Engineering Applications of Artificial Intelligence 159 (2025) 111697

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Research paper

Data driven-based thesis defense scheduling system: A theoretical framework and empirical development

Yanjie Zhou [a], Zhicheng Zhang [a], Ningning Song [b], Jincan Zhang [a,*]

[a] School of Management, Zhengzhou University, Zhengzhou, China
[b] Ocean College, Zhejiang University, Zhoushan, China

ARTICLE INFO

Keywords:
Data-driven
Genetic algorithm
Thesis defense

ABSTRACT

The thesis defense, essential for students to obtain their degree during the undergraduate program, is a formal event for presenting students' theses to committees. This paper studies the thesis defense scheduling problem, considering equality and efficiency after the committees are determined. A data-driven method calculates the matching similarity degree between students and committees. A nonlinear mathematical model is proposed to formulate the studied problem. A linearized model is introduced to make the model solvable by mixed-integer solvers. A mathematical formulation-based genetic algorithm is proposed to solve large-scale problems. This paper developed a decision-support system for solving the studied problem by integrating the proposed solution method. Various experiments were conducted, and the experimental results showed that the proposed method could effectively solve the studied problem for large-scale instances. Finally, a web-based system is presented to show the capability of applying the proposed method.

1. Introduction

Thesis defense is an important event for students, including undergraduate and graduate students, to obtain their degrees. The defense committees who review the thesis are very important for determining whether the students could obtain their degree or not (Siering et al., 2018). To ensure the committee is making a comprehensive evaluation, the committee should be very familiar with the thesis. Many previous studies have focused on how to find good reviewers for reviewing a paper (Hartvigsen et al., 1999).

This paper focuses on solving undergraduate students' thesis defense scheduling problem when the committees are determined, in which the university administrative staff needs to determine the schedule of undergraduate students' thesis defenses considering the equality that is an important criterion in many scheduling problems (Gu et al., 2018; Zhou and Lee, 2020). Usually, university administrative staff need to prepare a thesis defense schedule, especially for undergraduate students. Zhengzhou University, which has over 55000 full-time undergraduates, is the largest university in China in terms of the number of students. Every year, more than 11,000 full-time undergraduates must attend thesis defenses held for several days simultaneously. If the committee is familiar with the student's thesis, the students will obtain a fair and comprehensive evaluation.

To obtain the familiarity between students and committees, this paper adopted a data-driven method to calculate the degree of familiarity between students and committees. Data-driven is a modern method that has been used for recommendation systems in many areas (Park and Lee, 2021; Mousavi et al., 2023; Bi et al., 2024). Using the degree of familiarity between students and committees, the university administrative staff needs to make a schedule for student defenses. Undergraduate students' thesis defense scheduling in this paper is a challenging combinatorial optimization problem. This paper aims to propose a solution method to solve the undergraduate student's thesis defense scheduling problem and develop a decision-support system.

The main contributions of this paper are summarized as follows. (1) This paper studies the thesis defense scheduling problem considering equality with three different objective functions. (2) A nonlinear mathematical model is developed to formulate the studied problem. To make the nonlinear mathematical model solvable, the nonlinear mathematical model is linearized by using linearization methods. (3) The mathematical model could only solve small-size problems. A mathematical formulation-based genetic algorithm is introduced. (4) Various experiments were conducted to verify the proposed mathematical model and mathematical formulation-based genetic algorithm. The experimental results showed that the proposed solution method is effective. (5) This paper proposes a data-driven decision-support framework based on the proposed method. A web-based decision-support system tool is developed.

* Corresponding author.
E-mail addresses: ieyjzhou@zzu.edu.cn (Y. Zhou), clarkzzzzz@gs.zzu.edu.cn (Z. Zhang), songningn@zju.edu.cn (N. Song), jincanz@zzu.edu.cn (J. Zhang).

https://doi.org/10.1016/j.engappai.2025.111697
Received 4 December 2024; Received in revised form 23 May 2025; Accepted 3 July 2025
Available online 19 July 2025

Table 1
Comparative table of literature methods.

Paper	Research focus	Methodology	Contribution	Shortcomings
Ferilli et al. (2006)	Automatically identify paper topics to assist reviewer assignment	Latent Semantic Indexing	Automated topic identification, Reduced manual effort	Limited accuracy in topic identification
Mimno and McCallum (2007)	Expertise modeling for matching papers with reviewers	Author-Persona-Topic (APT) model	Improved reviewer-paper matching by modeling multiple personas per author	Lack of practical constraints such as reviewer load and conflicts of interest
Charlin et al. (2011)	Optimize the matching of papers and reviewers	Integer Programming, Learning Models	Framework for optimizing reviewer assignments with learning and matching	Restricted generalization ability of the model
Li and Watanabe (2013)	Automatic paper-to-reviewer assignment	Preference-based approach, Topic-based approach	Combined preference-based and topic-based approaches for efficient assignment	Small-scale matrix algorithms may not be able to achieve optimal solutions
Charlin and Zemel (2013)	Automated reviewer assignment system	Machine Learning Models	Developed a paper matching system that automatically matches papers with reviewers	Rely on the specific workflow of the meeting
Liu et al. (2014)	Optimize the matching of papers and reviewers	Random Walk with Restart (RWR), Graph-based model	Balanced expertise, authority, and diversity in reviewer assignment	Lack of practical constraints such as reviewer load and conflicts of interest
Kou et al. (2015a)	Weighted coverage-based reviewer assignment	Branch-and-Bound Algorithm (BBA)	Improved assignment quality with weighted topic coverage and efficient algorithms	High dependency on topic modeling
Kou et al. (2015b)	Topic-based reviewer assignment system	Topic Extraction, Assignment Models	Automatically extract the characteristics of reviewers and papers and optimize the allocation through multiple models	High dependency on topic modeling
Anjum et al. (2019)	Automated paper-reviewer matching with vocabulary mismatch and partial topic overlap handling	Common topic space modeling with abstract topic vectors	A new matching method is proposed to deal with vocabulary mismatch and topic overlap problems effectively	Lack of topic interpretability
Kalmukov (2020)	Efficient automatic assignment of reviewers to papers	Heuristic algorithm	The propose algorithm had better time complexity than the maximum-weighted matching algorithm	Highly dependent on the quality of the similarity matrix
Tan et al. (2021)	Improved reviewer assignment considering incomplete reviewer data and non-manuscript-related papers interference	Word and semantic-based iterative model (WSIM)	Reduces overfitting to incomplete data and interference from non-related papers	Lack of actual system verification

The remainder of this paper is organized as follows. Section 2 introduces the literature review. Section 3 shows the information retrieval. Section 4 presents the mathematical model of the studied problem. The proposed solution approach is shown in Section 5. The experimental results are shown in Section 6. The developed application is presented in Section 7. The conclusions are given in Section 8. Finally, the Appendix is given.

2. Literature review

This section summarized the studies related to this paper.

The study of Ferilli et al. (2006) showed that the assignment work of conference papers was very complicated and time-consuming and suggested using the latent semantic indexing technique to automatically extract topics from the titles and abstracts of conference papers to use expert systems to match conference papers with reviewers with the corresponding expertise. Mimno and McCallum (2007) used the Author-Persona-Topic model to model the expertise of authors' paper documents to improve the matching accuracy. Charlin et al. (2011) proposed a framework to optimize the allocation of paper reviewers by using an integer programming approach with a suitability score to measure the pairing relationship between papers and reviewers. Li and Watanabe (2013) proposed an approach based on the match between reviewers and papers, combining preference-based and topic-based approaches to model reviewers. Charlin and Zemel (2013) developed a paper matching system that automatically matches papers with reviewers. The system has been used in machine learning and computer vision conferences. Liu et al. (2014) implemented an automatic paper reviewer recommendation system. A graph of potential reviewers and papers to be reviewed was proposed, which contains expertise and authority information. The random walk with restart model with sparsity constraints was applied to the proposed graph.

Kou et al. (2015a) proposed a new framework based on weighted coverage, which extracted knowledge from the papers published by reviewers and converted it into a set of topics using the relevance of the papers to the topics as weights. They introduced a comprehensive evaluation index based on the professional knowledge coverage of the paper topics within the reviewer population. Kou et al. (2015b) designed a reviewer assignment system that automatically extracts information about reviewers and submissions in the form of topic vectors. For each paper, the reviewer's topic vector was used to maximize the coverage of its topic, and then the corresponding reviewer was matched. Anjum et al. (2019) proposed a common topic model to jointly model common themes of submissions and reviewer profiles while relying on abstract topic vectors to handle the lexical mismatch between paper submissions and reviewer expertise. Kalmukov (2020) proposed a heuristic algorithm that provided roughly the same number of papers to reviewers. Experiments confirmed that the propose algorithm had better time complexity than the maximum-weighted matching algorithm. Tan et al. (2021) proposed a word and semantic-based iterative

Table 2
Comparative table of literature methods.

Paper	Research focus	Methodology	Contribution	Shortcomings
Battistutta et al. (2019)	Thesis defense timetabling in Italian universities	Integer Programming, Constraint Programming, Local Search	Instance generator; Method comparison	Lack of consideration of the fairness of matching committees with students' research directions
Su et al. (2020)	Defense grouping with conflict avoidance and gender balance constraints	Greedy-backtracking hybrid	Practical constraint handling	Lack of consideration of the fairness of matching committees with students' research directions
Dimitsas et al. (2022)	Three-stage mixed integer programming method	Three-stage hybrid optimization	New best solutions on benchmark dataset	Lack of consideration of the fairness of matching committees with students' research directions
Almeida et al. (2024)	Multi-objective MILP for defense scheduling with fairness	Two-stage exact solution	Multi-objective tradeoffs	Lack the deployment of actual system integration
Dimitsas and Gogos (2024)	Exact model for solving the problem using constraint programming	Constraint programming, Symmetry	Symmetry constraints reduce the search space	Lack of consideration of the fairness of matching committees with students' research directions
This article	Thesis defense scheduling problem	Integer Programming, Genetic algorithm	A complete solution from data acquisition to thesis defense scheduling output	

model for the reviewer assignment problem. This model solved two major constraints: incomplete reviewer data and interference from non-manuscript-related papers through an improved similarity calculation method. Experiments showed its advantages in improving the accuracy of reviewer recommendations (see Table 1).

The above previous studies focused on solving the reviewer and paper matching problem. However, the application of this matching information to thesis defense scheduling, while considering various influencing factors, remains underexplored. The effective scheduling of student thesis defenses is crucial for academic planning and resource allocation.

Several studies have examined thesis defense group scheduling. Battistutta et al. (2019) proposed a mixed-integer programming model with multidimensional constraints for the Italian university scenario. Their innovative instance generation method and multi-algorithm comparison framework serve as a methodological reference for similar studies. In addressing the grouping problem with complex constraints, Su et al. (2020) developed a greedy-backtracking hybrid algorithm to solve the defense grouping problem, incorporating multiple constraints such as advisor avoidance and gender balance. Their constraint-handling mechanism serves as a valuable reference for similar scheduling problems. Dimitsas et al. (2022) proposed an innovative three-stage mixed integer programming method, which effectively solved the scheduling problem of large-scale instances through constraint relaxation and variable aggregation technology. Almeida et al. (2024) established a multi-objective mixed integer programming model for the "single defense assignment" type (each committee is assigned to one defense), focusing on solving the coordinated optimization of committee allocation and time scheduling. Dimitsas and Gogos (2024) proposed an exact model for solving the problem using constraint programming, which enables an efficient search for near-optimal solutions on standard datasets (see Table 2).

Existing articles rarely consider the fairness of matching committees with students' research directions and lack the deployment of actual system integration. This paper establishes a fairness quantification indicator through data-driven matching similarity calculation, which greatly improves the matching quality. It extends the defense scheduling problem from theoretical research to actual system development, and realizes a complete solution from data acquisition to generating matching results.

3. Information retrieval

The graduation pipeline for undergraduate students is summarized as follows. (1) A student submits a defense application. (2) The university invites a thesis committee for the defense. (3) The university makes a schedule for student committee matching. The reviewers of academic papers for reviewing a paper are from different affiliations. The committees of the undergraduate student defense are usually from the same affiliations as the undergraduate students. To assign the committees to the undergraduate students, we need to know the degree of familiarity between each committee and undergraduate student. Fig. 1 shows the data collection pipeline for calculating the degree of familiarity between each committee and undergraduate student.

3.1. Data sources

The information of academic papers represents the committee's research direction and research fields. Academic papers are included in various paper databases, such as China National Knowledge Infrastructure (CNIK), Web of Science, and Scopus, which contain many academic papers. These included papers covering relatively cutting-edge academic achievements in various fields. In this paper, CNKI, Scopus, and Web of Science are selected as the data sources for obtaining academic papers, and the published academic papers are indexed through information such as the name and affiliation of committees.

3.2. Automatic data collection

3.2.1. Committee research information crawling

When the committee is determined, we need to collect the committee's publications. To ensure that all the academic papers are collected, we need to adopt all the affiliations of a committee and his/her name that has been used. Then, we can obtain all of the committee's publications.

3.3. Keyword extraction

Extracting keywords from a committee's publications is crucial for accurately calculating the similarity between students and the committee's research directions. This paper selects abstracts from academic papers as the information source for keyword extraction for committees and students, distinguishes academic papers in Chinese and English, constructs a unified language structure, and then conducts a subsequent keyword extraction. There are many keyword extraction methods. This paper adopts the TF–IDF and KeyBERT methods, which are summarized below. Fig. 2 shows the pipeline of Keyword extraction with TF–IDF and KeyBERT used in this paper.

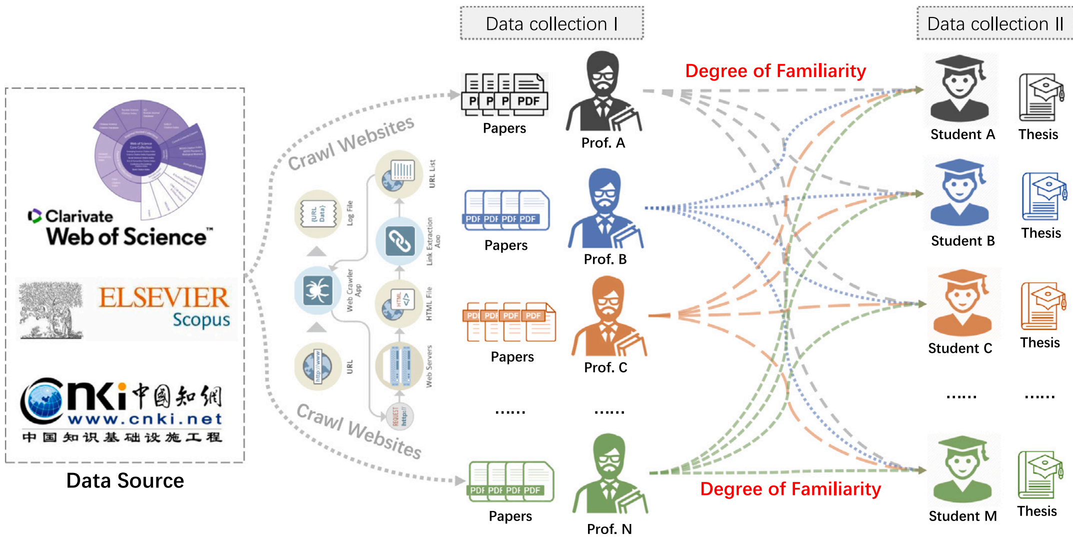
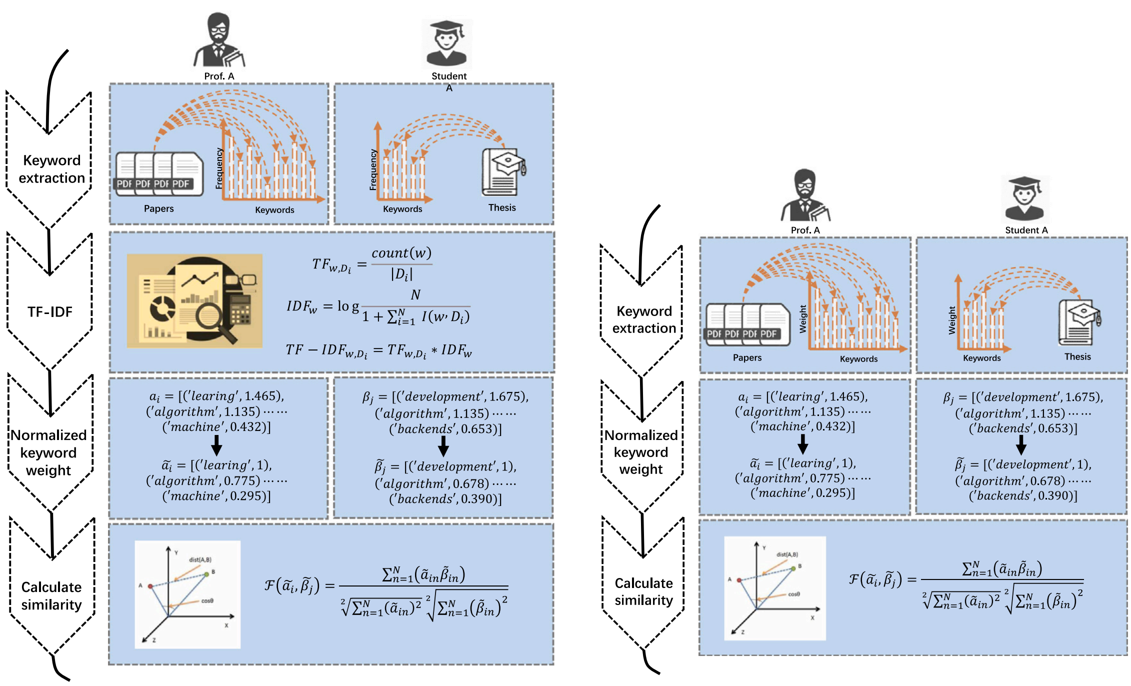


Fig. 1. Pipeline of data collection.



(a) TF-IDF (b) KeyBERT

Fig. 2. Keyword extraction with TF–IDF and KeyBERT.

3.3.1. TF-IDF

TF–IDF is a statistical measure that evaluates the frequency of keywords. Let l denote the committee l and s denote the student s. κ_l is the keyword vector for the committee l, which is defined as equation $\kappa_l = \langle \kappa_{l1}, \kappa_{l2}, \ldots, \kappa_{ln} \rangle$. κ_{ln} represents the nth keyword frequency for committee l and is a positive integer. $\varkappa_s$ is the keyword vector for student s, which is defined as equation $\varkappa_s = \langle \varkappa_{s1}, \varkappa_{s2}, \ldots, \varkappa_{sm} \rangle$. $\varkappa_{sm}$ represents the mth keyword frequency for student s and is a positive integer. A committee usually publishes more papers than an undergraduate student. In this paper, $n \gg m$ is held. After normalization, the κ_l and $\varkappa_s$ are $\bar{\kappa}_l \langle \bar{\kappa}_{l1}, \bar{\kappa}_{l2}, \ldots, \bar{\kappa}_{ln} \rangle$ and $\bar{\varkappa}_s = \langle \bar{\varkappa}_{s1}, \bar{\varkappa}_{s2}, \ldots, \bar{\varkappa}_{sm} \rangle$, respectively. Finally, we can calculate the inverse document frequency (IDF) by using the IDF equation.

3.3.2. KeyBERT

KeyBERT, unlike the TF–IDF method, is a minimal and easy-to-use keyword extraction technique.[1] KeyBERT extracts document embeddings using BERT. Next, word embeddings are extracted for N-gram words/phrases.

3.4. Degree of familiarity

After extracting keywords, we obtain the keyword vectors for both committees and students, denoted as $\bar{\kappa}_i$ and $\bar{\varkappa}_j$, respectively. The Cosine theorem formula of the vector space model is adopted to evaluate the similarity between keyword vectors of committees and students. Cosine

[1] https://github.com/MaartenGr/KeyBERT.

Table 3
Notations and decision variables.

Set	
L	Set of defense committee
S	Set of students
T	Set of round for defense
P	Set of session
Index	
l	Index of committee
s	Index of student
t	Index of round for defense
p	Index of session
Parameters	
c_{ls}	The degree of familiarity of the evaluation expert l with the research direction of the defense paper of the defense student s
a_{ls}	Bool, =1 means that there is a teacher–student relationship between the evaluation expert l and the respondent student s, otherwise it is 0
α_U	The maximum number of committees in a defense group
α_L	The minimum number of committees in a defense group
k_{tp}	Bool, =1 means that the defense can be arranged in the session p of the t round, otherwise it is 0
Decision variables	
x_{ltp}	Bool, =1 means that the evaluation expert l is arranged in the defense group of the session p of the t round, otherwise, it is 0
y_{stp}	Bool, =1 means that the defense student s is arranged in the defense group of the session p of the t round, otherwise it is 0

similarity is a mathematical metric for measuring two vectors in high-dimensional spaces. The Cosine similarity of $\tilde{\kappa}_i$ and $\tilde{\varkappa}_j$ is defined in Eq. (1).

$$\mathcal{F}\left(\tilde{\kappa}_i, \tilde{\varkappa}_j\right) = \frac{\sum_{n=1}^{|\rho|}\left(\tilde{\kappa}_{in}\tilde{\varkappa}_{in}\right)}{\sqrt{\sum_{n=1}^{|\rho|}\left(\tilde{\kappa}_{in}\right)^2}\sqrt{\sum_{n=1}^{|\rho|}\left(\tilde{\varkappa}_{in}\right)^2}} \tag{1}$$

The larger value of $\mathcal{F}\left(\tilde{\kappa}_i, \tilde{\varkappa}_j\right)$ means that committee i is more familiar with the thesis of student j. In this paper, we prefer larger value of $\mathcal{F}\left(\tilde{\kappa}_i, \tilde{\varkappa}_j\right)$.

4. Mathematical model

This section introduces the statement of the studied problem and the proposed mathematical models. Before introducing the mathematical model, the notations and decision variables used in this paper are summarized in Table 3.

4.1. Problem statement

A set of undergraduate students, S, have already prepared their theses for defense. A set of committee L has already been invited and confirmed to attend the defense for the set of undergraduate students. The university is planning the defense schedule for undergraduate students. Due to the large number of undergraduate students, the defense process is divided into different sessions. Let P denote the set of sessions. Each session has multiple rounds, and T is the set of rounds for defense. An undergraduate student could only be assigned to one session and one round. A committee could be assigned to multiple sessions.

As mentioned in the previous section, each committee has a different research area, and a committee may not be familiar with all the undergraduate students' theses. The main goal is to maximize the familiarity degree c_{ls} for each round of all the sessions, where $c_{ls} = \mathcal{F}\left(\tilde{\kappa}_i, \tilde{\varkappa}_j\right)$ represents the degree of familiarity of the committee l with the student s.

4.2. A nonlinear mathematical model

This section introduces the proposed nonlinear mathematical model.

4.2.1. Objective functions

One of the primary goals is to find the committees that are the most familiar with the student's thesis. Eq. (2) defines the primary goal to maximize the summation of familiarity between committees and students.

$$P_1 \qquad Max \quad \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{l=1}^{L}\sum_{s=1}^{S}\left(x_{ltp}y_{stp}c_{ls}k_{tp}\right) \tag{2}$$

4.2.2. Constraints

The constraints considered in this paper are summarized as follows.

1. The university usually sets the minimum and maximum number of committees during the defense. The upper and lower limits of the number of committees in each round should meet the requirements. Eqs. (3) and (4) define the upper and lower limits of the number of committees, respectively.

$$\sum_{l=1}^{L} x_{ltp} \le \alpha_U k_{tp} \qquad \forall t \in T, p \in P \tag{3}$$

$$\sum_{l=1}^{L} x_{ltp} \ge \alpha_L k_{tp} \qquad \forall t \in T, p \in P \tag{4}$$

2. To maximize the number of committees per defense round within the required range and ensure that the number of committees in the same session is as balanced as possible, the difference in the number of evaluation experts per round is restricted to at most 1.

If $\sum_{p=1}^{P} k_{tp}\alpha_L < |L|$, Eq. (5) should be hold.

$$\sum_{p=1}^{P}\sum_{l=1}^{L} x_{ltp} = \sum_{p=1}^{P} k_{tp}\alpha_L \qquad \forall t \in T \tag{5}$$

Otherwise, when $\sum_{p=1}^{P} k_{tp}\alpha_1 \ge |L|$ is met:

$$\sum_{p=1}^{P}\sum_{l=1}^{L} x_{ltp} = |L| \qquad \forall t \in T \tag{6}$$

$$\left|\sum_{l=1}^{L} x_{ltp_1} - \sum_{l=1}^{L} x_{ltp_2}\right| k_{tp_1}k_{tp_2} \le 1 \qquad \forall t \in T, p_1, p_2 \in P \tag{7}$$

3. To ensure the workload and quality of each defense group, it is necessary to balance the defense time of each group. Generally, the school specifies the time allocated for student presentations and committee questioning, making the total defense duration dependent on the number of students. It is necessary to ensure that the number of students in each defense round is as equal as possible (that is, the difference between the number of defense students in each round is not more than 1).

When $S/(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp})$ is an integer, Eq. (8) is hold.

$$\sum_{s=1}^{S} y_{stp} = \frac{S}{\left(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}\right)} \qquad \forall t \in T, p \in P \tag{8}$$

Otherwise, when $S/(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp})$ is not an integer, Eq. (9) is hold.

$$\sum_{s=1}^{S} y_{stp} \le \left\lceil \frac{S}{\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}} \right\rceil \qquad \forall t \in T, p \in P \tag{9}$$

$$\sum_{s=1}^{S} y_{stp} \ge \left\lfloor \frac{S}{\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}} \right\rfloor \qquad \forall t \in T, p \in P \tag{10}$$

4. Each committee can only participate in one session of defense in each round.

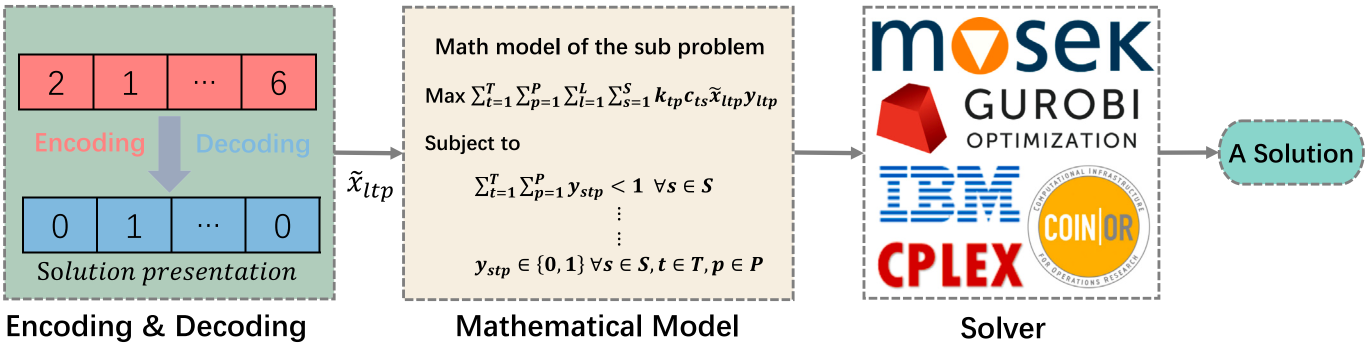


Fig. 3. The process of obtaining a solution with encoding and decoding.

$$\sum_{p=1}^{P} x_{ltp} \le 1 \qquad \forall l, t \tag{11}$$

5. Each student can only be assigned to one defense round.

$$\sum_{t=1}^{T}\sum_{p=1}^{P} y_{stp} = 1 \qquad \forall s \in S \tag{12}$$

6. In the defense, there are different restrictions on whether a committee can be assigned to a session that has a student under his/her supervision. If there is no teacher–student relationship between the committee and the student, the constraint can be expressed as (13).

$$x_{ltp} + y_{stp} + a_{ls} < 3 \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{13}$$

7. The decision variables are binary variables, which are defined in Eqs. (14) and (15).

$$x_{ltp} \in \{0,1\} \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{14}$$

$$y_{stp} \in \{0,1\} \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{15}$$

$$Model \quad P_1 \tag{16}$$

subject to constraints (3)–(15).

4.3. Linearization of the nonlinear model

P_1 is a nonlinear programming model transformed into a linear programming model, which cannot be directly solved by many solvers, such as Lingo. To solve the model P_1, we need to linearize the nonlinear $x_{ltp}y_{stp}c_{ls}k_{tp}$. Note that $c_{ls}k_{tp}$ is a constant coefficient and $x_{ltp}y_{stp}$ is nonlinear. This section introduces an auxiliary variable z_{lstp} and $z_{lstp} = x_{ltp} * y_{stp}$. After replacing the $x_{ltp}y_{stp}$ with z_{lstp}, the model P_1 be expressed as model $equ - obj1l$.

$$LP_1 \qquad Max \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{l=1}^{L}\sum_{s=1}^{S} (z_{lstp}c_{ls}k_{tp}) \tag{17}$$

subject to constraints (3)–(15) and

$$z_{lstp} \le x_{ltp}, \quad \forall l \in L, s \in S, t \in T, p \in P \tag{18}$$

$$z_{lstp} \le y_{stp}, \quad \forall l \in L, s \in S, t \in T, p \in P \tag{19}$$

$$z_{lstp} \ge x_{ltp} + y_{stp} - 1, \quad \forall l \in L, s \in S, t \in T, p \in P \tag{20}$$

$$z_{lstp} \in \{0,1\}, \quad \forall l \in L, s \in S, t \in T, p \in P \tag{21}$$

Theorem 1. *The model P_1 and the model LP_1 are equivalent.*

Proof. x_{ltp} and y_{stp} are two binary variables and $z_{lstp} = x_{ltp} * y_{stp}$ is also a binary. There are four cases for $x_{ltp} * y_{stp}$ including $x_{ltp} = y_{stp} = 0$, $x_{ltp} = y_{stp} = 1$, $x_{ltp} = 1, y_{stp} = 0$ and $x_{ltp} = 0, y_{stp} = 1$. When $x_{ltp} = y_{stp} = 0$, the constraints (18), (19) and (20) are $z_{lstp} \le 0$, $z_{lstp} \ge -1$. Thus, we can conclude that $z_{lstp} = 0$, which is the same as $z_{lstp} = x_{ltp} * y_{stp} = 0$. For all other three cases, the Eqs. (18), (19) and (20) are hold. Finally, we concluded that the model P_1 and the model LP_1 are equivalent. □

5. Mathematical formulation based genetic algorithm

The above-mentioned models shown in Section 4 are very time-consuming when the models have solved integer solvers for handling large-scale problems, such as CPLEX, Gurobi, Mosek, and LINGO, which are popular commercial software used for solving mixed integer programming models. Genetic algorithm (GA) is a very popular and efficient meta-heuristic that has been used for solving many combinatorial optimization problems (Gao et al., 2017; Ponce et al., 2025; Ghasemi et al., 2025). To solve the model more efficiently, this paper introduces a mathematical formulation-based genetic algorithm. The details of the genetic algorithm are summarized as follows.

5.1. Chromosome encoding and decoding

In this paper, the formulation-based genetic algorithm is divided into two phases. In the first phase, we use chromosomes to represent the decision variable x_{ltp}. After encoding and decoding, let $\tilde{x}_{ltp}$ denote the specified decision variable x_{ltp}. After that, we developed a mathematical model for obtaining the optimal value of $\tilde{y}^*_{stp}$ by the given $\tilde{x}_{ltp}$. Note that the mathematical model could be solved by any integer programming solvers, such as IBM CPLEX, CBC, Gurobi, and Mosek. Finally, we can obtain a feasible solution $(\tilde{x}_{ltp}, \tilde{y}^*_{stp})$. Fig. 3 shows the process of obtaining a solution with encoding and decoding.

5.1.1. Encoding

The encoding process,

The chromosome consists of different segments, and each segment denotes a session, which is an integer array. Each session is consisted of different parts, and each part denotes a round. An integer in each part denotes a committee ID, which means that the committee is assigned to a specified round and session.

Fig. 4 shows an example of chromosome decoding. In this example $[1,3,6]$ denotes the nth round defense in section A. $[2,4,5]$ denotes the $n+1$th round defense in section A. $[3,4]$ denotes the mth round defense in section B and $[2,5]$ denotes the $m+1$th round defense in section B.

5.1.2. Decoding

The encoding process determines the assigned session and round for each committee, as shown in Fig. 4. The genes in chromosomes are

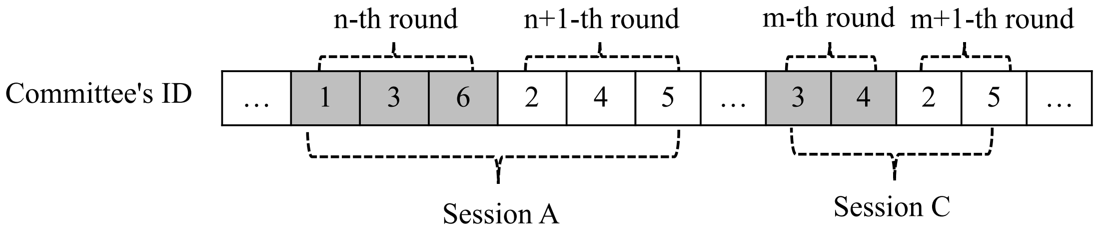


Fig. 4. An example of chromosome encoding.

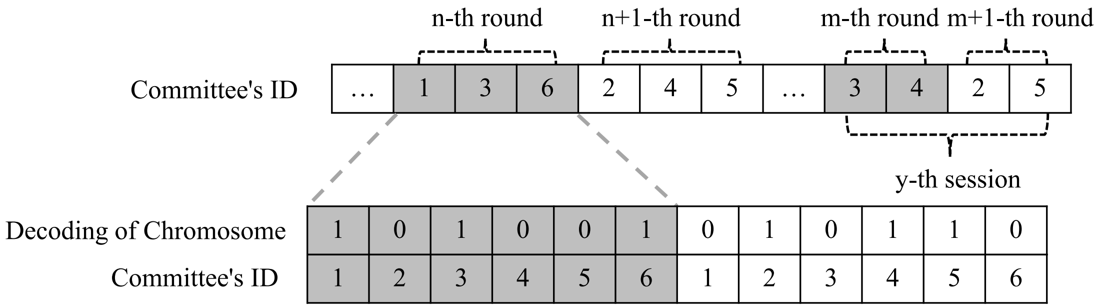


Fig. 5. An example of chromosome decoding.

integer values. As mentioned before, x_{ltp} is a binary value. We need to convert the chromosome into a binary value to determine x_{ltp}, which is called decoding processing in this paper.

The chromosome only represents the committees that are assigned in each round of a session. The total number of committees in each round will be different. x_{ltp} also needs to include the committees that are not assigned in each round of a session. Thus, for each round of a session, we set the total number of genes equaling the $|L|$. Each gene is a binary value for representing a committee that is assigned to this round or not. Fig. 5 shows an example of chromosome decoding. In this example, there $|L| = 6$.

5.2. Chromosome evaluation

As mentioned before, a chromosome only represents part of the solution. When a chromosome is given, we can only know the decision value of x_{ltp}. Let $\tilde{x}_{ltp}$ denote a chromosome after decoding. We need to know the decision value y_{stp} to calculate the objective function. Then, we can calculate the objective function by using the following formulation.

Thus, the objective function is can be expressed as:

$$\max \quad \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{l=1}^{L}\sum_{s=1}^{s}(\tilde{x}_{ltp} y_{stp} c_{ts} k_{tp}) \tag{22}$$

subject to

When $S/(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp})$ is an integer:

$$\sum_{s=1}^{S} y_{stp} = \frac{S}{\left(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}\right)} \qquad \forall t \in T, p \in P \tag{23}$$

Otherwise, when $S/(\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp})$ is not an integer:

$$\sum_{s=1}^{S} y_{stp} \le \left\lceil \frac{S}{\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}} \right\rceil \qquad \forall t \in T, p \in P \tag{24}$$

$$\sum_{s=1}^{S} y_{stp} \ge \left\lfloor \frac{S}{\sum_{t=1}^{T}\sum_{p=1}^{P} k_{tp}} \right\rfloor \qquad \forall t \in T, p \in P \tag{25}$$

(2) Restrictions on students' sole defense group.

$$\sum_{t=1}^{T}\sum_{p=1}^{P} y_{stp} = 1 \qquad \forall s \in S \tag{26}$$

(3) The decision variable of the defending student is a 0-1 variable.

$$y_{stp} \in \{0,1\} \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{27}$$

5.3. Population initialization

The chromosome group using chromosome form 1 represents a matching scheme of a review expert, and a chromosome group with a specified population size is randomly generated to form an initial population. According to the matching grouping model, ensure that each chromosome group in the population meets the model's constraints.

To ensure a balanced distribution of experts across different sessions and rounds, we first initialize the session and round assignments. The process considers constraints on the minimum and maximum number of committees in a defense group. The detailed procedure is described in Algorithm 1. After initializing the session and round assignments, we proceed with population initialization. The goal of this step is to construct a set of feasible assignments by selecting experts for each session. The initialization process is outlined in Algorithm 2, which incorporates Algorithm 1 as a subroutine.

5.4. Parent selection, crossover, and mutation operation

To generate offspring, parents are selected from the population pool using the tournament selection method. First, several individuals are randomly chosen from the population. Let k denote the total number of individuals selected from the population pool. Next, we find the best individual from the k individuals by comparing the fitness function value. Repeat this operation until a progeny population of the same population size is obtained.

In this paper, a two-point crossover operation is adopted to generate new offspring. Fig. 6 shows an example of a two-point crossover operation. One of the limitations of the two-point crossover operation is that it will generate infeasible solutions (offspring). Since the chromosomes of new individuals may violate constraint conditions after crossover, a repair operation is required.

All individual chromosome sets in the population are selected to mutate individuals according to mutation probability. In this paper, a two-point exchanging mutation is adopted. We randomly choose two genes and exchange those two genes. Fig. 7 shows an example of a

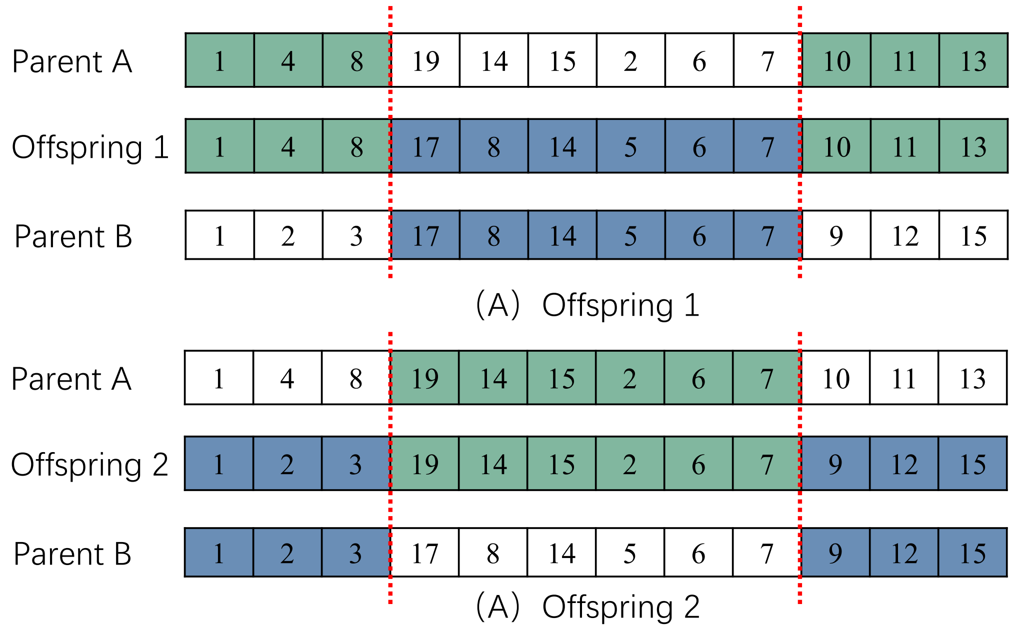


Fig. 6. An example of a two-point crossover.

Algorithm 1: Initialization of sessions and rounds.

```
Input: ExpertList;Session; α_L;α_U;
// ExpertList denotes list of all reviewer IDs.
   Session represents list of round numbers for each
   session. α_L and α_U are the minimum and maximum
   number of committees in a defense group,
   respectively.
Function InitSessionRound(ExpertList,Session,α_L,α_U):
    SessionRound = ∅;
    if (Max_{s∈Session}(s) × α_L > length(ExpertList)) then
        return ∅;
    end
    else
        if (Max_{s∈Session}(s) × α_U < length(ExpertList)) then
            for (each section) do
                // List(α_U) is function to create an empty
                   list with length α_U
                SessionRound = SessionRound ∪ List(α_U)
            end
        else
            for (each section) do
                SessionRound = SessionRound ∪ List(α_U)
            end
            for (each SessionList in SessionRound) do
                index = 0 ;
                while ( sum(SessionList) > length(ExpertList) )
                 do
                    SessionList[index] = SessionList[index] − 1;
                    index =(index+1) % length(SessionList) ;
                end
            end
        end
    end
    return SessionRound ;
```

Algorithm 2: Population initialization.

```
Function Init(ExpertList,PopulationSize,Session,α_L,α_U):
    Initialize empty Population ;
    SessionRound =
     InitSessionRound(ExpertList,Session,α_L,α_U) ;
    for (i = 1 to PopulationSize) do
        Initialize empty Chromosome ;
        for (each SessionList in SessionRound) do
            Initialize empty SessionChromosome ;
            for (each ExpertCount in SessionList) do
                // Function RandomSelectExpert():
                   Select the number of ExpertCount unique
                   review expert IDs from ExpertList
                SelectedExpert =
                 RandomSelectExpert(ExpertList,ExpertCount) ;
                SessionChromosome =
                 SessionChromosome ∪ SelectedExpert
            end
            Chromosome = Chromosome ∪ SessionChromosome;
        end
        Population = Population ∪ Chromosome;
    end
    return Population ;
```

mutation operation. In this example, genes 8 and 7 are selected (see before the mutation). Genes 7 and 8 exchange their corresponding positions (see after the mutation). When the two selected are from different sessions, an infeasible solution will be generated after exchanging their corresponding positions. Due to that, there will be two homogeneous genes in the same session.

5.5. Repair operation

After the crossover and mutation operations, the new offspring will be infeasible solutions. As shown in Fig. 6, offspring 1 has two 8 in its chromosome, which invalidates the constraint (11). A repair operation

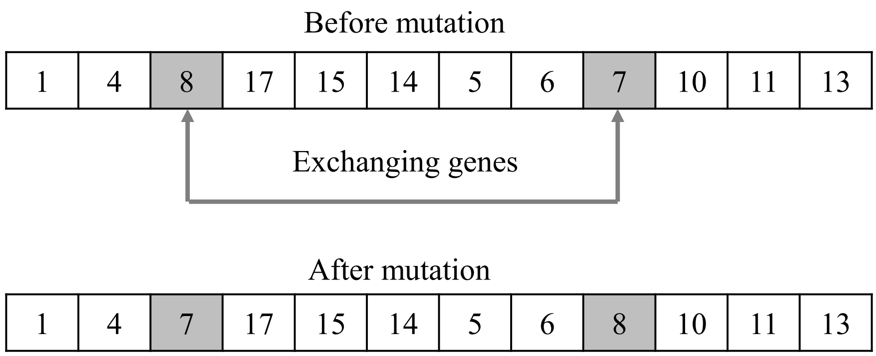
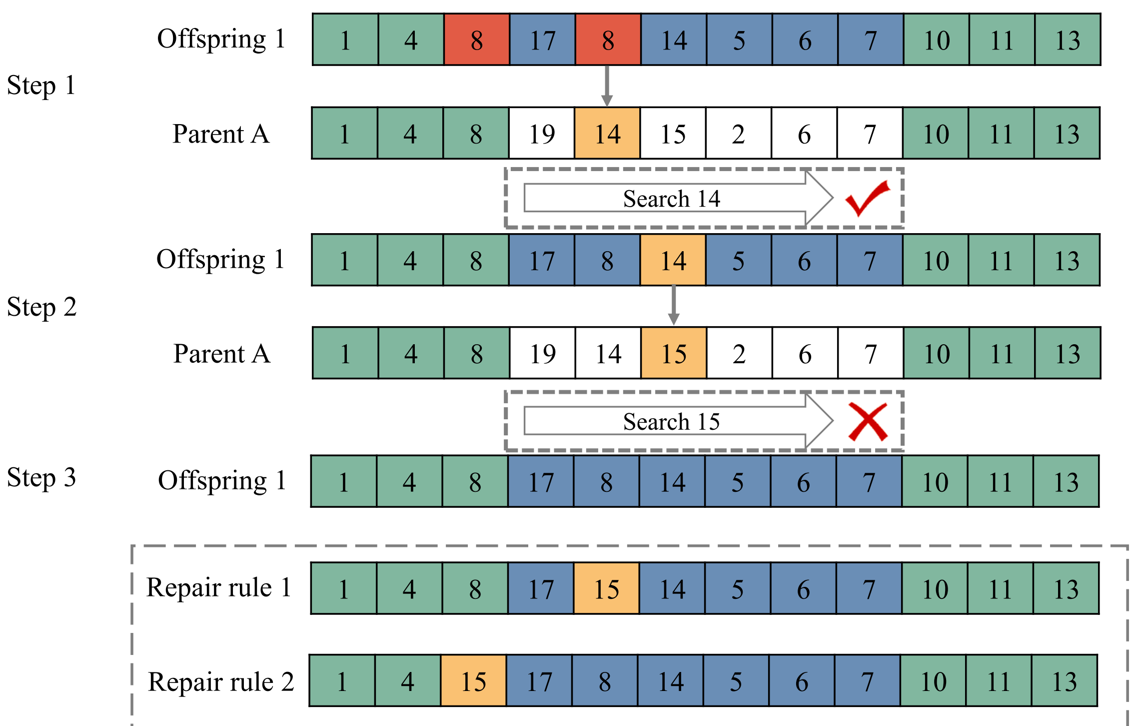
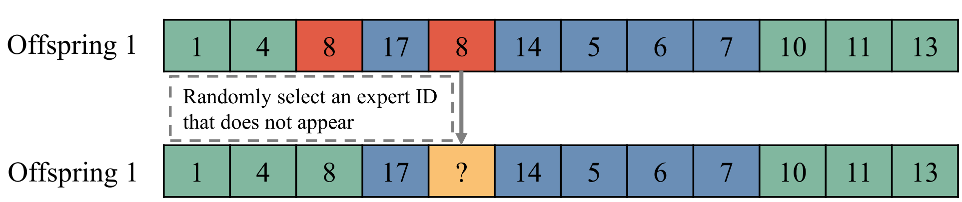


Fig. 7. An example of a mutation operation.



(a) Repair operation rules 1 and 2.



(b) Repair operation rule 3.

Fig. 8. Three different repair operation rules.

is designed to transform an infeasible solution into a feasible one. In this paper, three different repair operation rules are proposed.

Fig. 8 illustrates three different repair operation rules, all of which can be applied to infeasible solutions. The choice of a specific rule depends on how the repair operation is conducted.

Fig. 8(a) shows the repair operation rules 1 and 2. In Fig. 8(a), committee 8 appears twice. We need to find a new committee that is not Committee 8 to replace one of Committee 8, which corresponds to repair operation rules 1 and 2. The offspring 1 is from the crossover operation by Parents A and B (see Fig. 6). We search for the corresponding position of Parent A until there is a new committee that does not exist in offspring 1. Then we replace one of the committee 8. Another repair operation rule is totally randomly selected by a committee, which is called repair operation 3. Then, we replace one of the conflicted committees. Fig. 8(b) shows the repair operation rules 3.

6. Experimental results

This section introduces the experimental results to verify the proposed model and algorithm. The Benchmark data and experimental

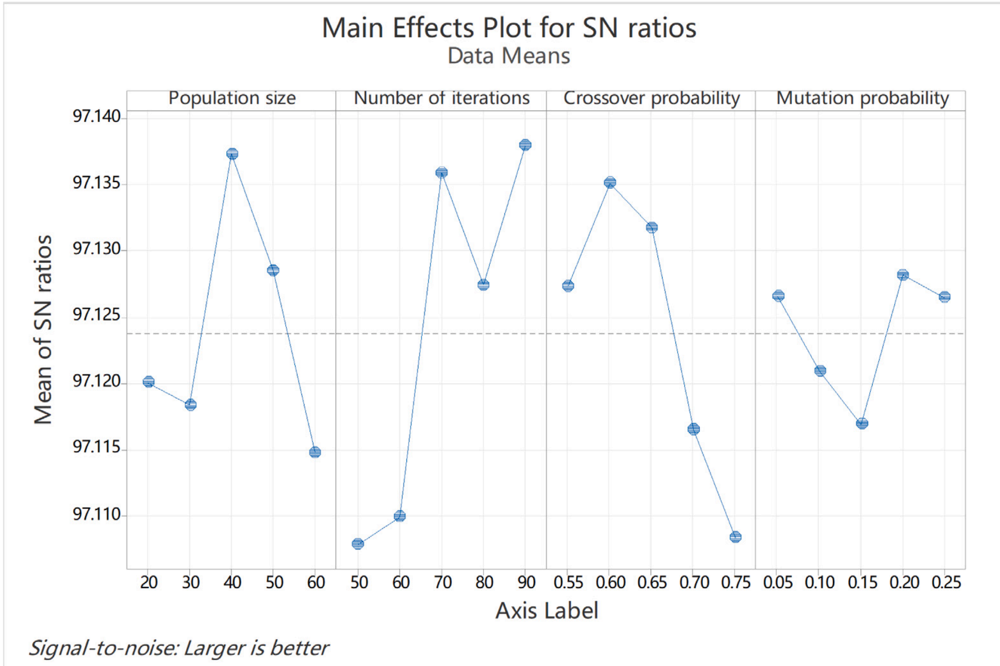


Fig. 9. Main effects plot for S/N ratios.

environment are presented in Section 6.1. Parameter tuning is shown in Section 6.2. The comparison between the genetic algorithm and the mathematical model is shown in Section 6.3. Section 6.4 analyzes the different objective functions. Finally, the algorithm analysis is given.

6.1. Benchmark data and environment

This section presents the benchmark data set. We adopt the undergraduate student's defense at the School of Management Engineering, Zhengzhou University, as the benchmark data set.[2] The School of Management Engineering has four departments, including Logistics Management, Industrial Engineering, Electronic Commerce, and Engineering Management. Around 200 undergraduate students graduate each year. In the following experiments, the maximum number of students is 200. According to the previous annual defense held by the School of Management Engineering at Zhengzhou University, the maximum number of review experts (Number of Review Experts, NRE) is 20. Table 4 shows the benchmark data set.

All the mathematical models proposed in this paper are solved by using IBM ILOG CPLEX Optimization Studio (version 22.1.1.0) with a Python interface to the CPLEX callable library. The Genetic algorithm is implemented in Python language (version 3.10). Both the models and algorithms are tested on an Ubuntu 22.04.4 LTS operating system with 13th Gen Intel Core i9-13900K and 16 GB RAM.

Table 4
Benchmark data set.

Experiment	NRE	Number of students	Sessions	Rounds	NRE per round
1	8	10	2	2	4
2	8	25	2	2	4
3	8	50	2	2	4
4	8	75	2	2	4
5	8	100	2	2	4
6	12	50	2	2	5
7	12	75	2	2	5
8	12	100	2	2	5
9	12	125	2	2	5
10	12	150	2	2	5
11	16	50	2	4	4
12	16	75	2	4	4
13	16	100	2	4	4
14	16	125	2	4	4
15	16	150	2	4	4
16	20	100	2	4	5
17	20	125	2	4	5
18	20	150	2	4	5
19	20	175	2	4	5
20	20	200	2	4	5

6.2. Parameter tuning

The genetic algorithm is not a deterministic algorithm, and its performance depends on parameter setting. The Taguchi method is the traditional method for parameter tuning that uses orthogonal tables to design experimental arrangements to obtain an optimized parameter. It was first proposed by Genichi Taguchi and has been in many algorithms to obtain better algorithm performance (Zhou and Lee, 2020). This paper adjusts four parameters, including population size, number of iterations, crossover probability, and mutation probability, by using the Taguchi method. Five levels are selected for each parameter, which is shown in Table 5.

According to the Taguchi method, the L_{25} orthogonal matrix is used to design the experimental arrangement, which is shown in Table 6

In order to determine the parameters of the genetic algorithm, the Taguchi method is used to establish the S/N ratio (Signal-to-Noise ratio) main effect diagram, which is defined in Eq. (34).

$$\frac{S}{N} = -10 \log \left(\frac{\sum_{i=1}^{n} \frac{1}{Y_i^2}}{n} \right) \tag{28}$$

[2] Now, the School of Management Engineering is School of Management. The official website of the School of Management Zhengzhou University is http://www7.zzu.edu.cn/glxy/.

Table 5
Parameter level of genetic algorithm.

Level	Parameters			
	Population size	Number of iterations	Crossover probability	Mutation probability
1	20	50	0.55	0.05
2	30	60	0.6	0.1
3	40	70	0.65	0.15
4	50	80	0.7	0.2
5	60	90	0.75	0.25

Table 6
Genetic algorithm L_{25} orthogonal experiment.

Experiment	Parameter			
	Population size	Number of iterations	Crossover probability	Mutation probability
1	20	50	0.55	0.05
2	20	60	0.60	0.10
3	20	70	0.65	0.15
4	20	80	0.70	0.20
5	20	90	0.75	0.25
6	30	50	0.60	0.15
7	30	60	0.65	0.20
8	30	70	0.70	0.25
9	30	80	0.75	0.05
10	30	90	0.55	0.10
11	40	50	0.65	0.25
12	40	60	0.70	0.05
13	40	70	0.75	0.10
14	40	80	0.55	0.15
15	40	90	0.60	0.20
16	50	50	0.70	0.10
17	50	60	0.75	0.15
18	50	70	0.55	0.20
19	50	80	0.60	0.25
20	50	90	0.65	0.05
21	60	50	0.75	0.20
22	60	60	0.55	0.25
23	60	70	0.60	0.05
24	60	80	0.65	0.10
25	60	90	0.70	0.15

Fig. 9 shows the Main effects plot for S/N ratios. According to the established signal-to-noise ratio main effect diagram, the four parameters of the genetic algorithm should take the values of population size 40, number of iterations 90, crossover probability 0.6, and mutation probability 0.2.

6.3. Comparison of genetic algorithm with mathematical modeling

In the following experiments, we compare the solutions obtained by the genetic algorithm with mathematical modeling. The parameters of the genetic algorithm are shown in Section 6.2. The linearized mathematical model is solved by the IBM Cplex. We used the default parameter setting of IBM Cplex. When the problem size is large, integer programming solvers usually run out of memory. To prevent CPLEX from running out of memory and running large-size problems, this paper modifies the parameter cplex.nodefileind = 3 to store the node file on the hard disk and compress it. The genetic algorithm is not deterministic, and this paper runs the genetic algorithm 10 times. Table 7 shows the solutions obtained by the genetic algorithm and mathematical model. From Table 7, we can find that the mathematical model could find the optimal solution for instances 1, 2, and 3. In other instances, the mathematical model could not obtain the optimal solutions within four hours. The genetic algorithm developed by this paper could obtain the optimal solutions for instances 1, 2, and 3. However, the computational time of the genetic algorithm is worse for instances 1 and 2.

To evaluate the Gaps of the objective function values and computational time obtained from the genetic algorithm and mathematical model, this paper defines the Gap_o and Gap_t, which are formulated in Eqs. (35) and (36), respectively.

$$Gap_o = \frac{Obj_{GA} - Obj_{MM}}{Obj_{MM}} * 100\% \tag{29}$$

$$Gap_t = \frac{Time_{MM} - Time_{GA}}{Time_{MM}} * 100\% \tag{30}$$

The following Fig. 10 shows the comparison of Gap_o and Gap_t for different instances. $Gap_o = 0$ means that the genetic algorithm mathematically obtains the same objective function value. Gap_o and Gap_t are positive values, which means that the performance of the genetic algorithm is better than the mathematical model. Otherwise, the performance of the mathematical model is better than that of the proposed genetic algorithm.

6.4. Comparison with different objective functions

The mathematical model, LP_1, maximizes all undergraduate students' total degree of familiarity. This section introduces two new objective functions, minimizing the degree of similarity differences between groups and minimizing the sum of the degree of similarity differences within each group, which are defined in models LP_2 and LP_3, respectively.

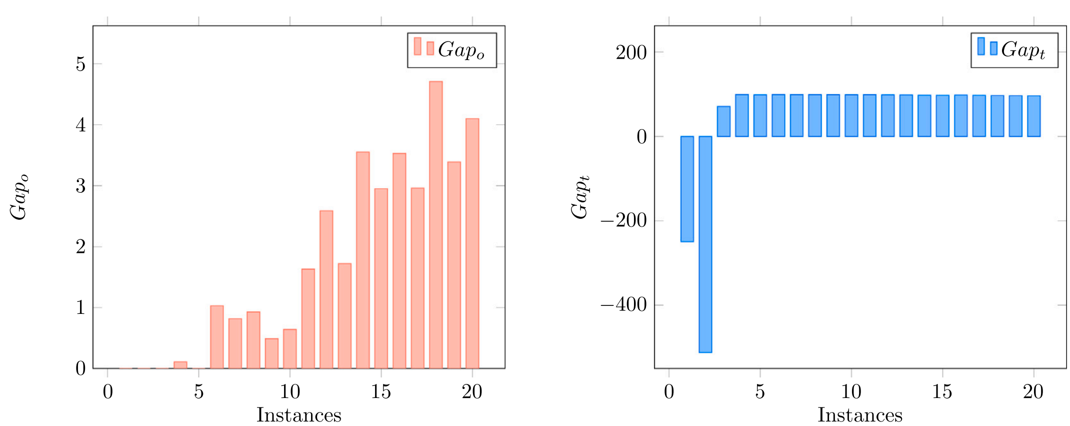
$$LP_2 \qquad Min \max_{t\in T,p\in P,l\in L,s\in S}\left\{z_{lstp}c_{ls}k_{tp}\right\} - \min_{t\in T,p\in P,l\in L,s\in S,x_{ltp}+y_{stp}+k_{tp}=3}\left\{z_{lstp}c_{ls}k_{tp}\right\} \tag{31}$$

subject to constraints (3)–(15) and (18)–(21).

Table 7
Solutions obtained by genetic algorithm and mathematical model.

Instances	Genetic algorithm						Linear programming model	
	Objective value			Time (s)			Objective value	Time (s)
	Max	Min	AVG	Max	Min	AVG		
1	**2892**	2889	2890.2	14.18	13.97	14.06	2892	4
2	**7067**	7033	7053.4	24.82	24.48	24.67	7067	4
3	**13 818**	13 795	13 815.7	41.60	41.21	41.45	13 818	142
4	20 495	20 495	20 495	59.43	58.16	58.80	20 472[a]	14 400
5	**27 024**	26 995	27 013.6	72.66	71.95	72.19	27 024	5672
6	17 264	17 140	17 209.4	52.41	51.76	51.98	17 088[a]	14 400
7	25 742	25 488	25 616.9	74.31	73.42	73.88	25 532[a]	14 400
8	33 979	33 653	33 798.8	93.49	91.81	92.66	33 666[a]	14 400
9	42 194	41 921	42 055.9	119.08	118.28	118.70	41 987[a]	14 400
10	50 693	50 363	50 518.6	142.06	141.08	141.50	50 372[a]	14 400
11	15 083	14 912	15 000.9	114.45	111.85	112.85	14 841[a]	14 400
12	22 501	22 201	22 368.5	166.07	164.65	165.45	21 934[a]	14 400
13	29 683	29 485	29 569.9	219.38	216.90	218.24	29 180[a]	14 400
14	37 201	36 716	36 940.2	273.22	271.23	271.74	35 926[a]	14 400
15	44 632	44 280	44 447.2	328.12	325.71	326.76	43 351[a]	14 400
16	36 617	36 147	36 335.9	260.73	258.33	258.98	35 370[a]	14 400
17	45 379	45 103	45 229.5	324.23	322.60	323.32	44 073[a]	14 400
18	54 438	53 897	54 206	389.82	387.79	388.87	51 990[a]	14 400
19	63 734	63 043	63 380	453.93	451.21	452.37	61 642[a]	14 400
20	72 304	71 901	72 113.3	505.48	502.76	503.94	69 457[a]	14 400

[a] It took 4 h to solve using CPLEX and still failed to find the solution. The solution was obtained after interruption.



(a) (b)

Fig. 10. Comparison of Gap between the proposed algorithm and mathematical model.

$$LP_3 \; Min \quad \sum_{t=1}^{T}\sum_{p=1}^{P}\left\{\max_{l\in L, s\in S}\left\{z_{lstp}c_{ls}k_{tp}\right\} - \min_{l\in L, s\in S, x_{ltp}+y_{stp}+k_{tp}=3}\left\{z_{lstp}c_{ls}k_{tp}\right\}\right\} \tag{32}$$

subject to constraints (3)–(15) and (18)–(21).

Models LP_2 and LP_3 are two fairness models that consider the fairness of receiving thesis evaluation. The objective function value of the models LP_1, LP_2 and LP_3 are Z_1, Z_2 and Z_3, respectively. The optimal objective function value of the models LP_1, LP_2 and LP_3 are Z_1^*, Z_2^* and Z_3^*, respectively. Table 8 shows the comparison of the solutions obtained by mathematical models LP_1, LP_2, and LP_3. In the following experiment, the mathematical models LP_1, LP_2, and LP_3 are solved by CPLEX, and the maximum running time of CPLEX is one hour. As mentioned before, the studied problem is difficult to solve by using the mathematical modeling approach for large-scale problems. To ensure the mathematical model could obtain the optimal solutions in an acceptable time, Table 8 only adopts the small-scale instances 1, 2, and 3. The instances 1-1, 1-2, and 1-3 are modified versions of instance 1 by changing the degree of similarity.

Table 8
Comparison of the solutions obtained by mathematical models LP_1, LP_2 and LP_3.

Instances	Model LP_1			Model LP_2			Model LP_3		
	Z_1^*	Z_2	Z_3	Z_1	Z_2^*	Z_3	Z_1	Z_2	Z_3^*
1-1	**2892**	75	231	2727	**55**	210	2105	60	**155**
1-2	**2781**	66	231	2417	**65**	219	2219	71	**178**
1-3	**2919**	72	257	2727	**62**	217	2367	82	**166**
2-1	**7067**	73	265	5820	**70**	250	5654	74	**227**[a]
2-2	**7118**	83	287	6627	**69**	261	6732	79	**222**[a]
2-3	**6572**	84	309	5144	**73**	282	5103	86	**281**[a]
3-1	**13 818**	89	343	12 577	**74**	282	12 221	88	**252**[a]
3-2	**13 288**	86	314	10 938	**77**	299	11 466	87	**249**[a]
3-3	**13 528**	89	347	10 476	**78**	307	10 849	89	**284**[a]

[a] Denotes the solution obtained in 1 h.

The values of Z_2 and Z_3 in the second column with model LP_1 are calculated using the optimal solutions of Z_1^*. The values of Z_1 and Z_3 in the third column with model LP_2 are computed using the optimal solutions of Z_2^*. The values of Z_1 and Z_2 in the fourth column with model LP_3 are calculated using the optimal solutions of Z_3^*. Z_1^* is the maximum value in each row. Z_2^* and Z_3^* are the minimum values in

Table 9
Solutions obtained by genetic algorithm and mathematical model by using repair rule 2.

Instances	Genetic algorithm						Model LP_1	
	Objective value			Time (s)			Objective value	Time (s)
	Max	Min	AVG	Max	Min	AVG		
1	2892	2885	2890.4	14.30	13.97	14.07	2892	4
2	7067	7033	7060.2	24.85	24.45	24.66	7067	4
3	13 818	13 795	13 815.7	41.52	41.12	41.37	13 818	142
4	20 495	20 495	20 495	58.90	58.35	58.64	20 472#	14 400
5	27 024	27 007	27 022.3	72.53	71.91	72.12	27 024	5672
6	17 263	17 117	17 210.9	52.14	51.66	51.94	17 088#	14 400
7	25 742	25 385	25 588.1	74.15	73.59	73.89	25 532#	14 400
8	33 854	33 617	33 735.3	92.97	92.26	92.67	33 666#	14 400
9	42 120	41 953	42 037	119.72	118.58	119.04	41 987#	14 400
10	50 660	50 247	50 501	142.50	140.93	141.66	50 372#	14 400
11	15 104	14 935	15 018.2	113.51	112.00	112.45	14 841#	14 400
12	22 529	22 202	22 339.3	165.79	164.92	165.19	21 934#	14 400
13	29 704	29 470	29 588	218.85	217.81	218.27	29 180#	14 400
14	37 206	36 978	37 053.9	273.54	271.37	272.13	35 926#	14 400
15	44 516	44 240	44 414.9	327.64	326.33	326.88	43 351#	14 400
16	36 444	35 983	36 211.7	260.39	258.07	258.92	35 370#	14 400
17	45 599	45 104	45 315.8	323.72	321.45	322.52	44 073#	14 400
18	54 491	53 961	54 178.9	389.28	387.55	388.41	51 990#	14 400
19	63 552	62 991	63 349.8	457.31	452.96	454.33	61 642#	14 400
20	72 223	71 781	72 041.9	507.08	505.78	506.38	69 457#	14 400

each row. The Table 8 shows that the model LP_3 is challenging to solve compared with the model LP_1 and LP_3.

6.5. Analysis of repair operation

This section analyzes the proposed algorithm with different repair operation rules defined in the previous section.

6.5.1. With and without repair operation

First, this paper examines the variation in the number of feasible solutions in each iteration. Fig. 11 shows the number of feasible solutions for each instance. From Fig. 11, we can find that the number of feasible solutions dramatically decreased at the proposed algorithm's beginning. The waves in Fig. 11 are due to the crossover or mutation operation for generating feasible solutions. Fig. 12 presents a violin plot illustrating the distribution of feasible solutions across generations in ten experiments conducted on Instance 20 without repair operations. Fig. 13 presents the algorithm's convergence curve for Instance 20. The blue and orange lines indicate cases with and without repair operations, respectively. Fig. 14 presents the average number of repair operations per generation for each instance.

6.5.2. Repair operation rules analysis

This subsection analyzes three different repair operation rules regarding solution quality and computational time. The solutions obtained by the genetic algorithm and mathematical model using repair rule 1 are shown in Table 7. Solutions obtained by the genetic algorithm and mathematical model using repair rules 2 and 3 are shown in Tables 9 and 10, respectively. In Tables 9 and 10, # denotes that the best solution was obtained after an interruption for running 4 h by using CPLEX.

Fig. 15 shows the comparison of Gap_o for different repair operation rules.

From Fig. 16, we can find that the proposed algorithm has a similar computational time for the three different repair rules.

7. Applications

This section introduces the application of the studied problem. The proposed algorithm could be used for undergraduate students' defense assignments. This section presents how to develop a system by using the proposed algorithm.

7.1. System overview

This system is a Web-based application developed using the front-end and back-end separation development model. The front end uses the Vue open-source framework, the back end is developed using the Django open-source web framework, and Celery and Redis are used to implement asynchronous tasks. The system database uses the MySQL database. The developed system is deployed on a server with a 3.00 GHz Intel Core i9-13900K CPU and 16 GB of memory. Fig. 17 shows the screenshot of the developed system.

7.2. System framework

The proposed system framework is shown in Fig. 18. From Fig. 18, we can find that the proposed system framework is divided into four parts, including expert management, student management, assignment management, and visualization management.

7.2.1. Expert management

The expert management module is designed to determine the committee and their publications. The users of the developed system could upload the expert list, including their names and affiliations. After the users upload the expert list, the developed system automatically collects the experts' academic papers. Python language is selected to crawl the data of the review committee's academic papers. According to the data requirements, the papers published by the committee are retrieved through the name and affiliation of the expert. After that, the Jieba library is adopted for text segmentation. Finally, we adopted spaCy and KeyBERT libraries to calculate the degree of similarity between experts and students. All those data are saved into the database.

7.2.2. Student management

The student management module is used for collecting students' information, including thesis information. This module is designed to extract the keywords from the student's thesis, which is prepared for calculating the degree of similarity between experts and students. Different from the expert management module, the student's thesis information is directly uploaded to the developed system by the students themselves. The student management module uses the same methods that are used in the expert management module for text segmentation.

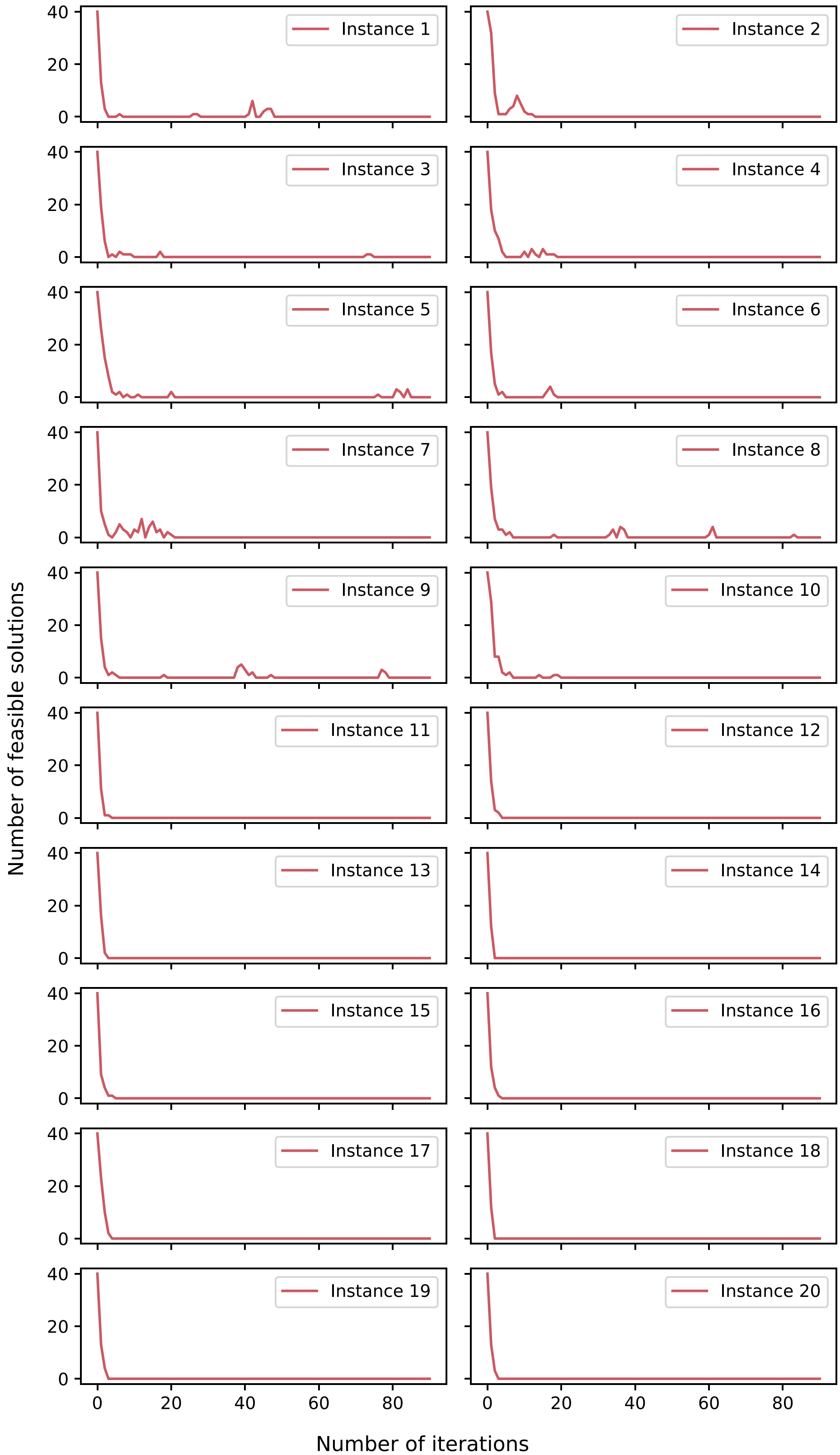


Fig. 11. The average number of feasible solutions for each instance.

7.2.3. Assignment management

The developed system enables concurrent multi-user access. The Celery, which is a package that is easy to use and maintain, is adopted for asynchronous task queues/job queues. In the developed system, we used the Celery for assignment management. In Fig. 18, the solver denotes the proposed mathematical formulation based genetic algorithm. After the solver obtains the solution, the scheduling results are stored in the database.

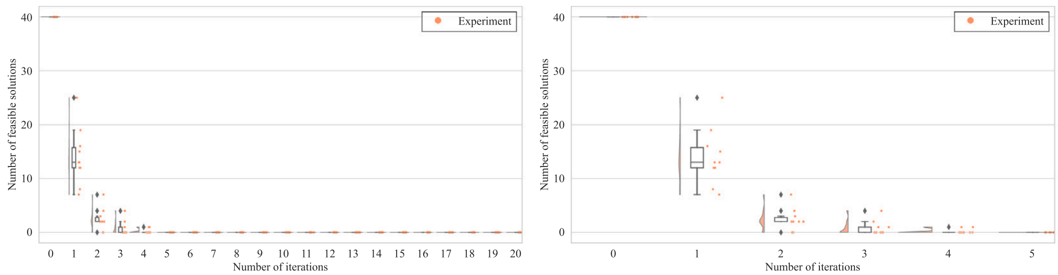


Fig. 12. Violin plot of the number of feasible solutions for Instance 20.

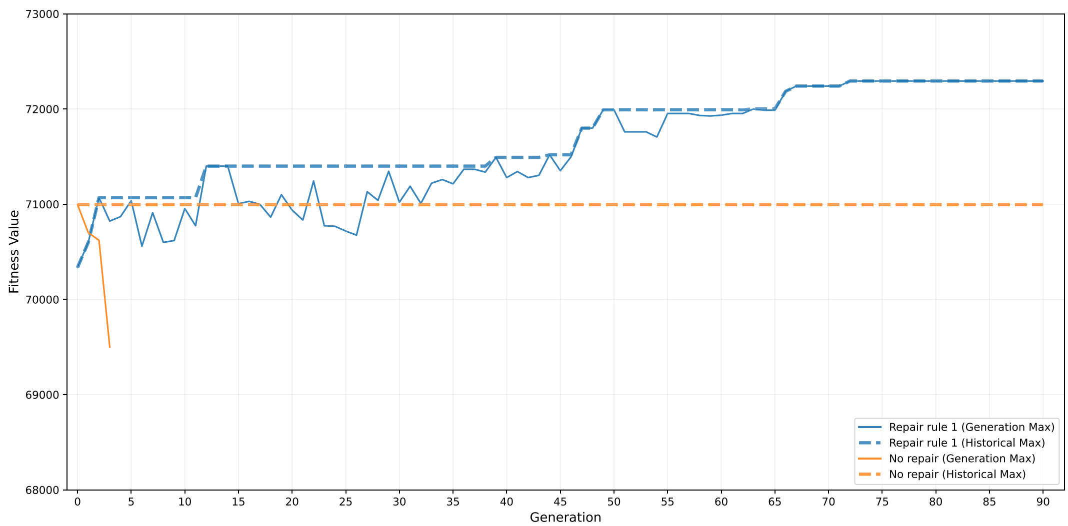


Fig. 13. Genetic Algorithm Convergence Trends.

Table 10
Solutions obtained by genetic algorithm and mathematical model by using repair rule 3.

Instances	Genetic algorithm						Model LP_1	
	Objective value			Time (s)			Objective value	Time (s)
	Max	Min	AVG	Max	Min	AVG		
1	2892	2889	2891.4	14.34	14.10	14.23	2892	4
2	7067	7023	7055.8	25.12	24.82	25.03	7067	4
3	13 818	13 798	13 814	42.54	41.84	42.21	13 818	142
4	20 495	20 495	20 495	59.98	59.47	59.79	20 472#	14 400
5	27 024	26 995	27 016	74.21	73.52	73.82	27 024	5672
6	17 263	17 065	17 222.1	53.32	52.81	53.03	17 088#	14 400
7	25 658	25 445	25 568.5	75.89	75.02	75.38	25 532#	14 400
8	33 840	33 521	33 705.3	95.62	94.47	94.80	33 666#	14 400
9	42 120	41 953	42 050.3	121.95	120.91	121.49	41 987#	14 400
10	50 660	50 163	50 313.7	145.95	144.30	144.88	50 372#	14 400
11	15 107	14 828	14 977	115.40	114.59	114.89	14 841#	14 400
12	22 519	22 199	22 348.3	169.78	168.61	169.27	21 934#	14 400
13	29 723	29 281	29 547.7	224.03	222.70	223.36	29 180#	14 400
14	37 061	36 562	36 863.7	279.72	278.48	279.09	35 926#	14 400
15	44 510	44 025	44 282.2	335.94	334.37	335.35	43 351#	14 400
16	36 341	35 968	36 186.7	267.50	264.69	265.55	35 370#	14 400
17	45 619	44 831	45 228.3	331.60	329.96	330.97	44 073#	14 400
18	54 709	53 927	54 229.4	399.64	396.42	398.24	51 990#	14 400
19	63 588	63 061	63 354.7	466.11	463.44	464.88	61 642#	14 400
20	72 587	71 864	72 136.2	521.11	518.45	519.69	69 457#	14 400

7.2.4. Visualization management

Visualization management is one of the critical tools used in many decision support systems for providing data analysis. In the developed system, a visualization management module is integrated to display the scheduling results.

7.3. Expert feedback

The developed system was tested using the undergraduate student's thesis defense from the School of Management at Zhengzhou University. We invited administrative staff from Zhengzhou University to use the developed system to schedule the thesis defense. After using

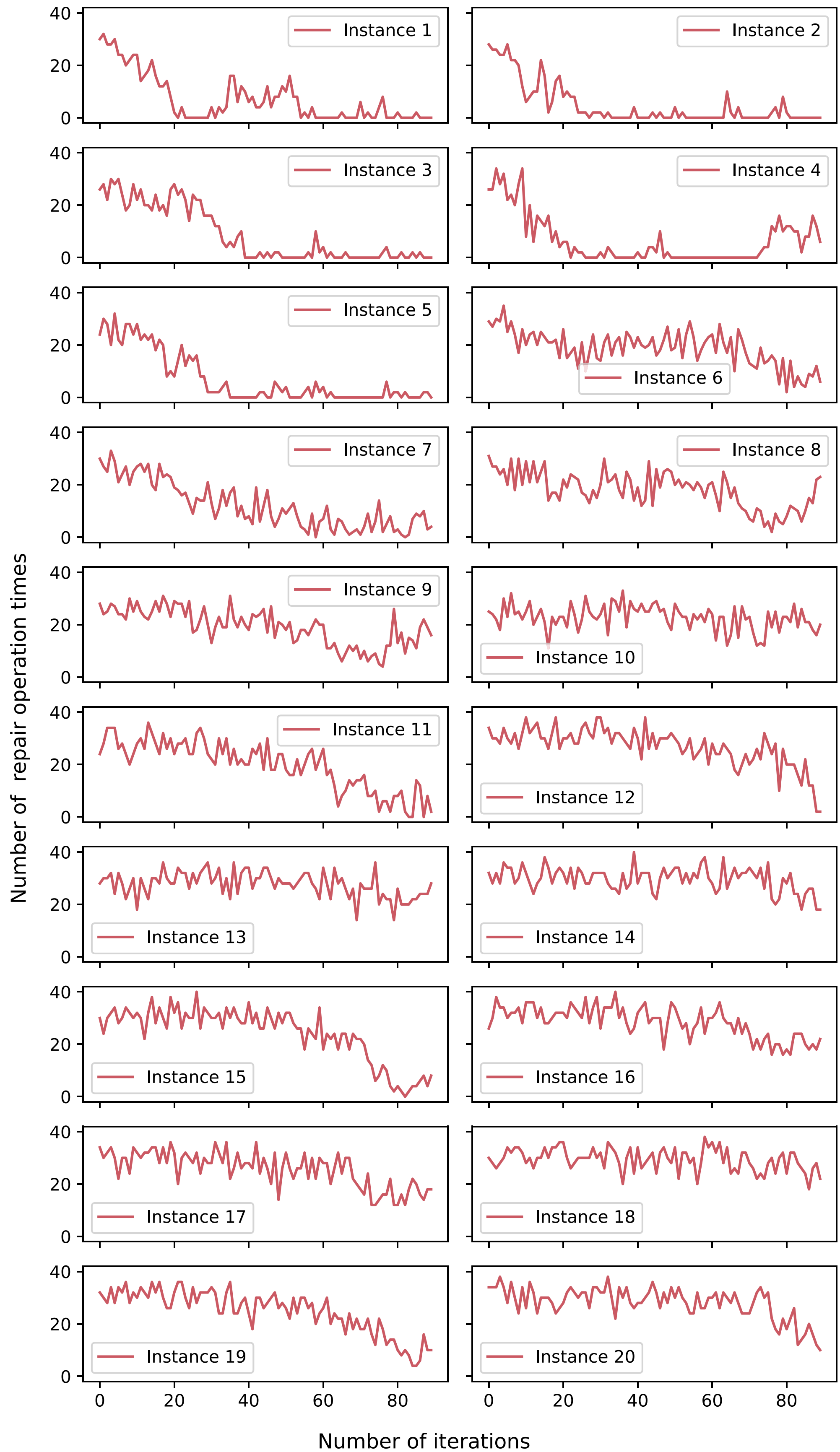


Fig. 14. The average number of times repair operations are performed for each instance.

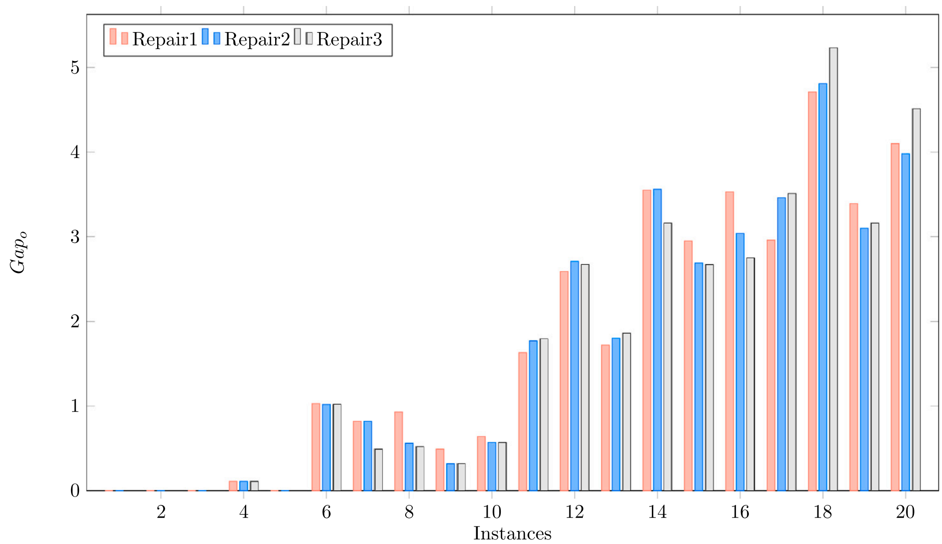


Fig. 15. Comparison of Gap_o for different repair operation rules.

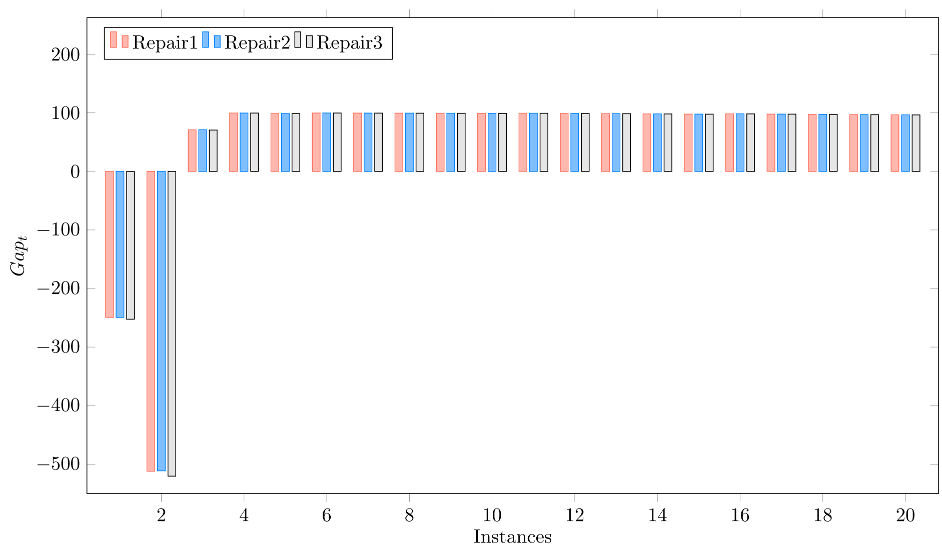


Fig. 16. Comparison of Gap_t for different repair operation rules.

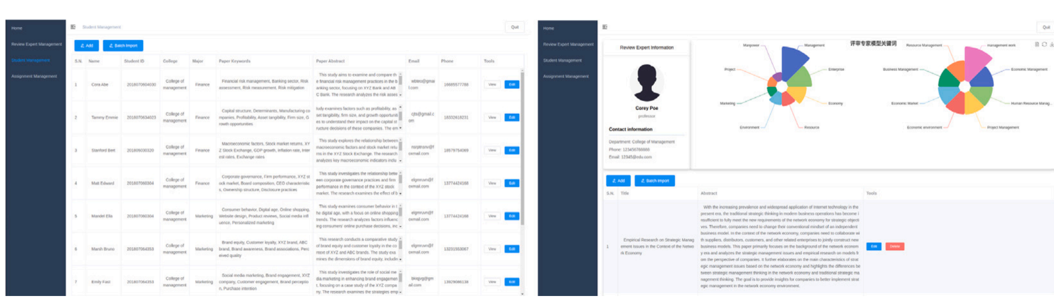


Fig. 17. Screenshot of the developed system.

it, we conducted an in-depth interview with the administrative staff. The administrative staff has a positive attitude towards the developed system. They said the system interaction is very user-friendly, with the committee list uploaded and the scheduling results returned. Compared with the previous method, the system saved a lot of work.

7.4. Discussions

This subsection discusses the limitations of this paper. A committee could or could not be assigned to the group with a student under their supervision, which is two different rules. The proposed model considered that a committee could not be assigned to the group with a

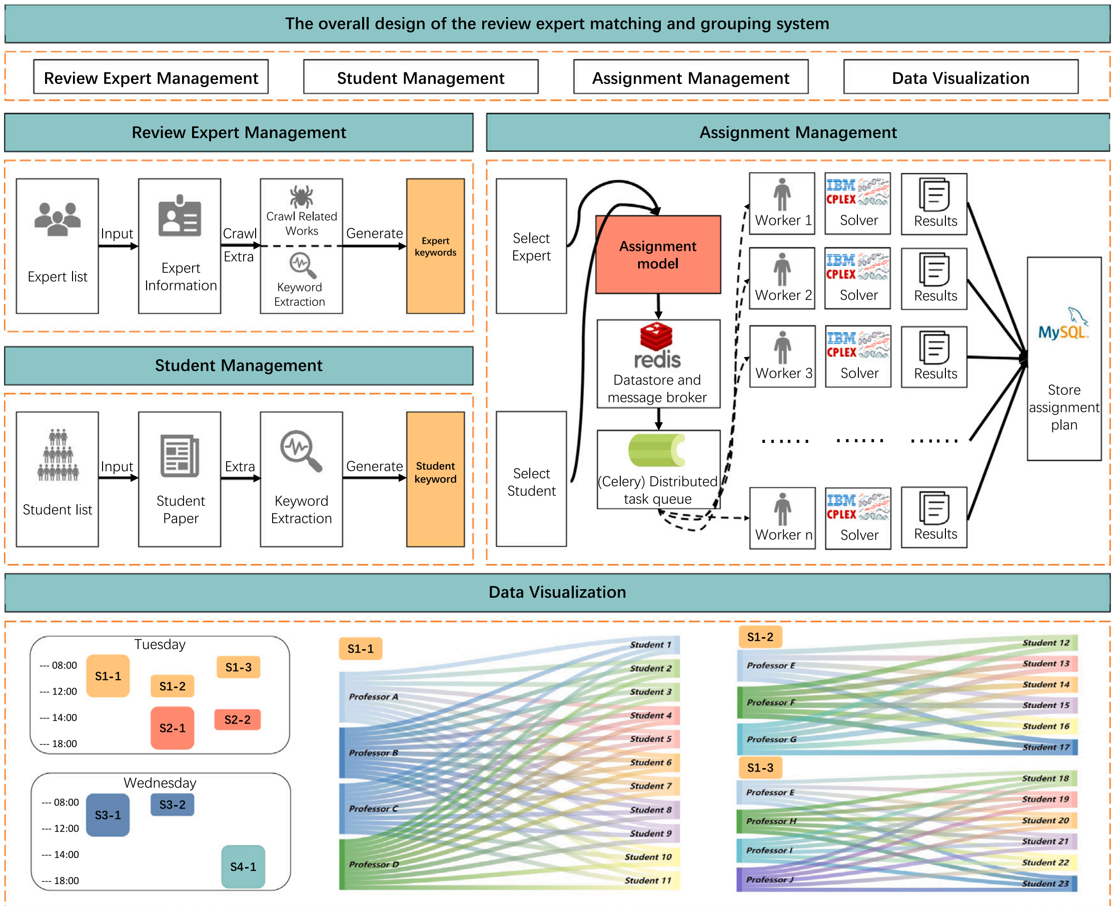


Fig. 18. The proposed system framework.

student under their supervision. However, if the committee is required to be assigned to the defense team, the constraint (33) should be added to the studied model, making the model more complex and challenging to solve.

$$(x_{ltp} + y_{stp} + a_{ls})a_{ls}x_{ltp}y_{stp} = 3a_{ls}x_{ltp}y_{stp} \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{33}$$

In the above mathematical model, we assume all the committees have time available during the defense period. However, some committees may only attend some rounds, so they may not have time for the defense period. In future system development, we could consider adding the time available constraints for the defense committees.

8. Conclusions

Undergraduate student's thesis defense is essential for students' graduation. This paper studies the scheduling of undergraduate students' thesis defenses by giving a set of committees considering equality and efficiency. Three nonlinear integer programming models are developed to formulate the studied problem with different objective functions. To obtain the optimal solutions, linearization methods are adopted to linearize the proposed nonlinear integer programming models. Due to the computational complexity, a mathematical formulation-based genetic algorithm is proposed. A comprehensive experiment is conducted, and the results show the effectiveness and efficiency of the proposed algorithm. Finally, a web-based system is developed for the studied problem using the proposed algorithm.

The future studies could be extended by considering the following aspects: (1) A committee could be assigned to the group with a student under their supervision; (2) Time availability of the committee could be considered as a constraint; (3) Multiple objective functions could be considered simultaneously. The system developed by this paper has some limitations. We will continuously develop the system by considering future studies.

CRediT authorship contribution statement

Yanjie Zhou: Writing – review & editing, Supervision, Methodology, Conceptualization, Writing – original draft, Project administration, Funding acquisition. **Zhicheng Zhang:** Visualization, Methodology, Writing – original draft, Software, Data curation. **Ningning Song:** Visualization, Formal analysis, Writing – original draft, Methodology. **Jincan Zhang:** Writing – review & editing, Validation, Methodology, Writing – original draft, Supervision.

Declaration of competing interest

All authors disclosed no relevant relationships.

Acknowledgments

This research was supported by National Natural Science Foundation of China (72201252) and Henan Zhongyuan Medical Science and Technology Innovation and Development Foundation, China (24YCG2006).

Appendix A. Simulated annealing algorithm

To verify the effectiveness and advantages of the genetic algorithm employed in this study, we conducted a control experiment using

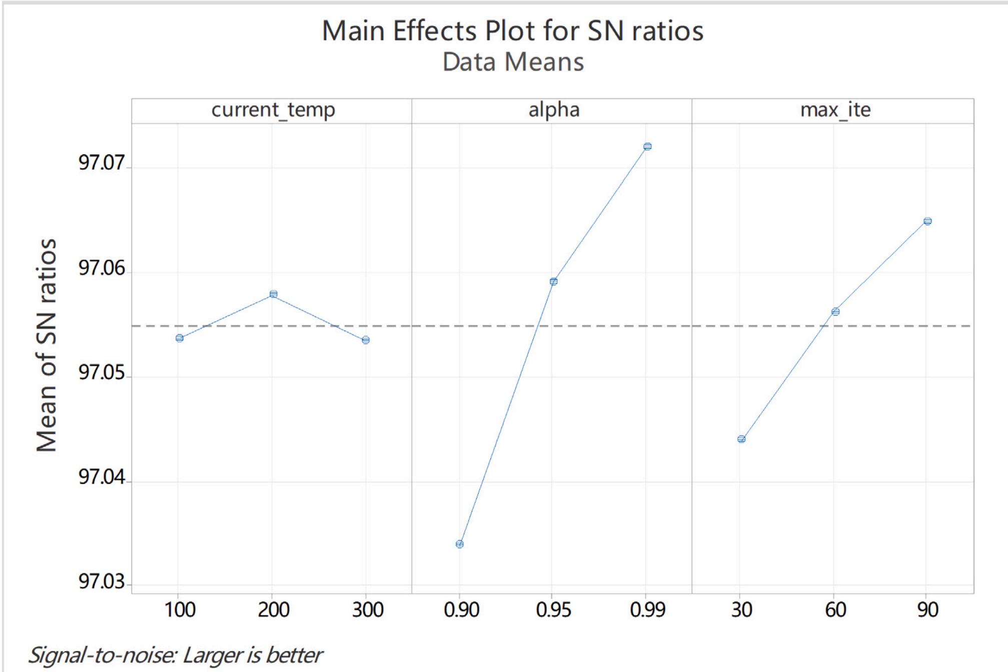


Fig. 19. Main effects plot for S/N ratios.

Table 11
Parameter level of simulated annealing algorithm.

Level	Parameters		
	Initial temperature	Cooling coefficient	Number of iterations at each temperature level
1	100	0.90	30
2	200	0.95	60
3	300	0.99	90

Table 12
Simulated annealing algorithm L_9 orthogonal experiment.

Experiment	Parameter		
	Initial temperature	Cooling coefficient	Number of iterations at each temperature level
1	100	0.90	30
2	100	0.95	60
3	100	0.99	90
4	200	0.90	60
5	200	0.95	90
6	200	0.99	30
7	300	0.90	90
8	300	0.95	30
9	300	0.99	60

the simulated annealing algorithm (SA), another widely recognized heuristic optimization technique.

Since simulated annealing is a heuristic method that does not guarantee an exact solution, we similarly apply the Taguchi method to fine-tune its parameters. In our study, the Taguchi method optimizes key parameters, including the initial temperature, the cooling coefficient, and the number of iterations at each temperature level, to improve the overall performance and convergence of the algorithm. Che (2012) Three levels are selected for each parameter, which is shown in Table 11.

According to the Taguchi method, the orthogonal matrix L_9 is used to design the experimental arrangement, which is shown in Table 12

In order to determine the parameters of the simulated annealing algorithm, the Taguchi method is used to establish the S/N ratio main effect diagram, which is defined in Eq. (34).

$$\frac{S}{N} = -10\log\left(\frac{\sum_{i=1}^{n}\frac{1}{Y_i^2}}{n}\right) \tag{34}$$

Fig. 19 shows the Main effects plot for S/N ratios. According to the established signal-to-noise ratio main effect diagram, the three parameters of the simulated annealing algorithm should take the values of initial temperature 200, cooling coefficient 0.99, and number of iterations at each temperature level 90.

In the following experiments, we compare the solutions obtained by the simulated annealing algorithm with the genetic algorithm. The simulated annealing algorithm is not deterministic, and this paper runs the simulated annealing algorithm 10 times. Table 13 shows the solutions obtained by the genetic and simulated annealing algorithms.

To evaluate the Gaps of the objective function values and computational time obtained from the genetic algorithm and simulated annealing algorithm, this paper defines the Gap_o and Gap_t, which are formulated in Eqs. (35) and (36), respectively.

$$Gap_o = \frac{Obj_{GA} - Obj_{SA}}{Obj_{SA}} * 100\% \tag{35}$$

$$Gap_t = \frac{Time_{SA} - Time_{GA}}{Time_{SA}} * 100\% \tag{36}$$

Fig. 20 illustrates the comparison of Gap_o and Gap_t for various instances. Both Gap_o and Gap_t are positive, indicating that the genetic algorithm outperforms the simulated annealing algorithm. In particular, the genetic algorithm achieves a superior target value and requires less computational time compared to simulated annealing.

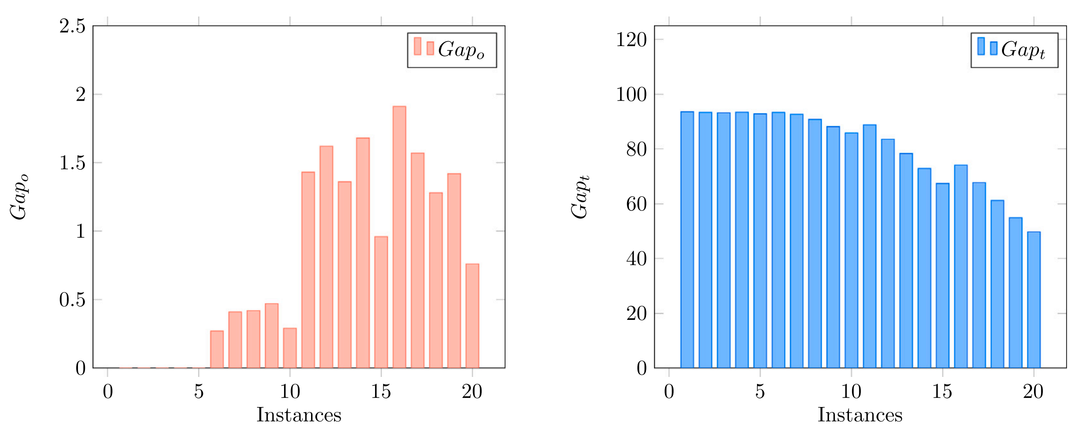
Appendix B. Extended model

This appendix presents an extension of the proposed model to accommodate additional scheduling rules that may apply in other universities. In the base model, we assume that a student's advisor cannot be on the defense committee of their own student, to avoid conflicts of interest. However, in some universities, the advisor is required to be part of the defense group. To accommodate this rule, the following

Table 13
Solutions obtained by simulated annealing algorithm and genetic algorithm.

Instances	Genetic algorithm						Simulated annealing algorithm					
	Objective value			Time (s)			Objective value			Time (s)		
	Max	Min	AVG	Max	Min	AVG	Max	Min	AVG	Max	Min	AVG
1	2892	2889	2890.2	14.18	13.97	14.06	2892	2892	2892	225.35	219.18	222.47
2	7067	7033	7053.4	24.82	24.48	24.67	7067	7067	7067	420.46	369.30	378.71
3	13 818	13 795	13 815.7	41.60	41.21	41.45	13 818	13 818	13 818	637.67	607.85	619.08
4	20 495	20 495	20 495	59.43	58.16	58.80	20 495	20 495	20 495	998.90	883.35	909.41
5	27 024	26 995	27 013.6	72.66	71.95	72.19	27 024	27 024	27 024	1000[a]	1000[a]	1000[a]
6	17 264	17 140	17 209.4	52.41	51.76	51.98	17 218	17 031	17 130.4	826.40	779.63	789.82
7	25 742	25 488	25 616.9	74.31	73.42	73.88	25 636	25 388	25 526.4	1000[a]	1000[a]	1000[a]
8	33 979	33 653	33 798.8	93.49	91.81	92.66	33 836	33 544	33 679.6	1000[a]	1000[a]	1000[a]
9	42 194	41 921	42 055.9	119.08	118.28	118.70	41 995	41 720	41 862.7	1000[a]	1000[a]	1000[a]
10	50 693	50 363	50 518.6	142.06	141.08	141.50	50 548	50 092	50 242.1	1000[a]	1000[a]	1000[a]
11	15 083	14 912	15 000.9	114.45	111.85	112.85	14 870	14 715	14 776.6	1000[a]	1000[a]	1000[a]
12	22 501	22 201	22 368.5	166.07	164.65	165.45	22 143	21 950	22 010.1	1000[a]	1000[a]	1000[a]
13	29 683	29 485	29 569.9	219.38	216.90	218.24	29 284	29 156	29 195.9	1000[a]	1000[a]	1000[a]
14	37 201	36 716	36 940.2	273.22	271.23	271.74	36 585	36 428	36 488.4	1000[a]	1000[a]	1000[a]
15	44 632	44 280	44 447.2	328.12	325.71	326.76	44 208	43 782	43 954.2	1000[a]	1000[a]	1000[a]
16	36 617	36 147	36 335.9	260.73	258.33	258.98	35 932	35 625	35 713.4	1000[a]	1000[a]	1000[a]
17	45 379	45 103	45 229.5	324.23	322.60	323.32	44 677	44 452	44 566.9	1000[a]	1000[a]	1000[a]
18	54 438	53 897	54 206	389.82	387.79	388.87	53 752	53 447	53 560.8	1000[a]	1000[a]	1000[a]
19	63 734	63 043	63 380	453.93	451.21	452.37	62 841	62 353	62 544.6	1000[a]	1000[a]	1000[a]
20	72 304	71 901	72 113.3	505.48	502.76	503.94	71 760	71 132	71 323.3	1000[a]	1000[a]	1000[a]

[a] It took 1000 s to solve and still failed to find the solution. The solution was obtained after interruption.



(a) (b)

Fig. 20. Comparison of *Gap* between the genetic algorithm and simulated annealing algorithm.

constraint can be added:

$$Model \quad P_1 \tag{37}$$

subject to constraints (3)–(12), (14)–(15), and

$$(x_{ltp} + y_{stp} + a_{ls})a_{ls}x_{ltp}y_{stp} = 3a_{ls}x_{ltp}y_{stp} \qquad \forall l \in L, s \in S, t \in T, p \in P \tag{38}$$

Integrating this constraint increases the complexity of the model, especially in large-scale scenarios with many students and advisors, as it introduces strong coupling between student-group and committee-group assignments. However, this extension demonstrates the flexibility and generalizability of the proposed model, showing that it can be customized for various university rules regarding committee composition.

Data availability

Data will be made available on request.

References

Almeida, J., Santos, D., Figueira, J.R., Francisco, A.P., 2024. A multi-objective mixed integer linear programming model for thesis defence scheduling. European J. Oper. Res. 312 (1), 92–116.

Anjum, O., Gong, H., Bhat, S., Hwu, W.-M., Xiong, J., 2019. PaRe: A paper-reviewer matching approach using a common topic space. arXiv preprint arXiv:1909.11258.

Battistutta, M., Ceschia, S., De Cesco, F., Di Gaspero, L., Schaerf, A., 2019. Modelling and solving the thesis defense timetabling problem. J. Oper. Res. Soc. 70 (7), 1039–1050.

Bi, X., Yang, M., Adomavicius, G., 2024. Consumer acquisition for recommender systems: A theoretical framework and empirical evaluations. Inf. Syst. Res. 35 (1), 339–362. http://dx.doi.org/10.1287/isre.2023.1229.

Charlin, L., Zemel, R., 2013. The toronto paper matching system: an automated paper-reviewer assignment system.

Charlin, L., Zemel, R.S., Boutilier, C., 2011. A framework for optimizing paper matching. In: UAI. Vol. 11, pp. 86–95.

Che, Z., 2012. Clustering and selecting suppliers based on simulated annealing algorithms. Comput. Math. Appl. 63 (1), 228–238.

Dimitsas, A., Gogos, C., 2024. Finding near optimal solutions to the Thesis Defense Timetabling problem by exploiting symmetries. In: Operations Research Forum. Vol. 5, Springer, p. 65.

Dimitsas, A., Gogos, C., Pappa, E., 2022. Better solutions for the Thesis Defense Timetabling problem using a three-phase approach. In: Proceedings of the 26th Pan-Hellenic Conference on Informatics. pp. 58–63.

Ferilli, S., Di Mauro, N., Basile, T.M.A., Esposito, F., Biba, M., 2006. Automatic topics identification for reviewer assignment. In: Advances in Applied Artificial

Intelligence: 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2006, Annecy, France, June 27-30, 2006. Proceedings 19. Springer, pp. 721–730.

Gao, X., Zhou, Y., Amir, M.I.H., Rosyidah, F., Lee, G., 2017. A hybrid genetic algorithm for multi-emergency medical service center location-allocation problem in disaster response. Int. J. Ind. Eng.: Theory Appl. Pr. 24, 663–679.

Ghasemi, P., Goli, A., Goodarzian, F., Ehmke, J.F., 2025. Simulation-based genetic algorithm for optimizing a municipal cooperative waste supply chain in a pandemic. Eng. Appl. Artif. Intell. 139, 109478. http://dx.doi.org/10.1016/j.engappai.2024.109478.

Gu, J., Zhou, Y., Das, A., Moon, I., Lee, G.M., 2018. Medical relief shelter location problem with patient severity under a limited relief budget. Comput. Ind. Eng. 125, 720–728. http://dx.doi.org/10.1016/j.cie.2018.03.027.

Hartvigsen, D., Wei, J.C., Czuchlewski, R., 1999. The conference paper-reviewer assignment problem. Decis. Sci. 30 (3), 865–876. http://dx.doi.org/10.1111/j.1540-5915.1999.tb00910.x.

Kalmukov, Y., 2020. An algorithm for automatic assignment of reviewers to papers. Scientometr. 124 (3), 1811–1850.

Kou, N.M., U, L.H., Mamoulis, N., Gong, Z., 2015a. Weighted coverage based reviewer assignment. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. pp. 2031–2046.

Kou, N.M., U, L.H., Mamoulis, N., Li, Y., Li, Y., Gong, Z., 2015b. A topic-based reviewer assignment system. Proc. VLDB Endow. 8 (12), 1852–1855.

Li, X., Watanabe, T., 2013. Automatic paper-to-reviewer assignment, based on the matching degree of the reviewers. Procedia Comput. Sci. 22, 633–642.

Liu, X., Suel, T., Memon, N., 2014. A robust model for paper reviewer assignment. In: Proceedings of the 8th ACM Conference on Recommender Systems. pp. 25–32.

Mimno, D., McCallum, A., 2007. Expertise modeling for matching papers with reviewers. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 500–509.

Mousavi, N., Adamopoulos, P., Bockstedt, J., 2023. The decoy effect and recommendation systems. Inf. Syst. Res. 34 (4), 1533–1553. http://dx.doi.org/10.1287/isre.2022.1197.

Park, J., Lee, B.K., 2021. An opinion-driven decision-support framework for benchmarking hotel service. Omega 103, 102415. http://dx.doi.org/10.1016/j.omega.2021.102415.

Ponce, J., Barbier, A., Palau, C.E., Guardiola, C., 2025. A novel approach in constructing virtual real driving emission trips through genetic algorithm optimization. Eng. Appl. Artif. Intell. 139, 109637. http://dx.doi.org/10.1016/j.engappai.2024.109637.

Siering, M., Muntermann, J., Rajagopalan, B., 2018. Explaining and predicting online review helpfulness: The role of content and reviewer-related signals. Decis. Support Syst. 108, 1–12. http://dx.doi.org/10.1016/j.dss.2018.01.004.

Su, P., Luo, B.X., Deng, F.Y., Xia, A.X., Guo, Y., 2020. Group strategy of dissertation defense based on greedy retrospective hybrid algorithm. In: Journal of Physics: Conference Series. Vol. 1634, IOP Publishing, 012077.

Tan, S., Duan, Z., Zhao, S., Chen, J., Zhang, Y., 2021. Improved reviewer assignment based on both word and semantic features. Inf. Retr. J. 24, 175–204.

Zhou, Y., Lee, G.M., 2020. A bi-objective medical relief shelter location problem considering coverage ratios. Int. J. Ind. Eng.: Theory Appl. Pr. 27, 971–988. http://dx.doi.org/10.23055/ijietap.2020.27.6.7603.