SURVEY

A Survey of Solution Methodologies for Exam Timetabling Problems

EMILY SING KIANG SIEW[ID][1], **SAN NAH SZE**[ID][1], **SAY LENG GOH**[ID][2],
GRAHAM KENDALL[3,4], **(Senior Member, IEEE), NASSER R. SABAR**[ID][5],
AND SALWANI ABDULLAH[ID][6]

[1]Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Kota Samarahan, Sarawak 94300, Malaysia
[2]Optimization and Visual Analytics Research Group, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Labuan International Campus, Labuan 87000, Malaysia
[3]School of Engineering and Computing, MILA University, Putra Nilai, Negeri Sembilan Darul Khusus 71800, Malaysia
[4]School of Computer Science, University of Nottingham, University Park, NG7 2RD Nottingham, U.K.
[5]Department of Computer Science and Information Technology, La Trobe University, Melbourne, VIC 3086, Australia
[6]Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor 43600, Malaysia

Corresponding authors: Emily Sing Kiang Siew (sskiang37@gmail.com), San Nah Sze (snsze@unimas.my), and Say Leng Goh (slgoh@ums.edu.my)

ABSTRACT Exam timetabling is a prominent topic in academic administration management as it ensures the effective utilization of resources and satisfies the requirements and preferences of stakeholders, which leads to a productive academic environment, contributing to the institution's overall success. Given the myriad of solution methodologies explored across diverse exam timetabling problems and constraints, both in studied benchmark datasets and real-life cases over the last decade, it is imperative to undertake a comprehensive survey. This survey paper aims to comprehensively describe the exam timetabling problem (ETP), including its variants, constraints, and benchmark dataset. We look at different methods to solve ETP problems from 2012 to 2023. These methods include mathematical optimization, heuristics, metaheuristics, hyper-heuristics, hybrid approaches, and matheuristics. Finally, we discuss the review findings and potential research directions. By doing so, we hope to facilitate a deeper understanding of ETP and offer valuable insights for future research.

INDEX TERMS Educational timetabling, exam scheduling, exam timetabling, solution methodologies.

I. INTRODUCTION

Educational timetabling represents a significant instance among challenging combinatorial optimization problems [89]. This intricate problem is conventionally classified into three principal types: school timetabling, course timetabling, and examination timetabling [1]. Each involves assigning events (e.g., meetings, exams, lectures, tutorials, classes) to limited resources (e.g., timeslots and rooms) while adhering to predefined constraints. Despite similarities, solving one problem type with a method does not guarantee success with another. Consequently, each problem type has been addressed

independently, and each instance may differ significantly in constraints and dimensions.

Exam timetabling problem (ETP) was defined by Qu et al. [2] as "assigning a set of exams into a limited number of timeslots (periods) and rooms (of certain capacity) subject to a set of constraints." Constraints can vary in definition or weighting based on institutional requirements. The scientific community has been studying the ETP since the 1960s, with an early survey conducted by [3] focusing on practical applications of graph coloring heuristics from 1964 to 1984. Carter and Laporte [4] extended the survey by categorizing algorithms into sequential, generalized search, cluster, and constraint-based methods, while Schaerf [1] reviewed the early approaches for resolving the ETP. Most algorithms primarily addressed the basic

---

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad J. Abdel-Rahman[ID].

timetabling problem with minimal constraints during that period.

The area of timetabling problems, including exam timetabling, has a thriving research community, with notable biannual conference series such as Practice and Theory of Automated Timetabling (PATAT) dedicated to timetabling practices and their applications. Exam timetabling constantly evolves, leading to rapid developments in theory and practice. With more standardized benchmark datasets, researchers are exploring ways to place the timetabling problem in a real-world context. Some studies focus on the optimality gap when solving benchmark instances, while others prioritize the time to find a feasible solution.

Many studies have investigated educational timetabling; however, the breadth of the field restricts each to addressing specific problem subsets. In 2008, Lewis [70] focused his study on metaheuristic techniques, while Pillay [72] focused on hyper-heuristics within educational timetabling in 2014. In 2020, Bashab et al. [73] specifically concentrated on university timetabling using metaheuristic techniques. In 2022, Ceschia et al. [74] focused on benchmarks and state-of-the-art results in educational timetabling. Numerous survey papers in educational timetabling have delved into specific subdomains, such as course timetabling [87] and school timetabling [88], each with its associated methods [83], [84], [85], [86], respectively.

For dedicated surveys focusing on exam timetabling, Qu et al. [2] conducted a comprehensive study from 1995 to 2008, discussing key research achievements and trends in exam timetabling, encompassing algorithmic strategies, benchmark datasets, and emerging challenges. The survey by Bashar et al. in 2019 [5] exclusively concentrates on one formulation of ETP, namely the Uncapacitated ETP. Notably, the need for dedicated surveys addressing the exam timetabling problem is even more pronounced when compared to other educational timetabling subdomains. This scarcity encompasses the analysis of benchmark datasets or specific methods and the investigation of real-world problem constraints and requirements. Examining existing literature is crucial to discern gaps in exam timetabling research, considering substantial advancements in problem understanding and solution methodologies over the past decade.

This survey is essential for bridging the gap between benchmark datasets introduced over a decade ago and recent real-world cases, providing valuable insights to researchers, educators, and policymakers to navigate the evolving landscape of exam timetabling challenges. The contributions of this survey paper are:

- We present an overview of the terminologies, problem descriptions, variants, and constraints related to the ETP. The review outlines the commonly used ETP benchmark datasets, detailing their respective characteristics and state-of-the-art methods.
- We provide an up-to-date overview of recent solution methodologies employed in ETPs, encompassing both

benchmark and real-world ETPs from 2012 to 2023. Furthermore, we have structured the article based on the methods and techniques in chronological order, aiding readers in comprehending the methodology timeline in the field.

- We categorize solution methods into six categories: mathematical optimization, heuristics, matheuristics, metaheuristics, hybrid approaches, and hyper-heuristics. The tables and figures in this article categorize these methods, facilitating researchers in selecting categories of interest for further study.
- We discuss the classification of the recent methodologies used in the ETP field and propose future research directions.

The remainder of the paper is structured as follows: Section II outlines the scope of the reviewed paper. Section III defines the problem variants and constraints. Section IV classifies and offers an overview of solution methodologies for ETPs. The categorization of these solution methodologies is discussed in Section V. Section VI highlights trends in benchmark ETP. Section VII discusses potential future directions. The paper concludes with Section VIII.

II. SURVEY SCOPE

We opted to initiate the search period in 2012 since it marked the commencement of widespread publications on exam timetabling problems, employing various solution methodologies. Bibliographic information was obtained from the Science Citation Index Expanded (SCIE) and Social Science Citation Index (SSCI) within the Web of Science Core Collection by Clarivate Analytics. The search involved looking for phrases related to exam timetabling in the topic field from 2012 to 2023 in the SSCI and SCIE databases. Additionally, further refinement of publications was carried out by selecting articles that prominently featured the search keywords in their front page elements, including the article title, abstract, author keywords, and year published. The search terms and research strategy utilized for exam timetabling were elaborated upon, encompassing the following criteria:

Article title: "exam timetabling" or "examination timetabling" or "university timetabling" or "exam timetabling problem" or "exam scheduling" or "examination scheduling" or

Abstract: "exam timetabling" or "examination timetabling" or

Keywords: "exam timetabling" or "examination timetabling" and

Year: "2012-2023"

The search results (last accessed: January 9, 2024) resulted in 1,955 publications meeting these criteria, which were subsequently retrieved for further analysis. Apart from that, publications from the journal OPSEARCH are also included. After excluding specific publication types such as dissertations, conference proceedings, surveys, reviews, book chapters, partial exam-related studies, and those with fewer

Journal	Paper
Annals of Operations Research	11
Computers and Operations Research	5
European Journal of Operational Research	5
Journal of Scheduling	5
Applied Soft Computing	4
Expert Systems with Applications	3
IEEE Transactions on Evolutionary Computation	3
Journal of the Operational Research Society	3
Applied Intelligence	2
Information Sciences	2
OPSEARCH	2
Turkish Journal of Electrical Engineering and Computer Sciences	2
Others	12
Total	59

than five pages, we obtained 59 journal articles, as listed in Table 1.

III. EXAMINATION TIMETABLING
A. PROBLEM DEFINITION
An ETP involves allocating a predefined set of exams, each with a known number of students, into specific timeslots within a designated exam session, adherence to a defined set of feasibility constraints (hard constraints), and optimization of quality based on specified criteria (soft constraints). Soft constraints can be breached, resulting in a penalty for the objective function.

B. TERMINOLOGIES
In this survey, we establish the meanings of several key terms within the exam timetabling domain, which will be consistently applied throughout this survey.

- *Exam*: To schedule exams, a specific period and room must be assigned for each one.
- *Exam session*: An exam session consists of the entire period during which exams occur or several periods over a defined time frame.
- *Timeslot/ Period*: The exam session spans a certain number of days, each of these days further segmented into distinct timeslots. Each combination of day and timeslot designates a specific period within the exam session.
- *Room*: Each room is assigned a designated capacity, indicating the maximum number of available student seats. Certain exams may necessitate one or multiple rooms, while others can be conducted using an online platform and do not require physical spaces.
- *Invigilator*: Also known as a proctor or supervisor, oversees students during exams.

C. ETP VARIANTS AND DATASETS
In the existing literature, two main variants of ETP are recognized: Uncapacitated and Capacitated.

- *Uncapacitated ETP* imposes no room capacity restrictions during scheduling and assumes there will consistently be a room with ample space for all participating students.

- *Capacitated ETP* entails considering limited room/slot capacity per period.
  - *Room-Capacitated ETP:* a capacity constraint is imposed on each period.
  - *Slot-Capacitated ETP:* a capacity constraint is imposed on available rooms, specifying the maximum seating capacity each room can accommodate during a given period.

For each variant, we present an overview of the problem, constraints, and objectives, along with the main characteristics of the respective benchmarks.

1) UNCAPACITATED ETP VARIANT
The Uncapacitated variant represents the traditional version of ETP introduced by Carter et al. [59]. It features a dataset comprising 13 real-world cases collected at universities, primarily in Canada, known as the Toronto dataset. Furthermore, the dataset was expanded by Bilgin et al. [69] to establish a Slot-Capacitated variant. This modified dataset enforces a maximum seating capacity for each timeslot, featuring three daily exam slots.

a: CONSTRAINTS
Consider one hard constraint for clash-free exams and one soft constraint for maximizing the distribution of conflicting exams.

b: OBJECTIVE FUNCTIONS
Minimize the sum of proximity costs per student, representing the distance between conflicting exams. Students taking exams with a time gap encounter varying proximity costs, determined by weights: 16 for no gap, 8 for one interval, 4 for two intervals, 2 for three intervals, and 1 for four intervals.

c: DATASETS
Accessible through http://www.cs.nott.ac.uk/ pszrq/data.htm, the dataset includes 13 instances, each with two plain text files–exams and student enrollments. Two versions exist for the five instances, varying in student count, enrollments, exams, and conflict density. A program with its source code for penalty computation can be found on the website. Table 2 illustrates the characteristics of the Toronto version 1 dataset, including the modified Toronto data in the last two columns.

2) ROOM-CAPACITATED ETP VARIANT
The 2nd International Timetabling Competition (ITC 2007) organizers provided 12 instances from British universities for track 1 exam timetabling [61] that were used to evaluate the results, per the competition regulations. This problem formulation effectively bridges the gap between research efforts and institutional requirements by introducing many practical aspects of real-world cases.

a: CONSTRAINTS
Consider five hard constraints: exam conflicts, room capacity, period length, precedence, and room exclusivity, as well as

TABLE 2. Features of Toronto Datasets; symbol definitions: IN - Instances, EX - Exams, ST - Students, ER - Enrolments, CD - Conflict Density, PD - Periods, DY - Days, CP - Capacities.

IN	EX	ST	ER	CD	PD	DY	CP
CAR-S-91	682	16925	56877	0.13	35	17	1550
CAR-F-92	543	18419	55522	0.14	32	12	2000
EAR-F-83	190	1125	8109	0.27	24	8	350
HEC-S-92	81	2823	10632	0.42	18	6	650
KFU-S-93	461	5349	25118	0.06	20	7	1955
LSE-F-91	381	2726	10918	0.06	18	6	635
PUR-S-93	2419	30032	120681	0.03	42	10	5000
RYE-F-92	486	11483	45051	0.07	23	8	2055
STA-F-83	139	611	5751	0.14	13	4	3024
TRE-S-92	261	4360	14901	0.18	23	10	655
UTA-S-92	622	21266	58979	0.13	35	12	2800
UTE-S-92	184	2750	11793	0.08	10	3	1240
YOR-F-83	181	941	6034	0.29	21	7	300

seven soft constraints: two exams in a row, mixed duration, undesired period, spread, frontload, undesired room, and two exams in a day.

b: OBJECTIVE FUNCTIONS

Minimize the penalty that measures the violation of the seven terms associated with the soft constraints. The weightings vary between instances and are specified in the input file.

c: DATASETS

Available for download at https://www.unitime.org/ITC2007/, the solver and its corresponding source code are also accessible. The dataset is also accessible at https://opthub.uniud.it/problem/timetabling/edutt/ett/itc-2007-ett, where solutions can be validated. Each instance is inputted with a text format specifying exam details, hard constraints, and institutional weightings. Table 3 illustrates the characteristics of the ITC 2007 datasets.

TABLE 3. Features of ITC 2007 Exam Track Datasets; symbol definitions: IN - Instances, EX - Exams, ST - Students, ER - Enrolments, CD - Conflict Density, PD - Periods, RM - Rooms, CP - Capacities.

IN	EX	ST	ER	CD	PD	RM	CP
EXAM1	607	7891	32380	0.05	54	7	802
EXAM2	870	12743	37379	0.01	40	49	4076
EXAM3	934	16439	61150	0.03	36	48	5212
EXAM4	273	5045	21740	0.15	21	1	1200
EXAM5	1018	9253	34196	0.01	42	3	2395
EXAM6	242	7909	18466	0.06	16	8	2050
EXAM7	1096	14676	45493	0.02	80	15	2530
EXAM8	598	7718	31374	0.05	80	8	922
EXAM9	169	655	2532	0.08	25	3	170
EXAM10	214	1577	7853	0.05	32	48	1914
EXAM11	934	16439	61150	0.03	26	40	4924
EXAM12	78	1653	3685	0.18	12	50	1525

3) SLOT-CAPACITATED ETP VARIANT

The dataset from Yeditepe University (Faculty of Engineering) in Turkey includes real-world problems spanning eight semesters across three consecutive years. Initially presented by Özcan and Ersoy [75], it underwent subsequent modifications by Bilgin et al. [69].

a: CONSTRAINTS

The problem includes two hard constraints (exam conflict and slot capacity) and one soft constraint (exam spread).

b: OBJECTIVE FUNCTIONS

Minimize the instances of students undergoing two consecutive exams on a single day.

c: DATASETS

The datasets, available on http://www.cs.nott.ac.uk/ pszajp/timetabling/exam/, consist of eight real-life instances that have been converted into the ITC 2007 format by Parkes and Özcan [60]. To align with the ITC 2007 formulation, a considerably large room with a high cost, though having no impact on solution cost [32], was introduced for all Yeditepe instances except one. Table 4 illustrates the characteristics of the Yeditepe datasets.

TABLE 4. Features of Yeditepe Datasets; symbol definitions: IN - Instances, EX - Exams, ST - Students, ER - Enrolments, CD - Conflict Density, PD - Periods, CP - Capacities.

IN	EX	ST	ER	CD	PD	CP
yue20011	126	559	3486	0.18	6	450
yue20012	141	591	3708	0.18	6	450
yue20013	26	234	447	0.25	2	150
yue20021	162	826	5755	0.18	7	550
yue20022	182	869	5687	0.17	7	550
yue20023	38	420	790	0.2	2	150
yue20031	174	1125	6714	0.15	6	550
yue20032	210	1185	6833	0.14	6	550

4) OTHER DATASETS

The other ETP datasets that are available to be accessed are listed in Table 5.

TABLE 5. Datasets for ETP with web links.

Datasets	Web link	Reference
Nottingham	http://www.cs.nott.ac.uk/~pszrq/data.htm	[62]
Melbourne	http://www.cs.nott.ac.uk/~pszrq/data.htm	[63]
Purdue	https://www.unitime.org/exam_datasets.php	[38]
UniudData	https://bitbucket.org/satt/examtimetablinguniuddata/	[79]

D. CONSTRAINTS

Table 6 lists primary constraints categorized into exam-related, period-related, and room-related constraints. It provides their appearances across three benchmark instances and other real-world cases surveyed in the literature. These constraints may be classified as either hard or soft, contingent on the requirements or preferences of the institutions.

Invigilator-related constraints have received relatively less attention in the reviewed papers. These constraints have yet to be incorporated into any benchmark dataset, attributed to the diverse and significantly varying nature of constraints across different universities and the preferences that can represent each university's unique challenges. Table 7 lists invigilator-related constraints from real-world problems in the surveyed literature.

TABLE 6. Common Constraints for ETP; symbol definitions: T - Toronto Benchmark, EX - ITC 2007 Exam Benchmark, Y - Yeditepe Benchmark, H - Hard Constraint, S - Soft Constraint.

Constraints	T	I	Y	References
Exam related constraints				
Each exam is allocated a clash-free timeslot.	H	H		[68], [76]
Only one exam can be taken by a student at any given time.		H	H	[8], [31], [37], [51], [52], [77]
Each course's exams should be scheduled only once and within a single time slot.				[36], [37], [43], [46], [52], [56], [77], [78]
Sets of exams are mandated to be held at the same timeslot.		H		[8], [46], [51], [56], [68]
Exams are scheduled based on priority value, determined by the exam size.		S		
The exam can only be assigned to an available room or timeslot.				[38], [51], [76]
No consecutive exam timeslots (e.g., taking two exams on the same day, taking three exams on the same day, having two exams back-to-back)		S	S	[8], [38], [43], [51], [52], [56], [68], [77], [78]
Period related constraint				
Exam duration must not exceed the designated timeslot.		H		[68]
Minimize the total number of days on which exams are held.		S		[68]
All exams should be scheduled within the specified exam session.				[8], [52], [68]
Enforces the assignment of exams to specified or preferred slots or days.				[51], [52], [56], [68], [76]
Spread student exams evenly across the exam session.	S	S		[8], [31], [36], [37], [77]
Exams with a mixed duration may be scheduled in the same time slot.		S		
Consider timeslot precedence (ordering) restrictions, e.g., scheduling Exam B after Exam A.		H		[76]
Daily exam scheduling is restricted to user-identified exam slots.				[37], [68]
Courses in a given semester should have different timeslots assigned for their exams.				[37], [43]
Room related constraints				
The total number of students taking the exam remains within the limits of the capacity of the room(s).		H	H	[8], [25], [31], [36], [38], [43], [46], [51], [52], [56], [68], [76]–[78]
Exam room allocation constraints, such as Exam A being assigned to Room 101.		H		
Certain exams must be scheduled in the preferred room or specific room type.				[38], [51], [52], [56], [76]
Each room accommodates only a single examination. No room sharing allowed.				[25], [31], [38], [43], [51], [52], [68], [77]
A room cannot be used during timeslots in which it is unavailable.				[38], [51]
Same-exam rooms should be located proximate to each other.				[31], [38], [43], [46]
An exam can be split across multiple rooms.				[31], [78]
Minimize exams with varying durations being held in the same room.		S		
Maximize utilization of the room capacity.				[38], [78]
The limit for splitting an exam into multiple rooms is not exceeded.				[38], [52], [56]

TABLE 7. Invigilator-related constraints.

Constraints	References
Invigilators should not oversee their exams.	[22]
Only lecturers can be chief invigilators, as per university policy.	[22]
Invigilators cannot have overlapping duties per timeslot.	[22], [66]
No invigilator can invigilate more than three exams per exam session.	[22]
Assign the necessary number of invigilators and appoint one chief for each room.	[22]
Invigilator workloads should be balanced across all invigilators.	[22], [66]
Invigilators must not have more than two consecutive duties.	[66]
Invigilators cannot be assigned additional exams during the timeslots when they are responsible for invigilator duties.	[66]

IV. SOLUTION METHODOLOGIES

This section delves into the solution methodologies employed in the context of the ETP in this survey paper, categorizing these methods into one or more of the six categories:

- **Mathematical Optimization (MO)**: also known as mathematical programming, involves optimizing a mathematical model. It is categorized into Linear Programming (LP), which deals with linear functions and integer variables, and Nonlinear Programming (NLP), which involves general functions with discrete and continuous variables. This category encompasses integer linear programming, goal programming, mixed integer programming, and constraint programming.
- **Heuristics (HE)**: find high-quality solutions within an acceptable computational cost but do not guarantee optimality. This category comprises constructive heuristics, graph coloring, and decomposition.

- **Metaheuristics**: high-level methodologies that combine underlying heuristics, employing intelligent strategies for exploring and exploiting the search space and concurrently guiding heuristic procedures with learning techniques. The classification of a metaheuristic as either single-solution or population-based is contingent upon the number of solutions examined in each iteration of the exploration procedure.
  - **Single solution-based (MH-S)**: employ iterative processes to enhance an individual candidate solution using transformation operators within its neighborhood, employing local search techniques.
  - **Population-based (MH-P)**: commence by establishing an initial collection of candidate solutions, iteratively manipulating them to improve search, and replacing some with new solutions based on quality criteria to explore and exploit effectively.
- **Matheuristics (MAH)**: hybridizes metaheuristics with mathematical programming techniques, utilizing mathematical programming models in a heuristic framework.
- **Hyper-heuristic (HH)**: explores a search space composed of low-level heuristics or heuristic components, delineating two major types:
  - **Generation hyper-heuristics**: create new heuristics using components and operators.
  - **Selection hyper-heuristics**: enhance solutions by selecting low-level heuristics from a provided set.

- **Hybrid Approaches**: a high level of integration, either by incorporating a metaheuristic algorithm into a search tool or integrating various metaheuristic algorithms into a unified framework.

A. MATHEMATICAL OPTIMIZATIONS

1) LINEAR PROGRAMMING

Arbaoui et al. [32] presented a mixed-integer mathematical model and used the CPLEX 12.5 solver to test it on both the ITC 2007 benchmark and the Yeditepe benchmark. In the preprocessing stage, the exam-based conflict graph was utilized to reveal implicit conflict constraints between exams. The approach incorporated valid inequalities derived from cliques and employed a data-dependent dual-feasible function to improve the model. Experimental results demonstrated the superior performance of the enhanced model in most instances of both datasets compared to the original model. In 2019, [50] proposed a generic mixed integer programming (MIP) model with compact formulations having an equation count proportional to the number of exams. The lower bounds were improved by focusing on two exam spacing soft constraints. Computational analysis on the same datasets showed a substantial enhancement in lower bounds compared to previous results and outperformed other available formulations.

Woumans et al. [36] developed two column generation (CG) models for scheduling multiple exam versions while ensuring even distribution. These models were implemented using IBM ILOG CPLEX and tested with real-world data at the KU Leuven campus and the Toronto benchmark. Model 1 used columns to represent exam schedules for unique student groups, while Model 2 represented mask schedules for each group. Post-processing employed heuristic and binary IP approaches to assign exam versions to student groups. Model 1 focused on minimizing spread costs, even if it led to higher total costs, and performed well on both Toronto datasets. Model 2 aimed for the lowest total cost but could only solve one dataset within the time limit.

Cataldo et al. [43] tackled a curriculum-based ETP for Universidad Diego Portales in Chile using three mathematical programming models. Unlike other ETPs, the exam schedule conflict numbers, in this case, are uncertain and depend on curriculum configurations. The first model allocated timeslots and room patterns to group courses, whereas the second model allocated timeslots and room patterns to each course. In the final model, specific rooms were allocated to each exam based on the outcomes of the second model. These models were implemented using GAMS, with CPLEX serving as the solver. Compared to the university's manual process, the schedules demonstrated improved efficiency, with fewer conflicts in most situations and no requirement for rescheduling.

Keskin et al. [46] proposed a mixed MIP model with a 2-stage heuristic approach to address a real-world ETP at Pamukkale University's Faculty of Engineering. The scheduling process involves two stages: determining the exam time in the first stage and assigning classrooms and students to those exams in the second stage. The proposed two-stage method yielded better feasible solutions and faster computation times than commercial software (i.e., CPLEX and Gurobi). Additionally, it shortened the projected final week by one day.

Güler and Gecici [51] developed a decision support system (DSS) using a spreadsheet and MIP model to generate balanced schedules for Yıldız Technical University's Industrial Engineering Department. To create the DSS, the MIP model was integrated into Microsoft Excel with the Solver-Studio add-in and Gurobi. In 2021, [56] presented two MIP models with a decomposition approach for solving exam and supervisor assignment problems at a vocational school that provides associate-level degrees. The MIP models were integrated into a web-based DSS, enabling a typical end-user to generate a timetable in less than two minutes using the CBC open-source solver.

Al-Hawari et al. [52] introduced a three-phase Integer Linear Programming (ILP) for addressing the ETP at the German Jordanian University (GJU) and Toronto benchmark. The approach decomposed the problem into three phases: first, allocating exams to timeslots; second, assigning timeslots to days; and third, assigning exams to rooms. ILP formulations were used to obtain solutions for each phase, with a CPLEX solver being utilized. Modularization simplified the formulation, resulting in easily generated, feasible timetables stored in multiple standard formats significantly faster than manual methods.

Godwin [78] utilized linear 0-1 integer mathematical programs, formulated using AMPL and solved with CPLEX, to generate a quality exam timetable considering faculty, room, and student-oriented aspects. Two surrogacy-based approaches were introduced: one assessed surrogacy levels among different objective functions, identifying potential surrogates for multi-objective problems, and the other created multiple solutions for quality and computational time comparisons. Tested on a business school ETP, the optimization models–faculty-oriented and room-oriented–exhibited quicker optimization times and served as effective surrogates for each other. However, they proved less suitable as surrogates for student-oriented objective functions.

Bazari et al. [77] implemented an ILP model using IBM ILOG CPLEX and its optimizer. CPLEX's innovative search tree traversal facilitates more efficient discovery of feasible solutions than relying solely on the branching and cutting method. Tested at Ferdowsi University Management Department, the model outperformed the manually prepared schedule, showcasing its superiority in optimal class utilization, avoiding exam conflicts, and achieving average exam spreads.

2) GOAL PROGRAMMING (GP)

Cavdur and Kose [37] designed a binary GP-based approach incorporating fuzzy logic to tackle the ETP tailored explicitly for the Industrial Engineering Department at Uludağ

University. Initially, fuzzy criticality levels for exams were determined by assigning them based on three factors: class credits, success ratios, and class types. Next, the criticality levels and pre-processed data were inputted into a GP model to develop an optimal exam schedule. A simple algorithm was then used to assign rooms and proctors. In all evaluated objectives, the model's generated schedule performs better than the human expert's produced schedule.

B. HEURISTICS

Burke et al. [14] evaluated a linear combination of primitive heuristics on an enhanced weighted graph that utilized continuously updated information. A new vertex-selection heuristics was developed based on saturation degree and largest weighted degree. The improved model enabled the combination of basic heuristics to create compound selectors and introduced ''badness'' thresholds for color assessment. Tested on Toronto datasets, it achieved optimal results in five cases and outcomes comparable to the best-known outcomes in seven others. Performance gains were observed when the linear combination strategy was combined with backtracking.

Abdul-Rahman et al. [21] proposed an adaptive decomposition and heuristic ordering strategy and experimented on the Toronto dataset. Exams were categorized into two subsets, difficult and easy, and ordered using various graph coloring heuristics to prioritize the assignment of exams to timeslots. To enhance the solution quality, the boundary set, positioned as a subset between the easy and difficult sets, underwent merging and swapping with the difficult set. If no improvement occurred, the roulette wheel selection-based shuffle strategy was applied. While the experimental results did not surpass six constructive approaches in their entirety, they remained competitive with other improvement strategies.

Kahar and Kendall [22] designed a two-phase constructive algorithm for addressing the ETP at Universiti Malaysia Pahang (UMP). In the first phase, exams were concurrently scheduled in timeslots and rooms, and the second phase involved invigilator scheduling. They enhanced the invigilation scheduling model with new constraints, including a minimum 2-day gap between duties, simultaneous oversight of exam papers by lecturers in the same location, and penalties for overseeing their papers. The algorithm met all constraints that the UMP proprietary system could not.

Kasm et al. [68] introduced a constructive heuristic, incorporating a color graphing algorithm, to address a capacitated ETP at Masdar Institute and Toronto benchmark. They considered a new constraint related to the limit on the number of exams assigned to a single student within a predetermined period. Utilizing an adjustable parameter known as the quantity of high-degree nodes, the heuristic impacted scheduling efficiency by giving preference to larger courses. It prioritized identifying the color group that harbors the most non-conflicting vertices, deviating from emphasizing minimizing the number of such groups. Experimental results showed its efficiency compared to the

IP formulation, yielding optimal or near-optimal outcomes in computational time and scheduling duration for both datasets.

C. METAHEURISTICS
1) SINGLE-SOLUTION BASED METAHEURISTICS
a: TABU SEARCH (TS)

Pais and Amaral [13] proposed a TS method utilizing a fuzzy inference rule-based system (FIRBS) for automated tabu tenure management. Two features, inactivity, and frequency, were employed to register elements on the tabu list. The definitions of these features varied depending on the neighborhood, taking into account both simple and Kempe chain moves. The study experimented with Toronto datasets in two phases. Initially, it compared FIRBS to TS with fixed tenure values, showing a significant improvement. In the second phase, FIRBS was compared with four other established methods using the same objective function, demonstrating its simplicity and suitability as an alternative to manually tuning tabu tenure. In 2016, [34] presented a multi-objective approach that utilized TS for multi-criteria ETP, assessed on the same dataset. Two features were introduced to enhance automation: FIRBS for tabu list's element tenure selection and MCR for flexible preference modeling. FIRBS achieved the best value for 3 of 4 objectives, and MCR's weighting functions provided a smoother balance between objective function values.

b: VARIABLE NEIGHBORHOOD SEARCH (VNS)

Elloumi et al. [25] developed an adapted VNS to minimize total capacity usage for an ETP at the Faculty of Economics and Management Sciences of Sfax. The first stage involved scheduling exams into timeslots, and the second stage assigned exams from timeslots to available classrooms. Two reduction procedures were employed: one involved sorting exam sizes and classroom capacities in a non-decreasing order, and the second used a dominance criterion. The VNS algorithm efficiently allocated classrooms using insertion and swap moves as neighborhood structures. Evaluating the VNS algorithm against this lower limit validated its ability to produce favorable outcomes.

c: SIMULATED ANNEALING (SA)

Battistutta et al. [39] introduced a single-stage SA method that utilized a feature-based parameter tuning technique designed for the ITC 2007 benchmark. The technique entailed the inclusion of non-feasible solutions within the search space and managing them through suitable penalty mechanisms. A two-step parameter tuning process identified crucial parameters and correlated parameters' values to instance features through a linear function using a regression model. Compared to the five ITC 2007 finalists, they globally outperformed them and achieved the best outcomes in two instances. The study highlighted that a properly tuned search method could effectively enhance generalization on new instances.

Leite et al. [48] designed an accelerated variant of SA, named FastSA, which automatically computes the cooling

rate. The approach comprised two phases: construction, using a saturated degree graph coloring variant to form the initial solution and optimization, employing the SA approach. Testing on ITC 2007 datasets showed that FastSA outperformed standard SA with fewer evaluations and less computation time. FastSA ranked third among five algorithms when pitted against state-of-the-art methods, exhibiting improvement in one out of twelve instances.

Bellio et al. [55] proposed a two-stage SA-based approach for an ETP based on the formulation of uncapacitated Toronto benchmarks. In Stage 1, they focused on conflict resolution to establish a feasible solution, and Stage 2 enhanced the solution by optimizing exam distances without introducing new conflicts and utilizing specific neighborhood relations. The results obtained matched or surpassed the best outcomes in 10 out of 13 instances compared to a compilation of the most renowned results in the literature. Also, using a mathematical model and "warm-starting" CPLEX with SA-derived solutions to try to improve the initial solutions for instances but run into problems, showing how hard it is to use exact algorithms for larger instances.

Van Bulck et al. [67] presented an SA algorithm with a multi-neighborhood approach to address the ITC 2007 benchmarks. The SA framework incorporated six neighborhoods: Move, Kempe, Kick, Shake, and two noteworthy additions: a beam search operator for exam-to-room assignments and an approach leveraging multiple disconnected components in the underlying conflict graph. Its parameters underwent fine-tuning using the irace package through iterated racing procedures. The ablation analysis demonstrated a preference for including all neighborhoods, mainly Move and Swap, while the infrequent selection of Kempe was attributed to computational costs. The algorithm excelled in 11 out of 12 instances compared to ITC 2007 competition finalists and achieved the best solutions in 5 out of 8 Yeditepe instances.

Carlsson et al. [76] investigated three distinct methods for addressing a real-world ETP proposed in [79]: (i) extended SA with a new neighborhood relation, SwapEvents; (ii) constraint programming (CP) solved using Gecode; and (iii) two MIP models solved with Gurobi (2-stage method, MIP2S, and encoded in MiniZinc, MIPMZ). The experimental results demonstrated that extended SA outperformed the original SA in all instances except for two and surpassed MIPMZ and MIP2S in all instances except for two. CP exhibited somewhat inferior performance, and while MIPMZ succeeded in some instances, it struggled or timed out in others.

d: HILL CLIMBING (HC)
Bykov and Petrovic [33] implemented a step-counting mechanism named the Step Counting Hill Climbing algorithm (SCHC). SCHC generated an initial feasible solution using the Saturated Degree heuristic for exam scheduling and then employed a heuristic search, utilizing various moves and the Kempe chain procedure iteratively to produce a candidate solution. Three variants of SCHC were developed:

SCHC-all (counting all moves), SCHC-acp (counting only accepted moves), and SCHC-imp (counting only improving moves). SCHC-acp outperformed SA, Late Acceptance Hill Climbing, Great Deluge, and other SCHC variants in longer runs when evaluated on ITC 2007 problems. Additionally, SCHC-amp outperformed previously published results on eight benchmark instances, including one pre-competition and two post-competition outcomes.

Burke and Bykov [40] presented a Late Acceptance Hill Climbing (LAHC) approach with an added greedy rule and enhanced history recording and evaluated it using benchmarks for the Travelling Salesman and ITC 2007 ETP. The algorithm initiated its process by generating an initial feasible solution at random, employing the saturation degree heuristic for exam room allocation. It then iteratively accepts or rejects candidates while ensuring feasibility by applying a range of moves. Throughout the entire search process, only feasible solutions were considered. The proposed approach demonstrated superior performance compared to basic LAHC, Simulated Annealing, threshold acceptance, and Great Deluge on most of the benchmark problems, particularly those with larger sizes.

e: GREAT DELUGE (GD)
Kahar and Kendall [31] implemented a modified GD algorithm (modified-GDA) for a capacitated ETP at UMP. The algorithm included a simple parameter that enabled dynamic adjustments to the boundary, decay rate, and desired value as the search process unfolded. The modified-GDA approach surpassed UMP's proprietary software, Dueck-GDA, and constructive heuristics by generating high-quality solutions that satisfied all constraints, which the proprietary software could not achieve.

Burke and Bykov [35] introduced an extension of GD called Flex-Deluge, featuring a flexible acceptance condition. This enhancement underwent testing on both the Toronto and ITC 2007 benchmarks. They introduced two concepts to improve search technique performance: slowing down uphill moves and adapting the algorithm to specific problem properties. Experimental results demonstrated that the adaptive Flex Deluge algorithms (AFDA), specifically AFDA/LD and AFDA/LD+LNS, which use the largest degree (LD) for replacement moves and the largest number of students (LNS) for room moves, outperformed other algorithms, such as GD, greedy HC, and fixed Flex-Deluge. AFDA variants achieved superior results compared to previous publications, obtaining nine top results for the ITC 2007 benchmark and nine out of 16 top results for the Toronto benchmark.

2) POPULATION BASED METAHEURISTICS
a: MEMETIC ALGORITHM (MA)
Lei et al. [44] presented an MA based on a multiobjective evolutionary algorithm with decomposition (MOEA/D) to address the uncapacitated Toronto ETP. In the initial phase, the algorithm used double hybrid initialization to generate diverse-length initial populations, followed by crossover

and mutation on randomly selected individuals. Two local search operators, period-supplement and bi-directed, were employed to enhance solution distribution and optimize two objectives simultaneously. The experimental results showed that double hybrid initializations worked better than random and hyper-heuristic ones, demonstrating the effectiveness and robustness of the designed operators.

Leite et al. [45] designed a cellular memetic algorithm (cMA) that hybridized a cellular evolutionary algorithm and a threshold acceptance metaheuristic to address ETP posed by Toronto and ITC 2007 benchmarks. This two-phase cMA first generated feasible solutions using saturated degree and then optimized them in the second phase using TA, Kempe chain neighborhoods, and a combination of mutation and crossover operators. The cMA method did better than other solutions that had been published before. It had the smallest average relative deviation compared to the best solutions on four out of thirteen cases in the Toronto dataset and three out of twelve cases in the ITC 2007 dataset.

b: HONEY-BEE MATING OPTIMIZATION (HBMO)
Sabar et al. [10] proposed a modified HBMO algorithm and evaluated its performance using the Toronto and Socha datasets for uncapacitated ETP and course scheduling problems. Their approach started by creating populations of feasible solutions using the largest enrolment, largest degree, and least saturation degree. In the crossover process, two timeslots were randomly chosen from the queen and drone, and events were then swapped between these selected timeslots. To prevent the drone's sperm from being reused, an event-swapping mutation operator was employed between timeslots. Mated drones were eliminated to sustain population diversity, and newly produced broods were introduced into the population for the subsequent mating cycle. This approach outperformed the original HBMO on Toronto datasets and achieved results comparable to or better than five population-based and eight non-population-based methodologies.

c: PREY-PREDATOR ALGORITHM (PPA)
Tilahun [49] introduced an extended PPA and tested it on six instances of uncapacitated Toronto datasets. Three classifications were assigned to the initial solutions: the top-performing solution was denoted as the best prey, the least effective solution was marked as the predator, and the rest were considered ordinary prey. The best prey exploited its neighborhood, whereas the predator explored the search space to locate a promising area. Both the exploration and exploitation search behaviors influenced the ordinary prey. Statistical comparisons were conducted between the results and seven selected methods from existing literature, demonstrating performance on par with the best.

d: FIREFLY ALGORITHM (FA)
Nand et al. [58] designed a discrete firefly algorithm (dFA) by introducing preference parameters, a stepping-ahead mechanism to tackle the uncapacitated Toronto ETP.

The algorithm consisted of three phases: Phase 1 used dFA to create an initial solution; Phase 2 improved it with simple moves; and Phase 3 employed sequential neighborhood operators for further refinements. The novel stepping-ahead mechanism utilized preference parameters to enhance optimization and thoroughly explored the solution space, identifying superior solutions. Compared to 10 other algorithms using identical stages, dFA-Step showed similar performance in 5 out of 12 datasets.

D. HYPER-HEURISTICS (HH)
1) SELECTION HYPER-HEURISTICS
Burke et al. [9] evaluated various Monte Carlo-based HHs on the capacitated Toronto dataset. They employed basic heuristic selection methods such as greedy, simple random, choice functions, and a learning scheme. Additionally, they applied three move acceptance techniques based on Monte Carlo methods: standard SA, SA with reheating, and exponential Monte Carlo. An experiment tested combinations of heuristic selection and move acceptance techniques in selection HH. The results indicated that the most effective performance was achieved when combining the SA approach with the reheating move acceptance method and utilizing the choice function technique.

Demeester et al. [8] proposed a tournament-based HH approach to a curriculum-based ETP at KAHO and two post-enrollment ETP benchmarks–Toronto and ITC 2007. The study examined various factors, including the random heuristic selection, tournament factor, and four move acceptance criteria: ''improving or equal'', SA, GD, and a modified late acceptance (LA). Results showed that combining LA with the steepest descent was the most effective move acceptance criterion, on par with SA. It outperformed the manual solution for KAHO and improved 7 of 13 instances in Toronto. While not surpassing the best-known approaches in ITC 2007, it remained competitive.

Pillay [6] used an evolutionary algorithm (EA) based HH with three heuristic representations for the Toronto ETP: fixed length (FHC), N-times (NHC), and variable length (VHC). These representations, using unique combinations of low-level heuristics, were assessed for their impact on EA performance. FHC performed less effectively than VHC and NHC. Combining all three representations yielded superior results compared to using FHC, VHC, or NHC individually.

Soghier and Qu [16] suggested an iterative HH method that used a low-level graph coloring heuristic to choose which exams to take and adaptively hybridized bin-packing heuristics to assign exam timeslots and rooms automatically. This approach dynamically selects and adjusts the level of hybridization for each heuristic to generate a spectrum of heuristic sequences with varying quality. Combining top-performing heuristics resulted in the best solutions across the ITC 2007 datasets, achieving feasibility in all eight instances and ranking 5th among the nine best approaches in the literature.

Sabar et al. [26] created a framework for gene expression programming-based HH and utilized it to tackle the ITC 2007 ETP and a dynamic vehicle routing problem. This framework represented an improvement-oriented approach, featuring a set of perturbative low-level heuristics and high-level techniques with two elements: selection of heuristics through a dynamic multi-armed bandit approach with extreme value-driven rewards and an automatically generated criterion for accepting moves using gene expression programming. This approach outperformed the ITC 2007 competition winner and post-ITC 2007 methods in 4 of 8 instances.

Qu et al. [27] developed an HH framework using an Estimation Distribution Algorithm (EDA) to tackle the uncapacitated Toronto ETP. The high-level search process relied on a basic EDA, guiding heuristic selection for various problem-solving scenarios. The performance of low-level heuristics was assessed through their probability distribution during solution construction, enhancing HH search. Despite lacking backtracking or local improvement techniques, this constructive approach showed promising results across all instances, matching the top performance among the ten others discussed in the literature.

Muklason et al. [41] designed a three-stage multi-objective selection HH approach to optimize standard objectives and fairness within student cohorts. In the initial stage, feasible solutions were created using the squeaky wheel technique. In the subsequent two stages, a selection HH was used, integrating the GD algorithm and reinforcement learning to control the move acceptance and 14 low-level heuristics, respectively. Stage 2 aimed to optimize the standard objective, while Stage 3 sought fairer solutions within a reasonable Pareto front range. Experiments on three benchmarks (Toronto, ITC 2007, and Yeditepe) showed that the approach works for multi-objective ETP, with performance similar to methods that have been reported before.

Hao et al. [53] introduced a unified evolutionary multitasking graph-based HH that employed evolutionary multitasking for high-level search and graph heuristics for low-level heuristics. This unified framework exhibited greater effectiveness and versatility than the single-task HH method based on experimental results on both Toronto ETP and graph coloring problems.

2) GENERATION HYPER-HEURISTICS

Sabar et al. [7] proposed a novel graph coloring constructive HH approach that utilized hierarchical fusion of four low-level heuristics: largest colored degree, saturation degree, largest enrollment, and largest degree. Combining these heuristics generated four ordered lists to calculate the scheduling difficulty index for the initial exams. Exams were chosen according to their difficulty index and assigned to timeslots using a roulette wheel selection. Despite its single-pass stochastic nature, this approach exhibited competitive performance when evaluated against other HH methods using single sequential graph coloring on the

Toronto dataset and constructive methods on the ITC 2007 dataset.

Sabar et al. [17] designed a grammatical evolution HH framework and tested its performance on the ITC 2007 and Toronto benchmarks. The framework utilized grammatical evolution to build an online solver, utilizing a grammar definition to define heuristic components (neighborhood structures, acceptance criteria, and combinations). It automatically evolves multiple templates of perturbation heuristics to solve a problem instance. Experimental findings indicated that integrating adaptive memory into grammatical evolution-HH improved performance compared to the version without it. The approach achieved competitive results when compared to state-of-the-art HHs and bespoke methods for Toronto datasets, surpassing both the winner of the ITC 2007 competition and post-ITC 2007 methods.

Burke et al. [20] designed an adaptive HH approach that hybridized low-level heuristic moves to enhance timetables. The Toronto and ITC 2007 datasets revealed the optimal combination of swap timetables (ST) and Kempe chains (KCM) through offline learning. Ordering exams violating soft constraints via saturation degree yielded superior outcomes. They subsequently crafted an adaptive ST-KCM hybridization in two stages: first, they generated and analyzed random heuristic sequences to pinpoint the best ones. Second, they randomly assigned heuristics to empty positions to determine the optimal sequence. This approach showcased robust generalization across two benchmark datasets, achieving performance comparable to the latest cutting-edge techniques.

Abdul-Rahman et al. [19] introduced an adaptive linear approach based on squeaky wheel optimization. Each exam received a difficulty score during the assignment process by calculating a modified weighted sum of low-level heuristics. Exams were ordered based on their difficulty score, with higher scores getting assigned timeslots and rooms before lower scores. This cycle continued until a feasible solution improved as resources were allocated to minimize penalties. When several heuristics were used with a modifier, the results were similar to other constructive methods for the Toronto datasets. Feasible results were obtained for the ITC 2007 datasets.

Pillay and Özcan [47] presented a generation construction HH that automates generating low-level construction heuristics. This study investigated two kinds of heuristics: arithmetic (AHH), evolved through genetic programming, and hierarchical (HHH), explored through three distinct approaches: HHH-GP (genetic programming), HHH-GA (genetic algorithms), and HHH-RG (random generation), each with its unique representation. Experimental results on the Toronto and ITC 2007 exam timetabling benchmark sets revealed AHH's superiority over HHH and traditional graph coloring heuristics. On the other hand, HHH-GA showed promise for the ITC 2007 course timetabling benchmark. Hence, the authors concluded that heuristic effectiveness varies based on specific educational timetabling problems.

Mweshi and Pillay [54] enhanced the framework of HH methods to create perturbative heuristics with a broader design. Initially, these heuristics came from basic actions and solution components using grammatical evolution. This extension added conditional constructs, incorporated data from the solution space, and broadened the syntax of fundamental actions to include a broader range of heuristics. They tested this approach on benchmark sets in three domains: boolean satisfiability problems, ETP, and vehicle routing problems. The effectiveness of these newly generated heuristics highlighted the approach's versatility and demonstrated its competitiveness with existing perturbative HH on the ITC 2007 exam benchmark.

E. HYBRID APPROACH

Abdullah and Turabieh [11] presented a tabu-based MA by integrating a genetic algorithm and a tabu search algorithm and evaluated its performance on the ITC 2007 exam and course problems. The algorithm employed various neighborhood structures for local search and integrated a tabu list to manage their selection. Experiments were conducted on ITC 2007 datasets using three different selections of neighborhood structures: "random," "best," and "general". The "best" sequence demonstrated superior performance in course timetabling, while the "general" sequence excelled in examination timetabling. The results surpassed the ITC 2007 exam track winners in 7 out of 8 datasets.

Ahandani et al. [12] proposed a structure integrating discrete particle swarm optimization (DPSO) and a two-stage HC local search for the Toronto ETP using three hybridization strategies. It involved both generation HH elements (associated with DPSO and heuristic updates) and selecting HH components (concerning low-level heuristic management and the use of two-staged HC). The DPSO algorithm used a mixed-population heuristic initialization method to create an initial population and iteratively improve it through DPSO and local search. The study's findings strongly supported the effectiveness of these DPSO algorithms with diverse structures and strategies as HH systems. They performed as well as or outperformed other HH methods and were on par with evolutionary algorithms in the literature.

Abdullah and Alzaqebah [18] created a hybrid self-adaptive bees algorithm (BA) by making three essential modifications: comparing selection strategies (tournament, disruption, rank), comparing local search algorithms (SA and LAHC) through parameter tuning, and looking into how a self-adjusting mechanism affects the effectiveness of neighborhood exploration. Experimental on the Toronto and ITC 2007 datasets identified the most effective BA algorithm enhancements, with the self-adaptive disruptive selection strategy combined with LAHC local search showing superior performance over other strategies (basic BA, rank selection, tournament selection, and SA local search). In 2015, [28] hybridized bee colony optimization with LAHC, disruptive selection, and self-adaptive, yielding results that are on par with the best-performing methods on Toronto datasets and

mirrored the success of the top five winners in the ITC 2007 datasets.

Al-Betar et al. [24] suggested using the harmony search algorithm to create an optimization framework that combines different parts of MA and tested how well it worked on the uncapacitated Toronto datasets. The three components incorporated were the recombination operator for global improvement (GIM), the randomness operator for random improvement (RIM), and the neighborhood operator for local improvement (LIM). The combination of GIM, RIM, and LIM components worked well for complicated timetabling problems and did better than 22 other comparison methods, such as local search-based MH, MH-P, H, and HH. This method yielded the best overall outcomes in two instances out of twelve.

Fong et al. [23] presented a novel approach that hybridized the artificial bee colony (ABC) and GD algorithms, involving two phases: initialization and enhancement. The enhancement stage was divided into three sub-stages–employed bee, onlooker bee, and scout bee phases. The assimilation policy controlled the search in the employed bee phase, while the Nelder-Mead simplex search approach refined the solutions in the onlooker bee phase. In addressing the Toronto ETP, this technique exceeded the basic ABC algorithm and attained three novel best results while retaining its competitiveness against state-of-the-art methods. In 2015, [29] integrated a modified neighborhood search function in the employed bee phase of a variant ABC algorithm, drawing inspiration from the "global best" model found in particle swarm optimization. Evaluation of the same datasets revealed that the results of this variant were almost identical to those of the previous version.

Li et al. [30] presented an Evolutionary Ruin and Stochastic Rebuild approach, validated using uncapacitated Toronto datasets. The method involved four phases: (i) solution decomposition for solution breakdown, (ii) evolutionary ruin for destructive modification, (iii) stochastic recreate for restoration, and (iv) solution acceptance for probabilistic acceptance determination. It outperformed graph-based HH algorithms but was inferior to local search algorithms based on experimental results.

Müller [38] designed a multi-phase search algorithm for a real-life ETP at Purdue University and introduced nine new benchmark datasets. The algorithm began with a construction stage employing an iterative forward search algorithm and incorporating conflict-based statistics. The following phase used an HC algorithm to arrive at a local optimum, then a GD technique to account for oscillations in the overall solution value bound. Experimental results affirmed the two-hour time limit's reliability for creating Purdue's timetables. UniTime enabled interactive manual adjustments with solver recommendations to maintain quality during modifications.

Lei et al. [42] developed an adaptive coevolutionary MA for addressing the Toronto benchmark. This algorithm conducted an evolutionary search in two domains: heuristic space for global search and solution space for local search.

The heuristic space began with an HH approach to create an initial population and used the basic crossover operator and mutation operator to identify possible heuristic lists. Tailored evolutionary operators were utilized in the solution space to expand the exploration scope and optimize the solution. An adaptive coevolutionary strategy was used to select the search space adaptively. Compared to 11 popular methodologies, this algorithm achieved top-five results across 12 instances.

Aldeeb et al. [57] introduced a hybrid intelligent water drops (IWD) algorithm for addressing the Toronto ETP. The algorithm involved six phases for hybrid IWD: static parameter initialization, dynamic parameter initialization, assigning a random timeslot to an individual exam for each IWD, constructing IWD solutions, enhancing solutions, and termination. In the final stage, an enhancement loop integrated a local search optimizer (LSO), employing Move, Swap, and Kempe Chain neighborhood structures. Based on the experimental results, the algorithm outperformed nine swarm-based approaches in three datasets and achieved the best results compared to 10 metaheuristic approaches in one dataset.

F. MATHEURISTIC

Gogos et al. [15] presented a multi-staged algorithmic approach for solving the ITC 2007 ETP, primarily managed by the Greedy Randomized Adaptive Search Procedure (GRASP). This approach comprised three phases: pre-processing to uncover hidden dependencies, construction for timetable creation, and an improvement stage for solution optimization. The improvement phase incorporated HC local search, SA local search, and an IP sub-problem solved using the GLPK solver. In cases where the current solution could not be enhanced, a shaking stage rescheduled suboptimal exams. Experimental results demonstrated that this approach ranked second among the ITC 2007 competition winners.

Çimen et al. [66] introduced a Mixed-Integer Linear Programming (MILP) model and a heuristic algorithm to address the invigilator scheduling problem at Hacettepe University and Gazi University. When tested on eight real-life cases, both demonstrated effectiveness by outperforming schedules created by the faculty. The experimental results demonstrate that the MILP-generated schedules showcase an enhanced distribution of invigilation hours, avoidance of successive assignments, and prioritization of department-based duties. Simultaneously, the heuristic algorithm excels at creating schedules with a fair distribution of invigilation hours and avoids successive assignments, proving its applicability to larger-sized examinations.

V. CATEGORIZATION OF ETP SOLUTION METHODOLOGIES

In this study, we reviewed solution methodologies for ETP presented in 59 journal articles spanning from 2012 to 2023, as illustrated in Table 8. The distribution of the chosen papers based on publication year and categories is depicted

in Figures 1 and 2, providing percentage and numerical perspectives. The publication trends over the years are illustrated in Figure 1, revealing fluctuating patterns. The peak publication rate, with ten publications representing 17% of the total, occurred in 2012, while the lowest percentage, with two publications reflecting 3% participation, was observed in 2013, 2018, 2020, and 2021. The categorization of these methodologies is depicted in Figure 3.
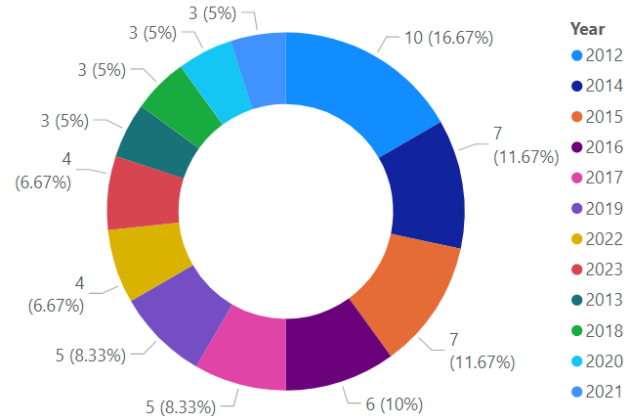

FIGURE 1. Distribution of the ETP studies by publication year.
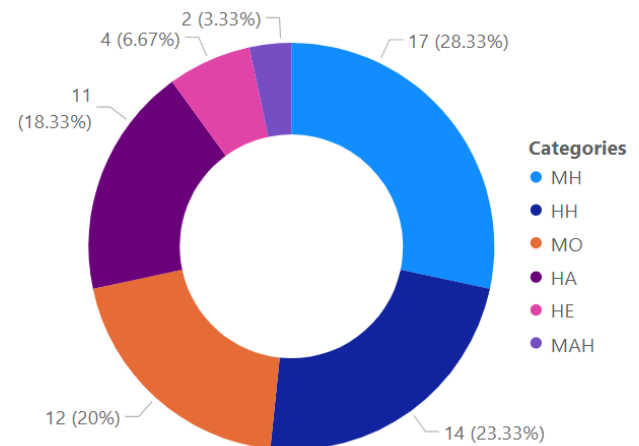

FIGURE 2. Distribution of the ETP studies by category.

As indicated in Table 8, there are fourteen hyper-heuristics (HH), twelve mathematical optimizations (MO), twelve metaheuristics based on single solutions (MH-S), five approaches employing population-based metaheuristics (MH-P), eleven hybrid approaches (HA), four heuristics (HE), and two matheuristics (MAH). The highest figure is observed in the HH within these categorizations, while the least falls under the MAH. HH strives to elevate the level of generality in the operation of optimization systems. They operate on a search space defined by lower-level heuristics or even metaheuristics for the addressed problem. The primary focus of HH is the thoughtful selection or generation of the appropriate (meta-)heuristic for any given situation.
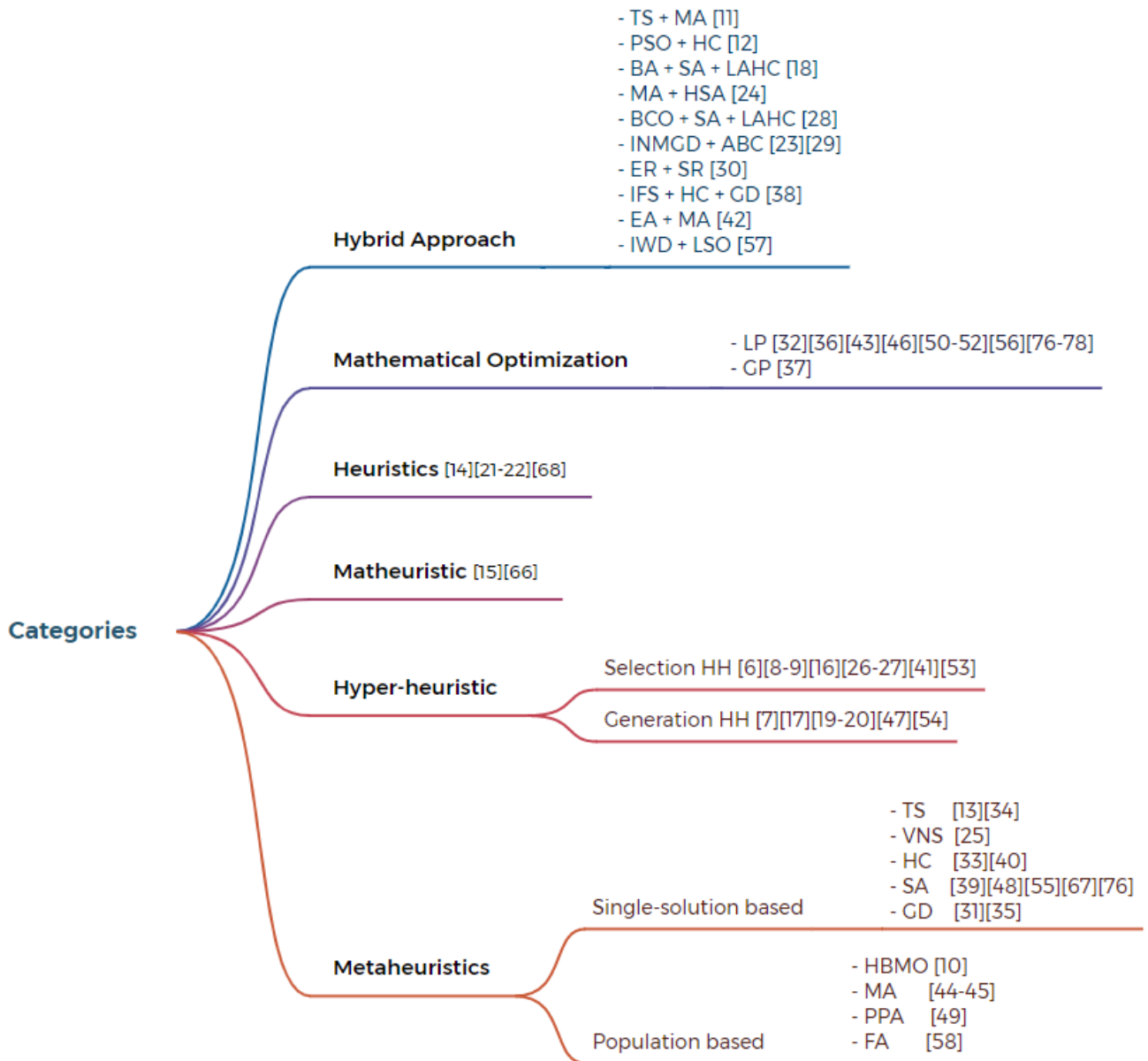
FIGURE 3. The categorization of solution methodologies for ETP.

Various selection HH has been explored in ETP, such as Monte Carlo Search [9], multiple move acceptance criteria [8], swarm-based computational intelligence integration [12], evolutionary algorithms combined with low-level heuristics citeb6, and estimation distribution algorithm [27]. Additionally, multi-objective selection HH with GD and reinforcement learning [41] and multitask selective HH [53] have been investigated. Adaptive approaches were implemented in HH for selecting heuristics, ordering heuristics, or hybridizing heuristics based on a learning mechanism [16], [19], [20], [21]. On the other hand, generation HH employed in ETP encompasses generation construction heuristics [47] and perturbative heuristics [54]. Genetic programming, which includes gene expression programming [26] and grammatical

evolution [17], [54], was employed to evolve acceptance criteria and generate selection heuristics.

However, as indicated in Table 8, there has been a discernible shift in research focus from a predominant emphasis on HH to an increasing inclination towards metaheuristics and MO over the timeline from 2012 to 2023. Figure 2 reveals a heightened interest in metaheuristic categories, constituting 28% of the selected studies: 20% in MH-S and 8% in MH-P. It is evident from Figure 3 that many of the selected metaheuristic algorithms encompass widely recognized and frequently employed ones, such as the MA, LAHC, TS, and GD. These algorithms' performance and experimental outcomes have demonstrated their efficiency and effectiveness. Notably, SA has received extensive coverage in publications,

surpassing the previously mentioned methods. The hybrid approach has demonstrated exceptional performance by combining metaheuristics with (meta-)heuristics or incorporating local search methods into population-based methods. This approach significantly benefits by enhancing diversity within a population, thereby boosting the search capabilities of the resultant hybrid algorithm [24], [80]. Notably, it has gained substantial traction, as evidenced by its 18% publication record.

The MO emerges as another highly utilized approach, with a substantial participation rate of 56%, as depicted in Figure 4, which illustrates the distribution of categories for ETP real-world problems. MO approaches are extensively employed in addressing real-world datasets, primarily owing to their effectiveness in handling small to medium-sized exam timetabling scenarios at the faculty or departmental level, mainly attributed to their assurance of optimal solutions and straightforward implementation through software integration with efficient heuristic algorithms. However, their efficiency diminishes when dealing with large-scale problems [55]. In such instances, the complexity of MO solutions escalates exponentially, classifying these problems as NP-hard with exponential time complexity, making MO approaches suitable for the ETP when time and space complexity are not critical considerations.
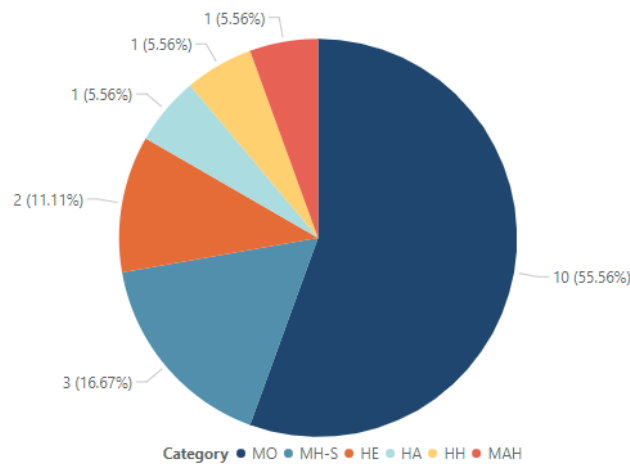
FIGURE 4. Distribution of categories for real-world dataset.

VI. CURRENT TRENDS IN BENCHMARK ETP
Between 2012 and 2023, benchmark problems, particularly the Toronto benchmark (44%) and the ITC 2007 exam benchmark (29%), have been extensively studied in the exam timetabling domain, as illustrated in Figure 5(a). In contrast, the Yeditepe benchmark has received less attention, with only 5% of the studies. Meanwhile, real-world datasets have also gained attention, constituting 22% of the studies. Additionally, there has been a noticeable increase in exploring other real-world problems, as evidenced in Figure 5(b). Table 9 presents a compilation of benchmark datasets with their respective categories and references.

A. TORONTO DATASETS
Table 9 highlights that hyper-heuristics (HH) constitute 32%, and hybrid approaches (HA) account for 23%, emerging as the most frequently employed methods for addressing large-sized Toronto datasets. Meanwhile, mathematical optimization (MO) approaches have a less significant presence, comprising only 6%. Despite their utilization in the Toronto benchmark problem, MO approaches exhibit limited effectiveness, with one solving only two small-size instances and another primarily focusing on method validation without demonstrating associated costs.

For this dataset, a total of 34 unique methodologies have been proposed. Among these, the state-of-the-art results are predominantly attributed to the multi-neighborhood SA approach by Bellio et al. [55] in most instances. Furthermore, Leite et al.'s cellular memetic [45] and Burke & Bykov's flex-deluge [35] each exhibit the best results in specific instances. Specifically, in [35], the average run time was 5.1 hours at a CPU speed of 3.2 GHz. In [45], a run time of 31.4 hours at a CPU speed of 2.0 GHz. Additionally, [55] achieved their best record running times under conditions of equal running time and CPU speed, as reported in [45]. Before 2012, the hybrid variable neighborhood approach by Burke et al. [90] in 2010 exhibited state-of-the-art results in a specific instance.

The outcomes for this problem are highly sensitive to running time, where longer durations yield better results. Due to the heterogeneous nature of running times and processor speeds in the literature, achieving fair comparisons is challenging. Consequently, depending solely on the total iteration count as the sole termination criterion for the competition proves inadequate. Instead, attention must be given to the relative computational costs, which vary based on the specific instance, adding complexity to the comparison process.

B. ITC 2007 (TRACK 1) DATASETS
According to Table 9, HH dominates with 44%, followed by metaheuristics single-solution (MH-S) at 26%, as the most commonly adopted methods for the ITC 2007 exam benchmark problem. For comparability, the running time aligns closely with the ITC 2007 benchmarking tool's specified time limit. For this dataset, a total of 23 unique methodologies have been proposed. The state-of-the-art results reported within the competition time limit are predominantly associated with a step-counting hill climbing method introduced by Bykov and Petrovic [33] and a flex-deluge approach by Burke and Bykov [35], prevailing in the majority of instances. Additionally, a distributed scatter search method by Gogos et al. [81] excels in two instances. In contrast, an HH by Sabar et al. [26] achieved the best result in a single instance, similar to [81]. Notably, the parallel distributed scatter search metaheuristic, introduced by Gogos et al. [82] in 2010, demonstrated state-of-the-art results across several instances. A detailed discussion on state-of-the-art results for the Toronto benchmark and the ITC 2007 exam benchmark can be found in the survey paper [74].

Year	Methodology	Category	Datasets	Ref
2012	Evolutionary Algorithm based Hyper-heuristics	HH	Toronto	[6]
	Graph Coloring Constructive Hyper-heuristics	HH	Toronto, ITC 2007	[7]
	Hyper-heuristics with Annealing Simulation + Great Deluge + Late Acceptance	HH	Toronto, ITC 2007, Real-world	[8]
	Monte Carlo Hyper-heuristics	HH	Toronto	[9]
	Honey-bee Mating Optimization	MH-P	Toronto	[10]
	Tabu Search + Memetic Algorithm	HA	ITC 2007	[11]
	Discrete Particle Swarm Optimization + Two Staged Hill Climbing	HA	Toronto	[12]
	Tabu Search	MH-S	Toronto	[13]
	Constructive Heuristics	HE	Toronto	[14]
	Greedy Randomized Adaptive Search Procedure + Hill Climbing + Simulated Annealing + Integer Programming	MAH	ITC 2007	[15]
2013	Hyper-heuristics	HH	ITC 2007	[16]
	Grammatical Evolution Hyper-heuristics	HH	Toronto, ITC 2007	[17]
	Bees Algorithm + Simulated Annealing + Late Acceptance Hill Climbing	HA	Toronto, ITC 2007	[18]
2014	Adaptive Linear Combination of Heuristics	HH	Toronto, ITC 2007	[19]
	Adaptive Selection of Heuristics	HH	Toronto, ITC 2007	[20]
	Graph Coloring Heuristics	HE	Toronto	[21]
	Constructive Heuristic	HE	Real-world	[22]
	Nelder-Mead Great Deluge + Artificial Bee Colony	HA	Toronto	[23]
	Memetic Computing + Harmony Search Algorithm	HA	Toronto	[24]
	Variable Neighborhood Search	MH-S	Real-world	[25]
2015	Hyper-heuristics + Multi-Armed-Bandit + Gene Expression Programming	HH	ITC 2007	[26]
	Hyper-heuristics	HH	Toronto	[27]
	Bee Colony Optimization + Simulated Annealing + Late Acceptance Hill Climbing	HA	Toronto, ITC 2007	[28]
	Artificial Bee Colony + Nelder-Mead + Great Deluge	HA	Toronto	[29]
	Evolutionary Ruin + Stochastic Rebuild	HA	Toronto	[30]
	Great Deluge	MH-S	Real-world	[31]
	Mixed Integer Programming	MO	ITC 2007, Yeditepe	[32]
2016	Step Counting Hill Climbing	MH-S	ITC 2007	[33]
	Tabu Search	MH-S	Toronto	[34]
	Flex-Deluge	MH-S	Toronto, ITC 2007	[35]
	Integer Programming + Column Generation	MO	Real-world, Toronto	[36]
	Goal Programming	MO	Real-world	[37]
	Iterative Forward Search + Hill Climbing + Great Deluge	HA	Real-world	[38]
2017	Simulated Annealing	MH-S	ITC 2007	[39]
	Late Acceptance Hill Climbing	MH-S	ITC 2007	[40]
	Selection Hyper-heuristics + Great Deluge	HH	Toronto, ITC 2007, Yeditepe	[41]
	Adaptive Coevolutionary + Memetic Algorithm	HA	Toronto	[42]
	Mixed Integer Programming	MO	Real-world	[43]
2018	Memetic Algorithm	MH-P	Toronto	[44]
	Cellular Memetic Algorithm	MH-P	Toronto, ITC 2007	[45]
	Mixed Integer Programming	MO	Real-world	[46]
2019	Arithmetic Hyper-heuristics + Hierarchical Hyper-heuristics	HH	Toronto, ITC 2007	[47]
	Simulated Annealing	MH-S	ITC 2007	[48]
	Prey-Predator Algorithm	MH-P	Toronto	[49]
	Mixed Integer Programming	MO	ITC 2007, Yeditepe	[50]
	Graph Coloring Heuristics	HE	Real-world, Toronto	[68]
2020	Mixed Integer Programming	MO	Real-world	[51]
	Integer Linear Programming	MO	Real-world, Toronto	[52]
	Selective Hyper-heuristics	HH	Toronto	[53]
2021	Grammatical Evolution Hyper-heuristics	HH	ITC 2007	[54]
	Simulated Annealing	MH-S	Toronto	[55]
	Mixed Integer Programming	MO	Real-world	[56]
2022	Intelligent Water Drops + Local Search Optimizer	HA	Toronto	[57]
	Stepping Ahead Firefly Algorithm	MH-P	Toronto	[58]
	Mixed-Integer Linear Programming + Heuristic	MAH	Real-world	[66]
	Integer Linear Programming	MO	Real-world	[78]
2023	Simulated Annealing	MH-S	ITC 2007, Yeditepe	[67]
	Integer Linear Programming	MO	Real-world	[77]
	Constraint Programming, Mixed Integer Programming, and Simulated Annealing	MO and MH-S	Real-world	[76]

C. YEDITEPE DATASETS

The Yeditepe Benchmark, comprising only four studies, involved applying MO approaches in two. The dataset's focus is restricted, possibly owing to its simpler formulation and the relatively diminutive scale of its instances compared to other benchmarks. The state-of-the-art method for this dataset is multi-neighborhood SA, introduced by Buick et al. [67]. Both the HH by Muklason et al. [41] and the MIP approach by Arbaoui et al. [50] achieve the best results in the same two instances, similar to Bulck et al.
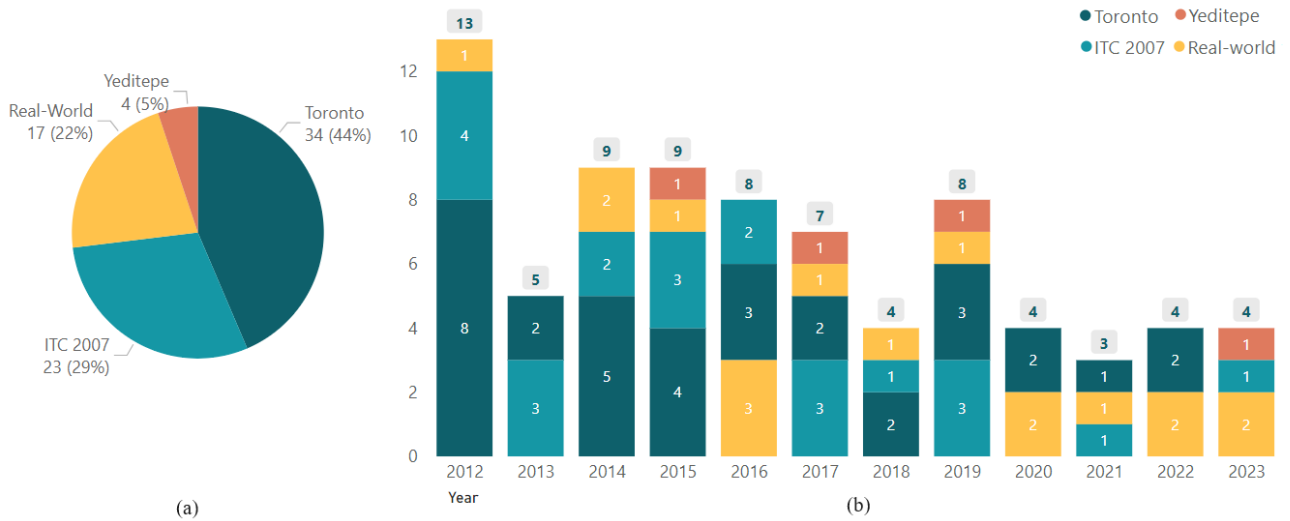
FIGURE 5. (a) Distribution of studied datasets (b) Distribution of studied datasets over time.

TABLE 9. Solution methodologies utilized for benchmark datasets.

Benchmark (Total)	Category	No. of Publication	%	References
Toronto (34)	HH	12	35	[6]–[9], [17], [19], [20] [27], [41], [47], [53], [57]
	HA	8	23	[12], [18], [23], [24], [28]–[30], [42]
	MH-P	5	15	[10], [44], [45], [49], [58]
	MH-S	4	12	[13], [34], [35], [55]
	HE	3	9	[14], [21], [68]
	MO	2	6	[36], [52]
ITC 2007 (23)	HH	10	44	[7], [8], [16], [17], [19], [20], [26], [41], [47], [54]
	MH-S	6	26	[33], [35], [39], [40], [48], [67]
	HA	3	13	[11], [18], [28]
	MO	2	9	[32], [50]
	MH-P	1	4	[45]
	MAH	1	4	[15]
Yeditepe (4)	MO	2	50	[32], [50]
	MH-S	1	25	[67]
	HH	1	25	[41]

The features, source code, instances, and new best solutions for the Toronto benchmark [55], ITC 2007 benchmark [39], [67], and Yeditepe benchmark [67] are publicly accessible for this dataset on the Exam Timetabling ETT section of OPTHUB (https://opthub.uniud.it) (last accessed: January 20, 2024). This platform allows researchers to verify and upload new instances, along with their solutions and corresponding scores, thereby fostering comparison, collaboration, and reproducibility.

VII. FUTURE DIRECTIONS

Despite being more realistic than the incapacitated variant and featuring a more complex formulation than the Yeditepe benchmark, ITC 2007 (Track 1) still reveals a gap between benchmark standards and the practical implementation of ETP in terms of requirements. This gap, highlighted in Table 6, is marked by differences in benchmark and other real-world constraints, particularly period-related and room-related constraints, indicating a need for more consideration for all practical features. Addressing

this discrepancy could be facilitated by introducing additional benchmark datasets that closely emulate real-world exam timetabling scenarios. These datasets should capture practical constraints, especially period-related and room-related constraints, crucial for effective resource management in institutions. Furthermore, ongoing investigations are impeded by the absence of crucial metadata, including details like the student's year and course, exam school, and faculty, which are essential for defining cohorts in existing benchmarks. Integrating this metadata into public datasets would significantly enhance the refinement of formulations and algorithms, aligning them more closely with student preferences [41].

Fairness holds paramount importance and is typically assessed through the spread of exam periods among all students. [36] introduced a formulation that approaches this issue from a student-centric perspective, considering scheduling specific examinations multiple times to improve fairness among diverse student groups. In a complementary fashion, [41] integrated fairness as an objective function within a cohort of students in a multi-objective optimization strategy designed to enhance overall fairness. Future research endeavors could delve into the students' demonstrated preference for timetables that are inherently fair, potentially leading to the development of examination schedules that align with these preferences and contribute to an elevated level of student satisfaction [41].

The primary limitation of metaheuristics lies in the necessity to configure a multitude of parameters, and the effectiveness of a metaheuristic hinges on parameter tuning. Despite the time-consuming nature of this process, which involves multiple runs to assess performance across instances with varied parameter settings, the universality of parameter tuning is advantageous. The competitive results were attained in [55] notably through SA, which performed tuning on artificial instances while avoiding excessive fine-tuning on benchmarks. Future research can explore the feasibility

of implementing feature-based tuning, adjusting parameters aligned with each given instance's distinctive features. It is preferable to enable algorithms to autonomously adapt their behavior for optimal solutions or use automatic parameter tuning approaches such as REVAC, SMAC, ParamILS, and F-Race [39], [55], [67].

The hybridization of metaheuristics with machine learning methods, such as deep learning or reinforcement learning, indicates a growing opportunity for cross-fertilization. Another potential direction involves the parallel implementation of a metaheuristic algorithm [45]. Burke et al. [9] showcased the effectiveness of integrating reinforcement learning with SA featuring reheating within a hyper-heuristic. Meanwhile, Muklason et al. [41] incorporated reinforcement learning and the great deluge algorithm into their hyper-heuristic.

The investigation of diverse hybrid techniques, designed to leverage the distinct advantages inherent in various search methods and metaheuristics, constitutes a notable and burgeoning trend in ETP. It may also involve employing metaheuristics for initializing the MIP or CP. Alternatively, delve into more intricate interaction mechanisms, such as the matheuristic paradigm, to further enhance the effectiveness of these methods [76].

The efficiency and promptness of a local search method depend on various factors, with the selection of both neighborhood structure and search space standing out as the most crucial determinants. Numerous implementations face challenges when their neighborhood structures prove to be overly simplistic. A notable trend observed in several state-of-the-art techniques [26], [33], [35], [45], [55], [67] applied to benchmark datasets is the integration of sophisticated neighborhood structures, particularly during the improvement phase. For instance, [55] achieved optimality by employing a combination of neighborhoods based on various atomic moves, suggesting the exploration of dynamic neighborhood probabilities through an online learning mechanism for future research. To boost local intensification, employ an adaptation mechanism to choose an appropriate neighborhood structure and oversee the search process [18], [28]. Future research could focus on developing neighborhood operators in conjunction with Kempe-chain operators [48], [58]. Introducing diverse neighborhood variations, such as shuffling periods instead of relocating exams, may prove valuable. Additionally, investigating the adaptation of these neighborhoods to handle weakly connected subgraphs could be an intriguing avenue for exploration [67].

The prevailing trend in timetabling research emphasizes a noticeable inclination towards metaheuristic methods and their hybridizations and MO approaches. Specifically, the judicious fine-tuning of the metaheuristic approach has showcased superior efficacy across most instances. However, MO approaches, proven more efficacious in a few instances [77], continue to be extensively utilized in addressing real-world timetabling scenarios, primarily attributed to the prevalence of small to medium-sized faculty-level exam timetabling instances. This preference stems

from the inherent scalability of such methods, enabling the efficient generation of high-quality solutions within manageable timeframes. Despite the assurance of optimal solutions offered by MO approaches, their widespread adoption faces hindrances due to limited scalability in dealing with large-scale problems. However, a phased approach in MO can be employed for problem decomposition [52]. Nevertheless, recent strides in computational power and solver efficacy are progressively expanding the potential and practicality of MO in addressing larger real-world timetabling challenges [78].

Most real-world timetabling in the reviewed studies is conducted on a faculty or departmental basis. Addressing challenges across various departments within an institution through an integrated approach and resolving them constitutes a potential research area aiming to foster efficient and cohesive management. Local search algorithms are notable for their remarkable efficacy in addressing large-scale problems with only moderate CPU time requirements [40]. The hybridization of vital elements from local search-based methods with population-based methods [24] and hyper-heuristics has proven particularly effective in handling extensive datasets, contributing to optimizing solutions for complex, real-world scenarios.

Instead of uniformly distributed study periods, it is pertinent to recognize that students devote more time to preparation for demanding examinations characterized by substantial content, which underscores the need for researchers to construct a framework endorsing an extended study duration before intricate exams, juxtaposed with a briefer preparatory period for more straightforward exams [77].

Another avenue for advancing the exam timetabling field involves conducting technically enriched surveys through the implementation of methodologies that emphasize state-of-the-art algorithms and rigorously test their performances on benchmarks.

VIII. CONCLUSION

The survey primarily concentrates on high-quality journal articles published between 2012 and 2023. We identify emerging solution methodologies and problem variants within the ETP domain, providing potential avenues for further development to interested readers. The methodologies are categorized based on temporal intervals, the analytical techniques employed, and the datasets utilized. The problem formulations articulated in this paper are concomitant with a dedicated benchmark dataset meticulously tailored to the specific constraints inherent in the considered ETP variant. Despite over a decade since their publication, benchmark instances for the specified exam variations remain persistently challenging. Hence, there is a recognized need for new benchmarks that consider resource-related constraints to improve resource management, optimize utilization, and narrow the gap between applicable strategies and theoretical achievements. We hope that this survey will assist researchers in identifying unexplored and challenging problem variants or emerging, innovative solution methods within the expansive and rapidly evolving domain of the ETP.

REFERENCES

[1] A. Schaerf, "A survey of automated timetabling," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 87–127, 1999, doi: 10.1023/a:1006576209967.

[2] R. Qu, E. K. Burke, B. McCollum, L. T. G. Merlot, and S. Y. Lee, "A survey of search methodologies and automated system development for examination timetabling," *J. Scheduling*, vol. 12, no. 1, pp. 55–89, Feb. 2009, doi: 10.1007/s10951-008-0077-5.

[3] M. W. Carter, "OR practice—A survey of practical applications of examination timetabling algorithms," *Oper. Res.*, vol. 34, no. 2, pp. 193–202, Apr. 1986, doi: 10.1287/opre.34.2.193.

[4] M. W. Carter and G. Laporte, "Recent developments in practical examination timetabling," in *Proc. Int. Conf. Pract. Theory Automated Timetabling*, 1995, pp. 1–21, doi: 10.1007/3-540-61794-9_49.

[5] B. A. Aldeeb, "A comprehensive review of uncapacitated university examination timetabling problem," *Int. J. Appl. Eng. Res.*, vol. 14, no. 24, pp. 4524–4547, 2019. Accessed: Oct. 15, 2023. [Online]. Available: https://www.ripublication.com/ijaer19/ijaerv14n24_14.pdf

[6] N. Pillay, "Evolving hyper-heuristics for the uncapacitated examination timetabling problem," *J. Oper. Res. Soc.*, vol. 63, no. 1, pp. 47–58, Jan. 2012, doi: 10.1057/jors.2011.12.

[7] N. R. Sabar, M. Ayob, R. Qu, and G. Kendall, "A graph coloring constructive hyper-heuristic for examination timetabling problems," *Int. J. Speech Technol.*, vol. 37, no. 1, pp. 1–11, Jul. 2012, doi: 10.1007/s10489-011-0309-9.

[8] P. Demeester, B. Bilgin, P. De Causmaecker, and G. V. Berghe, "A hyper-heuristic approach to examination timetabling problems: Benchmarks and a new problem from practice," *J. Scheduling*, vol. 15, no. 1, pp. 83–103, Feb. 2012, doi: 10.1007/s10951-011-0258-5.

[9] E. K. Burke, G. Kendall, M. Mısır, and E. Özcan, "Monte Carlo hyper-heuristics for examination timetabling," *Ann. Oper. Res.*, vol. 196, no. 1, pp. 73–90, Jul. 2012, doi: 10.1007/s10479-010-0782-2.

[10] N. R. Sabar, M. Ayob, G. Kendall, and R. Qu, "A honey-bee mating optimization algorithm for educational timetabling problems," *Eur. J. Oper. Res.*, vol. 216, no. 3, pp. 533–543, Feb. 2012, doi: 10.1016/j.ejor.2011.08.006.

[11] S. Abdullah and H. Turabieh, "On the use of multi neighbourhood structures within a tabu-based memetic approach to university timetabling problems," *Inf. Sci.*, vol. 191, pp. 146–168, May 2012, doi: 10.1016/j.ins.2011.12.018.

[12] M. A. Ahandani, M. T. V. Baghmisheh, M. A. B. Zadeh, and S. Ghaemi, "Hybrid particle swarm optimization transplanted into a hyper-heuristic structure for solving examination timetabling problem," *Swarm Evol. Comput.*, vol. 7, pp. 21–34, Dec. 2012, doi: 10.1016/j.swevo.2012.06.004.

[13] T. C. Pais and P. Amaral, "Managing the Tabu list length using a fuzzy inference system: An application to examination timetabling," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 341–363, Apr. 2012, doi: 10.1007/s10479-011-0867-6.

[14] E. K. Burke, N. Pham, R. Qu, and J. Yellen, "Linear combinations of heuristics for examination timetabling," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 89–109, Apr. 2012, doi: 10.1007/s10479-011-0854-y.

[15] C. Gogos, P. Alefragis, and E. Housos, "An improved multi-staged algorithmic process for the solution of the examination timetabling problem," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 203–221, Apr. 2012, doi: 10.1007/s10479-010-0712-3.

[16] A. Soghier and R. Qu, "Adaptive selection of heuristics for assigning time slots and rooms in exam timetables," *Int. J. Speech Technol.*, vol. 39, no. 2, pp. 438–450, Sep. 2013, doi: 10.1007/s10489-013-0422-z.

[17] N. R. Sabar, M. Ayob, G. Kendall, and R. Qu, "Grammatical evolution hyper-heuristic for combinatorial optimization problems," *IEEE Trans. Evol. Comput.*, vol. 17, no. 6, pp. 840–861, Dec. 2013, doi: 10.1109/TEVC.2013.2281527.

[18] S. Abdullah and M. Alzaqebah, "A hybrid self-adaptive bees algorithm for examination timetabling problems," *Appl. Soft Comput.*, vol. 13, no. 8, pp. 3608–3620, Aug. 2013, doi: 10.1016/j.asoc.2013.04.010.

[19] S. A. Rahman, A. Bargiela, E. K. Burke, E. Özcan, B. McCollum, and P. Mcmullan, "Adaptive linear combination of heuristic orderings in constructing examination timetables," *Eur. J. Oper. Res.*, vol. 232, no. 2, pp. 287–297, Jan. 2014, doi: 10.1016/j.ejor.2013.06.052.

[20] E. K. Burke, R. Qu, and A. Soghier, "Adaptive selection of heuristics for improving exam timetables," *Ann. Oper. Res.*, vol. 218, no. 1, pp. 129–145, Jul. 2014, doi: 10.1007/s10479-012-1140-3.

[21] S. Abdul-Rahman, E. K. Burke, A. Bargiela, B. McCollum, and E. Özcan, "A constructive approach to examination timetabling based on adaptive decomposition and ordering," *Ann. Oper. Res.*, vol. 218, no. 1, pp. 3–21, Jul. 2014, doi: 10.1007/s10479-011-0999-8.

[22] M. N. M. Kahar and G. Kendall, "Universiti Malaysia Pahang examination timetabling problem: Scheduling invigilators," *J. Oper. Res. Soc.*, vol. 65, no. 2, pp. 214–226, Feb. 2014, doi: 10.1057/jors.2012.6.

[23] C. W. Fong, H. Asmuni, B. McCollum, P. Mcmullan, and S. Omatu, "A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems," *Inf. Sci.*, vol. 283, pp. 1–21, Nov. 2014, doi: 10.1016/j.ins.2014.05.039.

[24] M. A. Al-Betar, A. T. Khader, and I. A. Doush, "Memetic techniques for examination timetabling," *Ann. Oper. Res.*, vol. 218, no. 1, pp. 23–50, Jul. 2014, doi: 10.1007/s10479-013-1500-7.

[25] A. Elloumi, H. Kamoun, B. Jarboui, and A. Dammak, "The classroom assignment problem: Complexity, size reduction and heuristics," *Appl. Soft Comput.*, vol. 14, pp. 677–686, Jan. 2014, doi: 10.1016/j.asoc.2013.09.003.

[26] N. R. Sabar, M. Ayob, G. Kendall, and R. Qu, "A dynamic multiarmed bandit-gene expression programming hyper-heuristic for combinatorial optimization problems," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 217–228, Feb. 2015, doi: 10.1109/TCYB.2014.2323936.

[27] R. Qu, N. Pham, R. Bai, and G. Kendall, "Hybridising heuristics within an estimation distribution algorithm for examination timetabling," *Appl. Intell.*, vol. 42, no. 4, pp. 679–693, Jun. 2015, doi: 10.1007/s10489-014-0615-0.

[28] M. Alzaqebah and S. Abdullah, "Hybrid bee colony optimization for examination timetabling problems," *Comput. Oper. Res.*, vol. 54, pp. 142–154, Feb. 2015, doi: 10.1016/j.cor.2014.09.005.

[29] C. W. Fong, H. Asmuni, and B. McCollum, "A hybrid swarm-based approach to university timetabling," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 870–884, Dec. 2015, doi: 10.1109/TEVC.2015.2411741.

[30] J. Li, R. Bai, Y. Shen, and R. Qu, "Search with evolutionary ruin and stochastic rebuild: A theoretic framework and a case study on exam timetabling," *Eur. J. Oper. Res.*, vol. 242, no. 3, pp. 798–806, May 2015, doi: 10.1016/j.ejor.2014.11.002.

[31] M. N. M. Kahar and G. Kendall, "A great deluge algorithm for a real-world examination timetabling problem," *J. Oper. Res. Soc.*, vol. 66, no. 1, pp. 116–133, Jan. 2015, doi: 10.1057/jors.2012.169.

[32] T. Arbaoui, J.-P. Boufflet, and A. Moukrim, "Preprocessing and an improved MIP model for examination timetabling," *Ann. Oper. Res.*, vol. 229, no. 1, pp. 19–40, Jun. 2015, doi: 10.1007/s10479-015-1832-6.

[33] Y. Bykov and S. Petrovic, "A step counting Hill climbing algorithm applied to university examination timetabling," *J. Scheduling*, vol. 19, no. 4, pp. 479–492, Aug. 2016, doi: 10.1007/s10951-016-0469-x.

[34] P. Amaral and T. C. Pais, "Compromise ratio with weighting functions in a Tabu search multi-criteria approach to examination timetabling," *Comput. Oper. Res.*, vol. 72, pp. 160–174, Aug. 2016, doi: 10.1016/j.cor.2016.02.012.

[35] E. K. Burke and Y. Bykov, "An adaptive flex-deluge approach to university exam timetabling," *INFORMS J. Comput.*, vol. 28, no. 4, pp. 781–794, Nov. 2016, doi: 10.1287/ijoc.2015.0680.

[36] G. Woumans, L. De Boeck, J. Beliën, and S. Creemers, "A column generation approach for solving the examination-timetabling problem," *Eur. J. Oper. Res.*, vol. 253, no. 1, pp. 178–194, Aug. 2016, doi: 10.1016/j.ejor.2016.01.046.

[37] F. Cavdur and M. Kose, "A fuzzy logic and binary-goal programming-based approach for solving the exam timetabling problem to create a balanced-exam schedule," *Int. J. Fuzzy Syst.*, vol. 18, no. 1, pp. 119–129, Feb. 2016, doi: 10.1007/s40815-015-0046-z.

[38] T. Müller, "Real-life examination timetabling," *J. Scheduling*, vol. 19, no. 3, pp. 257–270, Jun. 2016, doi: 10.1007/s10951-014-0391-z.

[39] M. Battistutta, A. Schaerf, and T. Urli, "Feature-based tuning of single-stage simulated annealing for examination timetabling," *Ann. Oper. Res.*, vol. 252, no. 2, pp. 239–254, May 2017, doi: 10.1007/s10479-015-2061-8.

[40] E. K. Burke and Y. Bykov, "The late acceptance hill-climbing heuristic," *Eur. J. Oper. Res.*, vol. 258, no. 1, pp. 70–78, Apr. 2017, doi: 10.1016/j.ejor.2016.07.012.

[41] A. Muklason, A. J. Parkes, E. Özcan, B. McCollum, and P. McMullan, "Fairness in examination timetabling: Student preferences and extended formulations," *Appl. Soft Comput.*, vol. 55, pp. 302–318, Jun. 2017, doi: 10.1016/j.asoc.2017.01.026.

[42] Y. Lei, M. Gong, L. Jiao, J. Shi, and Y. Zhou, "An adaptive coevolutionary memetic algorithm for examination timetabling problems," *Int. J. Bio-Inspired Comput.*, vol. 10, no. 4, p. 248, 2017, doi: 10.1504/ijbic.2017.087918.

[43] A. Cataldo, J.-C. Ferrer, J. Miranda, P. A. Rey, and A. Sauré, "An integer programming approach to curriculum-based examination timetabling," *Ann. Oper. Res.*, vol. 258, no. 2, pp. 369–393, Nov. 2017, doi: 10.1007/s10479-016-2321-2.

[44] Y. Lei, J. Shi, and Z. Yan, "A memetic algorithm based on MOEA/D for the examination timetabling problem," *Soft Comput.*, vol. 22, no. 5, pp. 1511–1523, Mar. 2018, doi: 10.1007/s00500-017-2886-y.

[45] N. Leite, C. M. Fernandes, F. Melício, and A. C. Rosa, "A cellular memetic algorithm for the examination timetabling problem," *Comput. Oper. Res.*, vol. 94, pp. 118–138, Jun. 2018, doi: 10.1016/j.cor.2018.02.009.

[46] M. E. Keskin, A. Döyen, H. Akyer, and M. G. Güler, "Examination timetabling problem with scarce resources: A case study," *Eur. J. Ind. Eng.*, vol. 12, no. 6, p. 855, 2018, doi: 10.1504/ejie.2018.096394.

[47] N. Pillay and E. Özcan, "Automated generation of constructive ordering heuristics for educational timetabling," *Ann. Oper. Res.*, vol. 275, no. 1, pp. 181–208, Apr. 2019, doi: 10.1007/s10479-017-2625-x.

[48] N. Leite, F. Melício, and A. C. Rosa, "A fast simulated annealing algorithm for the examination timetabling problem," *Expert Syst. Appl.*, vol. 122, pp. 137–151, May 2019, doi: 10.1016/j.eswa.2018.12.048.

[49] S. L. Tilahun, "Prey-predator algorithm for discrete problems: A case for examinationtimetabling problem," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 27, no. 2, pp. 950–960, Mar. 2019, doi: 10.3906/elk-1809-175.

[50] T. Arbaoui, J.-P. Boufflet, and A. Moukrim, "Lower bounds and compact mathematical formulations for spacing soft constraints for university examination timetabling problems," *Comput. Oper. Res.*, vol. 106, pp. 133–142, Jun. 2019, doi: 10.1016/j.cor.2019.02.013.

[51] M. G. Güler and E. Geçici, "A spreadsheet-based decision support system for examination timetabling," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 28, no. 3, pp. 1584–1598, May 2020, doi: 10.3906/elk-1909-14.

[52] F. Al-Hawari, M. Al-Ashi, F. Abawi, and S. Alouneh, "A practical three-phase ILP approach for solving the examination timetabling problem," *Int. Trans. Oper. Res.*, vol. 27, no. 2, pp. 924–944, Mar. 2020, doi: 10.1111/itor.12471.

[53] X. Hao, R. Qu, and J. Liu, "A unified framework of graph-based evolutionary multitasking hyper-heuristic," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 35–47, Feb. 2021, doi: 10.1109/TEVC.2020.2991717.

[54] G. Mweshi and N. Pillay, "An improved grammatical evolution approach for generating perturbative heuristics to solve combinatorial optimization problems," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113853, doi: 10.1016/j.eswa.2020.113853.

[55] R. Bellio, S. Ceschia, L. Di Gaspero, and A. Schaerf, "Two-stage multi-neighborhood simulated annealing for uncapacitated examination timetabling," *Comput. Oper. Res.*, vol. 132, Aug. 2021, Art. no. 105300, doi: 10.1016/j.cor.2021.105300.

[56] M. G. Güler, E. Geçici, T. Köroglu, and E. Becit, "A Web-based decision support system for examination timetabling," *Expert Syst. Appl.*, vol. 183, Nov. 2021, Art. no. 115363, doi: 10.1016/j.eswa.2021.115363.

[57] B. A. Aldeeb, M. A. Al-Betar, N. M. Norwawi, K. A. Alissa, M. K. Alsmadi, A. A. Hazaymeh, and M. Alzaqebah, "Hybrid intelligent water drops algorithm for examination timetabling problem," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 8, pp. 4847–4859, Sep. 2022, doi: 10.1016/j.jksuci.2021.06.016.

[58] R. Nand, B. Sharma, and K. Chaudhary, "An introduction of preference based stepping ahead firefly algorithm for the uncapacitated examination timetabling," *PeerJ Comput. Sci.*, vol. 8, p. e1068, Sep. 2022, doi: 10.7717/peerj-cs.1068.

[59] M. W. Carter, G. Laporte, and S. Y. Lee, "Examination timetabling: Algorithmic strategies and applications," *J. Oper. Res. Soc.*, vol. 47, no. 3, pp. 373–383, Mar. 1996, doi: 10.1057/jors.1996.37.

[60] A. J. Parkes and E. Zcan. (2010). *Properties of Yeditepe Examination Timetabling Benchmark Instances*. Accessed: Oct. 15, 2023. [Online]. Available: https://www.patatconference.org/patat2010/proceedings/3_19.pdf

[61] B. McCollum, A. Schaerf, B. Paechter, P. Mcmullan, R. Lewis, A. J. Parkes, L. D. Gaspero, R. Qu, and E. K. Burke, "Setting the research agenda in automated timetabling: The second international timetabling competition," *INFORMS J. Comput.*, vol. 22, no. 1, pp. 120–130, Feb. 2010, doi: 10.1287/ijoc.1090.0320.

[62] E. K. Burke, J. P. Newall, and R. F. Weare, "A memetic algorithm for university exam timetabling," in *Proc. Int. Conf. Pract. Theory Automated Timetabling*, 1995, pp. 241–250, doi: 10.1007/3-540-61794-9_63.

[63] L. T. Merlot, N. Boland, B. D. Hughes, and P. J. Stuckey, "A hybrid algorithm for the examination timetabling problem," in *Proc. Int. Conf. Pract. Theory Automated Timetabling*, Gent, Belgium. Cham, Switzerland: Springer, Aug. 2002, pp. 207–231, doi: 10.1007/978-3-540-45157-0_14.

[64] E.-G. Talbi, "Combining metaheuristics with mathematical programming, constraint programming and machine learning," *Ann. Oper. Res.*, vol. 240, no. 1, pp. 171–215, May 2016, doi: 10.1007/s10479-015-2034-y.

[65] B. McCollum, P. McMullan, A. J. Parkes, E. K. Burke, and R. Qu, "A new model for automated examination timetabling," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 291–315, Apr. 2012, doi: 10.1007/s10479-011-0997-x.

[66] M. Sel, "Invigilators assignment in practical exam timetabling problems," *Int. J. Ind. Eng., Theory Appl. Pract.*, vol. 29, no. 3, pp. 351–371, 2022, doi: 10.23055/ijietap.2022.29.3.6943.

[67] D. Van Bulck, D. Goossens, and A. Schaerf, "Multi-neighbourhood simulated annealing for the ITC-2007 capacitated examination timetabling problem," *J. Scheduling*, 2023, doi: 10.1007/s10951-023-00799-1.

[68] O. A. Kasm, B. Mohandes, A. Diabat, and S. El Khatib, "Exam timetabling with allowable conflicts within a time window," *Comput. Ind. Eng.*, vol. 127, pp. 263–273, Jan. 2019, doi: 10.1016/j.cie.2018.11.037.

[69] B. Bilgin, E. zcan, and E. E. Korkmaz, "An experimental study on hyper-heuristics and exam timetabling," in *Proc. Int. Conf. Pract. Theory Automated Timetabling*, Brno, Czech Republic. Cham, Switzerland: Springer, Sep. 2006, pp. 394–412, doi: 10.1007/978-3-540-77345-0_25.

[70] R. Lewis, "A survey of metaheuristic-based techniques for university timetabling problems," *OR Spectr.*, vol. 30, no. 1, pp. 167–190, Nov. 2007, doi: 10.1007/s00291-007-0097-0.

[71] E. Zcan, "Towards an XML-based standard for timetabling problems: TTML," in *Multidisciplinary Scheduling: Theory and Applications*, Nottingham, U.K. Cham, Switzerland: Springer, Aug. 2005, pp. 163–185, doi: 10.1007/0-387-27744-7_8.

[72] N. Pillay, "A review of hyper-heuristics for educational timetabling," *Ann. Oper. Res.*, vol. 239, no. 1, pp. 3–38, Apr. 2016, doi: 10.1007/s10479-014-1688-1.

[73] A. Bashab, A. O. Ibrahim, E. E. AbedElgabar, M. A. Ismail, A. Elsafi, A. Ahmed, and A. Abraham, "A systematic mapping study on solving university timetabling problems using meta-heuristic algorithms," *Neural Comput. Appl.*, vol. 32, no. 23, pp. 17397–17432, Dec. 2020, doi: 10.1007/s00521-020-05110-3.

[74] S. Ceschia, L. Di Gaspero, and A. Schaerf, "Educational timetabling: Problems, benchmarks, and state-of-the-art results," *Eur. J. Oper. Res.*, vol. 308, no. 1, pp. 1–18, Jul. 2023, doi: 10.1016/j.ejor.2022.07.011.

[75] E. Ozcan and E. Ersoy, "Final exam scheduler–FES," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2005, pp. 1356–1363, doi: 10.1109/CEC.2005.1554848.

[76] M. Carlsson, S. Ceschia, L. Di Gaspero, R. Ø. Mikkelsen, A. Schaerf, and T. J. R. Stidsen, "Exact and metaheuristic methods for a real-world examination timetabling problem," *J. Scheduling*, vol. 26, no. 4, pp. 353–367, Aug. 2023, doi: 10.1007/s10951-023-00778-6.

[77] S. Bazari, A. Pooya, O. Soleimani Fard, and P. Roozkhosh, "Modeling and solving the problem of scheduling university exams in terms of new constraints on the conflicts of professors' exams and the concurrence of exams with common questions," *OPSEARCH*, vol. 60, no. 2, pp. 877–915, Jun. 2023, doi: 10.1007/s12597-023-00638-z.

[78] T. Godwin, "Obtaining quality business school examination timetable under heterogeneous elective selections through surrogacy," *OPSEARCH*, vol. 59, no. 3, pp. 1055–1093, Sep. 2022, doi: 10.1007/s12597-022-00590-4.

[79] M. Battistutta, S. Ceschia, F. De Cesco, L. Di Gaspero, A. Schaerf, and E. Topan, "Local search and constraint programming for a real-world examination timetabling problem," in *Proc. Int. Conf. Integr. Constraint Program., Artif. Intell., Oper. Res.* Cham, Switzerland: Springer, 2020, pp. 69–81, doi: 10.1007/978-3-030-58942-4_5.

[80] T. O. Ting, X.-S. Yang, S. Cheng, and K. Huang, "Hybrid metaheuristic algorithms: Past, present, and future," in *Studies in Computational Intelligence*. Cham, Switzerland: Springer, 2015, pp. 71–83, doi: 10.1007/978-3-319-13826-8_4.

[81] C. Huang, Y. Li, and X. Yao, "A survey of automatic parameter tuning methods for metaheuristics," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 201–216, Apr. 2020, doi: 10.1109/TEVC.2019.2921598.

[82] C. Gogos, G. Goulas, P. Alefragis, V. Kolonias, and E. Housos, "Distributed scatter search for the examination timetabling problem," in *Proc. PATAT*, 2010, p. 211.

[83] S. Abdipoor, R. Yaakob, S. L. Goh, and S. Abdullah, "Meta-heuristic approaches for the university course timetabling problem," *Intell. Syst. With Appl.*, vol. 19, Sep. 2023, Art. no. 200253, doi: 10.1016/j.iswa.2023.200253.

[84] J. S. Tan, S. L. Goh, S. Sura, G. Kendall, and N. R. Sabar, "Hybrid particle swarm optimization with particle elimination for the high school timetabling problem," *Evol. Intell.*, vol. 14, no. 4, pp. 1915–1930, Dec. 2021, doi: 10.1007/s12065-020-00473-x.

[85] S. L. Goh, G. Kendall, and N. R. Sabar, "Monte Carlo tree search in finding feasible solutions for course timetabling problem," *Int. J. Adv. Sci., Eng. Inf. Technol.*, vol. 9, no. 6, pp. 1936–1943, Dec. 2019, doi: 10.18517/ijaseit.9.6.10224.

[86] C. H. Wong, S. L. Goh, and J. Likoh, "A genetic algorithm for the real-world university course timetabling problem," in *Proc. IEEE 18th Int. Colloq. Signal Process. Appl. (CSPA)*, May 2022, pp. 46–50, doi: 10.1109/CSPA55076.2022.9781907.

[87] 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.

[88] J. S. Tan, S. L. Goh, G. Kendall, and N. R. Sabar, "A survey of the state-of-the-art of optimisation methodologies in school timetabling problems," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113943, doi: 10.1016/j.eswa.2020.113943.

[89] A. Hamdan, S. S. Nah, G. S. Leng, C. K. Leng, and T. W. King, "Recent evolutionary algorithm variants for combinatorial optimization problem," *Appl. Model. Simul.*, vol. 7, pp. 214–238, Dec. 2023.

[90] E. K. Burke, A. J. Eckersley, B. McCollum, S. Petrovic, and R. Qu, "Hybrid variable neighbourhood approaches to university exam timetabling," *Eur. J. Oper. Res.*, vol. 206, no. 1, pp. 46–53, Oct. 2010, doi: 10.1016/j.ejor.2010.01.044.

EMILY SING KIANG SIEW received the bachelor's degree in computer science (specializing in computer networks and security) from the University of Technology Malaysia and the master's degree in advanced information technology from Universiti Malaysia Sarawak, where she is currently pursuing the Ph.D. degree with the Faculty of Computer Science and Information Technology. Her current research interests include meta-heuristic algorithms, with a particular focus on timetabling problems.

SAN NAH SZE received the Ph.D. degree from The University of Sydney, in 2011. She is currently an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak (UNIMAS), and boasts over a decade of experience in research and development. She has presented her work at numerous conferences and contributed to publications in ISI and Scopus-indexed journals. Her research interests encompass vehicle routing, educational timetabling, and heuristics and meta-heuristics.

SAY LENG GOH received the B.I.T. degree (Hons.) from Universiti Malaysia Sabah, Malaysia, the M.Sc. degree from Imperial College London, and the Ph.D. degree from the University of Nottingham. He is currently an Academician with the Faculty of Computing and Informatics, Universiti Malaysia Sabah. His research contributions include publications in prestigious journals, such as the *European Journal of Operational Research*, the *Journal of the Operational Research Society*, and *Expert Systems with Applications*. His current research interests include operational research, artificial intelligence, discrete optimization, timetabling, scheduling, and meta-heuristics.

GRAHAM KENDALL (Senior Member, IEEE) holds the degree (Hons.) in computation from the University of Manchester Institute of Science and Technology (UMIST), the Ph.D. degree in computer science from the University of Nottingham, in 2000, transitioning to an academic role during the Ph.D. studies, spending over two decades split evenly between the U.K. and Malaysia.

He serves as the Deputy Vice Chancellor (Research and Quality Assurance) at MILA University, Malaysia. Previously, he held positions at the University of Nottingham Malaysia (UNM), serving as a Vice-Provost for Research and Knowledge Exchange from 2011 to 2016, followed by Provost/CEO from 2016 to 2021, and also as Pro-Vice Chancellor of the Global University of Nottingham. Before his current role, he served as the Chief Executive of the Good Capitalism Forum (GCF) and as a Senior Vice President of the overarching group, both based in Malaysia. Before relocating to Malaysia in 2011, he was a Professor of computer science at the University of Nottingham. Starting his career as a computer operator, he progressed to senior positions in the Information Technology Sector. In addition to his research pursuits in operations research and artificial intelligence, he has developed interests in publication ethics and bibliometrics, including predatory publishing. His research primarily focuses on applying operations research and artificial intelligence methods based on principles of natural evolution, particularly those inspired by Charles Darwin.

Dr. Kendall was a fellow of the British Computer Society and the Operational Research Society, relinquishing these positions upon leaving academia in 2021. He maintains Honorary and Distinguished Professorial roles at Universities, Hong Kong and India. With about 300 peer-reviewed papers to his credit, he has served on various Editorial Boards, including as Editor-in-Chief of the IEEE Transactions of Computational Intelligence and AI in Games and as an Associate Editor of the *Journal of the Operational Research Society*.

NASSER R. SABAR is currently a Senior Lecturer with the Computer Science and IT Department, La Trobe University, Australia. He has published more than 76 papers in international journals and peer-reviewed conferences. His current research interests include hyper-heuristic frameworks, evolutionary computation, and hybrid algorithms, with a specific interest in big data optimization problems, cloud computing, dynamic optimization, and scheduling problems.

SALWANI ABDULLAH received the B.Sc. degree in computer science from the University of Technology Malaysia, the master's degree in computer science from the National University of Malaysia, and the Ph.D. degree in computer science from the University of Nottingham, U.K. She is currently a Professor of computational optimization with the Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia. Her research interests include artificial intelligence and operation research, particularly computational optimization algorithms (heuristics and meta-heuristics, evolutionary algorithms, and local search techniques. Her work addresses various real-world applications, dealing with single and multi-objective continuous and combinatorial optimization challenges, such as timetabling, scheduling, routing, nurse rostering, dynamic optimization, data mining problems (including feature selection, clustering, classification, and time-series prediction), intrusion detection, and search-based software testing.

• • •