Teaching Assignment Based on NASH Equilibrium and Genetic Algorithm

Le Phuong Chi
[1]ICT Department, FPT University, Hanoi, Vietnam
[2]School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi, Vietnam
chilp2@fe.edu.vn

Huynh Quyet Thang
School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi, Vietnam
thanghq@soict.hust.edu.vn

Vu Thi Huong Giang
School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi, Vietnam
giangvth@soict.hust.edu.vn

Ngo Tung Son
ICT Department, FPT University, Hanoi, Vietnam
sonnt69@fe.edu.vn

Bui Ngoc Anh
ICT Department, FPT University, Hanoi, Vietnam
anhbn5@fe.edu.vn

***Abstract*—Teaching assignment refers to the process of allocating classes to instructors for a given academic semester. Its outcome is a schedule that must adhere to various constraints, such as ensuring that all classes have assigned instructors, avoiding schedule conflicts among instructors, and adhering to predetermined limits on the number of classes assigned to each instructor. It is a complex task due to the involvement of multiple stakeholders such as instructors, and academic departments, each with their own interests. In many cases, these interests may conflict with each other. In this study, we propose using the NASH equilibrium, a branch of game theory, to analyze conflicts of interest among stakeholders and modeling the teaching assignment as a multi-objective optimization problem. Furthermore, we develop a genetic algorithm to solve this problem. Our algorithm is experimented with a real-world dataset consisting of 153 classes and 25 instructors. The experimental results demonstrate the feasibility and effectiveness of our proposition.**

***Keywords*—Game theory, NASH Equilibrium, Teaching assignment, University timetabling, Meta heuristic, Genetic algorithm.**

I. Introduction

A. Background

The scheduling of university education, also known as the university timetabling problem (UTP), represents a critical and formidable challenge in university education management [1]. Recently, education has been focused on and prioritized the rights of learners, so the scheduling process at universities is usually carried out in three small steps. Firstly, based on the learning needs of students and the capacity of the university's facilities, the academic department will proceed with class allocation for the students. A class is understood as a specific set of students studying a specific subject in a classroom during a specific time slot. This class allocation is called the Enrollment Timetabling Problem [2]. Secondly, the allocated classes need to be assigned to the corresponding instructors. This is the task of the Teaching Assignment Problem. Finally, at the end of the semester, the academic department will schedule exams for the students. This problem is called the Examination Timetabling Problem [3].

In this study, we focus on the second problem of the three aforementioned problems, namely the **Teaching Assignment Problem (TAP)**.

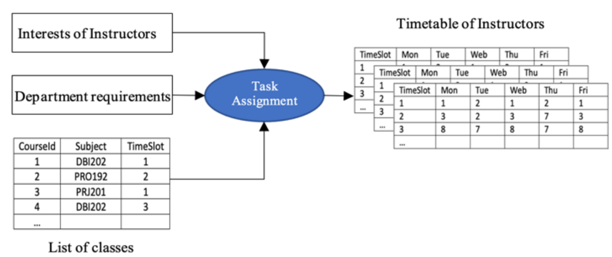


Fig. 1. University teaching assignment problem.

As described in Fig. 1, the input of the problem consists of a list of classes that need to be taught, a list of instructors being managed by the academic department, as well as all the general requirements of the teaching process that the department has to ensure. The output is a schedule indicating the list of classes that each instructor is assigned to teach.

Similar to the two remaining problems, TAP is difficult to solve because it involves multiple stakeholders, each of whom seeks different benefits. Firstly, the participation of instructors is crucial. Each instructor has their own preferences and needs regarding their subject, the time they spend teaching, and the number of classes assigned to them. They expect to receive a teaching schedule that is suitable for their preferences and needs. Meanwhile, from a holistic perspective, the academic department aims to provide a teaching schedule that ensures all classes are assigned to instructors and that there are no classes without an instructor, while also ensuring the overall quality of education in the university. The quality of education can be seen as a benefit that the university, represented by the academic department, desires. The quality of education is obtained through evaluating the quality of teaching in each class. Here, we differentiate between two criteria: an instructor's preference for a certain subject and their quality of teaching in that subject. These are two independent criteria. The enjoyment an instructor derives from teaching a subject does not guarantee that the teaching quality for that subject will be good. Similarly, an instructor who teaches a subject well is not obligated to like that subject. The independence of these two criteria creates a conflict of interests between instructors and the academic department, as each party seeks to achieve their own benefits. Conflicts between instructors and the academic department also arise when an instructor wants to teach a class during their preferred time slot, but their teaching quality for that particular class is not high, or vice versa. Each instructor also has a need for the number of classes they want to teach, but this need may contradict the desire for all classes to be assigned to instructors by the academic department. Conflicts of interest also arise between instructors themselves when they share a liking for a particular subject or time slot, resulting in many instructors wanting to teach some classes in a particular subject or time slot. Due to the complex conflicts of interest among the parties involved, an effective tool is needed to identify and prioritize these benefits.

The objective of our research is to address the TAP problem in a manner that not only provides an acceptable timetable by satisfying the constraints, but also maximizes the interests of all participating parties. Our research employs NASH equilibrium, a branch of game theory, to analyze the benefits of each party and model the conflicting interests, thereby transforming the stated problem into a multi-objective optimization problem. We then use the weighted sum method to transform the multi-objective optimization problem into a single Fitness function. Additionally, we design a genetic algorithm to search for the optimal solution for this Fitness function.

In summary, our study constructs a solver based on NASH and metaheuristic to address the TAP problem, including a complete process from conflict analysis (Step 1), using NASH equilibrium to model conflicts into a multi-objective optimization problem (Step 2), transformation a multi-objective optimization to a single objective optimization (SOP) using weighted sum approach (Step 3), building an evolutionary algorithm to solve the optimal model (Step 4), and finally analyzing the obtained results (Step 5). The proposed process is explained in Fig 2 below.

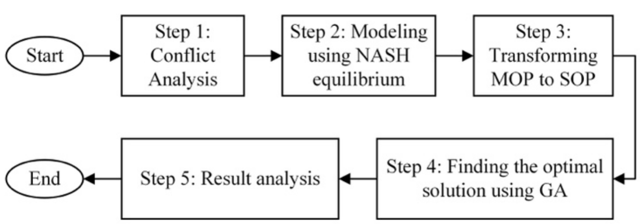


Fig. 2. The proposed process for TAP.

B. Related research

We conduct a comprehensive examination of existing research from three angles: an analysis of studies on the TAP problem, an exploration of research on multi-objective optimization problems with an emphasis on the utilization of NASH equilibrium, and an investigation into the algorithms employed to address multi-objective optimization problems.

First of all, TAP is a common problem in university timetabling, and there have been several studies on the topic. H. Algethami et al. used a mixed-integer multi-objective programming model with IBM ILOG CPLEX to optimize scheduling for less than 60 classes [5]. Thatchai Thepphakorn and Pupong Pongcharoen used self-adaptive Cuckoo search to optimize course scheduling costs based on resource compatibility [6]. Arratia-Martinez et al. used IBM ILOG CPLEX to solve a course-instructor allocation problem with three objectives: minimizing unscheduled courses, minimizing instructor overload, and minimizing new instructor hiring, with a dataset of 28 classes [7]. Ngo Tung Son et al. developed a genetic algorithm approach to optimize scheduling for instructors at FPT University, considering instructor preferences and availability but not accounting for departmental rights [8].

Secondly, TAP can be viewed as a multi-objective optimization problem. In general, for multi-objective optimization problems, there are four main approaches to solving them: interactive [9], preferences [10], no preferences, and posteriority[11] [12] and [13]. There are also some authors who use hybrid methods for multi-objective optimization problems based on compromise programming, such as the author Ngo in solving Coursera timetabling [14] and Examination timetabling problems [3]. There are many other studies that also use this approach in various fields, including logistics [15], scheduling [16], selection [17], and more.

Current approaches to multi-objective optimization problems often face difficulties in achieving fairness among the objective functions. This is because the objective functions often have different characteristics and cannot be accurately evaluated for priority. Additionally, prioritizing the objectives also depends on the perspectives and influences of the relevant stakeholders. Therefore, creating fairness among the objective functions is a major challenge in multi-objective optimization problems.

The concept of NASH equilibrium is a term in game theory, used to describe a state in which no player has an incentive to change their strategy while the other players remain unchanged, in order to achieve a greater benefit. In situations with conflicting interests, NASH equilibrium is employed to identify a state of balanced benefits among the parties. Therefore, NASH equilibrium can be considered as an impartial mediator to evaluate the balanced benefits of the parties, ensuring that no party has a complete advantage over the others. This promotes fairness in the final decision-making process and aligns with objective evaluation criteria.

Researchers have applied the NASH equilibrium model to various problems, including competition for medical supplies during the COVID-19 pandemic [18], interactions between users and small electricity providers in the Smart grid system [19], scheduling projects involving multiple contractors [20][21], scheduling the production process of an electroplating factory [22], and conflicts of interest in information technology investment project management [23][24]. NASH equilibrium has proven to be an effective tool for analyzing conflicts and proposing optimal solutions.

So far, no study has applied NASH equilibrium to the university class scheduling problem. In this study, we aim to leverage the advantages of NASH equilibrium to analyze the conflicts of interest between the department and the faculty, as well as between instructors, in order to solve the teaching assignment problem, a small problem of the larger university class scheduling problem.

Finally, contributing to the overall results of the problem, besides proposing an optimal model, the application of different algorithms to solve the model has a significant impact on the final outcome. Chen M and colleagues in their study divided the methods used to solve the UTTP problem into six categories, including: operational research, single-solution-based meta-heuristic algorithms (Tabu search, Variable Neighborhood Search, Simulated Annealing), population-based meta-heuristic algorithms (genetic algorithm GA, ant colony optimization ACO, and swarm optimization in general), hyper-heuristic, multi-objective optimization, and hybrid approaches. Chen M and colleagues' study demonstrated that meta-heuristic approaches are the most common [25]. We chose to design a genetic algorithm to solve the optimal model found using NASH equilibrium.

C. Contributions

Our work presents an approach to building a system to support instructor assignment, which is a task in the business process of the academic department in universities and also an approach to fixed task assignment problems in general.

This research also extends further to provide an approach to the problem of scheduling training in universities. Specifically, we use NASH equilibrium to model the problem and then apply evolutionary algorithms to solve the model and find the optimal solution.

Our study contributes to demonstrating the effectiveness of NASH equilibrium in analyzing conflict issues in general and scheduling problems in universities in particular.

II. METHODOLOGY

A. Problem formulation

We describe the scheduling problem for instructors. The inputs to the problem include the following information:

- N_S , the number of subjects taught in the university.
- N_C, the number of classes that need to be assigned to instructors.
- N_T, the number of time slots.
- N_G, the number of instructors.
- Matrix $S = \{s_{x,y}, 1 \le x \le N_C, 1 \le y \le N_S\}$, in which $s_{x,y} = 1$ if class x studies subject y, $s_{x,y} = 0$ if class x does not study subject y.
- Matrix $T = \{t_{x,y}, 1 \le x \le N_C, 1 \le y \le N_T\}$, in which $t_{x,y} = 1$ if class x is taught during timeslot, otherwise $t_{x,y} = 0$.
- Matrix $A = \{a_{g,s} \in [0; 10], 1 \le g \le N_G, 1 \le s \le N_S\}$, in which $a_{g,s}$ indicates the level of preference of instructor g for subject s.
- $E = A * S^{-1} = \{e_{g,c} \in [0; 10], 1 \le g \le N_G, 1 \le c \le N_c\}$ given the level of preference in terms of subjects of the instructor g with class c.
- Matrix $B = \{b_{g,s} \in [0; 10], 1 \le g \le N_G, 1 \le s \le N_S\}$, in which $b_{g,s}$ indicates the teaching quality of instructor g for subject s.
- $F = B * S^{-1} = \{f_{g,c} \in [0; 10], 1 \le g \le N_G, 1 \le c \le N_C\}$, in which $f_{g,c}$ indicates the teaching quality of instructor g for class c.
- Matrix $C = \{c_{g,t} \in [0; 10], 1 \le g \le N_G, 1 \le s \le N_T\}$, in which $c_{g,t}$ indicates the level of preference of instructor g for time slot t.
- $H = C * T^{-1} = \{h_{g,c} \in [0; 10], 1 \le g \le N_G, 1 \le c \le N_C\}$, in which $h_{g,c}$ indicates the level of preference in term of time slots of instructor g for class c.
- Vector $K = \{k_g \in [0, 10], 1 \le g \le N_G\}$, in which k_g is the number of classes that instructor g wants to teach.
- Vector MinC = $\{minCourse_g \in [0, 10], 1 \le g \le N_G\}$, in which $minCourse_g$ the minimum number of classes that instructor g needs to be assigned.
- Vector MaxC = $\{maxCourse_g \in [0, 10], 1 \le g \le N_G\}$, in which $maxCourse_g$ the maximum number of classes that instructor g can be assigned.

A teaching schedule is a matrix $D = \{d_{c,g} \in [0; 1],\ 1 \le c \le N_C,\ 1 \le g \le N_G\}$, where $d_{c,g} = 1$ if instructor g is assigned to teach class c, $d_{c,g} = 0$ if instructor g is not assigned to teach class c. A selected teaching schedule must ensure the rights of all involved parties.

B. NASH equilibrium approach

According to the analysis in section I, the interests of the parties (the academic department and the instructors) conflict with each other. A solution that benefits one party will bring losses to the other. Therefore, it is necessary to find a scheduling solution that balances the interests of the participating parties, including the department and each instructor.

We have found that the problem here is suitable for using the Unified Game-based model proposed by author Trinh Bao Ngoc and colleagues [13]. We describe the teaching allocation problem as a game with $(N_G + 1)$ players, including N_G players, who are the N_G instructors, and one special player. The special player is not a specific person but a hypothetical player representing the common interests of the academic department. Each player will have different strategies, and for each strategy, they receive a corresponding reward/punishment [26]. We call the reward/ punishment that a player receives for each strategy the player's payoff function. This function represents the benefits that a player receives corresponding to a scheduling solution.

Special player P_0 represents the collective interest of the academic department.

The set of strategy of P_0: $G_0 = \{g_{01}, g_{02}, \dots, g_{0N_c}\}$, where g_{0x} $(1 \le x \le N_c)$ represents a class that needs to be assigned a instructor, which is an information structure consisting of:

- Vector indicating the subject that this class belongs to: $S_x = \{s_{x,y} \in [0,1],\ 1 \le y \le N_S\}$, $s_{x,y} = 1$ if class x belongs to subject y, otherwise $s_{x,y} = 0$ if class x does not belong to subject y.
- Vector indicating the time slot that this class belongs to: $T_x = \{t_{x,y} \in [0,1], 1 \le y \le N_T\}$, $t_{x,y} = 1$ if class x belongs to time slot y, otherwise $t_{x,y} = 0$ if class x does not belong to time slot y.
- Vector indicating the instructor assigned to teach this class: $D_x = \{d_{x,g} \in [0;1], 1 \le g \le N_g\}$, $d_{x,g} = 1$ if class x is assigned to instructor g, otherwise $d_{x,y} = 0$ if class x is not assigned to instructor g.

Payoff function of player P_0

The benefit of player P_0 is the teaching quality of the entire system. This quality is represented by the following payoff function $Payoff_{P_0}$.

Let $V = D * F = \{v_{i,k} \in [0,10], 1 \le i, k \le N_C\}$, where:

$$v_{i,k} = \begin{cases} = 0, \forall\ i \ne k \\ \ge 0, \forall\ i = k \end{cases}$$

$v_{i,i}$ $(\forall\ 1 \le i \le N_C)$ is the teaching quality of class c in case of using schedule D.

We derive the payoff function of player P_0:

$$Payoff_{P_0} = \sum_{i=1}^{N_C} v_{i,i} = \sum_{i=1}^{N_C} \sum_{j=1}^{N_G} (d_{i,j} * f_{j,i}) \tag{1}$$

Player Pi $(1 \le i \le N_G)$, represents the i-th instructor.

Each player P_i has the set of strategy: $G_i = \{g_{i1}, g_{i2}, \dots, g_{ij}\}$ where g_{ij} $(1 \le i \le N_G,\ 1 \le j \le N_c)$ is an information structure consisting of:

- The level of interest of instructor i in terms of subject for class j: $e_{i,j}$
- The level of interest of instructor i in terms of time slot for class j: $h_{i,j}$
- The teaching quality of instructor i for class j: $f_{i,j}$

Payoff function of player P_i

Player P_i desires to teach classes that have subjects they interest, during their preferred time slots, with the number of classes as close as possible to the desired amount.

- Level of preference in term of subject for the classes in the assigned schedule of player P_i:

$$ls_i = \sum_{j=1}^{N_C} e_{i,j} * d_{j,i} \tag{2}$$

- Level of preference in term of time slot for the classes in the assigned schedule of player P_i:

$$lt_i = \sum_{j=1}^{N_C} h_{i,j} * d_{j,i} \tag{3}$$

- Level of teaching quantity for the classes in the assigned schedule of player P_i:

$$la_i = 10 - \left| k_i - \sum_{j=1}^{N_C} d_{j,i} \right| \tag{4}$$

We deduce that the payoff function of P_i:

$$Payoff_{P_i} = w_1 * ls_i + w_2 * lt_i + w_3 * la_i \tag{5}$$

where w_1, w_2, w_3 are coefficients determined by experts.

Each player seeks to protect their own interests, namely maximizing the value of their payoff functions. Therefore, in order to ensure the interests of all parties, we need to find the NASH equilibrium point, which is the point (in this case, a set of payoff function values and the corresponding strategies that generate those values) where no player has an advantage over any other player. The strategy profile that constitutes the NASH equilibrium point is the one we need to find.

In 1955, H. Nikaido and K. Isoda generalized the NASH equilibrium problem in non-cooperative games and showed that the NASH equilibrium point is the point where the overall payoff value of all players is maximized [27]. We can calculate this overall payoff value, called the Fitness function, using the weighted sum method as follows:

Payoff of all players P_i (∀i)

$$\begin{gathered} Payoff_{AllPi} = \sum_{i=1}^{N_G} Payoff_{P_i} \\ = w_1 \sum_{i=1}^{N_G} \sum_{j=1}^{N_C} e_{i,j} * d_{j,i} + w_2 \sum_{i=1}^{N_G} \sum_{j=1}^{N_C} h_{i,j} * d_{j,i} + \\ w_3 \sum_{i=1}^{N_G} (10 - |k_i - \sum_{j=1}^{N_C} d_{j,i}|) \end{gathered} \tag{6}$$

The overall payoff function of the game, also known as Fitness function, is the aggregation of the payoffs of the special player $\boldsymbol{P_0}$ and all other players $\boldsymbol{P_i}$.

The Fitness function

$$\begin{gathered} F = w_4 * Payoff_{P_0} + w_5 * Payoff_{AllPi} \\ = w_4 * \sum_{i=1}^{N_C} \sum_{j=1}^{N_G} (d_{i,j} * f_{j,i}) + w_5 * (w_1 * \\ \sum_{i=1}^{N_G} \sum_{j=1}^{N_C} e_{i,j} * d_{j,i} + w_2 * \sum_{i=1}^{N_G} \sum_{j=1}^{N_C} h_{i,j} * d_{j,i}\ w_3 * \\ \sum_{i=1}^{N_G} (10 - |k_i - \sum_{j=1}^{N_C} d_{j,i}|)) \end{gathered} \tag{7}$$

We can see that if $F*$ is the maximum value of the F function in (7), then any change in the strategy choice of a player P_i $(0 \le i \le N_G)$ leads to a value $F' <= F*$. Meanwhile, the other players P_j $(j \ne i)$ do not change their strategies, which means that the reward/penalty they receive remains the same. From this, we can conclude that $Payoff'_{P_i} \le Payoff^*_{P_i}$. The corresponding equilibrium point for F^* is the NASH equilibrium point of the players in this game.

Thus, the solution that maximizes the F function is also the one that yields the NASH equilibrium point. To solve the problem, one needs to find a matrix D that maximizes the Fitness function, which is a single-objective optimization problem. In Section 2.3 below, we will use the GA to solve this single-objective optimization problem. Before that, we identify the constraints that any solution to the decomposition problem must satisfy, including:

- All classes must be assigned an instructor: $\sum_{x=1}^{N_C}\sum_{g=1}^{N_G} d_{x,g} = N_C$
- Each class can only be assigned to one instructor: $\sum_{g=1}^{N_G} d_{x,g} = 1 \quad \forall\, x = 1..N_C$
- An instructor cannot be assigned to teach different classes the same time slot: $\sum_{y=1}^{N_T}\sum_{x=1}^{N_C}(d_{x,g} * t_{x,y}) = 1 \quad \forall\, g = 1 \dots N_G$
- Do not assign instructors to teach the subjects they have absolutely no interest in teaching (the level of preference equal 0): $\sum_{x=1}^{N_G} d_{x,g} * e_{g,x} > 0 \quad \forall x = 1 \dots N_C$
- Do not assign an instructor to teach the subjects they cannot ensure the quantity (the level of quantity equal 0): $\sum_{x=1}^{N_G} d_{x,g} * f_{g,x} > 0 \quad \forall x = 1 \dots N_C$
- Do not assign an instructor to teach in time slots they cannot teach (the level of preference equal 0): $\sum_{x=1}^{N_G} d_{x,g} * h_{g,x} > 0 \quad \forall x = 1 \dots N_C$
- Do not assign an instructor to teach fewer classes than the minimum number they are required to teach: $\left(\sum_{x=1}^{N_C} d_{x,g}\right) - minCourse_g \geq 0 \quad \forall i = 1 \dots N_G$
- Do not assign an instructor to teach more classes than the maximum number they are allowed to teach: $maxCourse_g - \left(\sum_{x=1}^{N_C} d_{x,g}\right) \geq 0 \quad \forall i = 1 \dots N_G$

C. Genetic Algorithm

The Genetic Algorithm (GA) is an optimization search algorithm based on modeling natural processes such as reproduction, inheritance, mutation, etc. It is a metaheuristic algorithm. Below is our proposed GA algorithm.

ALGORITHM 1: Proposed Generic Algorithm

---

1. Generate the initial population P consisting of π individuals.
    1.1. Loop π times: create individual p by:
        1.1.1. For each class:
            Create a set F include all instructors that satisfies the set of constraints mentioned in Section 2.2
            Choose a random instructor from F, assigned from current class.
        1.1.2. Add p into P.
2. For each individual p in P, calculate $p.fitness$.
3. Loop G times:
    3.1. Sort population P by descending order of $p.fitness$
    3.2. Create population P' by crossover and mutation.
        3.2.1. Loop μ times:
            Choose an unselected pair (p_{father}, p_{mother}) dividuals from P.
            Create p_{child} by combine top haft rows of p_{father} and bottom haft rows of p_{mother}
            Add p_{child} to P'.
        3.2.2. Loop β times
            Choose an unselected p' from P'.
            Swap the assigned instructors of two randomly selected classes of p'.
    3.3. For each invididual p' in P', calculate $p'.fitness$
    3.4. Create P'' by choosing the first ϕ individuals from P and (300- ϕ) individuals from P'.
    3.5. Set $P = P''$.
4. Choose the individuals p that has maximum $p.fitness$ from P.

End.

For everyone p of P has the fitness as:

$$p.fitness = \begin{cases} F(p) & if\ C(p) = 0 \\ \dfrac{F(p)}{\varepsilon} & if\ C(p) = 1 \end{cases}$$

Where $C(p) = 1$ if p violates any constraints and 0 otherwise, F is the fitness function mentioned in formular (7) above.

III. Experiment and Result

The proposed algorithm was tested on real data from FPT University – Ha Noi during the Spring 2022 semester, which included 153 classes and 25 instructors across 13 subjects and 10 time slots. The input data for the algorithm included lists of classes, time slots, subjects, and instructors with relevant indices. The coefficients w_4 and w_5 were set to 0.5 to equally optimize for departmental and individual needs, while the coefficients w_1, w_2, and w_3 were all set to $\frac{1}{3}$ to equally optimize for subject preferences, time slot preferences, and number of classes preferences for each instructor. The GA algorithm was run 15 times, with the best result identified in the fourth run, consisting of 300 generations. The chart below shows the best Fitness value obtained in each generation for the selected run:

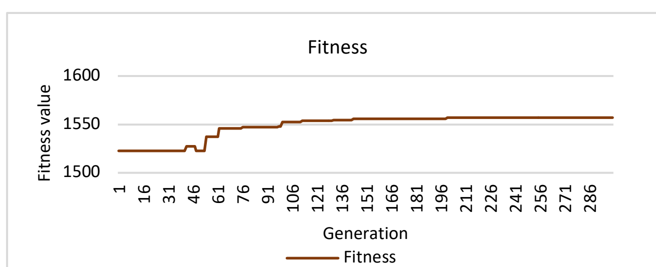
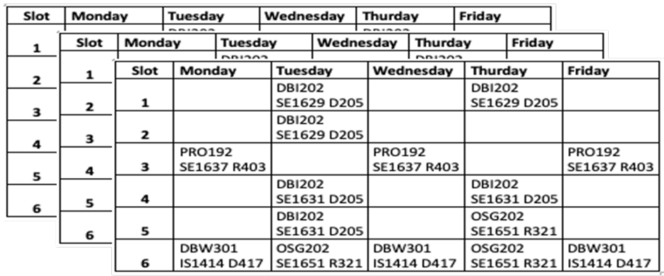


Fig. 3. The fitness values through each generation of the algorithm run have yielded the best solution.



Slot	Monday	Tuesday	Wednesday	Thurday	Friday
1		DBI202 SE1629 D205		DBI202 SE1629 D205	
2		DBI202 SE1629 D205			
3	PRO192 SE1637 R403		PRO192 SE1637 R403		PRO192 SE1637 R403
4		DBI202 SE1631 D205		DBI202 SE1631 D205	
5		DBI202 SE1631 D205		OSG202 SE1651 R321	
6	DBW301 IS1414 D417	OSG202 SE1651 R321	DBW301 IS1414 D417	OSG202 SE1651 R321	DBW301 IS1414 D417

Fig. 4. The selected schedule of all instructors.

The experiments were conducted on a Windows personal computer with the following configuration: Core i7 1255U 3.5GHz, 8GB RAM. The average time to run one iteration of the algorithm is 120.4 seconds. Running time like this is entirely feasible.

Regarding the level of entire teaching quality, we calculated the average quality score (across classes) of each subject, according to the assigned schedule, and then divided it by the teaching quality score of the best-performing instructor in that subject, obtaining the compliance rate for the quality aspect of each subject. On average across all subjects, the average quality score was 6.37, equivalent to meeting 67% of the maximum expected level. This rate can be known as the entitlement of the entire academic department.

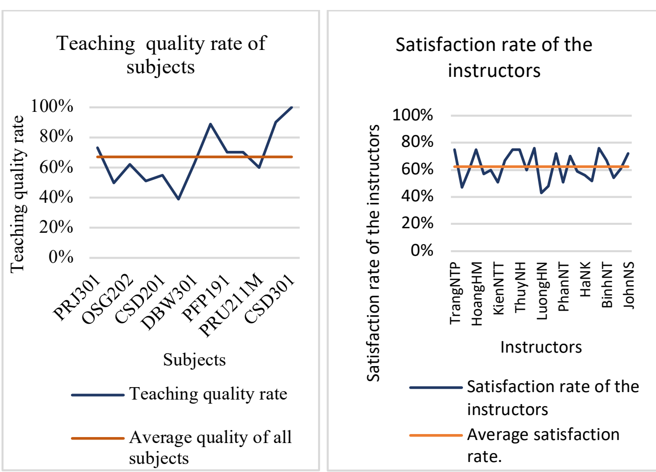


Fig. 5. The result solution created by GA. A) shows teaching quality rate of each subjects across classes. B) shows satisfaction rate of each instructors across assigned classes.

In terms of the satisfaction rate of the instructors, we constructed two comparison. We compared the number of classes assigned to each instructor with the minimum, maximum, and expected number of classes they desired. On average, the instructors deviated by 1.6 classes from their desired number of classes. We also created a table comparing the satisfaction level with the assigned schedule based on the criteria of subject and preferred time slot. We calculated the average liking score for each subject according to the given schedule, then took the percentage ratio to the highest expected value of the liking score for each subject that each instructor desired. The highest expected level was obtained by multiplying the number of assigned classes by the highest liking score for each subject in the input data. On average, the satisfaction rate of all instructors based on the criteria of liking for the subjects is 63%. Similar calculations were also performed with the criterion of preferred time slots. The result yielded an average satisfaction level of all instructors, based on the preferred time slots, of 62%.

Through the comparisons above, we see that among the participants in the problem, according to different criteria, no one achieves absolute or near absolute satisfaction. One of the main reasons for this is the input data itself: there are not enough instructors with high quality scores (8-10 points) for the number of classes needed to teach each subject, or an instructor may have high preferences for certain subjects, but there are not enough classes for that subject, the same goes for time frames. However, the results of our satisfaction assessment show on the other hand that all the participants in the problem (the entire department, all instructors), on all aspects (favorite subjects, favorite time slots) achieved a similar level of satisfaction, around 60-70%. This shows that no target is superior to another target, in other words, all optimal targets are almost evenly balanced.

Through the conducted experiments, we have observed that the proposed approach has achieved several advantages. Firstly, analyzing the problem using NASH equilibrium allows us to not only construct a schedule that satisfies the constraints or optimizes the organization's benefits but also considers the interests of the participating individuals. Secondly, NASH equilibrium demonstrates itself as an approach that effectively models the conflicts of interests in a clear, specific, and comprehensive manner. Thirdly, current approaches to multi-objective optimization problems often struggle to generate fairness among the objective functions. As indicated earlier, all participants in the problem have achieved a similar level of satisfaction, indicating that NASH equilibrium can be considered as a mediator between the optimal targets. The proposed method can be readily applied to other scheduling problems or any problem requiring the balancing of conflicting interests among multiple parties.

IV. Conclusions

In this study, we present a new approach to the problem of scheduling for university instructors. We apply the NASH equilibrium theory to model conflicts in the problem, and use it to develop an optimal model. The proposed approach is feasible for solving general multi-objective optimization problems and scheduling problems in particular. It also demonstrates potential directions for analyzing complex conflicting issues among multiple stakeholders.

Additionally, we design a genetic algorithm to solve the proposed optimal model. The results show that the algorithm maintains population diversity and runs within reasonable time limits. As the scale of university training expands, our algorithm can fully meet the demands.

However, there are still many factors and criteria in the university scheduling problem that have not been considered in this study, such as optimizing salary and rest time for instructors between teaching hours. Therefore, a more comprehensive investigation of the problem is our future research direction.

References

[1] M. Hosny, "Metaheuristic Approaches for Solving University Timetabling Problems: A Review and Case Studies from Middle Eastern Universities," in Information Systems and Technologies to Support Learning, Á. Rocha and M. Serrhini, Eds., in Smart Innovation, Systems and Technologies, vol. 111. Cham: Springer International Publishing, 2019, pp. 10–20. doi: 10.1007/978-3-030-03577-8_2.

[2] M. Mazlan, M. Makhtar, A. F. K. A. Khairi, and M. A. Mohamed, "University course timetabling model using ant colony optimization algorithm approach," Indones. J. Electr. Eng. Comput. Sci., vol. 13, no. 1, p. 72, Jan. 2019, doi: 10.11591/ijeecs.v13.i1.pp72-76.

[3] S. T. Ngo, J. B. Jaafar, I. A. Aziz, G. H. Nguyen, and A. N. Bui, "Genetic Algorithm for Solving Multi-Objective Optimization in Examination Timetabling Problem," Int. J. Emerg. Technol. Learn. IJET, vol. 16, no. 11, p. 4, Jun. 2021, doi: 10.3991/ijet.v16i11.21017.

[4] Y.-J. Gong et al., "Distributed evolutionary algorithms and their models: A survey of the state-of-the-art," Appl. Soft Comput., vol. 34, pp. 286–300, Sep. 2015, doi: 10.1016/j.asoc.2015.04.061.

[5] H. Algethami and W. Laesanklang, "A Mathematical Model for Course Timetabling Problem With Faculty-Course Assignment Constraints," IEEE Access, vol. 9, pp. 111666–111682, 2021, doi: 10.1109/ACCESS.2021.3103495.

[6] T. Thepphakorn and P. Pongcharoen, "Performance improvement strategies on Cuckoo Search algorithms for solving the university course timetabling problem," Expert Syst. Appl., vol. 161, p. 113732, Dec. 2020, doi: 10.1016/j.eswa.2020.113732.

[7] N. M. Arratia-Martinez, P. A. Avila-Torres, and J. C. Trujillo-Reyes, "Solving a University Course Timetabling Problem Based on AACSB Policies," Mathematics, vol. 9, no. 19, p. 2500, Oct. 2021, doi: 10.3390/math9192500.

[8] S. T. Ngo, J. Jaafar, I. A. Aziz, and B. N. Anh, "A Compromise Programming for Multi-Objective Task Assignment Problem," Computers, vol. 10, no. 2, p. 15, Jan. 2021, doi: 10.3390/computers10020015.

[9] B. Xin, L. Chen, J. Chen, H. Ishibuchi, K. Hirota, and B. Liu, "Interactive Multiobjective Optimization: A Review of the State-of-the-Art," IEEE Access, vol. 6, pp. 41256–41279, 2018, doi: 10.1109/ACCESS.2018.2856832.

[10] H. Wang, M. Olhofer, and Y. Jin, "A mini-review on preference modeling and articulation in multi-objective optimization: current status and challenges," Complex Intell. Syst., vol. 3, no. 4, pp. 233–245, Dec. 2017, doi: 10.1007/s40747-017-0053-9.

[11] L. Wang, Z. Zhang, X. Guo, C. Sun, and Y. Lan, "Strong Pareto-based multiobjective differential evolution algorithm," In Review, preprint, Dec. 2022. doi: 10.21203/rs.3.rs-2339827/v1.

[12] N. Sedaghat, H. Tabatabaee-Yazdi, and M.-R. Akbarzadeh-T, "Pareto Front Based Realistic Soft Real-Time Task Scheduling with Multi-objective Genetic Algorithm in Unstructured Heterogeneous Distributed System," in Advances in Grid and Pervasive Computing, P. Bellavista, R.-S. Chang, H.-C. Chao, S.-F. Lin, and P. M. A. Sloot, Eds., in Lecture Notes in Computer Science, vol. 6104. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 268–279. doi: 10.1007/978-3-642-13067-0_30.

[13] L. F. Amorim, A. P. de Paiva, P. P. Balestrassi, and J. R. Ferreira, "Multi-objective optimization algorithm for analysis of hardened steel turning manufacturing process," Appl. Math. Model., vol. 106, pp. 822–843, Jun. 2022, doi: 10.1016/j.apm.2022.02.011.

[14] S. Tung Ngo, J. Jafreezal, G. Hoang Nguyen, and A. Ngoc Bui, "A Genetic Algorithm for Multi-Objective Optimization in Complex Course Timetabling," in 2021 10th International Conference on Software and Computer Applications, Kuala Lumpur Malaysia: ACM, Feb. 2021, pp. 229–237. doi: 10.1145/3457784.3457821.

[15] 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, p. 388, Mar. 2022, doi: 10.3390/e24030388.

[16] P. Henryk and G. Gawinowski, "Compromise in Scheduling Objects Procedures Basing on Ranking Lists," in Intelligent Decision Technologies, J. Watada, G. Phillips-Wren, L. C. Jain, and R. J. Howlett, Eds., in Smart Innovation, Systems and Technologies, vol. 10. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 843–852. doi: 10.1007/978-3-642-22194-1_83.

[17] 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, p. e1063, Aug. 2022, doi: 10.7717/peerj-cs.1063.

[18] A. Nagurney, M. Salarpour, J. Dong, and P. Dutta, "Competition for Medical Supplies Under Stochastic Demand in the Covid-19 Pandemic: A Generalized Nash Equilibrium Framework," in Nonlinear Analysis and Global Optimization, T. M. Rassias and P. M. Pardalos, Eds., in Springer Optimization and Its Applications, vol. 167. Cham: Springer International Publishing, 2021, pp. 331–356. doi: 10.1007/978-3-030-61732-5_15.

[19] B. A. Bhatti and R. Broadwater, "Distributed Nash Equilibrium Seeking for a Dynamic Micro-grid Energy Trading Game with Non-quadratic Payoffs," Energy, vol. 202, p. 117709, Jul. 2020, doi: 10.1016/j.energy.2020.117709.

[20] A. Agnetis, C. Briand, J.-C. Billaut, and P. Šůcha, "Nash equilibria for the multi-agent project scheduling problem with controllable processing times," J. Sched., vol. 18, no. 1, pp. 15–27, Feb. 2015, doi: 10.1007/s10951-014-0393-x.

[21] P. Šůcha, A. Agnetis, M. Šidlovský, and C. Briand, "Nash equilibrium solutions in multi-agent project scheduling with milestones," Eur. J. Oper. Res., vol. 294, no. 1, pp. 29–41, Oct. 2021, doi: 10.1016/j.ejor.2021.01.023.

[22] 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, p. 105130, Apr. 2021, doi: 10.1016/j.cor.2020.105130.

[23] T. N. Bao, H. Q. Thang, and N. Thuy Linh, "Research on Genetic Algorithm and Nash Equilibrium in Multi-Round Procurement," in NEW TRENDS IN INTELLIGENT SOFTWARE METHODOLOGIES, TOOLS AND TECHNIQUES,

[24] D.-N. Le, G. N. Nguyen, T. N. Bao, N. N. Tuan, H. Q. Thang, and S. C. Satapathy, "MMAS Algorithm and Nash Equilibrium to Solve Multi-round Procurement Problem," in Advances in Systems, Control and Automations, A. K. Bhoi, P. K. Mallick, V. E. Balas, and B. S. P. Mishra, Eds., in Lecture Notes in Electrical Engineering, vol. 708. Singapore: Springer Nature Singapore, 2021, pp. 273–284. doi: 10.1007/978-981-15-8685-9_28.

[25] M. C. Chen, S. N. Sze, S. L. Goh, N. R. Sabar, and G. Kendall, "A Survey of University Course Timetabling Problem: Perspectives, Trends and Opportunities," IEEE Access, vol. 9, pp. 106515–106529, 2021, doi: 10.1109/ACCESS.2021.3100613.

[26] T. Q. Huynh, N. B. Trinh, and T. X. Nguyen, "NASH EQUILIBRIUM MODEL FOR CONFLICTS IN PROJECT MANAGEMENT," J. Comput. Sci. Cybern., vol. 35, no. 2, pp. 167–184, Jun. 2019, doi: 10.15625/1813-9663/34/3/13095.

[27] H. Nikaidô and K. Isoda, "Note on non-cooperative convex game," Pac. J. Math., vol. 5, no. 5, pp. 807–815, Dec. 1955, doi: 10.2140/pjm.1955.5.807.