TS	Tabu Search
IWD	Intelligent Water Drops

List of Figures

1. INTRODUCTION

1.1. Motivation

The Examination Timetabling Problem is a fundamental scheduling issue that institutions worldwide face, impacting both resource allocation and the academic performance of students. The increasing number of courses and students, coupled with limited resources, has made creating an efficient and fair exam schedule more challenging and complex. Traditional manual approaches to timetabling are labor-intensive and prone to human errors, resulting in suboptimal schedules that can cause unnecessary stress for students and inefficient use of resources.

The adoption of a Nash Equilibrium framework for the ETP is motivated by the potential to achieve a balanced solution that no participant can unilaterally improve upon, thereby ensuring fairness and efficiency. This concept from game theory provides a structured way to consider the preferences and constraints of various stakeholders simultaneously. By modeling the problem within this framework, we aim to ensure that the exam schedule is as fair as possible to all stakeholders involved, including students, faculty, and the administration.

1.2. Problem Overview

University operations require effective scheduling, which poses a multifaceted challenge that requires meticulous planning and resource allocation. The University Timetabling Problem (UTP) encompasses three primary types of scheduling dimensions [1]: Enrollment Timetabling Problem, Teaching Assignment Problem (TAP), and Examination Timetabling Problem (ETP). Each type presents its own set of complexities, from accommodating student preferences to optimizing faculty workload. The Enrollment Timetabling Problem involves scheduling courses and allocating students to classes during specific timeslots, considering factors such as course prerequisites and student preferences [2]. The Teaching Assignment Problem focuses on assigning instructors to courses based on their expertise and availability [3]. Finally, the **Examination Timetabling Problem (ETP)**, the focus of this paper, deals with scheduling exams for multiple courses to specific time slots and rooms strategically while satisfying various constraints such as avoiding conflicts for students [4], [5]. Our study also deals with efficiently assigning invigilators to oversee each examination session. The objective of our research is not only to ensure a fair and balanced distribution of the examination load but also to navigate through various constraints, including room capacities, invigilator availability, and other logistical considerations. Tackling these complexities within the ETP framework is crucial for optimizing the overall efficiency and fairness of university examination scheduling processes.

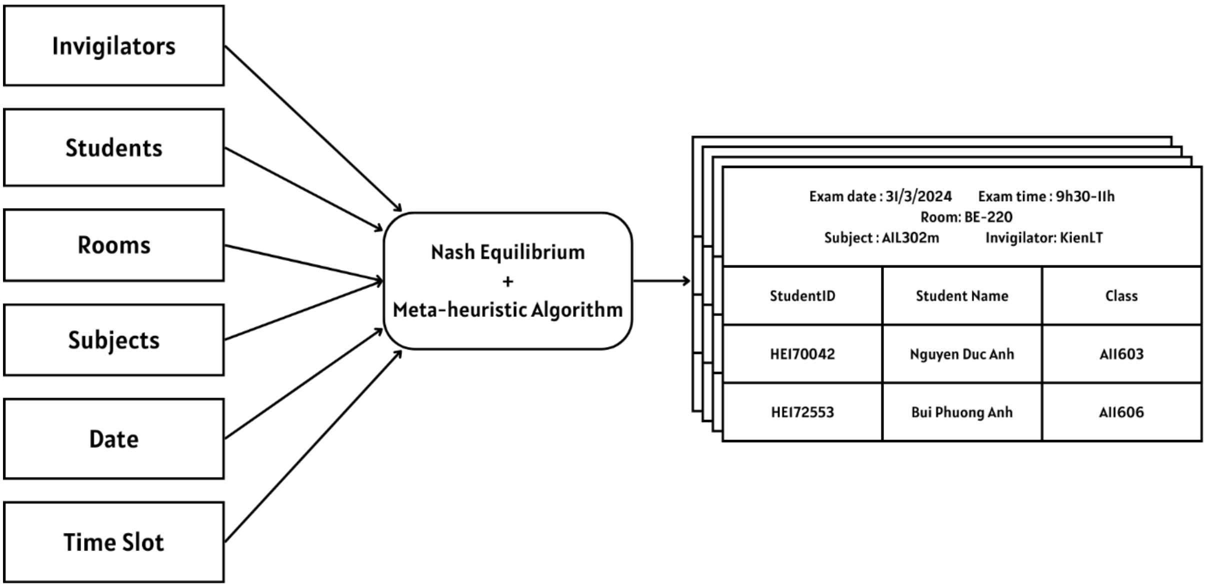


Figure 1. *Examination Timetabling Problem*

Traditional scheduling systems often focus solely on creating a timetable that satisfies certain constraints but typically do not consider balancing the interests among the participants. The stakeholders, including faculty, students, and the institution itself, often have conflicting interests. A schedule that benefits one student may disadvantage another—for instance, a student will have more time to review between two consecutive subjects than another student.

Therefore, our research aims to develop an examination timetable that not only satisfies the required constraints but also balances the interests of all stakeholders. We plan to use Nash Equilibrium as a tool to analyze the interests of each party. The concept of Nash Equilibrium describes a situation where no participant can gain by unilaterally changing their strategy if the strategies of the others remain unchanged [6]. Thus, a timetable corresponding to a Nash Equilibrium would ensure a balance of interests among all parties, which should lead to a more equitable examination scheduling system.

The Examination Timetabling Problem is a significant challenge in the educational sector due to its vast and complex problem space, making it an NP-difficult problem. To effectively address these complexities and find the Nash equilibrium point, meta-heuristics such as Genetic Algorithms [7], Simulated Annealing [8], and Tabu Search [9], [10], offer a versatile framework for exploring solution spaces and optimizing schedules. Our research aims to deliver an efficient, scalable algorithm that improves efficiency and fairness in university examination scheduling while minimizing conflicts and accommodating various constraints.

In the following sections, we will examine existing literature on timetabling problems, delve into the background of the Examination Timetabling Problem, outline our research methodology, and analyze the results. Our goal is to provide valuable insights to the field of educational scheduling by introducing a new and effective approach to tackling the complex challenges presented by the Examination Timetabling Problem.

1.2. Related Work

Throughout history, numerous researchers have extensively studied examination timetabling and contributed valuable insights to this field. This literature review explores research in three main areas: the constraints and objective functions utilized (Table 1), the modeling strategies employed to tackle the multi-objective problem especially the Nash Equilibrium model, and the application and results of meta-heuristic algorithms in resolving ETP.

Constraints \ Papers		[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
Soft Constraints	Maximize spacing between exams for better student exam preparation			x					x	
	Minimize the number of exam rooms				x		x			x
	Maximizing the preference over slots				x					
	Minimize deviations in assigning courses to the same day, different slots, and single floor				x					
	Minimize the number of students taking a medium exam with less than 1 day free and those taking a hard exam with less than 2 days free					x				

	Invigilator duties should be evenly spread among the lectures and staff						x		x	
	The number of students taking exams should be evenly distributed throughout the exam period.							x		
	Minimize the distance between exam rooms for the same exam							x		
Hard Constraints	The total number of students in a room must be less than the room's capacity	x	x			x		x	x	x
	Exams requiring multiple slots must be scheduled consecutively	x								
	No student should be required to sit two examinations simultaneously	x	x	x		x		x	x	x
	Limit on the number of time slots/days	x								
	Limitations of rooms each slot		x					x		
	Some exams must be scheduled pairwise		x							
	All exams must be scheduled and held only once			x		x		x	x	x
	Each room is used for only one exam at a time				x			x		

	Rooms for the same exam course must be on the same floor				x	x				
	Hard exams should not be schedule in the first two days					x				
	An invigilator is not allowed to oversee their own examination paper						x			
	Some exams only allow staff to serve as chief invigilators						x			
	Limit the number of exams for each invigilator						x			
	Invigilator is only assigned one invigilation duty at a time						x		x	
	Number of invigilators required for each room						x		x	

Table 1*. Soft and hard constraints of datasets in related research*

Scheduling for invigilator has not received as much attention as the examination timetabling for students, mainly due to the unavailability of datasets related to it. Moreover, an overlooked aspect of the examination scheduling problem is the simultaneous optimization for both students and invigilators. Recognizing this gap, my study aims to bridge the disparity by proposing an optimization approach that meets the needs of both students and invigilators. By adopting a balanced perspective, this study seeks to develop a comprehensive solution that not only enhances the examination experience for students but also streamlines the workload and preferences of invigilators.

The Examination Timetabling Problem is a multi-objective optimization challenge, requiring a careful balance between conflicting goals. Many researchers have tackled this problem with different strategies, focusing on meeting constraints to create feasible schedules. For instance, Mirhassani et al. explored various objective functions and solution methods for university timetabling, highlighting the importance of addressing both hard and soft constraints [20]. Another example is Bashab et al. ‘s comprehensive study on addressing university scheduling problems using hyper-heuristic algorithms, demonstrating the effectiveness of these methods in optimizing schedules based on constraints [21]. Moreover, Ngo et al. proposed a programmatic compromise to address examination timetabling problem [22] and Coursera timetabling [23]. Besides the education field, Ngo has also used this method to solve multi-objective optimization problems in various fields such as logistics [24] and selection [25].

However, these studies typically do not consider the interests of all stakeholders. Researcher Chi et al. [3] utilized the Nash theory to analyze conflicts and solve the university teaching assignment problem by balancing the interests of all parties involved. The Nash equilibrium has been applied in diverse scenarios such as scheduling the operational workflow in an electroplating facility [26], resource allocation in cloud computing [27], selecting and allocating the most suitable resources for the execution of jobs that users submit [28], and helping for the widespread implementation of price-responsive electrical appliances on a large scale [29]. Indeed, the NASH equilibrium has proven effective for conflict analysis and finding optimal solutions.

The field of examination timetabling has seen a variety of approaches due to the diverse constraints inherent to the problem. Meta-heuristics, including genetic algorithms, simulated annealing, and Tabu search, have emerged as powerful tools for navigating the complex solution spaces in examination timetabling.

A novel variant of the simulated annealing algorithm, termed FastSA, was introduced by Leite et al. [8], which employs an acceptance criterion that only evaluates exams with previously accepted moves. This method has been shown to significantly reduce simulation time and complexity in solving university examination timetabling problems. In a comparative study involving Salem [30] examined the efficacy of Genetic Algorithm (GA) against Constructive Heuristic model. The latter, which simulates manual scheduling processes, was found to outperform GA in creating optimized, conflict-free schedules. Further exploring genetic algorithms, Mohammed et al. [31] proposed a GA-based strategy that focuses on maximizing student achievement while adhering to the necessary hard constraints for timetable feasibility.

A hybrid approach that combines Intelligent Water Drops (IWD) with a local search algorithm was proposed by Aldeeb et al. [4]. This method enhanced the exploitation capabilities of the IWD algorithm and showed promising results across various datasets, surpassing other metaheuristic and swarm intelligence approaches. One form of metaheuristic algorithm that is based on a single solution is tabu search. Kendall et al. [32] demonstrated the effectiveness of this approach in scheduling by analyzing real-world data and generating solutions using a hyper-heuristic based on tabu search. Their findings revealed that their solution outperformed an existing manual solution by at least 80% in terms of proximity cost. Additionally, they conducted comparisons with a benchmark dataset to illustrate the ability of their method to yield high-quality results.

Scheduling can be a time-consuming process, especially when it involves balancing multiple objectives. During the literature review session, it was observed that there are limitations in considering the benefits for all stakeholders involved in ETP. Some researchers tend to prioritize optimizing the experience for specific groups, such as invigilators or students, while others focus solely on developing solutions tailored to their specific case studies.

1.3. Objectives

In this study, we will present an approach to create a comprehensive and well-organized timetable for students, invigilators, and academic departments that carefully balances the benefits of each stakeholder group. For students, our approach aims to minimize time gaps between exams for better preparation while reducing scheduling conflicts. Additionally, it seeks to consolidate invigilators' schedules into fewer days with supervised slots closely matching the required number. Furthermore, the goal is to ensure a balanced distribution of room usage across different time slots. Our methodology integrates the concept of Nash Equilibrium to analyze conflicts of interest among parties and determine the superior method between tabu search and genetic algorithm for resolving university examination timetable scheduling issues.

2. METHODOLOGY

2.1. Problem Formulation

To address the complexities of the examination timetabling problem, it is crucial to establish a clear understanding of the input data that governs the scheduling process. In this subsection, we explore the problem formulation by examining the necessary input data for constructing a comprehensive timetable. This includes:

- N_M: The number of Students
- N_I: The number of Invigilators
- N_R: The number of Rooms
- N_S: The number of Subjects
- N_D: The number of Days
- α: Maximum number of students in each room.
- β: The number of slots a day
- Matrix A = $\{a_{m,s}$, $1 \leq m \leq N_M$, $1 \leq s \leq N_s\}$,$a_{m,s}$ = 1 if student m will take subject s, otherwise 0.
- Matrix C = $\{c_{s,i}$, $1 \leq s \leq N_S$, $1 \leq i \leq N_I\}$, $c_{s,i}$ = 1 if subject s can be supervised by invigilator i.
- Vector L = $\{l_s$, $1 \leq s \leq N_S\}$, l_s denotes the length of subject s.
- Vector Q = $\{q_i$, $1 \leq i \leq N_I\}$, $q_i \in$ N, denotes number of slots inivigilator i need to supervise.
- $N_T = \beta * N_D$: denotes number of timeslots.
- Vector E = $\{e_m \mid e_m = \sum_{s=1}^{N_S} a_{m,s}$, $1 \leq m \leq N_M\}$, e_m denotes the number of subjects of each students need to take part in.
- Vector F = $\{f_s | \, f_s = \sum_{m=1}^{N_M} a_{m,s}$, $1 \leq s \leq N_s\}$, f_s denotes number of students that take subject s.
- Vector G = $\left\lceil \frac{F}{\alpha} \right\rceil$ = $\{g_s$, $1 \leq s \leq N_S\}$, g_s denotes number of rooms needed to hold for subject s.

The decision variable is a matrix D = {$d_{s,t,i}$, 1 ≤ s ≤ N_S, 1 ≤ t ≤ N_T, 1 ≤ i ≤ N_I}, Where $d_{s,t,i} = 1$ if subject s is held at slot t and supervised by invigilator i, $d_{s,t,i} = 0$ if subject s is not held at slot t and not supervised by invigilator i.

Some denotations can be inferred from decision variable:

- Matrix H = {$h_{s,t}$| $h_{s,t} = \begin{cases} 1 \; if \; \sum_{i=1}^{N_i} d_{s,i,t} > 0 \\ 0 \qquad otherwise \end{cases}$, 1 ≤ s ≤ N_s, 1 ≤ t ≤ N_T }, $h_{s,t}$ denotes if subject s is held at timeslot t.
- Vector X = {x_s| $x_s = \begin{cases} t \; if \; h_{s,t-1} = 0 \; and \; h_{s,t} = 1 \\ 0 \qquad otherwise \end{cases}$, 1 ≤ s ≤ N_S}, x_s denotes the slot start of subject s.
- Vector Y = X + L – 1 = {y_s, 1 ≤ s ≤ N_s}, y_sϵ [1, N_T] denotes the slot end of subject s.
- Matrix Z = A * X = {$z_{m,s}$|$z_{m,s} \in [1, N_T]$, 1 ≤ m ≤ N_M, 1 ≤ s ≤ N_s}, $z_{m,s}$ denotes the slot start for subject s of student m.
- Matrix K ={$k_{i,d}$| $k_{i,d} = \begin{cases} 1 \; if \; \sum_{t=\beta(d-1)+1}^{\beta d} \sum_{s=1}^{N_s} d_{s,t,i} > 0 \\ 0 \qquad otherwise \end{cases}$, 1 ≤ i ≤ N_I, 1 ≤ d ≤ N_D}, $k_{i,d}$ denotes the number of slots invigilator i supervises in day d.

2.2. Stakeholder Analysis using Nash Equilibrium

The examination timetabling process involves multiple parties, such as academic departments, students, and instructors. These parties often have conflicting interests that need to be carefully balanced in order to achieve fair outcomes without favoring any particular group. Therefore, it is crucial to prioritize fair distribution of benefits among all stakeholders involved in the scheduling process. To navigate this complex landscape of competing interests, we utilize the concept of Nash Equilibrium from game theory to understand strategic behaviors and interactions among the parties involved. We model the problem as a game with 3 groups: player P0 representing the academic department, group B representing players who are students, and group V representing players who are invigilators.

Special player P_0 represents the interest of academic department.

The benefit of player P_0 is the uniformity in the number of rooms between each time slot. This benefit is represented by the following payoff function, denoted as $Payoff_{P0}$.

Let μ is the mean of number of rooms in each slot:

$$\mu = \frac{\sum_{t=1}^{N_T} \sum_{s=1}^{N_s} \sum_{i=1}^{N_I} d_{s,t,i}}{N_T}$$

Then, we utilize the standard deviation equation to derive the payoff function of player P_0:

$$Payoff_{P0} = w_1 * \sqrt{\frac{\sum_{t=1}^{N_T} (\sum_{s=1}^{N_s} \sum_{i=1}^{N_I} d_{s,t,i} - \mu)^2}{N_T - 1}}$$

Player b_m ($1 \leq m \leq N_M$), represents the m-th student.

B = $\{b_1, b_2, \ldots, b_{N_M}\}$ is a set of players who are students. Each student wishes that his/her exam subjects be distributed evenly throughout the entire period.

Let $\frac{N_T}{e_{b_m}}$ be the time slot gap student b_m wants between two exams and let U = sort(Z) in descending order be the slot start for each exam, from the last exam to the first one that each student joined. We have payoff of a student b_m:

$$Payoff_{b_m} = \ln\left(\frac{1}{e_{b_m} - 1} * \sum_{i=1}^{e_{b_m}-1} e^{\left|u_i - u_{i+1} - \frac{N_T}{e_{b_m}}\right|}\right)$$

Payoff of all players b_m

$$Payoff_{AllB} = w_2 * \frac{1}{N'_M} * \sum_{m=1}^{N'_M} Payoff_{b_m}$$

$$= w_2 * \frac{1}{N'_M} * \sum_{m=1}^{N'_M} \left(\ln\left(\frac{1}{e_{b_m} - 1} * \sum_{i=1}^{e_{b_m}-1} e^{\left|u_i - u_{i+1} - \frac{N_T}{e_{b_m}}\right|}\right)\right)$$

N'_M: The number of students who have more than 1 exam.

Player V_i ($1 \leq i \leq N_I$), represents the i-th invigilator.

V = $\{v_1, v_2, \ldots, v_{N_I}\}$ is a set of players who are invigilators. Each invigilator v_i seeks to compress their invigilation schedule into as few days as possible:

$$g_{v_i} = \sum_{d=1}^{N_D} k_{v_i,d}$$

Furthermore, the actual number of exam slots for each invigilator v_i should be as similar as possible to the number of slots required.

$$h_{v_i} = \left| \sum_{s=1}^{N_s} \sum_{t=1}^{N_t} d_{s,t,v_i} - q_{v_i} \right|$$

We deduce the payoff function of a player v_i:

$$Payoff_{v_i} = w_4 * g_{v_i} + w_5 * \ h_{v_i}$$

Payoff of all players v_i

$$Payoff_{AllV} = w_3 * \frac{1}{N_I} * \sum_{i=1}^{N_I} Payoff_{v_i} = w_3 * \frac{1}{N_I} * \sum_{i=1}^{N_I} \left(w_4 * g_{v_i} + w_5 * \ h_{v_i} \right)$$

$$= w_3 * \frac{1}{N_I} * \sum_{i=1}^{N_I} \left(w_4 * \sum_{d=1}^{N_D} k_{v_i,d} + \ w_5 * \left| \sum_{s=1}^{N_s} \sum_{t=1}^{N_t} d_{s,t,v_i} - q_{v_i} \right| \right)$$

The general payoff function of the game, also referred to as the Fitness function, is derived from combining the payoffs of player P0, all players Bm, and all player Vi. The calculation of the fitness function can be determined using the weighted sum method:

The Fitness Function:

$$F = Payoff_{P0} + Payoff_{AllB} + Payoff_{AllV}$$

$$= w_1 * \sqrt{\frac{\sum_{t=1}^{N_T} (\sum_{s=1}^{N_s} \sum_{i=1}^{N_I} d_{s,t,i} - \ \mu)^2}{N_T - 1}}$$

$$+ w_2 * \frac{1}{N_M'} * \sum_{m=1}^{N_M'} \left(\ln \left(\frac{1}{e_{b_m} - 1} * \sum_{i=1}^{e_{b_m}-1} e^{\left| u_i - u_{i+1} - \frac{N_T}{e_{b_m}} \right|} \right) \right)$$

$$+ w_3 * \frac{1}{N_I} * \sum_{i=1}^{N_I} \left(w_4 * \sum_{d=1}^{N_D} k_{v_i,d} + \ w_5 * \left| \sum_{s=1}^{N_s} \sum_{t=1}^{N_t} d_{s,t,v_i} - q_{v_i} \right| \right)$$

H. Nikaido and K. Isoda [33] presented a concept of Nash equilibrium for non-cooperative games in 1955 and demonstrated that this equilibrium point occurs when the overall payoff value for all players is minimized. Therefore, the problem becomes finding the minimum of F, and solving this problem will lead to the Nash Equilibrium point. Before delving into solving F function, it's important to consider and follow a set of non-negotiable constraints that determine the feasibility and validity of any proposed solution. These hard constraints are essential guidelines, ensuring that the generated timetable not only optimizes the designated objective function but also meets critical operational and logistical considerations. Therefore, while minimizing function remains the ultimate goal, any feasible solution must also adhere to specific hard constraints:

With students:

- No student should be required to sit two examinations simultaneously.

$$\sum_{s=1}^{N_S} h_{s,t} * a_{m,s} \le 1 \qquad \forall m \in [1, N_M], t \in [1, N_T]$$

With invigilators:

- The number of supervisors must be equal to the number of rooms needed to organize each subject at each slot.

$$\sum_{i=1}^{N_I} d_{s,t,i} = g_s \ or\ 0 \qquad \forall s \in [1, N_S], t \in [1, N_t]$$

- No invigilator should be required to sit two examinations simultaneously.

$$\sum_{s=1}^{N_S} d_{s,t,i} \le 1 \qquad \forall i \in [1, N_i], t \in [1, N_t]$$

- Examiners only supervise subjects they are capable of supervising.

$$\sum_{i=1}^{N_I} \sum_{t=1}^{N_T} d_{s,t,i} * c_{s,i} = g_s * l_s \qquad \forall s \in [1, N_s]$$

With university:

- The number of rooms used in a slot must not exceed the allowed.

$$\sum_{s=1}^{N_S} \sum_{i=1}^{N_I} d_{s,t,i} \le N_R \qquad \forall t \in [1, N_t]$$

- Each subject must be held once.

$$\sum_{t=1}^{N_T} h_{s,t} = l_s \qquad \forall s \in [1, N_s]$$

- For each exam, the assigned invigilator needs to monitor all consecutive slots in which that exam takes place.

$$\sum_{t=x_s}^{y_s} d_{s,i,t} = l_s \ or \ 0 \qquad \forall s \in [1, N_s], i \in [1, N_i]$$

- For exams with L slot time, it cannot end later than the end of morning or afternoon.

$$\left\lfloor \frac{X}{\beta/2} \right\rfloor = \left\lfloor \frac{Y}{\beta/2} \right\rfloor$$

2.3. Proposed Algorithm

Due to the vast search space of the problem, classified as NP-hard, exact search methodologies are impractical. Therefore, we can only employ approximate search methods. Among these, metaheuristics stand out as a widely adopted and proven effective choice in navigating complex search landscapes. In this context, we have chosen to employ Genetic Algorithm (GA) and Tabu Search (TS) for their distinct advantages.

2.3.1. Genetic Algorithm

Genetic algorithms offer a robust approach to solving complex optimization problems by mimicking the process of natural selection and evolution. By leveraging evolutionary principles, we aim to iteratively refine and improve timetable solutions, ultimately arriving at optimal or near-optimal outcomes.

1. Initialize the population P with α individuals.
2. Repeat α times.
    Create individual p:
        Choose a random timeslot satisfying subject constraints.
        Choose a set of invigilators satisfying hard constraints.
    Add p to P.
3. Calculating fitness for each individual p from P.
4. Repeat β times:
    Sort all individuals p in P by ascending fitness order.

Select φ elite elements (top φ with minimum fitness)

Generate new population P' through crossover and mutation methods.

Add φ elite elements to new population P'.

Choose a parent pair $(p_{father},\ p_{mother})$ randomly from P.

Generate p_{child} using crossover and mutation:

If the random probability for crossover is less than the crossover rate:

Choose a random subject.

Create p_{child} by swapping slot and invigilators of that subject in $p_{father}\ and\ p_{mother}$

If the random probability for mutation is less than the mutation rate:

Choose a random subject from p_{father}

Replace the timeslot of chosen subject in p_{father} with another random timeslot.

For chosen subject:

Choose a random invigilator who supervises the subject.

Replace the chosen invigilator with another randomly chosen invigilator.

Add p_{child} to P'

Calculate fitness for each individual p' from P'.

Update P with P'

5. Choose the best individual p that has minimum fitness from P.

2.3.2. Tabu Search Algorithm

Tabu Search is a metaheuristic algorithm that utilizes memory-based techniques to address complex optimization problems. It uses a Tabu list to avoid revisiting certain moves or solutions, preventing cycling and promoting exploration of new solution spaces. This approach enables Tabu Search to effectively tackle various optimization challenges and provide high-quality solutions.

1. Initialize a solution s.
2. Initialize Tabu list.
3. Repeat β times:

Generate neighborhood for the current solution.

Repeat α times:

Choose two random subjects (s_1, s_2) in current solution.

Swap timeslot of s_1 and s_2

Change slot of s_1 to a new random slot.

Choose another two random subjects (s_1', s_2') in current solution.

Swap invigilators of s_1' and s_2'

Choose a random invigilator that supervise subject s_1'

Replace the chosen invigilator with a new random one.

Add new solution to the neighborhood.

Find the best neighbor in the neighborhood which is not in the tabu list.

Add the best neighbor to the Tabu list.

Check if the size of the tabu list is greater than the maximum size of tabu list.

Remove the first element in Tabu list.

Update the current solution with the best neighbor.

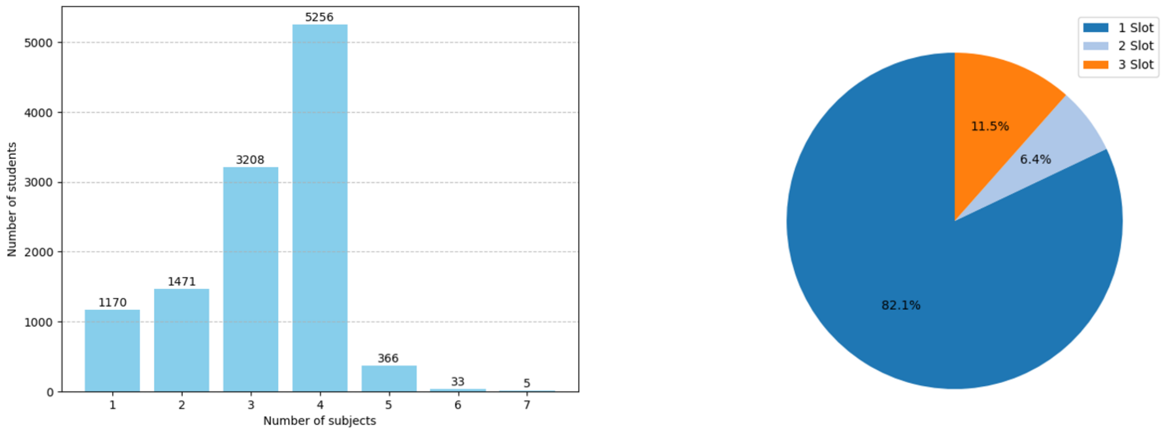
Update the best solution if a current solution is better than the best solution.

4. Return the best solution.

3. EXPERIMENT AND RESULT

3.1. Dataset preparation

In our exam scheduling problem, we need to organize exams for 11,509 students in 156 subjects using 275 instructors. The exam period is 9 days with each day divided into 6 slots of 90 minutes each. We have a maximum of 100 rooms, with a capacity limit of 22 students per room. Students are enrolled in between 1 to 7 subjects (figure 2a), and the duration of exams ranges from 1 to 3 time slots (figure 2b). These factors make timetable construction complex but important for managing individual student schedules efficiently while optimizing resource usage and minimizing conflicts.



(a) (b)

Figure 2: *(a) Distribution of Students by Number of Exams Taken. (b) Proportion of Exam Timeslots*

With our dataset information as above, after thorough testing, the weights for the fitness function were determined as follows: $w_1 = \frac{1}{20}, w_2 = \frac{6}{20}, w_3 = \frac{13}{20}$ to achieve a balance between the interests of the academic department, students, and invigilators. Furthermore, in order to compute the number of days invigilators need to be present at the university and guarantee alignment with the department's criteria for total slots equally, we selected $w_4 = \frac{1}{2}, w_5 = \frac{1}{2}$.

3.2. Algorithm Configuration

The genetic algorithm parameters were configured by running a total of 50,000 generations with a population size of 200 individuals. The crossover rate was set at 0.8, and the mutation rate was determined to be 0.4. In order to retain superior solutions across generations, 10% of the population (which represents the elite individuals) was preserved for future generations.

Simultaneously, the Tabu Search component executed over 50,000 iterations. The initial neighbor size for TS was established at 100 solutions with an incremental increase of 50 solutions for every 5,000 iterations while maintaining the length of the tabu list 500 solutions.

3.3. Experiment Setup

The experiments were conducted on a high-performance computer equipped with the following specifications:

- **CPU**: Intel Core i9 processor, running at a base frequency of 3.6 GHz.
- **Threads**: 16 threads, enabling parallel execution of processes.
- **RAM**: 64 GB of DDR4 memory, facilitating efficient handling of large datasets and complex calculations.

3.4. Result And Analysis

3.4.1. Comparison between Genetic Algorithm approach and Tabu Search Approach

Both the GA and TS were executed over a duration of 12 hours, facilitating a comprehensive comparison of their performance in optimization tasks. The chart below displays the optimal fitness value in each iteration of two algorithms:

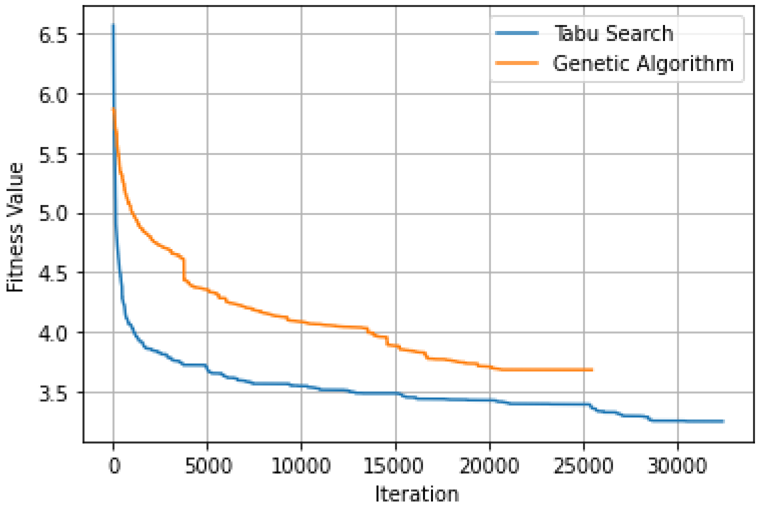


Figure 3: *The optimal fitness value for each iteration of two algorithms.*

After running for 12 hours, the Genetic Algorithm completed approximately 25,000 generations, while the Tabu Search reached almost 33,000 iterations. Figure 3 clearly shows that Tabu Search outperforms the Genetic Algorithm by finding a better solution with a lower fitness value. This conclusion is further supported by the hypervolume indicator [34] in Figure 4 which shows that Tabu Search achieved a value of 0.265 compared to GA's value of 0.199.

These results confirm not only TS's ability to find solutions with lower fitness but also its capacity to cover a broader and potentially more diverse set of feasible solutions.

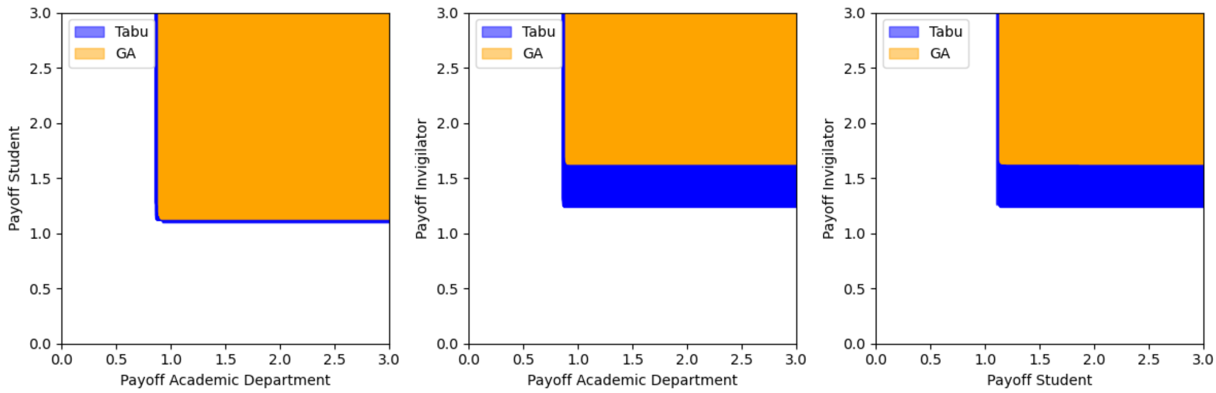


Figure 4: *Hypervolume comparison of GA (orange) and Tabu Search (blue)*

3.4.2. Best Result Evaluation

We use the best solution of Tabu Search to evaluate. We employed a metric for gauging student satisfaction based on the criteria of minimal exam clashes, suitable spacing between exams. The gap between two consecutive examinations for each student, which is considered to be optimal distance, depends on the total number of subjects of each student. This optimal gap specific to each student can be calculated by dividing the total slot by the number of their exams. For each student who has more than one exam, we divided the time slot gap between two consecutive examinations by its optimal distance and then average all the gaps of this student and multiply with 100. If the satisfaction rate calculated is smaller than 0, we will set this value equal to 0. The chart below shows the average satisfaction of students according to the number of exams:

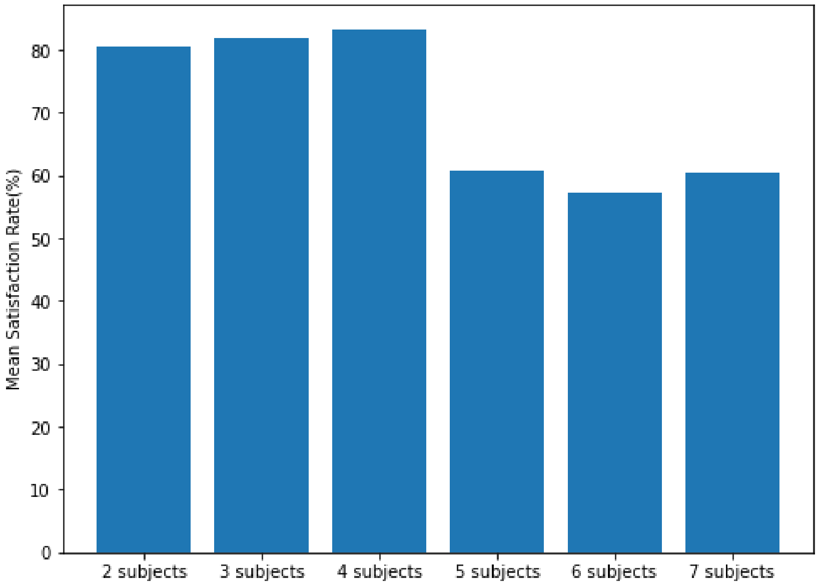


Figure 5: *The average satisfaction rate of students with different number of subjects.*

Figure 5 shows that students who have taken 2, 3, or 4 exams tend to report higher average satisfaction compared to those with 5, 6, or 7 exams. This could be due to the significantly larger number of students taking fewer exams (3 and 4) than those taking more (5, 6, or 7).

When considering invigilator benefits, we focus on two main criteria: minimizing required attendance days at the university and optimizing the deviation between assigned and required slots. To assess satisfaction for the first criterion, we determine each invigilator's number of unnecessary attendance days by subtracting their total assigned days from the total examination days. We then calculate the expected minimum attendance days for each invigilator based on input data by dividing their required slots by the number of slots per day. Finally, we compute an expected number of free days for each invigilator by comparing scheduled free days to our calculated expectation and multiplying by 100. For the second criterion, we use a different calculation. We find the absolute difference between the assigned and required slots, then divide this value by the required slots, and multiply by 100. By averaging these rates from both parts, we obtain an overall average satisfaction for an invigilator. If the satisfaction rate calculated is smaller than 0, we will set this value equal to 0.

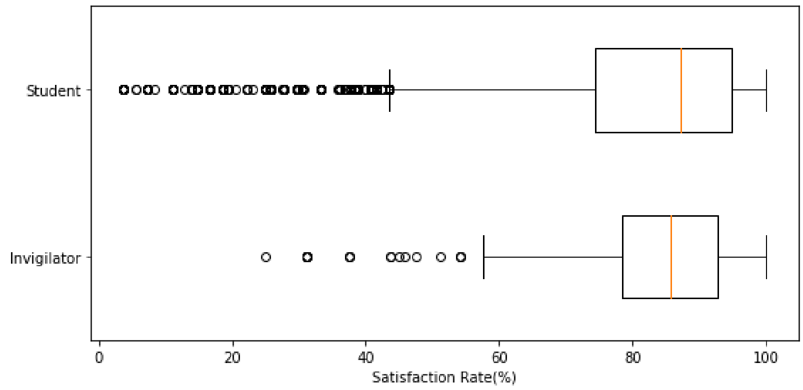


Figure 6: *The distribution of student satisfaction and invigilator satisfaction.*

Our academic department goals include evenly allocating the necessary rooms for each exam session, considering that some subjects may require a larger number of rooms compared to others. For example, certain subjects may need more than 70 out of 100 rooms with 22 students in each room. As a result, some sessions will require a higher allocation of rooms than others (Figure 7). To assess the satisfaction rate of the academic department, we first calculated the average number of rooms required per exam session if all rooms were evenly assigned. Then, we determine the maximum deviation in room assignments for each session by comparing it to this average. For every session, we calculate the absolute deviation between the assigned number of rooms and mean room count per session. This deviation is then normalized by dividing it by the maximum deviation and multiplying by 100.

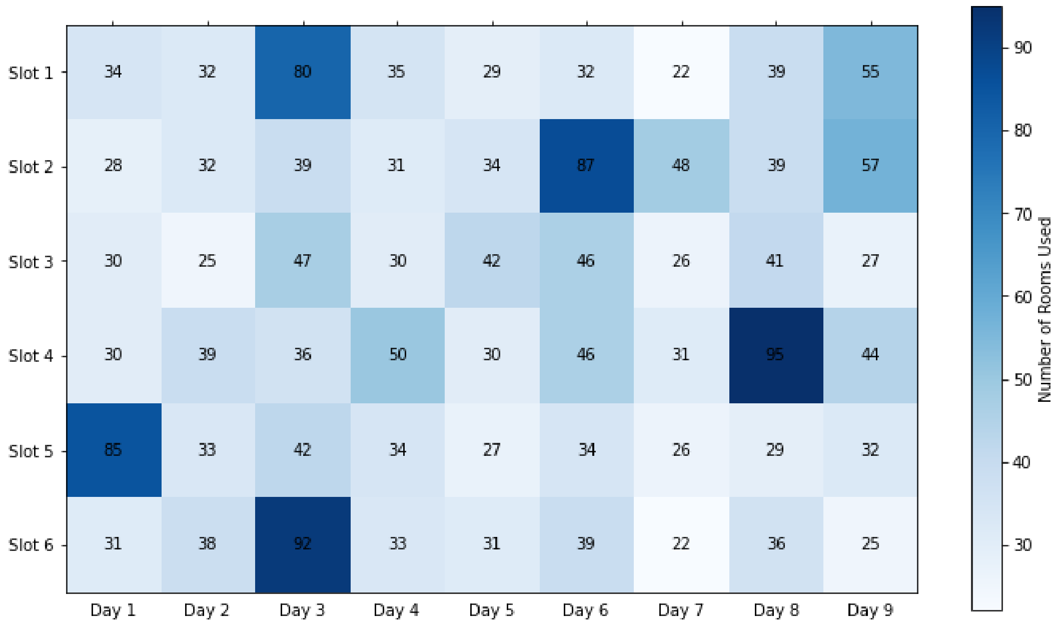


Figure 7*: Number of rooms in each timeslot.*

Throughout the experiment, we concluded that no single party could achieve an absolute or near-absolute satisfaction rate (as shown in Figure 8), and the average satisfaction rate of all stakeholders is little more than 80%.

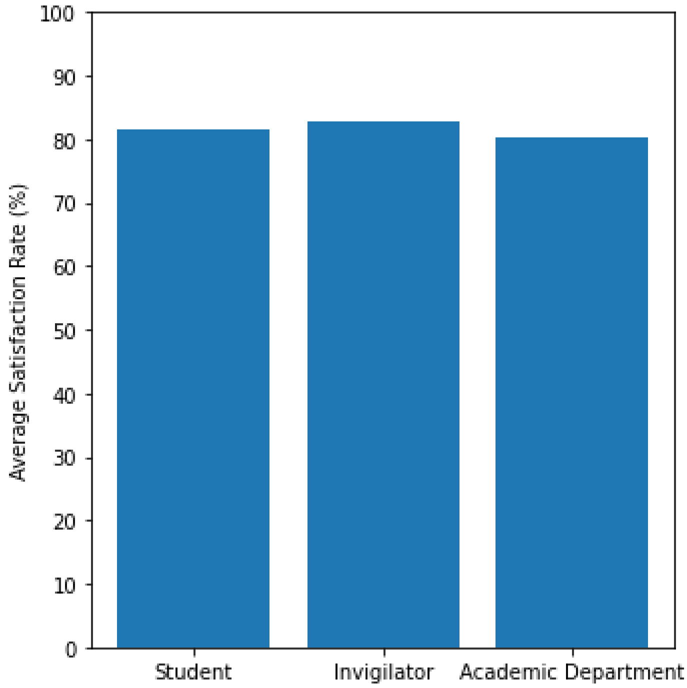


Figure 8: *The average satisfaction rate of all stakeholders.*

Our experiments demonstrate multiple advantages of the proposed approach. First, using Nash equilibrium analysis allows us to create timetables that meet strict requirements while ensuring fair benefits for all stakeholders. Additionally, genetic algorithms are shown to be a highly effective strategy for accurately and comprehensively modeling conflicts of interest compared to tabu search. Furthermore, this method can be applied to other scheduling problems or any situation where maintaining a balance between conflicting interests among multiple parties is crucial.

4. CONCLUSION AND FUTURE WORK

In summary, this study offers a comprehensive strategy for addressing the scheduling challenges of exams by integrating game theory principles with a Genetic Algorithm and Tabu Search framework. The conducted experimentation and analysis emphasize the importance of balancing the needs of academic departments, students, and invigilators in creating timetables that satisfy all stakeholders. Through careful parameter selection and adjustment, our method effectively optimizes objectives while adhering to strict constraints. The results illustrate the practicality and success of our proposed approach through high-quality timetables produced across various experimental scenarios. Additionally, the experiment highlights considering computational resources required for implementing this solution approach as demonstrated by hardware specifications used in this study. This research provides valuable insights that can guide future development of efficient timetabling systems prioritizing stakeholder interests and streamlining scheduling processes in educational institutions. Future research could explore a hybrid model that combine the strengths of GA and TS and incorporating a wider range of constraints and objectives, particularly by focusing on invigilator preferences alongside considerations for students and academic departments for real-world implementations to validate scalability and applicability of the proposed approach.

References

[1] M. Hosny, “Metaheuristic approaches for solving university timetabling problems: A review and case studies from middle eastern universities,” in *Smart Innovation, Systems and Technologies*, 2019. doi: 10.1007/978-3-030-03577-8_2.

[2] R. Perzina, “Solving the university timetabling problem with optimized enrollment of students by a self-adaptive genetic algorithm,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2006. doi: 10.1007/978-3-540-77345-0_16.

[3] L. P. Chi, H. Q. Thang, V. T. Huong Giang, N. T. Son, and B. N. Anh, “Teaching Assignment Based on NASH Equilibrium and Genetic Algorithm,” in *2023 IEEE Symposium on Industrial Electronics and Applications, ISIEA 2023*, 2023. doi: 10.1109/ISIEA58478.2023.10212338.

[4] B. A. Aldeeb *et al.*, “Hybrid intelligent water Drops algorithm for examination timetabling problem,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, 2022, doi: 10.1016/j.jksuci.2021.06.016.

[5] M. Carlsson, S. Ceschia, L. Di Gaspero, R. Ø. Mikkelsen, A. Schaerf, and T. J. R. Stidsen, “Exact and metaheuristic methods for a real-world examination timetabling problem,” *Journal of Scheduling*, vol. 26, no. 4, 2023, doi: 10.1007/s10951-023-00778-6.

[6] R. W. Rosenthal, “A class of games possessing pure-strategy Nash equilibria,” *International Journal of Game Theory*, vol. 2, no. 1, 1973, doi: 10.1007/BF01737559.

[7] M. A. Mohammed *et al.*, “A review of genetic algorithm application in examination timetabling problem,” *Journal of Engineering and Applied Sciences*, vol. 12, no. 20, 2017.

[8] N. Leite, F. Melício, and A. C. Rosa, “A fast simulated annealing algorithm for the examination timetabling problem,” *Expert Syst Appl*, vol. 122, 2019, doi: 10.1016/j.eswa.2018.12.048.

[9] F. H. Awad, A. Al-Kubaisi, and M. Mahmood, “Large-scale timetabling problems with adaptive tabu search,” *Journal of Intelligent Systems*, vol. 31, no. 1, 2022, doi: 10.1515/jisys-2022-0003.

[10] H. D. Lawal, I. A. Adeyanju, E. O. Omidiora, O. T. Arulogun, and O. I. Omotosho, “University Examination Timetabling Using Tabu Search,” *Int J Sci Eng Res*, vol. 5, no. 10, 2014.

[11] M. W. Carter, G. Laporte, and S. Y. Lee, “Examination timetabling: Algorithmic strategies and applications,” *Journal of the Operational Research Society*, vol. 47, no. 3, 1996, doi: 10.1057/jors.1996.37.

[12] L. T. G. Merlot, N. Boland, B. D. Hughes, and P. J. Stuckey, “A hybrid algorithm for the examination timetabling problem,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2740, 2003, doi: 10.1007/978-3-540-45157-0_14.

[13] M. Ayob, A. M. Ab Malik, S. Abdullah, A. R. Hamdan, G. Kendall, and R. Qu, “Solving a practical examination timetabling problem: A case study,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007. doi: 10.1007/978-3-540-74484-9_53.

[14] M. E. Keskin, A. Döyen, H. Akyer, and M. G. Güler, "Examination timetabling problem with scarce resources: A case study," *European Journal of Industrial Engineering*, vol. 12, no. 6, 2018, doi: 10.1504/EJIE.2018.096394.

[15] A. R. Komijan and M. N. Koupaei, "A new binary model for university examination timetabling: a case study," *Journal of Industrial Engineering International*, vol. 8, no. 1, 2012, doi: 10.1186/2251-712X-8-28.

[16] M. N. Mohmad Kahar and G. Kendall, "Universiti Malaysia Pahang examination timetabling problem: Scheduling invigilators," *Journal of the Operational Research Society*, vol. 65, no. 2, 2014, doi: 10.1057/jors.2012.6.

[17] M. N. M. Kahar and G. Kendall, "The examination timetabling problem at Universiti Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution," *Eur J Oper Res*, vol. 207, no. 2, 2010, doi: 10.1016/j.ejor.2010.04.011.

[18] S. Abdul-Rahman, N. S. Sobri, M. F. Omar, A. M. Benjamin, and R. Ramli, "Graph coloring heuristics for solving examination timetabling problem at Universiti Utara Malaysia," in *AIP Conference Proceedings*, 2014. doi: 10.1063/1.4903627.

[19] A. F. Khair, M. Makhtar, M. Mazlan, M. A. Mohamed, and M. N. A. Rahman, "Solving examination timetabling problem in UniSZA using ant colony optimization," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 2, 2018, doi: 10.14419/ijet.v7i2.15.11369.

[20] S. A. Mirhassani and F. Habibi, "Solution approaches to the course timetabling problem," *Artif Intell Rev*, vol. 39, no. 2, 2013, doi: 10.1007/s10462-011-9262-6.

[21] A. Bashab *et al.*, "A systematic mapping study on solving university timetabling problems using meta-heuristic algorithms," *Neural Computing and Applications*, vol. 32, no. 23. 2020. doi: 10.1007/s00521-020-05110-3.

[22] S. N. Tung, J. B. Jaafar, I. A. Aziz, H. G. Nguyen, and A. N. Bui, "Genetic Algorithm for Solving Multi-Objective Optimization in Examination Timetabling Problem," *International Journal of Emerging Technologies in Learning*, vol. 16, no. 11, 2021, doi: 10.3991/ijet.v16i11.21017.

[23] S. Tung Ngo, J. Jafreezal, G. Hoang Nguyen, and A. Ngoc Bui, "A Genetic Algorithm for Multi-Objective Optimization in Complex Course Timetabling," in *ACM International Conference Proceeding Series*, 2021. doi: 10.1145/3457784.3457821.

[24] T. S. Ngo, J. Jaafar, I. A. Aziz, M. U. Aftab, H. G. Nguyen, and N. A. Bui, "Metaheuristic Algorithms Based on Compromise Programming for the Multi-Objective Urban Shipment Problem," *Entropy*, vol. 24, no. 3, 2022, doi: 10.3390/e24030388.

[25] S. T. Ngo, J. Jaafar, A. A. Izzatdin, G. T. Tong, and A. N. Bui, "Some metaheuristic algorithms for solving multiple cross-functional team selection problems," *PeerJ Comput Sci*, vol. 8, 2022, doi: 10.7717/PEERJ-CS.1063.

[26] P. Yan, H. Miao, A. Che, and K. Yu, "Nash-equilibrium algorithm and incentive protocol for a decentralized decision and scheduling problem in sustainable electroplating plants," *Comput Oper Res*, vol. 128, 2021, doi: 10.1016/j.cor.2020.105130.

[27] A. Nezarat and G. H. Dastghaibifard, "Efficient nash equilibrium resource allocation based on game theory mechanism in cloud computing by using auction," *PLoS One*, vol. 10, no. 10, 2015, doi: 10.1371/journal.pone.0138424.

[28] M. O. Spata, “A Nash-equilibrium based algorithm for scheduling jobs on a grid cluster,” in *Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE*, 2007. doi: 10.1109/WETICE.2007.4407164.

[29] A. De Paola, D. Angeli, and G. Strbac, “On Distributed Scheduling of Flexible Demand and Nash Equilibria in the Electricity Market,” *Dyn Games Appl*, vol. 8, no. 4, pp. 761–798, Dec. 2018, doi: 10.1007/s13235-017-0237-3.

[30] D. Salem, “University Examination Timetable Scheduling Using Constructive Heuristic Compared to Genetic Algorithm,” *Fayoum University Journal of Engineering*, vol. 6, no. 1, 2023, doi: 10.21608/fuje.2022.157195.1018.

[31] M. F. Mohammed, R. A. Abobaker, A. M. Aljalil, and A. M. Maatuk, “Examinations Timetabling System Based on A Genetic Algorithm,” in *Proceedings - 2022 International Conference on Engineering and MIS, ICEMIS 2022*, 2022. doi: 10.1109/ICEMIS56295.2022.9914188.

[32] G. Kendall and N. M. Hussin, “A tabu search hyper-heuristic approach to the examination timetabling problem at the MARA university of technology,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2005. doi: 10.1007/11593577_16.

[33] H. Nlkaido and K. Isoda, “NOTE ON NONCOOPERATIVE CONVEX GAMES.”

[34] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, and L. Salomon, “Performance indicators in multiobjective optimization,” *European Journal of Operational Research*, vol. 292, no. 2. 2021. doi: 10.1016/j.ejor.2020.11.016.