Metaheuristic approach proposal for the solution of the bi-objective course scheduling problem

E. Cana,*, O. Ustuna, and S. Sgalmb

a. Department of Industrial Engineering, Kutahya Dumlupinar University, Kutahya, Turkey.
b. Department of Informatics, Kutahya Dumlupinar University, Kutahya, Turkey.

Received 25 January 2020; received in revised form 27 March 2021; accepted 5 July 2021

KEYWORDS
Multi-objective optimization; Conic scalarization; Tabu search algorithm; Experimental design; Timetabling.

Abstract. Timetabling problems are among the commonly encountered problems in real life, from education institutions to airline companies. It is generally difficult to obtain optimal solutions to timetabling problems that vary in terms of structures of constraints and objective functions, and these problems belong to the NP-hard category, which means that they cannot be solved in polynomial time in real life. In this study, a bi-objective mathematical model is proposed for a course scheduling problem at Kutahya Dumlupinar University, Department of Industrial Engineering. The first objective function aims to maximize the sum of the preferences of instructors determined using the Analytic Hierarchy Process method, and the second objective function is to minimize the cases of course overlap for students. Conic scalarization method is used to combine the objective functions. Due to NP-hard nature of the problem, the Tabu Search Algorithm as a metaheuristic approach is used to solve it. Using the obtained data and by considering the proposed bi-objective mathematical model, the Tabu Search Algorithm is designed for the problem and dealt with in the Excel Visual Basic program. The experimental results are evaluated through Analysis of Variance using Minitab Program. Based on the comparison of the results, satisfactory solutions are obtained.

© 2023 Sharif University of Technology. All rights reserved.

1. Introduction

Nowadays, despite the rapid development and advancement of management information technologies, many sectors still depend on manual execution instead of using computer software, which causes serious time losses. Eliminating the problems and losses occurring in the scheduling process and ensuring the flow without interruption are possible with optimizing the timetable according to the objectives of the system.

Timetabling is a decision-making process and is an approach that enables necessary allocations to resources within a certain time span and aims to use efficient time utilization. In general, creating a timetable is classified as the NP-hard problem and scheduling of staff, health, training is each examined as sub-headings of timetabling [1].

Course scheduling problems belong to the class of timetabling problems and they are based on determining the hours to be assigned for the courses for each class by providing the required constraints [2]. When creating the course schedule, it is a priority rule to determine the hours of all the courses of the
institution to be organized and that the prepared course schedule meets the defined needs and constraints [3]. These constraints are often defined as “compulsory (hard)” and “flexible (soft).” Hard constraints must be satisfied under any conditions, whereas soft constraints need to be satisfied as much as possible and are not critical [4]. Various methods such as mathematical modeling, heuristic approaches, local search, evolutionary algorithms, and constraint-based approaches are used for course scheduling problems. In order to solve the university-timetabling problem, Ozturk et al. [5] developed a multi-objective decision-making model and presented an approach considering class capacities and the preferences of instructors for course hours; then, they solved the problem by the weighted-sum scalarization method. Badri [6] developed a two-stage optimization procedure in his multi-objective 0-1 integer-programming model. In the first stage, the author assigned students to the courses by maximizing the course instructors’ preferences through weighting. In the second stage, they tried to maximize the allocation of course-classroom combinations to available time blocks. Upon proposing a three-stage process consisting of the Analytic Hierarchy Process (AHP), Conic Scalarization, and Modified Gradient Method, Ozlemir and Gasimov [7] used this integrated method for the first time to solve a non-convex multi-objective faculty course assignment problem. They evaluated the preferences and desires of instructors and administrators, which were in line with the priorities calculated by the AHP. Then, the objective functions were combined with the conic scalarization method. Using the Tabu Search Algorithm and proposing two different neighborhood structures in addition to simple and swap movements, Aladag et al. [8] solved the problem using four neighborhood structures and compared the results statistically. Akkan and Gulcu [9] investigated the course timetabling problem as a bi-criteria optimization problem and solved it by a hybrid multi-objective genetic algorithm, which makes use of hill climbing and simulated annealing algorithms besides the standard genetic algorithm approach. Jamili et al. [10] presented a multi-objective and comprehensive mathematical model in their study. The first objective function attempts to design a timetable for the instructors which take a longer time to do research works. The second objective function maximizes the sum of instructors’ day and hour preferences. Al-Yakooob and Sherali [11] proposed mathematical programming models; Al-Hadil et al. [12] aimed to reach a solution using a simulated annealing algorithm; and Al-Jarrah et al. [13] performed a timetabling system utilizing genetic algorithms. In their study involving solution methods for timetabling problems, Altunay and Eren [14] classified relevant studies in the literature through various approaches.

In this study, an integrated approach consisting of AHP, Tabu Search Algorithm, and Conic Scalarization is proposed for solving the problem of preparing a timetable, which is a common difficult problem usually encountered in educational institutions. A mathematical model is proposed for the problem of preparing the timetable of undergraduate program at the Department of Industrial Engineering, Kutahya Dumlupinar University and the Tabu Search Algorithm, as one of the metaheuristic approaches, is utilized for this purpose. A questionnaire is developed to determine the preferences of the instructors and analyzing the results using AHP in the first objective function of the bi-objective model formed within the constraints defined in the problem. The sum of day and hour priorities of the instructors strives to be maximized; in the second objective function, the total number of students whose courses overlapped is determined using the Student Information System (SIS) and is to be minimized. The preference priorities obtained by AHP are used in the structure of the objective function. The bi-objective function is turned into a single-objective function by using Conic Scalarization. In this study, the Tabu Search Algorithm is applied to the problem and the best solution is sought. Since an experimental design is needed to determine the factors and levels in the problem, a suitable experimental design is created for the designed model and the results of the experiments are compared and evaluated statistically by using the Analysis of Variance (ANOVA).

The course scheduling problem is tackled as a multi-objective problem and the objective functions are combined using the conic scalarization approach in this paper. The first objective function maximize the sum of the preferences of instructors, while the second objective function aims to minimize the students’ course overlap. Based on a review of the specialized literature, it is noted that the number of applications of the multi-objective metaheuristics to the class of course scheduling problems is scarce. In addition, there is no study on maximizing instructor preference and minimizing course overlap in multi-objective course scheduling problems in the literature. Furthermore, the use of a Tabu Search Algorithm was not encountered in solving multi-objective course scheduling problems. This paper offers a different perspective for researchers in this way.

2. Structure of the course scheduling problem and the mathematical model proposed for solution

The course schedule of Kutahya Dumlupinar University, Faculty of Engineering at the Department of Industrial Engineering is 5 days a week, 10 hours per day, between 08:00 am and 17:00 pm., totalling 50 hours a
week. The program includes 62 courses, 34 instructors, and 9 classrooms (6 classrooms, 1 laboratory, and 2 amphitheatres) to be scheduled. Some instructors give more than one course. The timetable of the department includes the 1st year, 2nd year, 3rd year, and 4th-year courses for spring semester.

At the department, the timetable is done manually by the research assistants assigned at the beginning of each semester. Based on the 2016–2017 Spring Semester course schedule of the department, physical facilities, the limitations of the academic staff and the teaching process of the department are examined and the following information is obtained:

- It is already known in which kind of physical environment (classroom or laboratory) the lessons will be taught;
- The estimated number of students to take the course is determined inferentially and physical environment with an appropriate class capacity is assigned for each course;
- Courses are grouped as Compulsory, Social Elective (Soc. Elect.), Technical Elective (Tech. Elect.), Engineering Project (Eng. Project), and Engineering Design (Eng. Design);
- The curriculum consists of 8 semesters. It is known which courses are given in each semester, and the courses that each student group is to take are determined in advance within the course plan;
- Research assistants in charge of preparing the course program get in touch with each instructor before the start of the semester to be informed about the lessons they will give and the time intervals of the courses;
- Fifty minutes of training and teaching is defined as a lesson hour;
- Courses can be conducted in more than one session depending on the course time and the number of the course sessions is determined by the department. 2- or 3-hour courses can be conducted in one session and 4-hour courses can be divided into two sessions as 2 hours in each session while 5-hour courses can be conducted in two sessions as 3 plus 2 hours. It is also possible to conduct some courses in 4 hours without any alteration;
- The number of sessions for each course and the duration of each session are known in advance.

As a result, in order not to disrupt the instructional plan, it is observed that all compulsory courses that should be opened in the related semester (spring or fall) in the course plan be opened; that the sufficient number of elective courses be opened; and that the course load of the instructors be completed as much as possible.

2.1. Bi-objective 0-1 Integer Mathematical Model for Course Timetabling (IMMCT)

Indices, decision variables, parameters, and clusters.

Indices:

- \( I = \{i | i = 1, 2, \ldots, m\} \) Set of courses
- \( J = \{j | j = 1, 2, \ldots, n\} \) Set of days
- \( K = \{k | k = 1, 2, \ldots, r\} \) Set of daily course hours
- \( N = \{n | n = 1, 2, \ldots, t\} \) Set of student groups
- \( L = \{l | l = 1, 2, \ldots, u\} \) Set of instructors
- \( I' = I, J' = J, K' = K \)

Decision variables

- \( x_{ij} = 1 \), if course \( i \) is assigned to the \( j \)th day and \( k \)th course hours; 0, otherwise:

Parameters

- \( D_i \) Weekly total course hours of the \( i \)th course
- \( V_{i,j'} \) Number of students with overlapping course \( i \) and course \( j' \)
- \( T_{ij} \) jth day preference priority of the instructor of the \( i \)th course
- \( H_{ik} \) kth hour preference priority of the instructor of the \( i \)th course
- \( K_{ij} \) Course hours that the instructor \( l \) does not want to teach on the \( j \)th day
- \( W_{ij} \) Fixed and constant course hours of instructor \( l \) on the \( j \)th day

Subsets used in the model

- \( I_w \) Courses with fixed days and hours
- \( I_l \) Courses given by the instructor \( l \)
- \( I_Z \) Compulsory courses
- \( I_S \) Elective courses

Constraints

While creating the mathematical model, in the light of the data, the instructors, the curriculum period, the non-overlapping of courses, and the course constraints are defined below:

a) Constraints of instructors

The instructors may submit the days and times they do not want to or will not give lessons in advance. Although this constraint is generally considered as an optional soft constraint, it is regarded as a strict constraint in our study (Eq. (1)).

Eq. (2) shows the courses with fixed and constant days and hours. The hours of the courses given by an instructor in a semester should not overlap
(Eq. (3)). This constraint is a strict constraint for all instructors:

$$x_{ijk} = 0, \forall i \in I, (j, k) \in K_{ij},$$

(1)

$$x_{ijk} = 1, \forall i \in I_w, (j, k) \in W_{ij},$$

(2)

$$\sum_{i \in I_w} x_{ijk} \leq 1, \forall (j, k).$$

(3)

strict constraints. Each course is given a number, and the compulsory and elective course clusters are expressed with these numbers given in Table 1.

The sets of the compulsory courses and elective courses are determined as follows:

$$IZ_1 = \{i| i = 1, 2, 3, 4, 5, 6, 7\},$$

$$IS_1 = \{i| i = 20, 21, 22\},$$

$$IZ_2 = \{i| i = 8, 9, 10, 11, 12\},$$

$$IS_2 = \{i| i = 23, 24, 25\},$$

b) Curriculum period course clusters

In the curriculum plan, the compulsory and elective course hours of fall or spring courses of a class must not overlap. This constraint is among the

Table 1. Compulsory and elective courses in the curriculum.

| Compulsory courses                                      | Course no. | Elective courses                                      | Course no. |
|---------------------------------------------------------|------------|------------------------------------------------------|------------|
| Mathematics II (Mat. II)                                | 1          | Soc. Elect. Course II/Entrepreneurship                | 20         |
| Physics II (Phys. II)                                   | 2          | Soc. Elect. Course II/Environmental management systems| 21         |
| Technical Drawing (Tech. Draw.)                         | 3          | Soc. Elect. Course II/Contemporary Approaches in Urban Construction | 22 |
| General Economics (Gen. Eco.)                           | 4          | Tech. Elect. Course II/Object-Oriented programming    | 23         |
| Turkish Language II                                     | 5          | Tech. Elect. Course II/Service systems                | 24         |
| Computer Programming (Comp. Prog.)                      | 6          | Tech. Elect. Course II/Business English II            | 25         |
| English II (Eng. II)                                    | 7          | Tech. Elect. Course IV/Database                       | 26         |
| Statistics I                                            | 8          | Management Systems                                    | 27         |
| Manufacturing methods                                   | 9          | Tech. Elect. Course IV/Logistic management            | 28         |
| Thermodynamics                                          | 10         | Tech. Elect. Course IV/Investment project analysis    | 29         |
| Numerical Analysis                                      | 11         | Tech. Elect. Course VI/Scheduling applications        | 30         |
| History of Ataturk's Principles and Reforms II (HAPR II)| 12         | Tech. Elect. Course VI/Experimental design            | 31         |
| Production management                                   | 13         | Eng. project/Supply chain project                     | 32         |
| Quality control                                         | 14         | Eng. project/Modern manufacturing systems project     | 33         |
| Operation Research II (Op. Res. II)                     | 15         | Eng. project/Manufacturing planning and economy project | 34 |
| Engineering Economics (Eng. Eco.)                       | 16         | Eng. project/Operational research project             | 35         |
| System Simulation (Syst. Sim.)                          | 17         | Eng. project/Management project                       | 36         |
| Management and organization                             | 18         | Eng. design/Supply chain design                       | 37         |
| Ergonomics                                              | 19         | Eng. design/Modern manufacturing systems design       | 38         |
|                                                          |            | Eng. design/Manufacturing planning and economy design | 39         |
|                                                          |            | Eng. design/Operational research design               | 40         |
|                                                          |            | Eng. design/Management design                         | 41         |
\[ I_{Z3} = \{i| i = 13, 14, 15, 16, 17\}, \]
\[ I_{S3} = \{i| i = 26, 27, 28, 29\}, \]
\[ I_{Z4} = \{i| i = 18, 19\}, \]
\[ I_{S4} = \{i| i = 30, 31\}, \]
\[ I = \{I_{Z1} \cup I_{Z2} \cup I_{Z3} \cup I_{Z4}\}, \]
\[ I_{S5} = \{i| i = 32, 33, 34, 35, 36\}, \]
\[ I_{S6} = \{i| i = 37, 38, 39, 40, 41\}, \]
\[ I = \{I_{S2} \cup I_{S3} \ldots \cup I_{S6}\} \].

In the compulsory course set, \( I_{Z1} \) denotes the first year spring semester compulsory courses, \( I_{Z2} \) the second year spring semester compulsory courses, \( I_{Z3} \) the third year spring semester compulsory courses, and \( I_{Z4} \) the fourth year spring semester compulsory courses.

In the elective course set, \( I_{S1} \) denotes the second year spring semester elective courses, \( I_{S2} \) the third year spring semester elective courses, \( I_{S3} \) the fourth year spring semester 1st group elective courses, \( I_{S4} \) the fourth year spring semester 2nd group elective courses, \( I_{S5} \) the fourth year spring semester 3rd group elective courses, and \( I_{S6} \) the fourth year spring semester 4th group elective courses. The fourth year 3rd and 4th group elective courses \( (I_{S5}, I_{S6}) \) in the curriculum are not included in the constraints since they are scheduled on Saturday.

c) Non-overlapping constraints of courses
Combination of the compulsory and elective course sets is equal to the whole set of courses \( (Z \cup S = I) \). Eqs. (4), (5), (6), and (7) refer to the non-overlapping constraints of the 1st, 2nd, 3rd, and 4th year courses in the curriculum, respectively:
\[
\sum_{i \in I_{Z1}} x_{ijk} \leq 1, \quad \forall (j, k), \quad (4)
\]
\[
x_{r^i j k} + \sum_{i \in I_{Z2}} x_{ijk} \leq 1, \quad \forall (j, k), \quad \forall r^i \in I_{S1}, \quad (5)
\]
\[
x_{r^i j k} + \sum_{i \in I_{Z3}} x_{ijk} \leq 1, \quad \forall (j, k), \quad \forall r^i \in I_{S1}, \quad (6)
\]
\[
x_{r^i j k} + x_{r^i j k} + \sum_{i \in I_{Z4}} x_{ijk} \leq 1, \quad \forall (j, k), \quad \forall r^i \in I_{S1}, \quad \forall r^{i'} \in I_{S4}. \quad (7)
\]
d) Course constraints
Eq. (8) shows that all the courses in the curriculum pre-determined by the department are assigned as many as the course hours. Under the assumption that there are 9 classrooms available for the department. Eq. (9) denotes the constraint of not having over 9 courses in the same class hour. Eq. (10) shows the sequence constraint which ensures that there is no other course between the course blocks in order not to disturb the integrity of the course. Eq. (11) indicates the constraint that guarantees assigning two-course hours consecutively. Eq. (12) denotes the constraint showing that the decision variable consists of 0-1 integer variables. These constraints are among the strict constraints in the model.
\[
\sum_{j=1}^{n} \sum_{k=1}^{r} x_{ijk} = D_i, \quad \forall i, \quad (8)
\]
\[
\sum_{i=1}^{m} x_{ijk} \leq 9; \quad \forall (j, k), \quad (9)
\]
\[
x_{ijk} + x_{ij(k+1)} - x_{i(jk-1)} \leq 0, \quad \forall (i, j, k), \quad (10)
\]
\[
x_{ijk} + x_{ij(k+1)} - x_{i(jk+2)} \leq 0, \quad \forall (i, j, k). \quad (11)
\]
\[
x_{ijk} \text{ : } 0 \leq \text{integer}, \quad \forall (i, j, k). \quad (12)
\]

Objective functions
Eq.(13) shows the first objective function, which enables the maximization of the sum of instructor preferences, and Eq.(14) indicates the second objective function, which enables the minimization of the sum of the overlapping number of students in the consecutive classes.
\[
\text{Max } f_1(x) = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{r} (T_{ij} + H_{ik}) x_{ijk}, \quad (13)
\]
\[
\text{Min } f_2(x) = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{r} \sum_{l=1}^{m} \sum_{n=1}^{r} \sum_{r'=1}^{m} \sum_{k'=1}^{r} V_{ij} x_{ijk} x_{ir'jkr'}. \quad (14)
\]

AHP method is used to determine the course day and class hour preferences of the instructors in the problem. A questionnaire is given to the instructors to make a paired comparison of the determined criteria based on the Saaty [15] 1-9 scale and the consistency ratios are analyzed. In order to minimize the total number of overlapping students in consecutive classes, the overlap numbers are determined from the SIS database and an overlap matrix is created.

Detailed solution steps of the proposed integrated approach using AHP, Tabu Search, and Conic Scalarization Methods are shown in Figure 1.

3. Determining the stochastic representation of the course timetabling by considering the IMMCT constraints
Taking into account the IMMCT constraints, the
stochastic representation of the course timetable is determined. The course list is created by enumerating each course. As the courses conducted in two sessions are considered as two separate groups, different numbers are given and listed. The first three-hour session of the Mathematics II course is ranked first, while the two-hour session is ranked third. Table 2 is created by assigning numbers to courses, starting from the first-class courses.

The number of random numbers generated is equal to the number of courses, and the first course in the list is assigned to the number with the smallest random number. This process is repeated for all courses in the list, creating a random course order. By arranging the assigned random numbers from small to large, the new sequence of the courses according to random numbers is obtained, as shown in Table 3.

In the course program, 37 one-hour gaps are assigned to have gaps between courses. If these empty hours are not defined, the program would assign the courses continuously and the lessons would stack within the first days of the week and the last days would be empty, leading to an incoherent timetable. The gaps are assigned to each class orderly as one hour.

**Figure 1.** Flowchart of the proposed approach.
Table 2. Numbering the sessions of the courses.

| Course name                  | Course hours | Course session no. | Course name                  | Course hours | Course session no. |
|------------------------------|--------------|--------------------|------------------------------|--------------|--------------------|
| Mathematics II               | 3            | 1                  | Social Elective Course II    | 3            | 15                 |
| Physics II                   | 3            | 2                  | Numerical Analysis           | 3            | 16                 |
| Mathematics II               | 2            | 3                  | HAPRI                        | 2            | 17                 |
| Technical Drawing            | 4            | 4                  | Production Management        | 3            | 18                 |
| Physics II                   | 2            | 5                  | Quality Control              | 4            | 19                 |
| General Economics            | 3            | 6                  | Operation Research II        | 3            | 20                 |
| Turkish Language II          | 2            | 7                  | Engineering Economics        | 3            | 21                 |
| Computer Programming         | 3            | 8                  | Technical Elective Course II | 3            | 22                 |
| Department 1                 |              |                    |                              |              |                    |
| Computer Programming         | 3            | 9                  | Operation Research II        | 2            | 23                 |
| Department 2                 | 4            | 10                 | System Simulation            | 3            | 24                 |
| English II                   | 2            | 11                 | Technical Elective Course IV | 3            | 25                 |
| Statistics I                 | 4            | 12                 | Management and Organization  | 3            | 26                 |
| Manufacturing Methods        | 3            | 13                 | Technical Elective Course VI | 3            | 27                 |
| Thermodynamics               | 2            | 14                 | Ergonomics                   | 4            | 28                 |

Table 3. The new sequence obtained by sorting random course numbers from small to large.

| Course no. | Random no. | Course no. | Random no. | Course no. | Random no. | Course no. | Random no. | Course no. | Random no. |
|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| 1          | 0.002      | 32         | 0.232      | 7          | 0.427      | 45         | 0.615      | 55         | 0.863      |
| 17         | 0.004      | 8          | 0.274      | 9          | 0.436      | 25         | 0.633      | 21         | 0.878      |
| 19         | 0.047      | 4          | 0.284      | 40         | 0.44       | 46         | 0.641      | 56         | 0.879      |
| 26         | 0.053      | 33         | 0.287      | 41         | 0.441      | 47         | 0.645      | 57         | 0.882      |
| 2          | 0.07       | 34         | 0.306      | 42         | 0.461      | 48         | 0.67       | 58         | 0.887      |
| 12         | 0.104      | 35         | 0.314      | 14         | 0.463      | 49         | 0.677      | 59         | 0.915      |
| 18         | 0.105      | 36         | 0.326      | 13         | 0.464      | 16         | 0.694      | 60         | 0.919      |
| 6          | 0.118      | 37         | 0.327      | 43         | 0.486      | 50         | 0.737      | 27         | 0.919      |
| 29         | 0.137      | 11         | 0.329      | 22         | 0.5        | 51         | 0.748      | 61         | 0.936      |
| 3          | 0.162      | 15         | 0.346      | 23         | 0.518      | 52         | 0.768      | 62         | 0.943      |
| 5          | 0.184      | 38         | 0.369      | 44         | 0.521      | 28         | 0.788      | 63         | 0.95       |
| 30         | 0.195      | 20         | 0.389      | 10         | 0.532      | 53         | 0.847      | 64         | 0.956      |
| 31         | 0.197      | 39         | 0.411      | 24         | 0.575      | 54         | 0.858      | 65         | 0.989      |

During the course assignment process, the overlap of the courses at distinct classes at the same time for the instructors with more than one course is prevented.

The courses listed according to the random numbers are placed in the classes in order of their numbers in Table 4. “Mathematics II” is placed at 08:00, 09:00, and 10:00 hours on Monday because the course’s number is ranked as number 1. Given that “History of Ataturk’s Principles and Reforms II” is a second-year course, it is assigned to 08:00 and 09:00 hours on Monday. The course assignments are carried out orderly, and each class is assigned an empty course for one hour when the course number 29 “vacant hour” is reached. Also, when assigning the courses with course hours more than 3 in two separate sessions to the program, positioning is done by considering the course with more course hours before. For example, 5-hour Mathematics II is divided into two sessions as 3 and 2 hours and the 3-hour session of this course has been assigned before.


| Courses                  | Course no. | Day   | Hours  | Class I         | Class II          | Class III         | Class IV                        |
|--------------------------|------------|-------|--------|-----------------|-------------------|-------------------|---------------------------------|
| Mathematics II           | 1          | 08:00 | Mathematics II | HAPR II 17      | Quality Control 19 | Management and Organization 26 |
| HAPR II                  | 17         | 09:00 | Mathematics II | HAPR II 17      | Quality Control 19 | Management and Organization 26 |
| Quality Control          | 19         | 10:00 | Mathematics II | Manufacturing   | Quality Control 19 | Management and Organization 26 |
| Organization             | 26         | 11:00 | Physics II   | Manufacturing   | Quality Control 19 | 29                        |
| Physics II               | 2          | 12:00 | Physics II   | Manufacturing   | Production Management 18 | 30                        |
| Manufacturing Methods    | 12         | 13:00 | Physics II   | Manufacturing   | Production Management 18 | 31                        |
| Production Management    | 18         | 14:00 | General Economics 6 | 29          | Production Management 18 | 32                        |
| General Economics        | 6          | 15:00 | General Economics 6 | 30          | 29                        | 33                        |
| Vacant                   | 20         | 16:00 | General Economics 6 | 31          | 30                        | 34                        |
| Mathematics II           | 3          | 17:00 | 29       | 32              | 31                        | 35                        |

4. Combining and computing objective functions with the conic scalarization method

First of all, an overlap matrix is created by comparing the lists of students who have enrolled in the courses of the successive classes and have courses of the previous spring semester. The preferences of instructors are computed with the AHP method [16]. In the schedule, starting from the second year, first class and third class courses are checked for overlap. The overlapping courses of the successive classes are determined and the optimum position is sought in which they would not overlap with the previous and next class courses in the current column and would best suit the instructor preferences. If not found, secondly, the optimum position is sought which would not overlap with at least one course in the previous and next classes and would best meet the instructor preferences. If no suitable place could be found in this stage, the appointment is carried out by placing the course in the best possible position with overlap check and taking the instructors’ preferences into consideration. After the position change of each course, the objective function is recomputed and a new value is obtained. By checking the overlapping in the neighborhood of the courses, the courses are assigned to the most appropriate places and the objective function is recomputed, yielding the best assignment. A single-objective function is obtained by combining two objective functions through Conic Scalarization. The conic scalarization method allows the finding of all the efficient solutions corresponding to the decision-maker’s preferences in multi-objective optimization problems without convexity and boundedness assumptions [17]. Weights $w_1$ and $w_2$ in the formula are determined by AHP and taken as 0.5. Using Eq. (15), the objective function is scalarized.

$$
\text{Min} KS(x) = \alpha \sum_{k=1}^{2} |f_k(x) - R_k| \\
+ \sum_{k=1}^{2} w_k (f_k(x) - R_k), \quad (15)
$$

Subject to Eqs. (1)-(14).

$(\alpha, w) \in \{0 \leq \alpha < w_k, k = 1, 2\}$ condition must be provided. Parameter values are $\alpha = Enk\{w_1, w_2\} - 0.01 = 0, 49, R_1 = 100, R_2 = 35$, and $w_1 = w_2 = 0.5$; thus, Conic Scalarization Objective Function becomes:

$$
KS(x) = 0.49 \times (|f_1(x) - 100| + |f_2(x) + 35|) \\
+ 0.5 \times (f_1(x) - 100) + 0.5 \times (-f_2(x) + 35).
$$
5. Optimization with the tabu search algorithm

The general steps of the algorithm for the problem-solving are as follows:

Step 1. A random start-up solution is created;
Step 2. The current solution is saved;
Step 3. Simple Move: A random course is selected and moved to a suitable space in its own column (from the same class);
Step 4. If there is no proper space, the course moving operation is cancelled;
Step 5. Randomly, two courses are chosen from the same column (from the same class);
Step 6. Equality of the number of courses’ hours is checked;
Step 7. Swap move: If the course hours are equal, the location of each course hour of the selected courses is swapped;
Step 8. If the course exchange is possible and the objective function improves, these courses are recorded in the tabu list;
Step 9. If the course hours are different from one another and there is no free time slot before and after for the course hours with more hours, there will be no change in courses;
Step 10. If the iteration number has not exceeded the maximum value requested, return to Step 3;

Step 11. After all the procedures are applied, all placement possibilities are checked for all the courses of all classes in the curriculum; when the course placement operation is performed and the maximum iteration is reached, the process is stopped.

5.1. Selection of parameters used in the tabu search algorithm

The success of optimization processes with the Tabu Search Algorithm depends on the correct determination of the used parameters given below:

a) Selection of movement

A movement selection mechanism is created when the algorithm is applied to the problem. Two types of movements are used for the course selection and taking courses to the tabu list. The first of these movements is the swap move. A random class is selected from a randomly generated course program. Two courses are selected randomly from the selected classes. The equality of the number of course hours for selected courses is checked. If the course hours are not equal, it will be checked whether there is a sufficient vacant time before and after the courses are replaced considering the hours of the course that would be replaced. If the calculated value is better than the objective function value of the random solution created at the beginning, the two courses selected in this stage would be kept constant after the swap and these courses would be taken into the tabu list (Figure 2).

The study is repeated with a small sample

| DAY | HOUR | CLASS I         | CLASS II         | CLASS III          | CLASS IV                        |
|-----|------|----------------|-----------------|--------------------|---------------------------------|
|     |      | Mathematics II | H.A.P.R.II       | Quality Control    | Management and Organization     |
|     | 8:00 | Mathematics II | H.A.P.R.II       | Quality Control    | Management and Organization     |
|     | 9:00 | Mathematics II | H.A.P.R.II       | Quality Control    | Management and Organization     |
|     | 10:00| Mathematics II |                 | Quality Control    | Management and Organization     |
|     | 11:00| Physics II     |                 | Quality Control    |                                 |
|     | 12:00| Physics II     |                 | Production Management|                              |
|     | 13:00| Physics II     |                 | Production Management|                              |
|     | 14:00| General Economics| Manufacturing Methods| Production Management|                              |
|     | 15:00| General Economics| Manufacturing Methods|                     |                                |
|     | 16:00| General Economics| Manufacturing Methods|                     |                                |
|     | 17:00|                 | Manufacturing Methods|                     |                                |
|     | 8:00 | Mathematics II |                 |                     |                                |
|     | 9:00 | Mathematics II | Statistics I     |                     |                                |
|     | 10:00| Physics II     | Statistics I     |                     |                                |
|     | 11:00| Physics II     |                 |                     |                                |
|     | 12:00|                 |                 |                     |                                |
|     | 13:00|                 |                 |                     |                                |
|     | 14:00|                 |                 |                     |                                |
|     | 15:00| Computer Programming| Soc. Elect. Course II| Operation Research II |                              |
|     | 16:00| Computer Programming| Soc. Elect. Course II| Operation Research II |                              |
|     | 17:00| Computer Programming| Soc. Elect. Course II| Operation Research II |                              |

**Figure 2.** Demonstration of the swap move on a small sample.
to see the effect of swap move on the program. Industrial Engineering fourth-year courses are chosen to be held on Monday through Friday to see the effects of swap move. Using only the swap move procedure, the program is run using the normal program for each tabu length and number of iterations. The obtained results are graphed by evaluating the first case and the situation after the experiments. In this way, the effect of swap move procedure on the conic objective function is seen. The tabu length has four levels (4, 6, 8, 12) and the iteration number has eight levels (10, 20, 100, 200, 400, 600, 800, 1000). While the tabu length is 4 and the iteration number is 10, these parameters are shown as (4-10) on the horizontal axis in Figure 3. This notation is repeated in all tabu length-iteration number combinations. Totally, 32 combinations of the tabu length and the iteration number are created in this way.

Another movement type is the simple move based on the movement of courses into any space. In this movement type, all hours of the randomly chosen course are randomly assigned to spaces equal to the number of hours per lesson (Figure 4).

b) Selection of initial course schedule
The experiments are carried out with three separate start-up programs. While a randomly initial appropriate solution is created, experiments are carried out with (a) a single randomly generated program for all experimental groups as the first option, (b) a course schedule optimized with random search for all the experiments as the second option, and (c) randomly generated programs for each experiment as the last option and the results are recorded.

c) Use of candidate list strategy
Because it will take a long time to examine all neighborhood structures to determine a specific solution, random class selection and random course selection are preferred to select the best one. A list
of candidates with as many solutions as the number of courses is used.

d) Selection of the movement rate
Movement assignments are carried out by assigning values between [0-1] for swap move (swapping the courses) and simple move (moving to a vacant position). For example, while the swap move took 0.1 value, a simple move that moved the course into a random space took 0.9 value and thus, the loop was completed. If the swap move can be expressed as $\beta$ (beta) referring to $\beta = 0, 1; 0.2; 0.5; 0.8$ and 0.9, then $1 - \beta$ gives simple move.

e) Tabu list length selection
The courses are taken to the tabu list by incorporating the specified tabu lengths. Experiments are conducted with tabu lists of 4, 6, 8, and 12 lengths. Programs are run through a number of iterations, and when the tabu length is completed, the courses are removed using the First-In-First-Out (FIFO) approach when the list reaches the tabu length, thus entering among the courses that are actively swapped in the program.

f) Tabu courses
There are some courses in the tabu list at the solution starts. The courses that would not be changed in the program are included in this tabu list and these courses kept in memory are not subject to change even if there is overlap because these courses are subject to strict constraints with regard to day and hour in accordance with the preferences of instructors. These courses are “General Economics”, “Numerical Analysis”, “Management and Organization”, and “Tech. Elect. Course IV”. Assignment of the overlapping courses is done on the condition that the location of these courses did not change.

The interface screens of the software created in Excel Visual Basic according to the defined algorithm of the program are shown in Figures 5 and 6. The “Program Assign” procedure is applied for the program run without using the Tabu Search Algorithm.

The “tabu list length” and “iteration number” counters under the “tabu search” screen are used for Tabu Search Algorithm. If the tests are to be performed by entering the move ratios, the “Movement with percentages” box is marked, and the “swap move” and “simple move” percentages under the “Movement Rates” tab are given in Figure 6 and the program is run.

6. Design of experiment for optimization of the conic objective function
Design of experiment is an approach used to determine the effects of the independent variable considered to be the cause of change in the dependent variable for statistical analysis [18]. In order to evaluate the interactions of each of the factors in the tests performed with the data obtained from the experimental results, the experimental design tools of Minitab 17.0 application belong to full factorial design. In the full factorial design, a combination is obtained by multiplying at least two or more factors by at least two or more levels of these factors [19].

In practice, a total 4-factor and 20-level experiment is designed involving four levels (4, 6, 8, 12) for Tabu Length, eight levels for Iteration Number (10, 20, 100, 200, 400, 600, 800, 1000), five levels for Experiment Formation Percentages (Beta) (0.1-0.9,
0.9-0.1; 0.2-0.8; 0.8-0.2; 0.5-0.5), and three levels for Start-up Solution Alternatives (normal program, good program, random program).

Two Experimental Application Procedures, namely ones without tabu and without tabu percentage, are determined except for the factors given in the experimental design. In the experimental procedure without tabu, in the case of each start-up solution, the program is run 15 times and the results are recorded. For each start-up solution alternative in the procedure without tabu percentage, the program is run through iterations without entering the test percentages.

The 4-factor and 20-level experimental design in the study is sequenced as $3^1 * 4^1 * 5^1 * 8^1$. The objective function is the dependent variable while tabu length, iteration number, program creation percentages, and start-up solution alternatives are considered as independent variables. The tests are carried out for approximately two months and together with the tests in “without tabu” and “without tabu percentage” procedures, a total of 621 tests are performed. The increase in the tabu list length and the iteration number causes the experiments to complete in a long time. However, it is determined that when the iteration number increases, the experimental time increases but good results are obtained.

The results obtained from the experimental study are statistically evaluated by ANOVA.

7. Analysis of Variance (ANOVA)

ANOVA is employed to compare the average of more than two populations with a normal distribution. The effects of independent variables on the dependent variable are investigated by ANOVA. The number of dependent and independent variables determines the type of ANOVA. “One-Way ANOVA” is used in ANOVA with one dependent variable number. In the One-Way ANOVA, it is assumed that each group comes from the normal distribution and that the variances of the groups are relatively homogeneous [19].

A hypothesis test is put forth for statistical analysis. Because hypothesis tests include comparison and select process, more than one hypothesis is required. These hypotheses are called alternative hypotheses. In this case, where $n$ is the number of population and $\mu_i$ is the average of test method $i$ and $(i = 1, 2, 3, ..., n)$, the initial hypothesis $H_0$ and alternative hypothesis $H_1$ are defined as follows:

$$H_0 : \mu_i = \mu_j \quad \forall i, j \neq i, i, j = 1, 2, 3, ..., n,$$

$$H_1 : \mu_i \neq \mu_j \quad \exists i, j, i \neq j.$$

A specific value is predicted for error margin in tests and expressed by alpha ($\alpha$), usually being as small

| One-way ANOVA: Objective Function versus Iteration Number |
|----------------------------------------------------------|
| Source | DF | SS | MS | F  | P |
| Iteration Number | 8  | 1770605 | 221326 | 1558 | 0.000 |
| Error     | 612 | 8692706 | 14204 |     |    |
| Total     | 620 | 10463311 |      |     |    |

$S = 119.2$

R-Sq = 16.92%

R-Sq(adj) = 15.84%

Figure 7. ANOVA of objective function and iteration number.

| One-way ANOVA: Objective Function versus Start-up Solution |
|-----------------------------------------------------------|
| Source | DF | SS | MS | F  | P |
| Start-up Solution | 3  | 4029770 | 2014885 | 193.55 | 0.000 |
| Error     | 618 | 6433541 | 10410 |     |    |
| Total     | 620 | 10463311 |      |     |    |

$S = 102.0$

R-Sq = 38.51%

R-Sq(adj) = 38.31%

Figure 8. ANOVA of objective function and start-up solution.

as 0.05 or 0.01. Statistical programs calculate the error margin that occurs as a result of a hypothesis test and this value is called “F” value. It is decided to accept or reject the hypothesis by comparing the F value with the predetermined $\alpha$ value. If $P \leq \alpha$, $H_0$ is rejected. If $P > \alpha$, $H_0$ is accepted [20]. The significance level is taken as $\alpha = 0.05$ in the study.

The results obtained from the experiments are subjected to ANOVA and pairwise comparison analysis of the variables is performed. The change values of four factors and objective function are evaluated. As a result of this analysis, it is observed that two factors have significant differences. These factors are determined to be the “Iteration Number” and “Start-up Solution Alternatives” and are shown with the explanations of the ANOVA results (Figures 7 and 8).

The test results of “Objective function versus Iteration Number” showed a significant difference in $\alpha = 0.05$ significance level, $n = 8$ level, and $P = 0.000 < 0.05$. Hypothesis $H_0$ is rejected. The test results of “Objective function versus Start-up Solution” showed a significant difference in $\alpha = 0.05$ significance level, $n = 3$ level and $P = 0.000 < 0.05$. $H_0$ hypothesis is rejected.

8. Results

In the initiation of the solution process, the AHP method is applied to determine the preferences of
instructors. The whole instructors fill the preferences form. The results are analyzed using the AHP and then, experiments are conducted to apply the parameters of Tabu Search Algorithm. The AHP and experiment processes take a long time to complete, almost two months. The Tabu Search Algorithm was designed by trying very different combinations (number of iterations, tabu length, etc.), thanks to the flexible structure of the software. In this way, the most suitable solution is determined. The statistical analysis is then initiated.

According to the results of the analysis, it is possible to determine between which factors the difference arose in the tests with a significant difference. “Iteration Number” and “Start-up Solution Alternatives” are determined as critical factors. In determining the variable effects on the objective function, the graphs in Figure 9 are used. The value of the iteration number 800 is seen to be minimum in the Iteration Number graph. The “Good Program” variable has the minimum value in the Start-up Solution Alternatives in the graph.

As a result of the experiments, the optimum objective function value is obtained as (-1.504) by incorporating 6 tabu lengths and iteration number 400 as well as selecting the initial solution as a good program without assigning the test percentages. As a result of this experiment, the course overlap is 20 and the instructor preference is 35.420. In the convergence graph obtained, it is seen that the conic objective function at the best objective value obtained from the experiment results converged to the iteration number 400 (Figure 10). In the 2016-2017 Spring Semester Course Schedule prepared manually, course overlap is 696, total instructor preference is 35.571, and

![Figure 9](image-url)

**Figure 9.** Effects of variables on the conic objective function.

![Figure 10](image-url)

**Figure 10.** The best conic objective function value according to the iteration number.

the scalarized objective function value is calculated as 589.334. In this case, a significant improvement is seen in the curriculum when the current course schedule is compared with the results of the best solution obtained by the proposed approach in the experiments, as shown in Table 5.

To determine the trade-off between the objective functions, the Tabu Search Algorithm is run 160 times for each of the initial solution as the normal program, the good program, and the random program, in order. The best results are obtained by selecting the initial solution as the good program. The points obtained using the good program are shown in Figure 11. The Pareto points are marked in dashed circles. There are four Pareto points among the obtained 160 points. It is shown that there is a trade-off between the points (55; 36,782) and (99; 38,638). While the student course overlap values change in the interval [55; 99], the values of the instructor preferences change in the
9. Conclusion and future directions

The course scheduling problem is a difficult optimization problem frequently encountered in the literature and various methods have been used to resolve it so far. In this study, the problem was analyzed step by step and a metaheuristic integrated approach was employed for the solution. A bi-objective mathematical model was developed for the problem; two objective functions were designed to maximize the preferences of the instructor and to minimize student-course overlaps. Instructor preferences were taken by a designed questionnaire and evaluated by the Analytic Hierarchy Process (AHP) method. Student-course overlaps were obtained from the Student Information System (SIS). The objective functions were transformed into a single objective function using the Conic Scalarization Method.

In order to solve the problem, the Tabu Search Algorithm, one of the heuristic approaches, was used in Excel Visual Basic programming language. Experiments were conducted using different tabu lengths, iteration numbers, and initial solutions. The results were analyzed. According to the test results, a significant improvement according to the conic objective function value and student-course overlaps was achieved by using the proposed approach. In addition, the outcome of the experiments demonstrated that the iteration number and start-up solutions had a significant difference in the minimization of the conic objective function.

For further study, one should compare the proposed approach with different metaheuristic methods and scalarization approaches. A decision support system can be designed to obtain data from instructors by web application and the SIS database.

References

1. Lewis, R. and Thompson, J. “Analysing the effects of solution space connectivity with an effective metaheuristic for the course timetabling problem”, European Journal of Operational Research, 240, pp. 637–648 (2015).
2. Song, T., Liu, S., Tang, X., et al. “An iterated local search algorithm for the university course timetabling problem”, Applied Soft Computing, 68, pp. 597–608 (2018).
3. Yasari, P., Ranjbar, M., Jamili, N., et al. “A two-stage stochastic programming approach for a multi-objective course timetabling problem with courses cancelation risk”, Computers & Industrial Engineering, 130, pp. 630–660 (2019).
4. Imran Hossain, S., Akhlaq, M.A.H., Shuvo, M.I.R., et al. “Optimization of university course scheduling problem using particle swarm optimization with selective search”, Expert Systems With Applications, 127, pp. 9–24 (2019).
5. Ozturk, Z.K., Kasimbevli, N., Ozdemir, M.S., et al. “University timetabling problem considering user preferences”, Journal of Industrial Engineering, 27(1), pp. 2–16 (2015).
6. Badri, M.A. “A two-stage multiobjective scheduling model for faculty-course-time assignments”, European Journal of Operational Research, 94, pp. 16–28 (1996).
7. Ozdemir, M.S. and Gasimov, R.N. “The analytic hierarchy process and multiobjective 0-1 faculty course assignment”, European Journal of Operational Research, 157, pp. 398–408 (2004).
8. Alagag, C.H., Hocaoglu, G., and Basaran, M.A. “The effect of neighborhood structures on tabu search algorithm in solving course timetabling problem”, Expert Systems with Applications, 36(10), pp. 12349–12356 (2009).
9. Akdan, C. and Gulcu, A. “A bi-criteria hybrid genetic algorithm with robustness objective for the course timetabling problem”, *Computers & Operations Research*, **90**, pp. 22-32 (2018).

10. Jamili, A., Hamid, M., Gharoun, H., et al. “Developing a comprehensive and multi-objective mathematical model for university course timetabling problem: A real case study”, *The 2nd European Intl. Conf. on Industrial Engineering and Operations Management*, Paris, France (2018).

11. Al-Yakoob, S.M. and Sherali, H.D. “A column generation mathematical programming approach for a class-faculty assignment problem with preferences”, *Computational Management Science*, **12**, pp. 297-318 (2015).

12. AlHaddid, I., Kaalneh, K., Tarawneh, H., et al. “Investigation of simulated annealing components to solve the university course timetabling problem”, *Italian Journal of Pure and Applied Mathematics*, **44**, pp. 291-301 (2020).

13. Al-Jarrah, M.A., Al-Suwalqah, A.A., and Al-Hamdan, S.F. “Developing a course timetable system for academic departments using genetic algorithm”, *Jordanian Journal of Computers and Information Technology*, **3**(1), pp. 25-36 (2017).

14. Altunay, H. and Eren, T. “A literature review for course scheduling problem”, *Panukkale University Journal of Engineering Sciences*, **23**(1), pp. 55-70 (2017).

15. Saaty, T.L. “How to make a decision: The analytic hierarchy process”, *European Journal of Operational Research*, **48**(1), pp. 9-26 (1990).

16. Can, E. “Metaheuristic approach proposal for the solution of the multiobjective course timetabling problem”, Master’s Thesis, Institute of Science And Technology, Katalaya Dumlupinar University (2019).

17. Ustun, O. “Interactive reference point procedure based on the conic scalarizing function”, *The Scientific World Journal*, **2014**(5), 242903 (2014).

18. Montgomery, D.C., *Design and Analysis of Experiments*, 5th Edn., John Wiley Sons, New York, US (1991).

19. Kocacalislan, I. and Bingöl, N.A. Biyoistatistik [Biostatistics], 3th Edn., Nobel Academic Publications, Ankara, Turkey (2017).

20. Scheffe, H., *The Analysis of Variance*, 1st Edn., Wiley-Interscience, New York, US (1990).

**Biographies**

**Esra Can** is currently a PhD Candidate at the Department of Industrial Engineering, Bursa Uludag University, Bursa, Turkey. She received MSc degree from the Industrial Engineering Department at Katalya Dumlupinar University, Katalya, Turkey. She received BSc degree from the Industrial Engineering from Yasar University, Izmir, Turkey. She received BSc degree from the Business Administration from Anadolu University, Eskisehir, Turkey. Her research interests include optimization, design of experiment, operations research, and multi-criteria decision. She has academic studies published as articles and book chapters.

**Ozden Ustun** is a Professor and the Head of the Industrial Engineering Department, Katalya Dumlupinar University, Turkey. He received his PhD, MSc, BSc degrees in Industrial Engineering at Esiksehir Osmangazi University, Eskisehir, Turkey. His research interests include operation research, multi-objectives modelling, and production systems. He has published in journals such as the International Transactions In Operational Research, International Journal of Fuzzy System Applications, Journal of Industrial and Management Optimization, Omega - The International Journal of Management Science, Computers & Industrial Engineering, Optimization, INFOR, Applied Mathematical Modeling, among others.

**Salhin Saglam** is a Lecturer at the Informatics Department, Katalya Dumlupinar University, Turkey. He is currently a PhD Candidate at the Department of Mathematics, Katalya Dumlupinar University, Katalya, Turkey. He received one MSc degrees from the Mathematics, Anadolu University and another from Computer Engineering, Anadolu University. He received BSc degree from Mathematics, Anadolu University. His areas of research include database modeling, computer programming, database management, software engineering, object-oriented programming, operating systems, and mathematics.