Optimizing University Course Timetabling for Metaverse Integration: A Human-Centered Decision Model in Medical Education

Seckin Damar[1], Gulsah Hancerliogullari Koksalmis[2,3*]

[1] Department of Industrial Engineering, Faculty of Management, Istanbul Technical University, Istanbul, Turkey, damar17@itu.edu.tr, ORCID: 0000-0002-2713-2738
[2] Department of Industrial Engineering, Faculty of Management, Istanbul Technical University, Istanbul, Turkey, ghancerliogullari@itu.edu.tr, ORCID: 0000-0002-2551-541X
[3] Department of Industrial Engineering and Management Systems, University of Central Florida, Orlando, FL 32816, USA, gulsah.hancerliogullarikoksalmis@ucf.edu

[*] Corresponding author. E-mail address: ghancerliogullari@itu.edu.tr Telephone number: +90 530 312 48 91 (G. Hancerliogullari Koksalmis)

Abstract

This paper addresses the University Course Timetabling Problem (UCTTP) in medical schools through a novel framework by developing a binary integer linear programming model that incorporates both metaverse and regular (non-metaverse) courses. Assignment decisions in the model are guided by weights that quantitatively represent professors' behavioral intentions to use metaverse technologies. These weights are derived using the Analytic Hierarchy Process (AHP), based on constructs obtained through Structural Equation Modeling (SEM). The weights are integrated into the objective function of the model to ensure that metaverse courses are assigned to the most appropriate professors in line with their level of intention towards metaverse technologies. In the solution process of the model, a comparative approach is adopted by employing the Greedy Reassignment and Assignment for Professor Equity (GRAPE) algorithm alongside the Simulated Annealing (SA) algorithm. The GRAPE is designed to rapidly generate initial solutions, which are subsequently improved using the SA approach. The parameters of the SA algorithm is tuned using the Taguchi Design of Experiments (DoE) method to determine the optimal configurations. A computational study is conducted on 45 synthetically generated problem instances of varying sizes. The results indicate that GRAPE performs efficiently in terms of computation time, whereas the SA algorithm yields higher solution quality, particularly for larger and more complex datasets. This study presents a comprehensive and innovative framework for integrating rapidly evolving metaverse-based teaching methods into course planning processes in medical education through a human-centered approach.

Keywords: University course timetabling . Educational technology integration . Metaverse in medical education . Integer linear programming . Analytic hierarchy process . Simulated annealing

1 Introduction

The University Course Timetabling Problem (UCTTP) addresses the challenge of assigning students, instructors, and courses to specific timeslots and classrooms while ensuring that the resulting schedules for students and faculty comply with technical constraints (Heidari et al., 2021). The timetabling problem fundamentally represents a resource allocation challenge, and it is widely recognized that such allocation tasks are combinatorial in nature and classified as NP-complete problems (Deris et al., 2000). The majority of NP-complete problems are difficult to solve optimally and efficiently due to the vastness of their search spaces, which expand exponentially with the increase in the number of variables (Parker & Rardin, 2014). Due to this complexity, exact solution approaches become computationally infeasible and require the adoption of heuristic and metaheuristic optimization strategies.

At the same time, medical education is experiencing a digital transformation with the emergence of the metaverse, which integrates virtual reality (VR), augmented reality (AR), and other immersive technologies within shared virtual environments (Lewis et al., 2024; Sandrone, 2022). A recent large-scale bibliometric analysis conducted by Damar and Koksalmis (2024) draws attention to metaverse applications in healthcare and highlights the growing research interest in this area within medical education. Within the metaverse context, medical students are able to engage deeply with highly realistic and interactive virtual environments that replicate clinical settings, including virtual hospitals, operating rooms, and patient interactions (Farrukh, 2024). These interactive and immersive environments provide students with a flexible and safe learning space while supporting active participation, student-centered approaches, and personalized learning experiences. In this respect, they align with the constructivist pedagogical principles proposed by Lam et al. (2021). Empirical studies have also highlighted the pedagogical value of these immersive environments. For example, Mergen et al. (2025) demonstrated that immersive VR-based simulations significantly improved medical students' perceived self-efficacy in performing skin cancer screenings. Furthermore, Bowen et al. (2021) found that virtual reality tools used in global health education enhanced students' emotional engagement, empathy, and awareness of humanitarian issues. In parallel, Campos et al. (2020) demonstrated the positive effects of simulation-based education (SE) on learning. Their study emphasized that SE, delivered through tools such as virtual laboratories and online platforms, enhanced students' critical thinking, decision-making, and collaboration skills.

AR and VR technologies also contribute significantly to specific domains of medical education. In surgical training, doctors can use AR to rotate specific anatomical structures during brain and cardiac surgeries, allowing for better visualization in the process of performing, planning and explaining the surgery. Anatomy education is supported by VR-based platforms where doctors can visualize and explain medical processes to other healthcare professionals and medical students can examine multiple organs and organ systems. Within classroom settings, AR tools, tools such as augmented reality pens can allow students to examine and learn concepts through three-dimensional images (Venkatesan et al., 2021). AR can also project virtual images onto physical models or cadavers during anatomical learning, providing additional visual information and labels throughout dissections. In medical imaging, AR enables three-dimensional scan data to be displayed directly on the patient's body to support visualization and surgical planning. Additionally, in procedural training, AR can assist medical students by overlaying step-by-step instructions on the patient or related medical instruments during practice. Hybrid simulations in medical education involve the integration of physical simulators or mannequins with virtual elements, enabling the practice of complex and realistic clinical scenarios and procedures. Additionally, virtual and augmented reality technologies are used in team-based training by simulating situations that require coordination among healthcare professionals, supporting the development of communication and collaboration skills in settings such as emergency care (Lewis et al., 2024). Furthermore, simulation-based medical education presents opportunities to minimize risks for both patients and students, strengthen student competence and confidence, improve patient safety, and contribute to the reduction of healthcare costs over time (Al-Elq, 2010). These technologies bring a new dimension to medical education by enabling interactive, scalable, and secure skill development across various domains.

Integrating the metaverse into medical education brings new course scheduling requirements. As medical schools add metaverse-enhanced courses and simulation sessions, new types of classes and resources (e.g. VR lab time, remote simulation workshops, specialized hardware) need to be scheduled. These additions impose extra constraints and scheduling requirements on UCTTP, such as the simultaneous allocation of virtual and physical resources. In fact, the already NP-hard nature of the scheduling problem is further compounded by the need to integrate metadata-driven learning into the timeline. In other words, the rise of metaverse-based medical education creates new scheduling challenges that make solving the NP-hard UCTTP even more critical for institutions aiming to coordinate traditional and virtual courses.

This study aims to develop a mathematical model that assigns regular courses and metaverse-based courses together. Here, regular courses refer to courses conducted through traditional, face-to-face instruction in standard physical classrooms. In contrast, metaverse courses involve immersive technologies such as VR, AR, or simulation-based platforms and require specialized virtual or metaverse-enabled environments. An important feature of the study is the integration of weights into the mathematical model, denoted as *w(i,p)*, derived from an Analytic Hierarchy Process (AHP) based on a previous structural equation model (SEM) (Damar & Koksalmis, 2023). The SEM was designed to assess medical educators' intention to adopt metaverse technologies in teaching and its important constructs were used as criteria to calculate these weights. By placing these weights in the objective function of a binary integer programming model, the study aims to prioritize the assignment of metaverse courses to professors who show greater willingness and adaptability to immersive teaching environments. This approach addresses emerging institutional scheduling needs as medical education increasingly incorporates metaverse-based learning methods.

The unique contribution of this study lies in integrating behavioral intention modeling with course scheduling optimization. By combining weights derived from an SEM and AHP, the proposed model extends traditional course scheduling approaches to account for faculty's behavioral intention to teach metaverse-based courses. This link between metaverse technology adoption and scheduling decisions enables more informed, personalized, and future-ready academic planning. To the best our knowledge, this is the first course planning model to combine regular and metaverse courses in a unified optimization framework driven by faculty weighting. As immersive technologies become increasingly central to medical education, this approach offers a novel mechanism for aligning instructional delivery with faculty readiness and institutional needs.

2 Problem definition

The UCTTP needs advanced mathematical models to meet the diverse and complex needs of academic institutions. Instead of focusing only on a specific case, this paper proposes a general model that includes different scenarios. The proposed model is flexible and inclusive and can be adapted to various educational needs. In particular, the inclusion of metaverse-based courses in this framework makes it possible to adapt to modern teaching methods in medical education and to adapt faster to technological developments.

The problem aims to create an efficient timetable within the university by assigning courses to available classes, time slots and days according to various constraints. This optimization process involves the development of a feasible schedule for regular courses, professors, and metaverse courses based on certain constraints. Accordingly, the model is described below:

- The term timeslot signifies a duration of one hour, encompassing nine such intervals within a day from 8:00 AM to 5:45 PM, occurring Monday through Friday.
- The courses are divided into regular courses and metaverse courses.
- Professors demonstrate a fit based on their field competencies by undertaking courses that overlap with their areas of expertise. In the case of metaverse-based courses, their

technological aptitude and potential to teach such courses were taken into account. In this context, professors' level of intention to use metaverse technology was determined through a previous SEM, and professors with a high level of intention were prioritized to be assigned metaverse courses. Thus, professors with a high capacity to adapt to technological innovations were encouraged to take on this new generation of courses.

- The challenges of implementing medical education in the metaverse are multifaceted. These include limitations on internet bandwidth and device (e.g. smartphones or computers) access, restrictions on the availability of AR/VR hardware, health-related usage considerations, restrictions on platform selection and integration, guidelines on the use of anonymity, and regulations to protect intellectual property (Sandrone, 2022). Additional concerns include cognitive overload, challenges related to the educational process, isolation and lack of social interaction, as well as motion sickness (Sakr & Abdullah, 2024). Addressing both hardware and cognitive challenges within the model is essential for ensuring that the developed framework accurately reflects the real-world problem. Accordingly, the relevant constraints representing these challenges must be carefully formulated. Integrating these constraints into the optimization model will help ensure that the resulting solution aligns with the practical limitations of implementing metaverse-based medical education and that these challenges are addressed in a comprehensive manner.

3 Literature review

University course scheduling has been widely recognized as an NP-hard combinatorial optimization problem that requires considerable attention due to its complexity. This task requires assigning courses to available time slots, rooms, and instructors while simultaneously satisfying both hard constraints, such as avoiding conflicts between students and instructors, and soft preferences, such as student satisfaction and resource utilization. A bibliometric analysis was conducted to comprehensively search for studies examining university course scheduling problems. Relevant studies were identified through a search of the Scopus database using a comprehensive set of keywords related to: (("timetable*" OR "schedule*" OR "course scheduling" OR "course allocation" OR "course planning" OR "course optimization" OR "scheduling problem*" OR "scheduling" OR "timetabling" OR "timetable scheduling" OR "timetable allocation" OR "schedule allocation" OR "timetable planning" OR "schedule planning" OR "timetable optimization" OR "timetabling problem*") AND ("metaheuristic*") AND ("universit*" OR "college*" OR "academy" OR "school*")). As a result of this search, 236 records were retrieved.

Subsequently, a co-occurrence analysis of author keywords was performed using VOSviewer and it was found that Simulated Annealing (SA) is the most common metaheuristic technique used for course scheduling problems (Fig. 1). This finding emphasized the important role of Simulated Annealing in addressing the inherently constrained and complex nature of university scheduling problems. Given the computational challenges of exact optimization for large-scale problems, metaheuristics and hybrid approaches are frequently used in the literature. In the following sections, important developments will be synthesized under three interrelated themes: classical metaheuristics, hybrid metaheuristic techniques and adaptive or hyper-heuristic strategies.

3.1 Classical metaheuristic methods

Significant research has focused on the application of classical metaheuristic algorithms to university course scheduling. The power of SA to effectively navigate large search spaces was demonstrated by Sylejmani et al. (2023) who proposed a two-stage SA framework in which the feasibility of solutions is prioritized in the first stage followed by the optimization of soft constraints in the second stage through penalization mechanisms designed to escape local optima. Although this strategy yielded promising results, it was mainly validated on the ITC2019 benchmark datasets, thus limiting the generalizability of their findings. Similarly, Bellio et al. (2016) introduced a feature-guided SA where parameter selection is guided by predefined features of the problem instance, improving performance in curriculum-based scheduling contexts. However, the dependency on fixed feature sets restricted its flexibility when applied to datasets with different features. Junn et al. (2017) conducted a comparative analysis of Simulated Annealing and the Big Flood algorithm, highlighting the different advantages and shortcomings of each method. Although valuable insights were provided, the potential benefits of integrating these two approaches into a hybrid model were not explored.

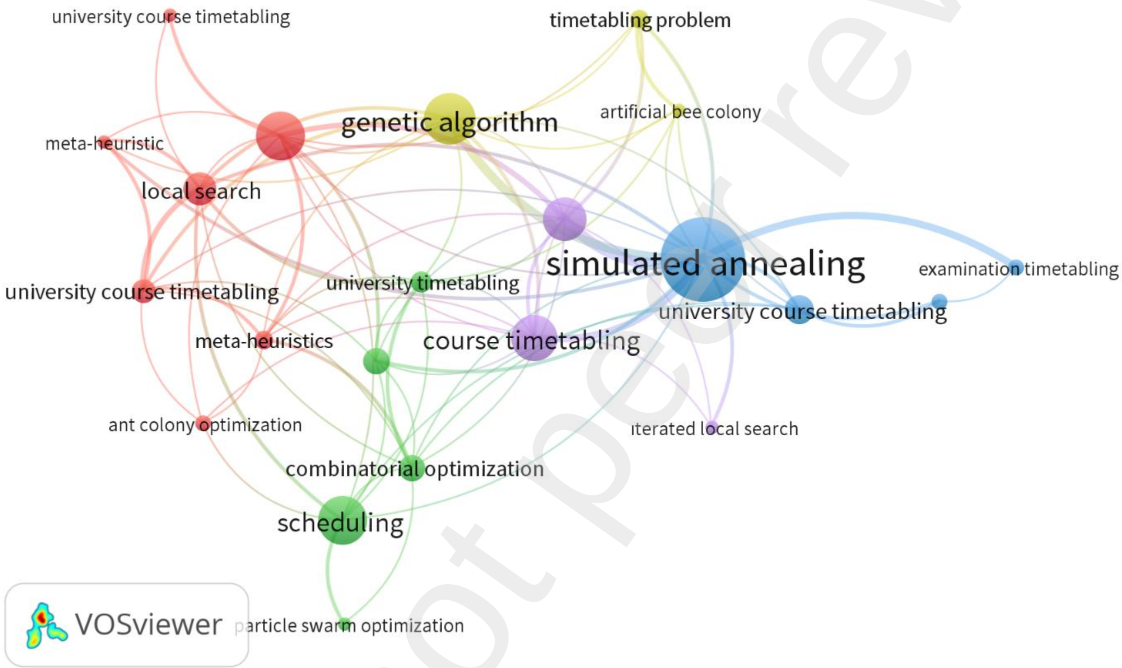

Fig. 1 Co-occurrence analysis of author keywords

Beyond SA and Great Deluge, the utility of other metaheuristics has also been explored. Hossain et al. (2019) extended the traditional Particle Swarm Optimization (PSO) framework by incorporating selective search and coercive trade-off operations, which significantly improved constraint satisfaction levels. However, the increased computational overhead associated with this approach posed challenges for scaling to larger datasets. Mahlous and Mahlous (2023) proposed a modified Genetic Algorithm in which student preferences are directly integrated into the scheduling process using repair and improvement functions. Their method achieved high satisfaction rates exceeding 90%, but the complexity of the algorithm raised concerns about its scalability to large problem instances. On a related topic, the effectiveness of Tabu Search was demonstrated by Laguardia and Flores (2022) through the assignment of professors to math courses where a greedy initialization enabled the method to efficiently escape local optima. However, similar to other studies, the scalability of their approach to larger or more complex scenarios has not been explored. Al-Betar and Khader (2012) developed a Harmony Search and its modified variant for the course scheduling problem, displaying competitive performance against established metaheuristics. Nonetheless, the reliance on manually set parameters and evaluations based on standard benchmarks rather than various real-world cases limited the findings' broader application.

These contributions have demonstrated that classical metaheuristics such as SA, PSO, Genetic Algorithm and Tabu Search can produce high quality schedules. However, several limitations remain, including the static nature of parameter settings, the limited variety of evaluation datasets, and the computational demands of large-scale applications. These challenges have motivated the development of hybrid metaheuristic frameworks.

3.2 Hybrid metaheuristic techniques

To address the limitations observed in single metaheuristic implementations, hybrid techniques that integrate complementary search strategies have been developed. Among the prominent contributions, Vianna et al. (2020) proposed a hybrid approach combining Variable Neighborhood Search (VNS) and Tabu Search, where systematic neighborhood modifications are used to improve global exploration, while densification is achieved through memory-based local search. Although significant improvements in solution quality were achieved, the lack of adaptive parameter control mechanisms limited the flexibility of the algorithm on different instances. Wong et al. (2022) proposed an approach that incorporates Tabu Search in the initialization phase of the Genetic Algorithm. Thanks to this integration, high quality initial populations are generated to improve the final solutions. However, the additional computational effort introduced by the Tabu Search phase highlighted the classic trade-off between solution quality and computational cost. A different hybridization strategy was pursued by Thepphakorn and Pongcharoen (2020), who proposed a Self-Adaptive Cuckoo Search algorithm in which critical parameters are dynamically adjusted to improve convergence. Although promising improvements have been reported, the lack of comparative analysis with alternative hybrids has made it difficult to fully assess their relative advantages. Badoni et al. (2023) introduced another hybrid technique that integrates Genetic Algorithms with Iterative Local Search. In this framework, global exploration is performed through Genetic Algorithm while local refinements are performed through iterative perturbations and refinements, effectively escaping local optima. Despite its robustness, the approach remained highly dependent on the quality of the initial solutions generated. Song et al. (2021) contributed by developing Competition-Driven Multi-neighborhood Local Search, where neighborhood operators compete based on their success rates, thus allowing the algorithm to dynamically select the most efficient operator during the optimization process. While this method has proven to be effective in handling complex constraints, the computational burden involved has raised concerns about its applicability in real-time scheduling environments. Bolaji et al. (2014) suggested a hybrid model in which the Artificial Bee Colony (ABC) algorithm was combined with Hill Climbing local search. This combination increased the ABC framework's exploitation capabilities and produced competitive results on Socha's benchmark datasets. However, the approach was primarily validated on small to medium instances, and its scalability to larger problems remained unexplored. Abdullah et al. (2012) suggested a hybrid metaheuristic that combined the Electromagnetism-Like Mechanism (EM) with the Great Deluge algorithm. Global exploration was accomplished using EM dynamics, while local search was improved utilizing GD's adaptive acceptance criteria. While their solution showed increased convergence on benchmark datasets, scalability issues and dependency on static tuning continued. De Causmaecker et al. (2009) proposed a decomposed hybrid metaheuristic strategy for dealing with real-world complications in university course scheduling. Their solution included a pre-processing stage in which related lectures were organized into "pillars" to limit the size of the search space, followed by a staged application of Tabu Search. Better performance was achieved by gradually adding more constraints across sequential optimization steps, as opposed to directly solving the fully constrained problem. While the breakdown improved computing efficiency and solution quality, there were questions about the pillar-based decomposition's applicability to other academic institutions, as the initial grouping necessitated manual design decisions.

Overall, hybrid metaheuristic methodologies demonstrate that integrating global and local search capabilities improves course scheduling performance. However, the complexity provided by hybridization, together with parameter setting and computational demands, continues to present challenges that must be overcome by increasingly advanced adaptive techniques.

3.3 Adaptive and hyper-heuristic approaches

Adaptive approaches dynamically modify algorithmic behaviors in response to the changing landscape of the search process, whereas hyper-heuristics operate at a meta-level, selecting or producing low-level heuristic. In this context, Tarawneh and Ayob (2013) proposed an adaptive Simulated Annealing framework where neighborhood selection is dynamically adjusted based on past performance. While this adaptive mechanism showed improvements in search efficiency, the lack of a direct comparison with static alternatives left the magnitude of its benefits somewhat unclear. Similarly, Alhuniti et al. (2020) introduced intelligent mutation strategies within a Genetic Algorithm, prioritizing mutations that are statistically more likely to lead to better solutions. While increasing the speed of convergence, the scalability of the method to large-scale scheduling problems remains largely untested. Prakasa et al. (2024) carried out a reheating technique within SA to further develop the attempts to maintain diversity and prevent local optima. The algorithm was able to revive the search and investigate new areas of the solution space by alternately raising the temperature. However, the lack of comparative benchmarks against alternative diversity preservation strategies has limited a wider validation of their approach. Kiefer et al. (2017) made significant progress in adaptivity by developing an Adaptive Large Neighborhood Search (ALNS) for curriculum-based course scheduling. Their strategy gradually reduced the amount of devastation during the search, constantly balancing exploration and exploitation. This adaptive feature allowed the system to exceed the best-known results in various ITC-2007 benchmark cases.

Taken together, adaptive and hyper-heuristic approaches provide promising ways to overcome the rigidity and sensitivity to parameter settings observed in previous metaheuristic and hybrid studies. However, further research is needed to fully realize their potential, especially through extensive benchmarking and scalability analyses.

4 Methodology

To address the university course scheduling problem in the context of metaverse-based education, an appropriate methodological framework is required. For this purpose, a multi stage approach has been developed. Figure 2 presents an overview of the proposed framework. The first step involves constructing a Structural Equation Model to identify the key factors influencing professors' intentions to use metaverse technologies based on the study by Damar and Koksalmis (2023). These factors are then used as criteria in the Analytic Hierarchy Process to derive professor weights. The resulting weights are incorporated into the objective function of a Binary Integer Programming model. To solve the model, a greedy algorithm referred to as Greedy Reassignment and Assignment for Professor Equity (GRAPE) and a Simulated Annealing (SA) algorithm are employed. In addition, the Taguchi Design of Experiments is used to determine the optimal parameters for the SA algorithm. Each step of the methodology is described in detail in the following subsections. The proposed methodology is illustrated in Fig. 2.

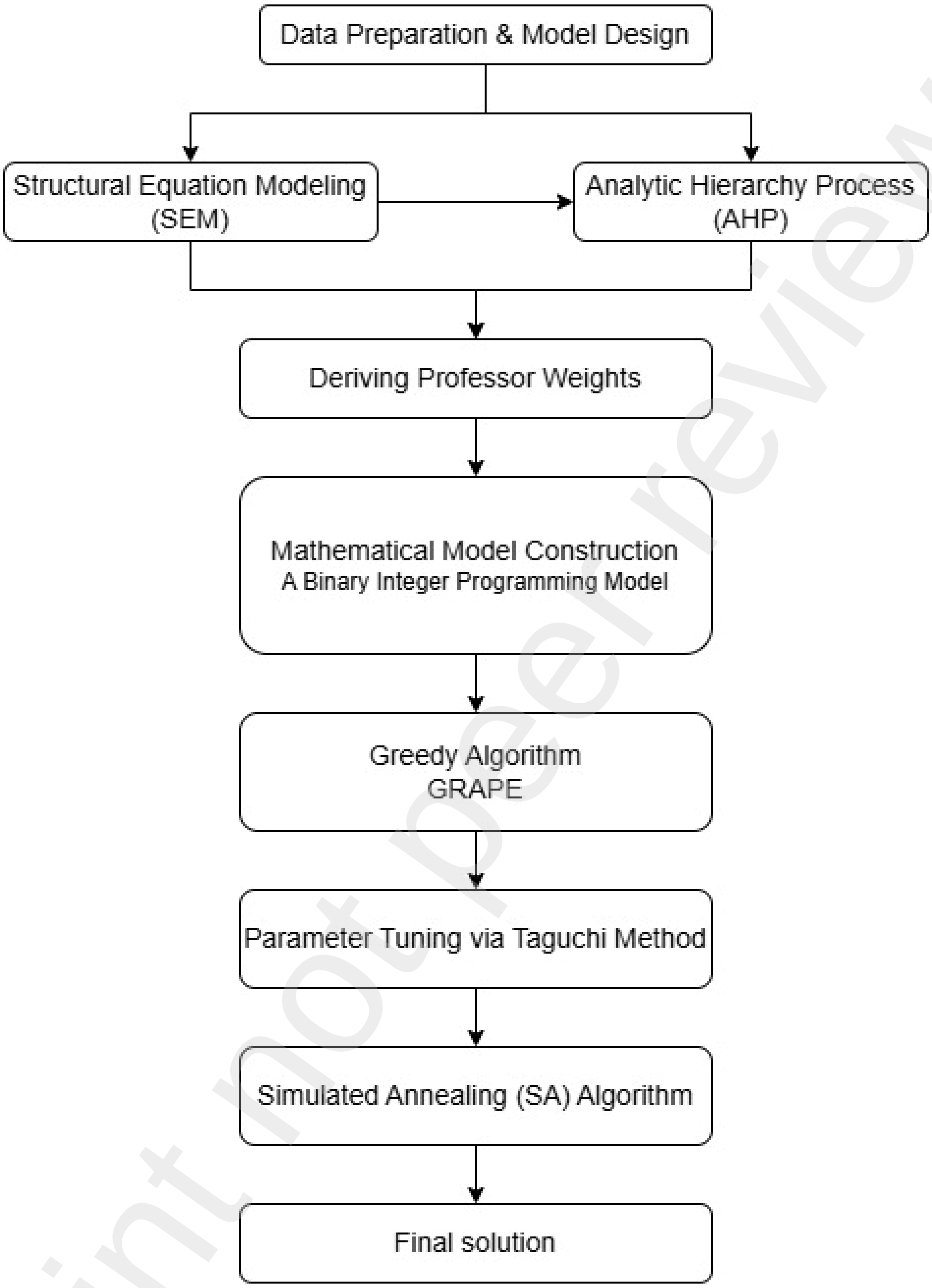

Fig. 2 Flowchart of the proposed methodology

4.1 Structural Equation Modeling approach

Structural Equation Modeling (SEM) refers to a suite of statistical methods used to analyze complex relationships among multiple independent and dependent variables, which may be either continuous or categorical. These variables can represent latent constructs (factors) or directly observed measures. SEM is also commonly known by several other terms, including causal modeling, causal analysis, simultaneous equation modeling, covariance structure analysis, path analysis, and confirmatory factor analysis. Notably, path analysis and confirmatory factor analysis are considered specific subtypes within the broader SEM framework (Ullman & Bentler, 2012).

SEM offers several advantages over traditional statistical methods. It enables the simultaneous analysis of multiple dependent variables and accommodates measurement errors in both independent and dependent variables. SEM allows for the concurrent estimation of latent variable structures and the relationships among them, thereby integrating factor analysis and path analysis within a unified framework. Moreover, it supports flexible measurement models, including those with indicators linked to multiple factors, and facilitates the assessment of model fit, providing a more comprehensive understanding of the data's underlying structure (Zhang, 2022).

4.2 Analytic Hierarchy Process approach

The Analytic Hierarchy Process (AHP) is a decision-making methodology designed to address problems involving multiple objectives and criteria. It utilizes a systematic pairwise comparison technique to establish a prioritized ranking among a set of alternatives based on relative preferences (Saaty, 1984). It is particularly effective in structuring complex decisions by decomposing them into a hierarchy consisting of a goal, evaluation criteria, and alternatives (Saaty, 1990). The method utilizes ratio-scale judgments through pairwise comparisons, and these judgments are mathematically synthesized by solving an eigenvalue problem to derive priority weights (Saaty, 1990).

In AHP, judgments are collected in the form of pairwise comparisons, which are used to construct a reciprocal matrix $A = [a_{ij}]$, where each entry a_{ij} denotes the relative importance or preference of element *i* over element *j* (Saaty, 1977). By definition, the matrix must satisfy the condition given in Eq. (1).

$$a_{ji} = \frac{1}{a_{ij}}, \quad a_{ii} = 1 \tag{1}$$

This ensures the matrix is positive and reciprocal, a foundational property in AHP.

If the pairwise comparisons are perfectly consistent, then the following condition holds for all *i*, *j*, *k* as shown in Eq. (2).

$$a_{ij}.a_{jk} = a_{ik} \tag{2}$$

In such cases, there exists a priority vector $w = [w_1, w_2, \ldots, w_n]^T$, as expressed in Eq. (3).

$$a_{ij} = \frac{w_i}{w_j} \tag{3}$$

This leads to the matrix form presented in Eq. (4).

$$A = \begin{bmatrix} 1 & \frac{w_1}{w_2} & \cdots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & 1 & \cdots & \frac{w_2}{w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \cdots & 1 \end{bmatrix} \tag{4}$$

To estimate the ratio-scale weights from real-world (and potentially inconsistent) judgments, the principal right eigenvector of the matrix A is obtained by solving the eigenvalue equation presented in Eq. (5).

$$Aw = \lambda_{max} w \tag{5}$$

The eigenvector *w* corresponding to the maximum eigenvalue λ_{max} is then normalized, as shown in Eq. (6).

$$\sum_{i=1}^{n} w_i = 1 \tag{6}$$

Saaty (1977) showed that if the matrix is fully consistent, then $\lambda_{max} = n$; otherwise, $\lambda_{max} > n$, and the Consistency Index (CI) can be used to quantify the degree of inconsistency, as shown in Eq. (7).

$$CI = \frac{\lambda_{max} - n}{n - 1} \tag{7}$$

This formulation ensures that even when small inconsistencies are present, reliable ratio-scale weights can still be extracted.

4.3 Illustrative study

An illustrative example is created to demonstrate the integration of SEM and AHP methodologies in the proposed optimization framework. This example aims to show how important latent constructs identified through SEM are transformed into decision criteria for AHP and then used to derive professor weights for metaverse courses, denoted as *w(i,p)*. In the example, professors eligible to teach metaverse courses are treated as alternatives and criteria are identified from SEM analysis based on constructs with statistically significant total effects. These constructs are then normalized to generate the relative weights used to make pairwise comparisons between eligible professors for each criterion.

Assume that professors a, b and c are eligible for metaverse courses. Here these professors will be used as alternatives. For these alternatives, pairwise comparison matrices will be created for each criterion. For example, the pairwise comparison matrix for the criterion "imagination" is given in Table 1. The pairwise comparison values represent the randomly assigned values between the three professors regarding their suitability for the "imagination" criterion. A scale from 1 to 9 was used to represent the importance of one professor over the other. On the scale, 1 means equal importance, 3 means moderate importance, 5 means strong importance, 7 means very strong importance and 9 means extreme importance. The weight of each criterion is shown in Table 2.

Table 1 Normalized weight of the criteria

Criteria	Total effects	Normalized total effects
Imagination	0.218	0.175
Perceived trialability	0.128	0.102
Technology anxiety	0.126	0.101
Technology readiness	0.073	0.058
Perceived usefulness	0.491	0.393
Perceived ease of use	0.213	0.170

Table 2 Pairwise Comparison matrix with respect to the imagination

	Professor a	Professor b	Professor c	Weights
Professor a	1	3	5	0.633
Professor b	1/3	1	3	0.260
Professor c	1/5	1/3	1	0.106

So, the relative scores of professor a, professor b, and professor c for imagination can be calculated as, professor a, (0.633*0.175 = 0.111), professor b, (0.260*0.175 = 0.046), professor c, (0.106*0.175 = 0.019). Similarly, the relative scores for the remaining criteria are calculated in the same manner. Accordingly, the professor weight (PW) for each professor p is computed using Eq. (8).

$$PW_p = \sum_{j=1}^{6} C_j b_{pj} \tag{8}$$

where C_j is the relative weightage for the criterion j, b_{pj} is the relative weightage for professor p with respect to j^{th} criterion, PW_p professor weight for professor p.

For example, in the aforementioned example, C_1 represents the weight assigned to the imagination criterion and is equal to 0.175. b_{a1} denotes the relative weight for professor a with respect to first criterion, which is 0.633. PW_a is then computed using Eq. (9).

$$PW_a = 0.175 * 0.633 + 0.102 * b_{a2} + 0.101 * b_{a3} + 0.058 * b_{a4} + 0.393 * b_{a5} + 0.17 * b_{a6} \tag{9}$$

In this example, six constructs were identified to measure intention to use and these constructs were evaluated as criteria. Using the AHP approach, 15 pairwise comparison questions are needed to calculate the relative weights of these criteria. However, when the SEM approach is used, there is no need to create such pairwise comparison matrices since the participants' responses are obtained directly through Likert scale items. Therefore, the SEM approach was preferred in this study in order to obtain the criteria weights in a more objective and practical way. Also, hypothetical values are used in this illustrative example to better demonstrate the SEM-AHP process. However, it is important to consider how such comparisons would be evaluated in real-world scenarios. In practical applications, pairwise comparisons between professors based on specific criteria could be conducted by academic decision-makers, such as department chairs, course scheduling coordinators, or curriculum committees.

4.4 Mathematical model

4.4.1 Sets

C = {1,2,….,n}: Set of regular courses
M = {1,2,….,m}: Set of metaverse courses
R: Set of rooms for regular courses
Rm: Set of rooms for metaverse courses
T: Set of available timeslots for both regular courses and metaverse courses
D = {1,2,….,l}: Set of days
P = {1,2,….,k}: Set of professors

4.4.2 Decision variables

- $\boldsymbol{x_{irtd}}$ is a binary variable representing whether course i is scheduled in room r at timeslot t on day d.
- $\boldsymbol{y_{ip}}$ is a binary variable representing whether course *i* is taught by professor *p*.
- $\boldsymbol{w_{iptd}}$ is a binary variable representing professor *p* teaches course *i* at timeslot *t* on day *d*.

4.4.3 Parameters

S(i): Shows the number of students/students enrolled in each course.
B(i, r): Binary parameter indicating eligibility of rooms for metaverse courses.
C(r): Shows the capacity of each room.
A(r, t, d): Binary parameter indicating room availability for metaverse courses.
Z(i, p): Binary parameter indicating the eligibility of professors for teaching courses.
w(i, p): The weight reflects the intention to use metaverse technology. These weights were derived based on the previously conducted SEM and were calculated based on data to measure the intention of medical physicians to use metaverse-based applications.

4.4.4 A binary integer programming formulation

$$Max\ Z = \sum_{i \in M} \sum_{p \in P} y_{ip} * w(i,p) \tag{10}$$

$$\sum_{r\in R\cup Rm}\sum_{t\in T}\sum_{d\in D} x_{irtd} = 1 \quad \forall i \in C \cup M \tag{11}$$

$$\sum_{i\in C\cup M} x_{irtd} \leq 1 \quad \forall r \in R \cup Rm, \forall t \in T, \ \forall d \in D \tag{12}$$

$$\sum_{i\in C\cup M} y_{ip} \geq 1 \quad \forall p \in P \tag{13}$$

$$\sum_{p\in P} y_{ip} = 1 \quad \forall i \in C \cup M \tag{14}$$

$$\sum_{i\in C\cup M} w_{iptd} \leq 1 \quad \forall p \in P, \forall t \in T, \qquad \forall d \in D \tag{15}$$

$$w_{iptd} \leq \sum_{r\in R\cup R_m} x_{irtd} \ \forall\, i \in C \cup M, \forall p \ \in P, \forall t \in T, \forall d \in D \tag{16}$$

$$w_{iptd} \leq \ y_{ip} \ \forall\, i \in C \cup M, \forall p \ \in P, \forall t \in T, \forall d \in D \tag{17}$$

$$w_{iptd} \geq \sum_{r\in R\cup R_m} x_{irtd} + y_{ip} - 1 \ \forall\, i \in C \cup M, \forall p \ \in P, \forall t \in T, \forall d \in D \tag{18}$$

$$\sum_{t\in T}\sum_{d\in D} x_{irtd} = B(i,r) \ \ \forall i \in M, \ \ \forall r \in Rm \tag{19}$$

$$\sum_{d\in D} x_{irtd} * S(i) \leq C(r) * B(i,r) \ \ \forall i \in M, \ \ \forall r \in Rm, \qquad \forall t \in T \tag{20}$$

$$\sum_{d\in D} x_{irtd} * S(i) \leq C(r) \quad \forall i \in C, \ \ \forall r \in R, \qquad \forall t \in T \tag{21}$$

$$\sum_{t\in T}\sum_{d\in D} x_{irtd} = 0 \ \ \forall i \in C, \ \ \forall r \in Rm \tag{22}$$

$$\sum_{t\in T}\sum_{d\in D} x_{irtd} = 0 \ \ \forall i \in M, \ \ \forall r \in R \tag{23}$$

$$\sum_{i\in M} x_{irtd} \leq A(r,t,d) \ \ \forall r \in Rm, \qquad \forall t \in T, \ \ \forall d \in D \tag{24}$$

$$y_{ip} \leq Z(i,p) \ \forall i \in C \cup M, \forall p \in P \tag{25}$$

$$x_{irtd}, y_{ip}, w_{iptd} \text{ binary } \forall i \in C \cup M, \forall r \in R \cup Rm, \forall t \in T, \forall d \in D, \forall p \in P \quad (26)$$

The objective function (10) aims to prioritize the assignment of metaverse courses to professors with a higher weight, in order to ensure that these courses are conducted in the most efficient way. Constraint (11) ensures that each course, whether regular or metaverse-based, is scheduled exactly once in a specific room, time slot, and day. Constraint (12) prohibits a course from being assigned to multiple rooms simultaneously during the same time slot and day. Constraint (13) guarantees that each professor is assigned to teach at least one course in the overall schedule. Constraint (14) enforces that each course is assigned to exactly one professor. Constraint (15), (16), (17), and (18) prevents any professor from being assigned to two overlapping courses within the same time and day. Constraint (19) ensures that metaverse courses are only scheduled in metaverse-eligible rooms. Constraint (20) restricts the total number of students in each metaverse-designated room from exceeding its capacity, considering the student size of each scheduled metaverse course. Constraint (21) applies the same room capacity restriction to regular courses. Constraint (22) prevents regular courses from being scheduled in metaverse-designated rooms. Conversely, constraint (23) prohibits metaverse courses from being assigned to rooms for regular courses. Constraint (24) enforces the use of only time slots and days during which metaverse-designated rooms are available, based on predefined availability. Constraint (25) ensures that professors are only assigned to courses for which they are eligible, according to subject expertise or institutional criteria. Finally, constraint (26) defines all decision variables as binary.

4.5 Solution methods

4.5.1 Greedy reassignment and assignment for professor equity (GRAPE)

A greedy algorithm, referred to as GRAPE, has been developed to generate an initial feasible solution to the UCTTP. The algorithm efficiently allocates both metaverse and regular courses while also ensuring a fair distribution of workload among professors. The algorithm works in two main phases: assignment and reassignment. In the first assignment phase, metaverse courses are prioritized, and professors are assigned based on the weights derived from the AHP. For each metaverse course, the algorithm selects the professor with the highest available weight who is both eligible and available during the relevant time period, and assigns the course to this professor along with an appropriate room. Once all metaverse courses are assigned, the algorithm proceeds to assign regular courses to ensure that each professor is assigned to at least one course. After this step, GRAPE proceeds to the reassignment phase to identify any unassigned professors. In this phase, the algorithm locates professors with multiple course assignments and checks whether one of their courses can be reassigned to an unassigned professor. During this process, all eligibility, availability, and capacity constraints are taken into account. If direct reassignment is not possible, the algorithm attempts to relocate courses by transferring assignments among eligible professors to balance the workload. The pseudo code of greedy algorithm is given below.

Set of regular courses C
Set of metaverse courses M
Set of rooms R for regular courses
Set of rooms R_m for metaverse courses
Set of timeslots T
Set of days D
Set of professors P

Eligibility matrix Z: Z[c, p] = 1 if professor p is eligible for course c
Room capacity $C_{capacity}$ (r) for R, $C_{m\_capacity}$ (r_m) for R_m
Course sizes S(c) for regular courses, S_m(m) for metaverse courses
Room availability matrix A(r_m, t, d) for metaverse courses, for timeslot t, day d
Professor weights w_p for metaverse courses (AHP-based)
Room availability matrix $room_{availability(r, t, d)}$
Professor availability matrix P_{avail}(p, t, d)

Step 1: Initialize schedule and availability:
room_availability$_{(r, t, d)}$ = 1 for all r in R, R_m, t in T, d in D
P_{avail} (p, t, d) = 1 for all p in P, t in T, d in D
Sched = {}
professor_assignment_count(p) = 0 for all p in P
Step 2: Assign metaverse courses by professor weight:
Sort professors by weight:
Sorted_Professors = sort(P, w_p, descending)
For each metaverse course m in M:
assigned = False
For each professor p in Sorted_Professors:
If Z[m, p] = 1 (professor is eligible) and P_{avail} (p, t, d) = 1:
For each timeslot t ∈ T and day d ∈ D:
If room $r_m \in R_m$ is eligible (B[m, r_m] = 1) and available ($room_{availability(r, t, d)}$ = 1 and A(r_m, t, d) = 1) and $C_{m\_capacity}(r_m) >= S_m(m)$:
Assign Sched[m] = (r_m, t, d, p)
Update availability:
P_{avail}(p, t, d) = 0, $room_{availability(r, t, d)}$ = 0
Increment professor_assignment_count(p) += 1
Break loops when assigned
Step 3: Ensure every professor has at least one course:
Sort professors by eligibility count:
Sorted_Professors_Z = sort(P, sum(Z[c, p] for c ∈ C + M), ascending)
For each professor p ∈ Sorted_Professors_Z:
If professor_assignment_count(p) = 0:
For each regular course c ∈ C:
If Z[c, p] = 1 and c is unassigned:
For each timeslot t ∈ T and day d ∈ D:
If P_{avail}(p, t, d) = 1 and room r ∈ R is available ($room_{availability(r, t, d)}$ = 1) and $C_{capacity}(r) >= S(c)$:
Assign Sched[c] = (r, t, d, p)
Update availability:
P_{avail}(p, t, d) = 0, $room_{availability(r, t, d)}$ = 0
Increment professor_assignment_count(p) += 1
Break loops when assigned.
Step 4: Reassign courses if needed:
For each professor p ∈ P with no courses assigned (professor_assignment_count(p) = 0):
Check regular courses first:
For each assigned professor $p_{assigned}$ with professor_assignment_count($p_{assigned}$) > 1:
For each course c ∈ C assigned to $p_{assigned}$:
If Z[c, p] = 1 and P_{avail}(p, t, d) = 1, reassign Sched[c] to p:
Update professor_assignment_count($p_{assigned}$) -= 1, professor_assignment_count(p) += 1
Update availability:
$P_{avail}(p_{assigned}, t, d) = 1$, P_{avail} (p, t, d) = 0
If no regular course is found, check metaverse courses:
Same reassignment logic for metaverse courses.
Step 5: Handle unassigned professors:
For each unassigned professor p ∈ P (where professor_assignment_count(p) = 0):

Loop through all courses $c \in C + M$:
Check if course c is assigned to another professor p_assigned:
If Z[c, p] = 1 (professor p is eligible for course c):
Check availability:
If professor p is available during the assigned timeslot t and day d for course c:
Relocate the course c from $p_{assigned}$ to p:
Sched[c] = (r, t, d, p)
Update availability:
$P_{avail}(p_{assigned}, t, d) = 1$, $P_{avail}(p, t, d) = 0$
Update assignment counts:
professor_assignment_count($p_{assigned}$) -= 1, professor_assignment_count(p) += 1
Mark the course as successfully relocated and break the loop

Continue this procedure until all unassigned professors p have at least one course assigned, or until no more relocations are possible.

4.5.2 Simulated annealing (SA) algorithm

SA a popular technique for finding global or near-global optima of complex cost functions. It is widely recognized for its rapid convergence behavior and straightforward implementation (He et al., 2014). Its robustness lies in its ability to escape local optima by probabilistically accepting worse solutions based on an acceptance probability $e^{\frac{-\Delta\theta}{T}}$, where Δ is the difference in objective function values between the current and candidate solutions, and T is a temperature control parameter. The algorithm models the objective function as an energy metric, which is iteratively minimized through a controlled reduction in temperature, mirroring the behavior of physical annealing processes (Şahin et al., 2020). When T is high, the algorithm is more exploratory, accepting most moves (better or worse). As T decreases, the algorithm becomes more exploitative, rejecting worse moves more frequently. To prevent premature convergence to a local minimum, the algorithm starts with a relatively high T.

In the context of this problem, the initial temperature is defined as a function of the objective function value of the initial schedule, making the starting temperature adaptive and reasonable. The SA algorithm then undergoes k temperature reductions according to the cooling function $T_k = \alpha T_{k-1}$, where α is the cooling rate. At each temperature level, the algorithm explores the neighborhood of the current solution to identify better or potentially acceptable solutions.

The logic behind the neighborhood search in this implementation is to avoid getting trapped in local optima. A swap neighborhood mechanism is used to perturb the current solution locally. The new neighbor is created by changing the professor assignments of two randomly selected courses. This ensures that constraints, such as workload limits and course coverage, are respected. If a metaverse course is involved, the objective value can improve by assigning it to a professor with a higher weight. Otherwise, the swap might still be accepted based on the acceptance probability, allowing exploration of less favorable areas in the solution space.

The proposed SA algorithm starts with an initial feasible schedule generated externally and iteratively improves upon it. At each temperature level, it evaluates the quality of candidate schedules using the objective function**,** which is defined as the sum of the weights of professors teaching metaverse courses. The schedule with the highest objective value encountered during the process is recorded as the best solution. By integrating constraints and using the swap neighborhood, the algorithm effectively balances feasibility with optimization.

T_initial_coeff: Initial temperature coefficient
α: Cooling rate
k_max: Maximum number of iterations
k_min: Minimum number of iterations per temperature

1. Start with an initial feasible schedule (S_current)
2. Compute the objective value (θ_current) for S_current
3. Set S_best = S_current and θ_best = θ_current
4. Initialize the temperature (T) as T_initial_coeff × θ_current
5. For Each Inner Iteration (k_min):
    6. Generate a new feasible schedule (S_new) using constraint-guided neighborhood generation
    7. Compute the objective value (θ_new) for S_new
    8. Calculate the change in objective value (Δθ = θ_current - θ_new)
    9. Decide whether to accept S_new:
        If Δθ < 0 or $rand[0,1] < e^{\frac{-\Delta\theta}{T}}$, accept S_new
    10. If S_new is accepted:
        Update S_current = S_new
        Update θ_current = θ_new
        If O_current > θ_best:
            Update S_best = S_current
            Update θ_best = θ_current
11. Update the temperature: T = T × α
12. Output the optimized schedule (S_best) and the final objective value (θ_best)

5 Computational study

The computational study conducted for the heuristic algorithms aims to evaluate the performance of the proposed methods across a wide range of problem sizes. The effectiveness and efficiency of the GRAPE and SA were tested using three different values for the number of regular courses (n = 20, 41, 83) and metaverse courses (m = 7, 14, 28), along with three different values for the number of professors. For each combination of n and m, five instances were generated, resulting in a total of 45 problem instances. The proposed algorithms were implemented in Python and executed on the Kaggle platform, which provides cloud-based computing resources for reproducible experimentation.

5.1 Data generation

To evaluate the performance of the proposed course scheduling problem, synthetic datasets of different sizes are generated. The data generation process was designed taking into account the number of courses, rooms and other parameters specific to the problem. The datasets were categorized into three scales: small, medium and large.

The small dataset is the basis for obtaining larger datasets. The following steps were followed to create the dataset:

1. The total number of courses was randomly selected from a uniform distribution between 20 and 30.
2. A binomial distribution with a success probability of 0.2 was used to determine the number of metaverse courses. The remaining courses were designated as regular course sets (C).
3. The total number of rooms is based on the total number of courses, with a proportional relationship. The number of rooms was calculated as approximately one fifth of the total number of courses and rounded to the nearest whole number.
4. The number of rooms for metaverse courses was determined using a binomial distribution with a success probability of 0.2. At least one room was ensured for each metaverse course. The remaining rooms were designated as room sets for regular courses (R).

5. The number of timeslots was fixed at 9 and number of days was fixed at 5.
6. The enrollment numbers for each course (S(i)) are assumed to follow a normal distribution around a mean value. This reflects the tendency for most courses to have enrolments close to the mean; there are fewer courses with very high or very low enrolments.
7. Room capacities (C(r)) are also assumed to follow a normal distribution. This assumption implies that most rooms have capacities close to the average, with a smaller number of rooms having significantly higher or lower capacities.
8. Eligibility of rooms (B(i, r)), room availability (A(r, t, d)), and eligibility of professors (Z(i, p)) are treated as binary variables (0 or 1). Random samples of these binary variables are generated to reflect the eligibility and availability of rooms and professors.

Finally, all numerical values in the small dataset were approximately doubled to create medium-sized datasets. To create large datasets, all numerical values in the medium dataset were approximately doubled**.**

5.2 SA parameter setting

The efficiency and effectiveness of the SA algorithm are significantly influenced by the selection of its parameter values. The Taguchi Design of Experiments (DoE) method was used to determine the optimal configuration for these parameters. This method allows a systematic analysis of the impact of each parameter and their interactions on performance metrics.

In this study, four key parameters of the SA algorithm were considered: the initial temperature coefficient, the cooling rate, the maximum number of iterations, and the number of iterations per temperature level. For each parameter, two levels were defined as follows: initial temperature coefficient ($k = 1, 10$), cooling rate $\alpha = 0.99, 0.999$), maximum number of iterations ($t_{max} = 1000, 2000$), and minimum number of iterations per temperature level ($i_{max} = 10, 20$). The levels of the two-level factorial design are shown in Table 3. The parameter space is explored using the L16 orthogonal array, which allows both main and interaction effects to be investigated efficiently. The experiments were performed on a large number of problem instances with different scales (e.g., $n = 27, k = 23; n = 27, k = 25; n = 55, k = 55$; and $n = 111, k = 103$). Each configuration was evaluated in terms of the average relative error (AVE) as the primary response variable. This approach aims to identify the most effective parameter combinations.

The results of the Taguchi design analysis revealed an excellent model fit, with an R^2 value of 0.998, indicating that the selected parameters explained almost all of the variability in the AVE. A significant interaction between the cooling rate and the maximum number of iterations was observed ($p = 0.025$), suggesting that the effect of cooling rate on the solution quality was dependent on the iteration level. Other interactions, including those involving the initial temperature coefficient and the minimum number of iterations, were found to be statistically insignificant.

Following the analysis, the configuration producing the most beneficial performance in terms of both solution quality and computational time was found to be: an initial temperature coefficient of $k = 1$, a cooling rate of $\alpha = 0.99$, a maximum number of iterations set to $t_{max} = 1000$, and a minimum of $i_{max} = 10$ iterations per temperature level. These settings were adopted for all subsequent computational experiments using the SA algorithm in this study.

Table 3 Factor levels' coded values

Level	Coded	Parameter

		k	α	t_{max}	i_{max}
Low	-1	1	0.99	1000	10
High	+1	10	0.999	2000	20

5.3 Experimental results

The performance of the GRAPE and SA algorithms was evaluated in terms of the average relative error and the computational time of the algorithms. The relative error was calculated for each test problem using Eq. (27).

$$Relatiive\ Error_{GRAPE} = \frac{Z^* - Z_{GRAPE}}{Z^*},\ \ Relatiive\ Error_{SA} = \frac{Z^* - Z_{SA}}{Z^*} \tag{27}$$

where Z^* is the optimal objective function value, Z_{GRAPE} is the objective function value obtained by the proposed greedy algorithm, and Z_{SA} is the objective function value produced by the Simulated Annealing algorithm for a test problem.

The optimal solutions are obtained using the binary integer programming formulation presented in subsection 4.4.4. The average relative errors and average computational times for GRAPE and SA are calculated by averaging 5 instances with different random seeds for each data set. The results in Table 4 show that GRAPE outperforms SA in terms of average computational times. However, SA method outperforms GRAPE in terms of both lower average relative error values and frequency of reaching the optimal solution. This becomes more evident especially in datasets where the problem size increases (e.g. n = 83, m = 28, k = 111). Another remarkable finding is that both heuristics provide significant advantages in terms of computational time compared to optimal solutions. While it may take hours to obtain optimal solutions in large data sets, solutions can be produced in seconds with GRAPE and SA. It is also observed that GRAPE produces high relative error values in some cases (e.g. $Average\ Relative\ Error_{GRAPE}$ = 0.457), indicating that GRAPE tends to deviate from the optimality at larger problem sizes, but may still be suitable for obtaining a fast initial solution due to its low computational cost. In general, the SA algorithm can be considered as a more reliable approach in terms of solution quality and closeness to the optimum solution, while the GRAPE algorithm has the potential to produce low-cost initial solutions.

Table 4 Results of the solution methods

n	*m*	*k*	Average relative error		# of optimal solutions		Average computational time (s)		
			GRAPE	SA	GRAPE	SA	Optimal	GRAPE	SA
20	7	17	0.000	0.000	5	5	15.832	0.001	1.617
		22	0.070	0.003	2	4	23.935	0.001	1.208
		27	0.287	0.000	0	5	48.316	0.001	0.860
41	14	46	0.030	0.010	2	2	273.726	0.002	2.397

		50	0.186	0.045	0	0	295.417	0.003	2.400
		55	0.335	0.009	0	0	464.415	0.016	2.189
83	28	93	0.072	0.028	1	1	2769.207	0.024	5.896
		101	0.222	0.028	0	0	2570.479	0.022	5.379
		111	0.457	0.021	0	0	4513.282	0.035	4.954

6 Discussion

The solution of the UCTTP is becoming increasingly difficult due to factors such as the specific needs of each educational institution and the increasing number of courses, lecturers and classrooms. Models developed for solving these problems sometimes take a very long time and sometimes become impossible to solve in large-scale problems where a large number of variables and constraints need to be considered together. Therefore, alternative solution methods such as heuristic and metaheuristic algorithms are employed to reduce computation time and obtain feasible results. Moreover, the integration of emerging technologies such as the metaverse into course content further complicates the timetabling problem not only from a computational perspective but also in terms of pedagogical and technological alignment. In this study, both of these issues are taken into account by specifically addressing the problem of course scheduling in the context of medical education, where the adoption of metaverse technologies is expected to become increasingly widespread. To this end, a more comprehensive and context-sensitive scheduling framework is created by integrating the behavioral intentions of medical educators towards the use of metaverse into the model.

The computational study conducted in this research is designed to evaluate the effectiveness of the proposed GRAPE and SA in solving the integrated UCTTP that includes both regular and metaverse courses. The performance of each algorithm is evaluated with synthetic datasets of different sizes and the results are evaluated in terms of average relative error, number of optimal solutions reached and computation time. The results reveal that SA achieves lower relative error values compared to GRAPE, especially for medium and large-scale samples. For example, on a dataset with $n = 83$, $m = 28$ and $k = 111$, SA achieved an AVE of 0.021 compared to GRAPE's significantly higher error of 0.457. This performance trend confirms that SA is more effective in navigating complex solution spaces and avoiding sub-optimal local optima. This finding is in line with previous studies (e.g., Sylejmani et al., 2023; Junn et al., 2017) where SA shows robustness on highly constrained scheduling problems. Moreover, SA's ability to maintain low AVE values at different scales emphasizes its scalability and reliability for large datasets in combinatorial scheduling contexts. This finding is also consistent with the results of the bibliometric analysis, which identified SA as one of the most frequently applied metaheuristic techniques in the literature. In contrast, GRAPE has produced almost instantaneous solutions, even for large problems, with computation times often below 0.05 seconds compared to SA's runtime of a few seconds. While this supports the use of the GRAPE for generating fast initial solutions, it seems that the tendency to deviate from optimality is greater as the problem size increases. However, the use of the GRAPE as a suitable initial solution generator for metaheuristic approaches is also widely supported in the literature (e.g., Laguardia and Flores, 2022).

Taguchi Design of Experiments was used to determine the optimal combination of SA algorithm parameters. The model exhibited a very high coefficient of determination (R^2 = 0.998), indicating that the selected parameters explain almost all the variance in AVE. A statistically significant interaction effect was found between the cooling rate and the maximum number of iterations, indicating that these parameters jointly influence the solution quality. This finding supports previous research on adaptive metaheuristics where parameter tuning is critical to optimize algorithmic performance (e.g., Alhuniti et al., 2020; Kiefer et al., 2023). These results also highlight the importance of integrating experimental design techniques into optimization processes.

An important contribution of the computational analysis is the demonstration of the feasibility and effectiveness of integrating the professor weights obtained from the SEM through AHP. This integration results in more realistic scheduling outcomes by prioritizing professors with a higher intention to teach in metaverse environments. The objective function that maximizes assignments based on weights represents a methodological innovation not found in previous scheduling studies by directly linking the empirical findings of behavioral intention modeling with optimization results.

These findings support the main goal of the study by confirming that integrating behavioral intention modeling into the scheduling process yields more informed and effective course allocation decisions, especially for emerging metaverse-based applications in medical education.

6.1 Theoretical implications

This work advances the theoretical landscape by combining combinatorial optimization with human-centered behavioral factors. The integration of SEM-derived constructs as criteria in an AHP framework and their subsequent inclusion as parameters in a binary integer programming model represents a significant methodological extension. In this way, the study not only addresses operational efficiency but also takes into account psychological variables that measure individuals' intentions to use technology. The combined consideration of this psychological and operational approach will be noteworthy given institutional trends towards immersive learning technologies that are increasingly used in medical education. Previous programming literature has largely ignored the inclusion of such latent psychological constructs in optimization problems. This work is thus intended to provide a foundation for future models that aim to incorporate faculty or student behavioral characteristics into scheduling or resource allocation mechanisms.

6.2 Practical implications

In terms of the practical contributions of the study, the proposed model offers significant benefits, especially for academic institutions that want to integrate technology-based courses such as metaverse into their curricula. The developed course assignment model guides institutions aiming to incorporate such innovative courses into their curricula. It enables the curriculum to be adapted to modern teaching approaches, especially metaverse-based learning processes. The research findings make it possible to dynamically assign metaverse courses to faculty members with a high intention to use these technologies, thus ensuring that those who are most pedagogically and psychologically prepared to teach these courses are included. This contributes to more effective use of immersive technologies and reduces resistance to the adoption of new systems. Moreover, the hybrid scheduling model has the flexibility to address both traditional constraints and emerging technological limitations such as special classroom availability, simulation lab planning, and virtual or augmented reality hardware. In this respect, it provides a powerful and applicable tool for decision-makers in resource-limited educational settings. Finally, the computational results also provide important practical insights for academic administrators. In cases where solution quality is a priority, SA produces more effective results, while in time-critical scenarios, GRAPE can be considered an effective option for fast initial assignments.

6.3 Limitations and future work

The study has some limitations. First, the weights used in the objective function of the optimization model are derived from a specific SEM study and may not be generalizable across different institutional or cultural contexts. Second, while using synthetic data sets provides control over the general timetabling model, real-world data sets representing various characteristics of institutions may impose additional constraints not accounted for here. Third, although the model prioritizes course allocations based on weights, it does not currently integrate student preferences or different constraints at the department level. Finally, while using the Taguchi design provides efficient parameter tuning, other experimental design approaches could be explored for better generalization or a comparative analysis could be conducted.

Future work could extend this study in several directions. First, in addition to professor weights, students' intentions to use metaverse technologies could be incorporated into the model, enabling a more user-centered course planning framework. Second, future studies could explore alternative objective functions to address diverse institutional priorities. For instance, multi-objective optimization models could be developed to balance goals such as maximizing student participation in metaverse courses and ensuring equitable course distribution among faculty members. Third, the weights derived from the SEM model could be dynamically updated based on faculty development programs or increasing technological proficiency, allowing for more adaptive scheduling systems. Finally, validating the model with real-world institutional data would offer deeper insights into its practical effectiveness and its applicability across various educational contexts.

7 Conclusion

The field of medical education has advanced significantly with the integration of emerging technologies. While educational strategies continue to evolve in response to global developments, one of the most significant innovations in recent years has been the emergence of the metaverse. This technology has started a new era in medical education by providing immersive and interactive learning experiences. These developments also necessitate the reorganization of course scheduling processes within medical faculties.

In this study, a binary integer linear programming model is developed for the UCTPP, in which metaverse and regular courses are considered together. Furthermore, various solution methods are proposed to solve the model. The proposed method consists of multiple stages. First, based on Damar and Koksalmis (2023), the constructs from the SEM model are incorporated into the AHP model as criteria, and the resulting weights are integrated as parameters into the objective function of the optimization model. These weights represent professors' intentions to use metaverse technologies, and the model aims to prioritize the assignment of professors with high intention levels to metaverse courses. Subsequently, a GRAPE is developed to solve the scheduling problem, and this algorithm is used as the initial solution for the SA method. To enhance the efficiency of the SA algorithm, optimal parameter values are determined using the Taguchi design of experiments.

The current model assumes that pairwise comparisons in the AHP stage are conducted by a single decision maker. However, in institutional settings, these evaluations may involve multiple decision makers. Future implementations could incorporate group decision-making mechanisms that support collaborative input from faculty committees or administrative managers. Additionally, the model could be extended to include fuzzy AHP, which allows for the representation of uncertainty and ambiguity in preference judgments. Such enhancements would improve the realism, flexibility, and institutional applicability of the decision-making process within the model.

The solution of UCTTP becomes increasingly complex as the problem size and the variety of institutional requirements grow. This is supported by the experimental findings obtained in this study. An analysis of the solution table reveals that while optimal solutions can be achieved within the defined constraints, solution times increase significantly as the problem size increases. The experimental results also highlight the importance of the proposed heuristic algorithms in addressing this complexity. While the GRAPE is highly effective in generating rapid initial solutions, the SA algorithm significantly enhances solution quality and approaches optimality, particularly in larger and more constrained instances. Together, these two methods offer both computational efficiency and high-quality outcomes, making them a practical and scalable solution for real-world academic scheduling problems. As immersive technologies such as the metaverse become increasingly integrated into medical education, the ability to design adaptive, human-centered, and institutionally feasible programs will become even more critical. The proposed framework presents a proactive and practical roadmap for meeting these emerging needs.

Declarations

Availability of data and materials

The all data generated and analyzed during the current study are available from the authors upon reasonable request.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Acknowledgments

Not applicable.

Ethics declarations

Conflict of interest

The authors declare that there are no interest conflicts to disclose.

Ethics approval and consent to participate

Not applicable.

Consent for publication

The authors have read and approved the manuscript in its current version.

References

Abdullah, S., Turabieh, H., McCollum, B., & McMullan, P. (2012). A hybrid metaheuristic approach to the university course timetabling problem. *Journal of Heuristics, 18*, 1-23. https://doi.org/https://doi.org/10.1007/s10732-010-9154-y

Al-Betar, M. A., & Khader, A. T. (2012). A harmony search algorithm for university course timetabling. *Annals of Operations Research, 194*, 3-31. https://doi.org/https://doi.org/10.1007/s10479-010-0769-z

Al-Elq, A. H. (2010). Simulation-based medical teaching and learning. *Journal of family and Community Medicine, 17*(1), 35-40. https://doi.org/https://doi.org/10.4103/1319-1683.68787

Alhuniti, O., Ghnemat, R., & El-Seoud, M. S. A. (2020). Smart university scheduling using genetic algorithms. Proceedings of the 9th International Conference on Software and Information Engineering,

Badoni, R. P., Sahoo, J., Srivastava, S., Mann, M., Gupta, D., Verma, S., Stanimirović, P. S., Kazakovtsev, L. A., & Karabašević, D. (2023). An exploration and exploitation-based metaheuristic approach for university course timetabling problems. *Axioms, 12*(8), 720. https://doi.org/https://doi.org/10.3390/axioms12080720

Bellio, R., Ceschia, S., Di Gaspero, L., Schaerf, A., & Urli, T. (2016). Feature-based tuning of simulated annealing applied to the curriculum-based course timetabling problem. *Computers & Operations Research, 65*, 83-92. https://doi.org/https://doi.org/10.1016/j.cor.2015.07.002

Bolaji, A. L. a., Khader, A. T., Al-Betar, M. A., & Awadallah, M. A. (2014). University course timetabling using hybridized artificial bee colony with hill climbing optimizer. *Journal of computational science, 5*(5), 809-818. https://doi.org/https://doi.org/10.1016/j.jocs.2014.04.002

Bowen, K., Barry, M., Jowell, A., Maddah, D., & Alami, N. H. (2021). Virtual Exchange in Global Health: an innovative educational approach to foster socially responsible overseas collaboration. *International Journal of Educational Technology in Higher Education, 18*, 1-11. https://doi.org/https://doi.org/10.1186/s41239-021-00266-x

Campos, N., Nogal, M., Caliz, C., & Juan, A. A. (2020). Simulation-based education involving online and on-campus models in different European universities. *International Journal of Educational Technology in Higher Education, 17*, 1-15. https://doi.org/https://doi.org/10.1186/s41239-020-0181-y

Damar, S., & Koksalmis, G. H. (2023). Investigating the influence of technology anxiety on Healthcare Metaverse Adoption. In M. A. Al-Sharafi, M. Al-Emran, G. W. Tan, & K. Ooi (Eds.), *Current and Future Trends on Intelligent Technology Adoption* (Vol. 1, pp. 85-99). Springer. https://doi.org/https://doi.org/10.1007/978-3-031-48397-4_5

Damar, S., & Koksalmis, G. H. (2024). A bibliometric analysis of metaverse technologies in healthcare services. *Service Business, 18*(2), 223-254. https://doi.org/https://doi.org/10.1007/s11628-024-00553-3

De Causmaecker, P., Demeester, P., & Vanden Berghe, G. (2009). A decomposed metaheuristic approach for a real-world university timetabling problem. *European Journal of Operational Research, 195*(1), 307-318. https://doi.org/https://doi.org/10.1016/j.ejor.2008.01.043
Deris, S., Omatu, S., & Ohta, H. (2000). Timetable planning using the constraint-based reasoning. *Computers & Operations Research, 27*(9), 819-840. https://doi.org/https://doi.org/10.1016/S0305-0548(99)00051-9
Farrukh, K. (2024). Metaverse in medical education: a paradigm shift. *Pakistan Journal of Medical Sciences, 40*(1), 255. https://doi.org/https://doi.org/10.12669/pjms.40.1.8752
He, Z., Wang, N., & Li, P. (2014). Simulated annealing for financing cost distribution based project payment scheduling from a joint perspective. *Annals of Operations Research, 213*, 203–220. https://doi.org/10.1007/s10479-012-1155-9
Heidari, P., Arkat, J., & Mohsenpour, B. (2021). Course timetabling in medical universities given physicians' educational and clinical tasks. *Scientia Iranica, (),* -. https://doi.org/https://doi.org/10.24200/sci.2021.57410.5226
Hossain, S. I., Akhand, M., Shuvo, M., Siddique, N., & Adeli, H. (2019). Optimization of university course scheduling problem using particle swarm optimization with selective search. *Expert systems with applications, 127*, 9-24. https://doi.org/https://doi.org/10.1016/j.eswa.2019.02.026
Junn, K. Y., Obit, J. H., & Alfred, R. (2017). Comparison of simulated annealing and great deluge algorithms for university course timetabling problems (UCTP). *Advanced Science Letters, 23*(11), 11413-11417. https://doi.org/https://doi.org/10.1166/asl.2017.10295
Kiefer, A., Hartl, R. F., & Schnell, A. (2017). Adaptive large neighborhood search for the curriculum-based course timetabling problem. *Annals of Operations Research, 252*, 255-282. https://doi.org/https://doi.org/10.1007/s10479-016-2151-2
Laguardia, J. J., & Flores, J. A. (2022). University class schedule assignment by a Tabu search algorithm. 2022 8th International Engineering, Sciences and Technology Conference (IESTEC),
Lam, P. L., Ng, H. K., Tse, A. H., Lu, M., & Wong, B. Y. (2021). eLearning technology and the advancement of practical constructivist pedagogies: Illustrations from classroom observations. *Education and Information Technologies, 26*, 89-101. https://doi.org/https://doi.org/10.1007/s10639-020-10245-w
Lewis, K. O., Popov, V., & Fatima, S. S. (2024). From static web to metaverse: reinventing medical education in the post-pandemic era. *Annals of medicine, 56*(1), 2305694. https://doi.org/https://doi.org/10.1080/07853890.2024.2305694
Mahlous, A. R., & Mahlous, H. (2023). Student timetabling genetic algorithm accounting for student preferences. *PeerJ Computer Science, 9*, e1200. https://doi.org/https://doi.org/10.7717/peerj-cs.1200
Mergen, M., Will, L., Graf, N., & Meyerheim, M. (2025). Feasibility study on virtual reality-based training for skin cancer screening: Bridging the gap in dermatological education. *Education and Information Technologies, 30*(4), 5251-5282. https://doi.org/https://doi.org/10.1007/s10639-024-13019-w
Parker, R. G., & Rardin, R. L. (2014). *Discrete optimization*. Elsevier.
Prakasa, T. A. D., Muklason, A., & Premananda, I. G. A. (2024). Automating School Timetabling: An Intelligent System Application Using Simulated Annealing. 2024 International Seminar on Intelligent Technology and Its Applications (ISITIA),
Saaty, T. L. (1977). A scaling method for priorities in hierarchical structures. *Journal of mathematical psychology, 15*(3), 234-281. https://doi.org/https://doi.org/10.1016/0022-2496(77)90033-5

Saaty, T. L. (1984). The analytic hierarchy process: Decision making in complex environments. In *Quantitative assessment in arms control: mathematical modeling and simulation in the analysis of arms control problems* (pp. 285-308). Springer.
Saaty, T. L. (1990). How to make a decision: the analytic hierarchy process. *European Journal of Operational Research*, *48*(1), 9-26. https://doi.org/https://doi.org/10.1016/0377-2217(90)90057-I
Sakr, A., & Abdullah, T. (2024). Virtual, augmented reality and learning analytics impact on learners, and educators: A systematic review. *Education and Information Technologies*, *29*(15), 19913-19962. https://doi.org/https://doi.org/10.1007/s10639-024-12602-5
Sandrone, S. (2022). Medical education in the metaverse. *Nature Medicine*, *28*(12), 2456-2457. https://doi.org/https://doi.org/10.1038/s41591-022-02038-0
Song, T., Chen, M., Xu, Y., Wang, D., Song, X., & Tang, X. (2021). Competition-guided multi-neighborhood local search algorithm for the university course timetabling problem. *Applied Soft Computing*, *110*, 107624. https://doi.org/https://doi.org/10.1016/j.asoc.2021.107624
Sylejmani, K., Gashi, E., & Ymeri, A. (2023). Simulated annealing with penalization for university course timetabling. *Journal of Scheduling*, *26*(5), 497-517. https://doi.org/https://doi.org/10.1007/s10951-022-00747-5
Şahin, R., Niroomand, S., Durmaz, E. D., & Molla-Alizadeh-Zavardehi, S. (2020). Mathematical formulation and hybrid meta-heuristic solution approaches for dynamic single row facility layout problem. *Annals of Operations Research*, *295*(1), 313–336. https://doi.org/10.1007/s10479-020-03704-7
Tarawneh, H., & Ayob, M. (2013). Adaptive neighbourhoods structure selection mechanism in simulated annealing for solving university course timetabling problems. *Journal of Applied Sciences*, *13*(7), 1087-1093. https://doi.org/https://ui.adsabs.harvard.edu/link_gateway/2013JApSc..13.1087T/doi:10.3923/jas.2013.1087.1093
Thepphakorn, T., & Pongcharoen, P. (2020). Performance improvement strategies on Cuckoo Search algorithms for solving the university course timetabling problem. *Expert systems with applications*, *161*, 113732. https://doi.org/http://dx.doi.org/10.1016/j.eswa.2020.113732
Ullman, J. B., & Bentler, P. M. (2012). Structural equation modeling. In *Handbook of psychology, second edition* (Second ed., Vol. 2).
Venkatesan, M., Mohan, H., Ryan, J. R., Schürch, C. M., Nolan, G. P., Frakes, D. H., & Coskun, A. F. (2021). Virtual and augmented reality for biomedical applications. *Cell reports medicine*, *2*(7), 100348. https://doi.org/https://doi.org/10.1016/j.xcrm.2021.100348
Vianna, D. S., Martins, C. B., Lima, T. J., Vianna, M. d. F. D., & Meza, E. B. M. (2020). Hybrid VNS-TS heuristics for university course timetabling problem. *Brazilian Journal of Operations & Production Management*, *17*(2), 1-20. https://doi.org/https://doi.org/10.14488/BJOPM.2020.014
Wong, C. H., Goh, S. L., & Likoh, J. (2022). A genetic algorithm for the real-world university course timetabling problem. 2022 IEEE 18th international colloquium on signal processing & applications (CSPA),
Zhang, H. (2022). Structural equation modeling. In *Models and methods for management science* (pp. 363-381). Springer.