University of Tehran Press

Industrial Management Journal

Print ISSN: 2008-5885
Online ISSN: 2423-5369

Homepage: https://imj.ut.ac.ir

A Mathematical Model for Reviewer Assignment Problem: Balancing Maximum Coverage, Fairness, and Expertise Matching

Jalil Heidary Dahooie ✉ [1], Raheleh Fathollahi [2], and Mohammad Gholami[3]

1. Corresponding author, Associate Prof., Faculty of Industrial Management & Technology, College of Management, University of Tehran, Tehran, Iran. E-mail: Heidaryd@ut.ac.ir
2. MSc., Faculty of Industrial Management & Technology, College of Management, University of Tehran, Tehran, Iran. E-mail: raheleh.fathollahi@gmail.com
3. MSc., Faculty of Industrial Management & Technology, College of Management, University of Tehran, Tehran, Iran. E-mail: mohammadgholami8@ut.ac.ir

Article Info

Article type:
Research Article

Article history:
Received November 06, 2024
Received in revised form June 08, 2025
Accepted June 09, 2025
Available online June 10, 2025

Keywords:
reviewer assignment, maximum coverage, fairness, expertise matching.

ABSTRACT

Objective: This study tackles the reviewer assignment problem by proposing a model that optimizes reviewer-proposal matching based on thematic coverage, fairness, and expertise, while considering workload balance and team size constraints. The model incorporates practical constraints such as limits on the number of proposals each reviewer can handle and team composition requirements. This approach is especially relevant to institutions like academic conferences, journals, and funding organizations, aiming to enhance the integrity and efficiency of the review process.

Methods: This study is classified as descriptive research with a practical orientation and relies on data collection through applied methods. The approach is grounded in mathematical modeling. Initially, the selected articles are grouped into clusters. Reviewers are then assigned to these clusters using a multi-objective binary integer programming model that incorporates all relevant criteria and constraints. To implement this model, 150 articles were selected through purposive sampling. The model was optimized using Python, employing both the branch-and-bound algorithm and a genetic metaheuristic algorithm to maximize the degree of reviewer-proposal matching within the proposed framework.

Results: The proposed model demonstrates strong practical relevance by closely reflecting real-world reviewer assignment challenges. By simultaneously optimizing thematic coverage, evaluation fairness, and reviewer expertise, the model captures the complexity of actual allocation scenarios. To validate its effectiveness, the model was solved using both the branch-and-bound algorithm and a genetic algorithm. The branch-and-bound method yielded an objective value of 177.349 in approximately one hour, while the genetic algorithm reached 120.35 in just seven minutes. Although branch-and-bound guarantees optimality, its longer runtime makes it less practical for larger datasets. Given the similarity of results, the genetic approach is a reliable and scalable alternative.

Conclusion: This study introduces a new allocation strategy and mathematical model for reviewer assignment, addressing often-overlooked factors such as reviewer expertise, grouping, and conflicts of interest. By integrating these elements, the proposed model better reflects real-world conditions. Future work is encouraged to expand on these findings with new frameworks and methods.

Cite this article: Heidary Dahooie, J., Fathollahi, R., & Gholami, M. (2025). A mathematical model for reviewer assignment problem: Balancing maximum coverage, fairness, and expertise matching. ***Industrial Management Journal***, *17*(2), 149-174. https://doi.org/10.22059/imj.2025.384968.1008200

Publisher: University of Tehran Press.

DOI: https://doi.org/10.22059/imj.2025.384968.1008200

Introduction

The fourth industrial revolution has triggered an unprecedented surge in global information and research activity, fueling the rapid expansion of the scientific community. As the volume of scientific publications grows, assigning qualified reviewers has become increasingly complex and has drawn significant academic attention (Hoang et al., 2021). Since article quality is directly influenced by reviewers' evaluations, peer review is now recognized as a critical step in the publication process. As a result, the Reviewer Assignment Problem (RAP) plays a vital role in identifying impactful research and enhancing the overall efficiency of academic systems. This growing importance has prompted extensive research into developing more effective reviewer assignment strategies (Aksoy et al., 2023; Ribeiro et al., 2023).

The review process consists of several steps, with the assignment of reviewers to submitted articles being the first and most crucial. It is evident that the method of this assignment significantly impacts the quality of the review results. Even a small number of inaccurate reviews can adversely affect the quality of published scientific standards and harm researchers' careers (Stelmakh et al., 2023).

Traditionally, journal or conference editors manually assigned reviewers to articles. However, as the number of submissions has increased and research fields have become more specialized, this task has become challenging, if not impossible, to manage manually. Consequently, the use of operations research techniques has gained popularity among researchers in this area (Cook et al., 2005).

From the perspective of Reviewer Assignment Problems (RAP), algorithms used for assigning reviewers can be broken down into three stages: building a candidate reviewer database, assessing the matching degree between each article-reviewer pair, and creating a reviewer assignment scheme based on the matching degree matrix (Zhao & Zhang, 2022).

The present paper focuses on the third stage, where the reviewer assignment scheme is developed based on the matching degree along with other constraints in various review scenarios, such as balancing the workload among reviewers and ensuring adequate article coverage. The RAP must be modeled considering different constraints and solved using various assignment optimization algorithms.

To effectively address these challenges, it is essential first to identify the criteria for selecting the members of the judging team. These criteria can be expressed as an objective function in mathematical modeling, with the limitations of the research being considered. Reviewer assignment represents an advanced subset of the assignment problem. The objective when assigning reviewers to an article is to form a k-member team capable of evaluating the article while meeting all specified criteria (Kolasa & Krol, 2011).One approach to solving assignment problems involves using integer linear programming (0-1 programming), which has attracted considerable interest from researchers in this field. While precise techniques like branch and bound can be utilized to optimize this model, the problem becomes increasingly complex and classified as NP-hard when the number of criteria for selecting the judging team, as well as the number of articles, reviewers, and constraints, grows. The time needed to reach a solution increases proportionally with the number of variables and constraints (Monteiro, 1997; Wang et al., 2013). Consequently, many researchers have opted for meta-heuristic algorithms, including genetic algorithms, tabu search, ant colony optimization, greedy algorithms, and simulated annealing, to efficiently find solutions and save time.

An examination of prior research reveals that issues such as thematic coverage of articles, work balance, and the number of group members have been taken into account by researchers. However, as time has progressed, new criteria and limitations have emerged to ensure better alignment with reality. Specifically, the grouping methods, reviewers’ skill levels, and potential conflicts of interest have not received adequate focus from researchers, even though they are important for the review assignment process. Criteria for selecting reviewers, the absence of conflicts of interest, and their expertise are vital for assessing scientific work. Overlooking these factors may result in the exclusion of many qualified reviewers, especially in emerging fields (Xu et al., 2010).This study examines the benefits of a grouping approach for assigning reviewers to articles. It considers several criteria for reviewer assignments, including maximizing subject coverage, ensuring fair evaluations, and matching reviewers' expertise to the topics at hand. Additionally, the model aims to reflect real-world conditions by factoring in workload balance and the number of group members. The research is structured into two primary phases. The first phase involves clustering the articles, while the second phase formulates the problem as a multi-objective integer programming model (utilizing binary variables) that addresses various criteria and applicable constraints. We conclude by presenting the results through a case study.

The structure of this study is organized as follows. Section 2 offers a concise review of the literature on grouping articles and outlines the criteria and limitations for selecting judging team members. It also provides the research background and highlights the gap our study aims to address. Section 3 presents an overview of the research framework, which utilizes the Dijkstra algorithm to develop a mathematical model, alongside implementation steps, branch and bound algorithms, and genetic meta-heuristics to solve the model. The research findings are detailed in Section 4. Finally, Section 5 analyzes and discusses the results of the research findings and draws conclusions.

Literature Background

Selecting the right team members is a key factor in the success of team projects (Wi et al., 2012). The aim of addressing member selection issues is to form a team of experts that not only possesses all the necessary skills for the project but also minimizes communication costs (Lappas et al., 2009). Conference organizers, journal editors, and research funding organizations often face the challenge of forming a judging team (Wang et al., 2013; Neshati et al., 2014). The primary applications of RAPs involve assigning reviewers for conference papers or major journals, as well as evaluating research project proposals. Although RAP is a relatively small research area, it has garnered significant attention from many researchers (Luo et al., 2024).

As previously mentioned, operations research techniques are particularly effective in addressing large-scale and repetitive problems, which has led to a growing interest in this approach in recent studies (Simon & Newell, 1958). Typically, the process of selecting a reviewer involves three main steps: grouping articles, establishing criteria and constraints for selecting members of the judging team, and ultimately solving the model to identify the optimal solution. This section will critically review research conducted on these three steps to highlight the existing research gap and illustrate the innovations presented in the current article.

Grouping of articles

In the initial step of addressing the reviewer assignment issue, researchers have identified two main strategies for evaluating articles. The first strategy, which encompasses a significant portion of the research, involves assigning articles individually (Karimzadehgan et al., 2008; Kolasa & Krol, 2011; Neshati et al., 2014). This means that each article is reviewed separately and assigned to relevant reviewers. However, this approach can be time-consuming, especially when there is a large number of articles and eligible reviewers, and it may sometimes be impractical due to tight deadlines (Das & Göçken, 2014).

To mitigate this problem, researchers have considered a second strategy that involves grouping articles and then assigning them to relevant reviewers. This approach not only saves time but also helps balance the workload among reviewers. Wang et al. (2013) examined the assignment of reviewers to articles through a group-to-group assignment model. They formulated the grouping and assignment problem using multi-objective integer programming and solved it with a stochastic-biased greedy algorithm. Their study utilized real data from several well-known business schools in China. Additionally, Fan et al. (2009) developed a binary model for grouping national foundation proposals, which they solved using a genetic algorithm. Xu et al. (2010) applied this grouping method and made assignments based on the degree of agreement between the reviewers and the groups of articles. The information provided indicates that utilizing a grouping approach for articles can enhance the efficiency of the review process. Research in this area has explored grouping using two different methodologies. The first group represented the grouping and assignment as a multi-objective model (Wang et al., 2013), while the second group approached the task in two distinct phases to simplify the problem: the first phase involves grouping the articles, followed by the second phase, which assigns groups of articles to reviewers (Fan et al., 2009).

Determining criteria and restrictions for selecting members of the judging team

In the second step, it's crucial to establish suitable criteria for selecting reviewers. Before defining these criteria, it’s important to recognize that they can be assessed in two ways. First, there is the explicit method, which involves directly asking both the reviewer and the article's author about the reviewer’s research field, the article's subject area, and the reviewer’s interests and skills. This approach is commonly used for manual appointments. Second, criteria can also be assessed implicitly by examining keywords in the submitted article alongside the reviewer’s research background. This method uncovers underlying aspects of the article's topic and the reviewer’s expertise.

The topic of reviewer assignment has been examined from various perspectives, with researchers targeting different criteria and limitations. One key criterion is article coverage, which evaluates how closely a reviewer’s expertise aligns with the article's content (Kalmukov, 2012; Li et al., 2017). This alignment is influenced by the reviewer’s knowledge in the specific subject area, determined by factors such as the novelty, quantity, and quality of their published work (Lee & Watanabe, 2013). Additionally, minimizing social connections between reviewers and authors is crucial to avoid potential conflicts of interest (Mimno & McCallum, 2007; Long et al., 2013).

These criteria are often represented as objective functions in quantitative models. Notably, recent research has sought to model these criteria in ways that better reflect the complexities of real-world scenarios. Traditionally, the focus has primarily been on thematic coverage of papers, frequently neglecting other significant criteria. As new researchers enter the field, a variety of criteria have been suggested; however, few studies have successfully integrated all of them into their reviewer assignment strategies.

Researchers have also examined various limitations, including the distribution of workload among reviewers (Karimzadehgan & Zhai, 2012; Neshati et al., 2014), the number of reviewers required for each article (Long et al., 2013; Li et al., 2017), and budgetary constraints (Das & Göçken, 2014). For example, Li et al. (2017) introduced an automated system for assigning reviewers to articles with the goal of maximizing thematic coverage. They evaluated thematic alignment by checking for common references between an article and a reviewer’s past publications, indicating a thematic connection.

Moreover, they underscored that a reviewer's expertise in a subject area is based on three factors: the quantity, quality, and novelty of their publications. This approach was tested using articles from the computer science field. In another study, Karimzadehgan et al. (2012) focused on the extent of subject coverage by reviewers and suggested employing probabilistic topic modeling methods to identify hidden topics in articles, thereby determining the necessary skill set required from the text. They utilized probabilistic latent semantic analysis to uncover these hidden areas and addressed the allocation problem by using integer linear programming and a greedy algorithm to find solutions.

Furthermore, Karimzadehgan et al. (2008) incorporated a maximum matching criterion and a confidence coefficient criterion that reflects the subject match between reviewers. They defined their performance function based on the harmonic mean of these two criteria and optimized their model using data from an ACM conference, applying a greedy algorithm in the process.

Solving the Optimization Model

The concluding step in addressing the reviewer assignment problem involves solving the developed mathematical model to determine the optimal solution. This issue can be framed as a binary integer programming problem (Karimzadehgan & Zhaei, 2012). Although some research has attempted to tackle the problem in smaller cases using exact algorithms, such as branch and bound (Monteiro, 1997; Zhou et al., 2012) and assignment algorithms (Li & Watanabe, 2013), it becomes an NP-Hard problem when applied to larger, real-world scenarios (Wang et al., 2013). Consequently, approximate algorithms, including heuristic and meta-heuristic methods, are frequently employed to derive solutions.

For instance, Karimzadehgan et al. (2012) analyzed the ACM conference dataset, modeling the reviewer assignment problem by factoring in coverage and confidence criteria through integer programming and utilizing a greedy algorithm for resolution. Kolasa et al. (2011) examined a conference management system and introduced a hybrid ant-colony genetic algorithm (ACO-GA) for reviewer assignments. Meanwhile, Li et al. (2017) investigated papers accepted at SIGMOD, implementing two algorithms—the refrigeration simulation algorithm and the maximum fit with minimum deviation algorithm—and comparing their results. Schirrer et al. (2007) focused on meta-heuristics in the context of international conferences, employing a memetic algorithm for assigning reviewers to papers. To highlight the existing research gap, the reviewed articles are summarized in Table 1.

Table 1. Summary of Judged articles in the field of arbitration allocation

Reference	Article assignment approach		Criteria			Limitations			Method
	Individual	Group	Coverage	Skill	Fairness	Budget	Work balance	Number of team members	
Leyton-Brown et al. (2024)	×		×		×		×	×	Large Conference Matching (LCM)
Bouanane et al. (2024)	×		×		×	×	×		Fairflow and FairIR algorithms
Carpenter et al. (2024)		×	×			-	-	-	Deploy and evaluate machine learning and optimization techniques
Rordorf et al. (2023)	×		×	×		-	-	-	Natural Language Processing (NLP) and Knowledge Engineering
Stelmakh et al. (2023)	×		×	×				×	Similarity - Calculation Algorithms
Hoang et al. (2021)		×	×				×	×	Decision support system
Li et al. (2017)	×		×		×		×	×	Simulated refrigeration

The table above illustrates that reviewer assignment has become an increasingly important topic for researchers in the field of operations research in recent years. Early studies mainly focused on criteria such as coverage, work balance, and the number of group members. However, as research has progressed, this approach has expanded to include new criteria and limitations that more accurately reflect real-world situations. Among these factors, the grouping approach, reviewers' expertise, and potential conflicts of interest are crucial elements that have been surprisingly overlooked in the literature. In this article, we explore the advantages of grouping by first categorizing the articles and then addressing all relevant assignment criteria in the next step. Our approach is guided by three key criteria: maximum subject coverage, fairness in judgment, and reviewers' expertise in the relevant areas. Additionally, to ensure that our model is closely aligned with practical realities, we also incorporate limitations related to work balance and the number of group members.

Materials and Methods

The focus of this study is on articles published in the Expert Systems Journal that are indexed in the Web of Science database. We implemented a targeted search strategy to gather articles from the years 1979 to 2023 that are relevant to the aims of our research. In total, the study encompasses 12,178 scientific articles from this database.

For our sampling approach, we employed purposive sampling, which entails selecting units based on specific characteristics related to the phenomenon being studied, rather than using random selection (Delavar, 2008). To identify the reviewers and authors, we constructed a collaboration network from the articles published in the Expert Systems Journal. We then chose individuals from this network who exhibited high centrality indices, as they are likely to possess a more extensive research background than their peers.

Centrality is a critical concept in social network analysis that evaluates the significance and influence of members within a network. We can assess the centrality of network nodes using four key indices: degree, closeness, betweenness, and eigenvector. In this paper, we apply Dijkstra's algorithm to evaluate the similarity among articles and utilize agglomerative hierarchical clustering for grouping these articles. After outlining the mathematical model, we first address it using the branch and bound algorithm to propose a judging team. Subsequently, we introduce a meta-heuristic model designed to tackle larger dimensional problems. Each of these algorithms will be explained in detail below.

Article clustering

To streamline operations, it's important to cluster articles based on their similarities before modeling this problem. The following methods are frequently used to assess semantic similarity, or distance, between keywords:

1. Based on a vocabulary network or other dictionaries.
2. Syntactic dependency relationships between the phrases and the words in the dictionaries.
3. Co-occurrence of the phrase with the words in the primary list across various document corpora.

This study examines a method for calculating the similarity between articles based on a network of keywords. First, we create this network, and then we determine the similarity or distance between the keywords using various criteria for measuring word similarity. One such criterion is the shortest distance between words, indicating that the greater the semantic relationships between words, the shorter the shortest path will be (Kamps et al., 2004). Keywords for this analysis were initially gathered from the Web of Science database and then visualized using VOSviewer software. To measure the distance between these keywords, we employed

Dijkstra's algorithm, which finds the least-cost path (in terms of length) from a starting node to every other node in a graph. Below is a summary of how the algorithm operates.

1. Assign an infinite value as the initial length to all nodes, except for the origin point, which takes a value of zero.

2. Create a vector with a length equal to the total number of nodes to store the previous node for each node. Set the value to null for all nodes, except for the origin point, which remains empty since it has no previous node.

3. Initially, place all points in the set of unvisited nodes.

4. Perform the relaxation process for all nodes that share an edge with the current node.

5. Once the nearest node to the current node is determined, remove the current node from the set of unvisited nodes.

6. If all nodes have been visited, the algorithm terminates. Otherwise, from the unvisited nodes, select the node with the shortest length as the next point, and then return to step 4.

After calculating the shortest distances between the keywords of the articles, we use this distance matrix as the foundation for clustering. In hierarchical clustering, the similarity or distance matrix between items is essential for the process, while other methods require a proximity matrix of items and features.

In this study, we employ the agglomerative hierarchical method to cluster the articles. This method treats each article as an individual cluster, which are then merged together to form a tree structure. For binary clustering, it is crucial to identify the best clusters to merge, which can be achieved by calculating the similarity at each step or by retaining the similarities from previous stages.

Typically, in hierarchical clustering, a stopping criterion is established to prevent the algorithm from running indefinitely, resulting in a defined number of clusters. In our case, the stopping criterion is based on the distance between the cluster centers. We set a threshold of 3 for clustering the articles, which resulted in a total of 22 clusters. The code for this algorithm was implemented in Python.

Identifying the criteria and limitations of the judging team

Once the articles have been clustered, a model will be established to identify the suitable judging team for each cluster. This model will focus on three primary criteria, which serve as the objective function.

Thematic coverage

Thematic coverage refers to the comprehensive evaluation of all thematic aspects of an article by the reviewers. To ensure that the reviewers' research background aligns closely with that of the submitted article, we follow a clustering approach to measure similarity. First, we gather keywords from both the reviewers' backgrounds and the submitted articles. We then create a network based on these keywords. Using Dijkstra's algorithm, we calculate the distances between these keywords. The average of the shortest paths between the keywords of the articles and the reviewers' backgrounds is used as the criterion for thematic coverage.

Fairness of judgment

It describes a scenario where individuals in trusted positions may have personal or group interests that conflict with their official responsibilities. In situations where a reviewer has a social relationship with the author of an article, there is a risk that the reviewer's assessment might be swayed by personal biases. To mitigate conflicts of interest, it is essential to analyze the cooperation network and identify the longest shortest path between the reviewer and the author (Wang et al., 2013). For example, Figure 1 displays a social network. When assigning one of three reviewers to the first cluster based on this network, we should utilize the concept of the shortest path. This means choosing the reviewer who is the farthest removed from the article's authors. To facilitate this process, we gathered the research backgrounds of both reviewers and authors from the Web of Science database and constructed their collaboration network with VOSviewer software. Using Dijkstra's algorithm, we then identified the shortest paths between nodes in this network. Ultimately, we calculate the average of these shortest paths between the reviewers and the set of authors within each cluster, which acts as the second criterion for reviewer selection.

The skill of the reviewers

The skill of reviewers should be assessed based on their expertise in the subject areas of the clusters. Therefore, it is essential to identify the subject area of each cluster prior to calculating this criterion. After clustering the articles, we create a keyword network for each cluster, selecting words with high centrality indices as the subject area for that cluster. The reviewer’s proficiency in these specific subject areas is then evaluated. The skill of the reviewers is broken down into two sub-criteria: quality and novelty, which will be explained further below. To assess the quality of an article, we assume that higher citation counts indicate higher quality. To determine the quality of a reviewer’s contributions in a certain field, we consider the average citations of their published articles within that subject area. Additionally, more recent research reflects a reviewer’s updated knowledge in the field. To calculate the novelty of an article, we define it as the difference between the average publication year of articles in that field and the year of the conference, all relative to a base year. The final reviewer skill score in a particular field is derived from the product of the quality and novelty factors. It is important to note that quality is a positive indicator, where a higher value is more desirable, and a shorter time interval in the novelty index is also more favorable. Therefore, we define the novelty index as $\exp^{(\frac{-(\mathrm{mean(t)}-\mathrm{T})}{5})}$ (Li & Watanabe, 2013).

Finally, the skill index is defined as equation 1:

$$\mathrm{skill} = \mathrm{mean}(c_j) * \exp^{(\frac{-(\mathrm{mean(t)}-\mathrm{T})}{5})} \quad (1)$$

Where mean (cj) is the average number of citations for the reviewer's papers within the specific subject area of the cluster, mean (t) denotes the average publication year of the papers reviewed in the same area. The parameter T refers to the year of the conference, which is 2024 in this case. To assess the reviewer's expertise in the cluster of papers, we calculated the average skill level across the subject areas covered by those papers.

Limitations

This model incorporates two types of constraints. The first type is related to the distribution of workload among the judges. The workload balance constraint is a significant factor in group formation issues, as it ensures that tasks are allocated evenly among all group members. To facilitate this, we can set a minimum and maximum number of articles that each reviewer can assess (Charlin et al., 2011; Neshati et al., 2014). The second type of constraint addresses the number of reviewers needed for each cluster. Given that the articles are categorized into clusters, the required number of reviewers for each cluster is proportional to the number of articles it contains.

General framework of the model

To create a judging team, reviewers must be assigned to clusters of articles according to the criteria specified earlier. This assignment problem can be framed as an integer programming problem using binary variables (0 and 1). Prior to discussing the model, we will define the variables, parameters, and model sets as detailed in Table 2.

Table 2. Symbology of the model

symbol	Description
I	A collection of articles
J	Reviewer's Collection
K	Collection of articles
B	Maximum number of reviewer's required for each cluster
A	Maximum number of referees required for each Minimum number of reviewer's required for each cluster
$Y_{k,j}$	Variable related to the assignment of the j judge to the k cluster
$S_{k,j}$	Topic similarity between reviewer j and cluster k
$I_{k,j}$	Interest of reviewer j in the subject area of cluster k
$D_{k,j}$	Average shortest path between reviewer j and authors cluster k
w_1	The weight of the first objective function
w_2	The weight of the second objective function
w_3	The weight of the third objective function
M	Number of articles
N	Number of reviewer's
G	Number of clusters
C	The minimum number of clusters to which the reviewer should be assigned
D	Minimum cluster distance to which the reviewer should be assigned
Z_1	The minimum cluster distance to which the first objective function reviewer is assigned is related to the subject matching of reviewer's and article clusters
Z_2	The second objective function is related to the fairness of allocation
Z_3	The third objective function is related to the reviewer's skill in the cluster's subject area

As mentioned, this model has three criteria for allocation, which are presented as an objective function in the modeling process discussed below.

$$\text{Min } z_1 = \sum_{k=1}^{g}\sum_{j=1}^{n} S_{k,j} * Y_{k,j} \qquad (2)$$

$$\text{MAX } z_2 = \sum_{k=1}^{g}\sum_{j=1}^{n} I_{k,j} * Y_{k,j} \qquad (3)$$

$$\text{MAX } z_3 = \sum_{k=1}^{g}\sum_{j=1}^{n} D_{k,j} * Y_{k,j} \qquad (4)$$

Several solutions have been proposed to address multi-objective modeling problems. One such method involves summing the objective functions to transform the multi-objective model into a single-objective model, which is the approach used in this modeling. To combine the first objective function with other criteria, its values have been inverted and rephrased as a maximization objective function.

$$\text{MAX } z_1 = \sum_{k=1}^{g}\sum_{j=1}^{n} \frac{1}{S_{k,j}} * Y_{k,j} \qquad (5)$$

The general framework of the model is as follows:

$$\text{MAX } \; w_1 z_1 + w_2 z_2 + w_3 z_3 \qquad (6)$$

Subject to:

$$a \le \sum_{j=1}^{n} Y_{k,j} \le b \tag{7}$$

$$c \le \sum_{k=1}^{g} Y_{k,j} \le d \tag{8}$$

$$Y_{k,j}=0,1 \tag{9}$$

$Y_{k,j}$ is equal to 1 if the reviewer j is assigned to the k group; otherwise, 0. Equation (6) shows the objective function that combines the three criteria of subject coverage, fairness, and reviewer interest with coefficients w1, w2, and w3. Equation (7) shows the number of reviewer's required for each cluster. Equation (8) shows the minimum and maximum number of clusters to which each reviewer can be assigned. Equation (9) expresses the binary nature of the decision variable.

Model solution

Once the model is developed, we address this problem using both the branch and bound algorithm and the genetic algorithm, which will be outlined below.

Branch and bound algorithm

The branch and bound algorithm is a general approach used to solve various optimization problems, particularly in combinatorial optimization. It was first introduced by Land et al. in 1960 to address discrete optimization challenges. This method explores the state space of potential solutions by representing the set of possible answers as a tree. The root of this tree corresponds to all possible solutions, while its branches represent subsets of those solutions. Before traversing the solution set of a specific sub-branch, the algorithm evaluates the branch against both lower and upper bounds relevant to the overall optimization problem. If a sub-branch is determined to be incapable of yielding a more optimal solution, the algorithm will avoid exploring that entire sub-branch. To develop an algorithm that minimizes the function f, we can use the function g as a lower bound for the value of f at the vertices of a subtree within the state space. It's important to note that by identifying the maximum value of g, we can determine the minimum value of f . The general structure of this method will be as follows:

1. First, we find an arbitrary solution x and set the value of B equal to f(x). From now on, the value of B will represent the best solution found up to this point of the search.

2. We consider a queue of state space vertices and add the root of the state space tree to it.

3. Repeat the following steps until the queue is empty.

4. Remove a vertex from the queue.

5. If this vertex represents a specific solution to the problem, say x, and f(x) < B, this solution is the best solution found so far; as a result, we place the value of f(x) in B.

6. Otherwise, for all branches, we call this vertex, for example, Ni.

7. If: g(Ni)< <B .

8. This branch may lead to a more optimal solution, so we add Ni to the list.

9. Otherwise, this branch would have no value, because the lower bound of its solutions is greater than the upper bound of the solution to the problem.

10. Return to command 3.

The recursive function terminates under two conditions: when the current proposed solution set S is narrowed down to a single element, or when the upper bound of the set S matches the lower bound. In both scenarios, each element in S represents the minimum value of the function within that set.Genetic algorithm

A genetic algorithm is a machine learning model inspired by the mechanisms of evolution found in nature. This method involves creating a population of individuals, each represented by chromosomes (Daraei, 2022). Essentially, the genetic algorithm serves as a random search algorithm, drawing on concepts from nature. In biological evolution, improved generations arise from the combination of superior chromosomes. Occasionally, mutations occur in these chromosomes, which can lead to enhancements in the subsequent generation. Genetic algorithms apply this concept to problem-solving. The process of utilizing genetic algorithms is as follows:

1. Introducing the problem solutions as chromosomes.

2. Introduction to the fitness function.

3. Gathering the initial population.

4. Introducing selection operators.

5. Introduction to reproduction operators (crossover).

In genetic algorithms, we start by generating a set of solutions to the problem, either randomly or using a specific algorithm. This set of solutions is referred to as the initial population, and each individual solution is called a chromosome. Using genetic algorithm operators, we then select the better-performing chromosomes, combine them, and introduce mutations. Subsequently, we merge the current population with a new population created through recombination and mutation of the chromosomes (Immanuel & Chakraborty, 2019). The problem we aim to solve is transformed into solutions through a process that mimics genetic evolution. Each solution is assessed as a candidate via a fitness function, and the algorithm terminates when the exit condition of the problem is met. In each generation, we select the fittest individuals rather than the absolute best. Each solution is represented as a list of parameters known as a chromosome or genome. While chromosomes are often represented as simple strings of data, other data structures can also be utilized. Initially, a set of traits is randomly generated to form the first generation. During each generation, each trait is evaluated, and its fitness value is calculated using the fitness function. The next step involves creating the second generation of the population based on the selection processes and genetic operators. This includes joining chromosomes and introducing changes. For each individual, a pair of parents is chosen, with a selection method that ensures even the weaker members have a chance to be selected, thereby avoiding the risk of converging on a local solution. Several selection patterns can be used, such as roulette wheel selection or competitive selection. Genetic algorithms typically implement a linkage probability, ranging from 0.6 to 1, indicating the likelihood of producing offspring. This probability determines how organisms recombine. The union of two chromosomes produces offspring that are added to the next generation, continuing until suitable candidates for the solution are identified. The next step is to mutate the new offspring. Genetic algorithms employ a small, fixed mutation probability, usually around 0.01 or lower. Based on this probability, the offspring chromosomes are randomly altered or mutated. This mutation process generates a new generation of chromosomes that differ from the previous one. The entire cycle is repeated: pairs are selected for mating, a third generation is created, and this process continues until we reach the final stage.

Research framework

Figure 1 illustrates the overall framework of the research. This study employs two integer programming algorithms to address the reviewer assignment problem, which can be defined and solved in ten steps. The first three steps involve selecting and displaying the relevant indicators and constraints of the problem. The next three steps focus on determining the weights for the defined features and constraints, followed by the presentation of the mathematical model. The subsequent two steps are concerned with solving the model using branch-and-bound algorithms and genetic metaheuristics. In the final two steps, the results obtained are compared, leading to a discussion and conclusion. The solution methods are clearly outlined and evaluated step by step in the research findings section.

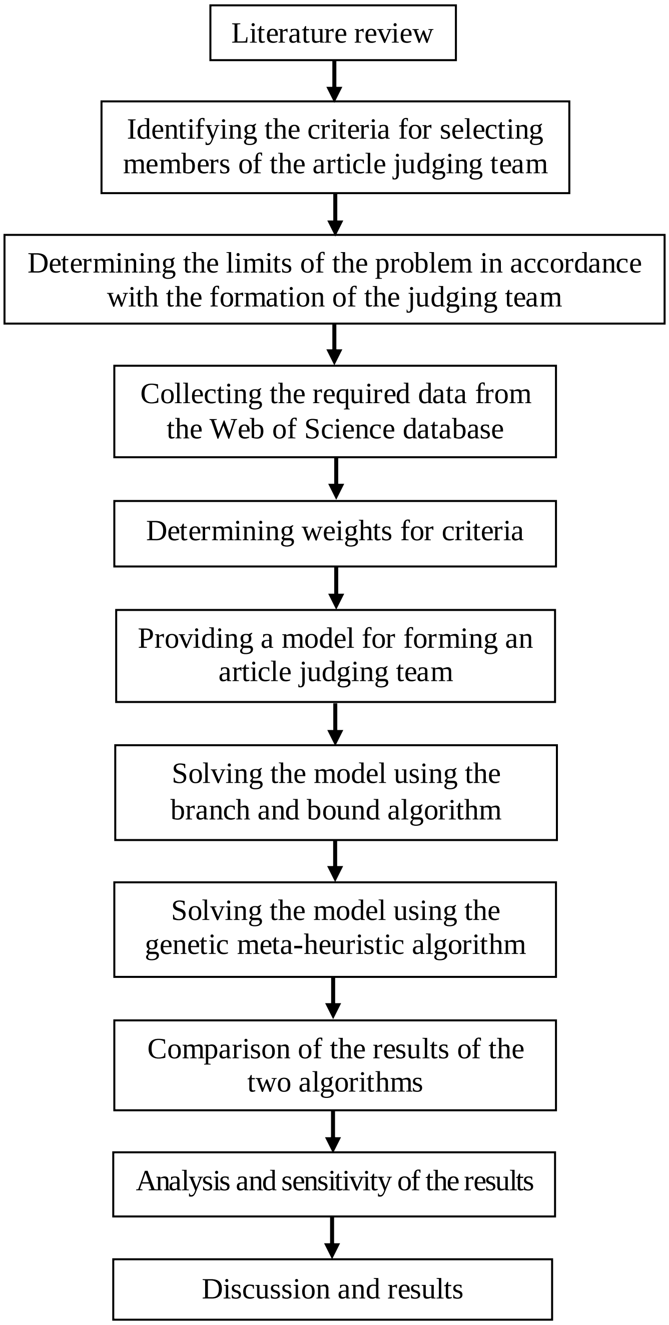
Literature review
Identifying the criteria for selecting members of the article judging team
Determining the limits of the problem in accordance with the formation of the judging team
Collecting the required data from the Web of Science database
Determining weights for criteria
Providing a model for forming an article judging team
Solving the model using the branch and bound algorithm
Solving the model using the genetic meta-heuristic algorithm
Comparison of the results of the two algorithms
Analysis and sensitivity of the results
Discussion and results

Figure 1. Research framework

Results

The data for this study was obtained from the Web of Science database. The focus of the research is on articles published in the journal *Expert Systems with Applications*. To compile this data, the journal name "Expert Systems with Applications" was searched, yielding around 12,000 articles published from 1979 to 2023. From this collection, 200 authors were selected through purposive sampling, with 150 identified as contributors of submitted articles and 50 as reviewers. To determine the authors and reviewers, a collaboration network of the published articles was created, allowing for the identification of authors with higher centrality indices. Following the selection process, keyword and collaboration networks were constructed. To create these networks, the publication history of the authors and reviewers was essential. Data on their research backgrounds was gathered from the Web of Science database. By entering the journal name "Expert Systems with Applications" along with the specified time range of 1979 to 2023, information on the reviewers and authors was obtained. One article was randomly chosen from each author's list of publications to serve as the submission for the journal or conference. This information was documented in text format, encompassing the title, year, authors, keywords, abstract, sources, and citations, for further analysis in subsequent phases. The VOSviewer software was then employed to generate thematic maps and analyze the social network. VOSviewer is recognized as one of the most dependable tools for visualizing and analyzing networks, including co-authorship and co-occurrence networks in the field of scientometrics. Figure 2 displays the co-occurrence network of keywords, and the output from VOSviewer was saved for future reference in utilizing this network.

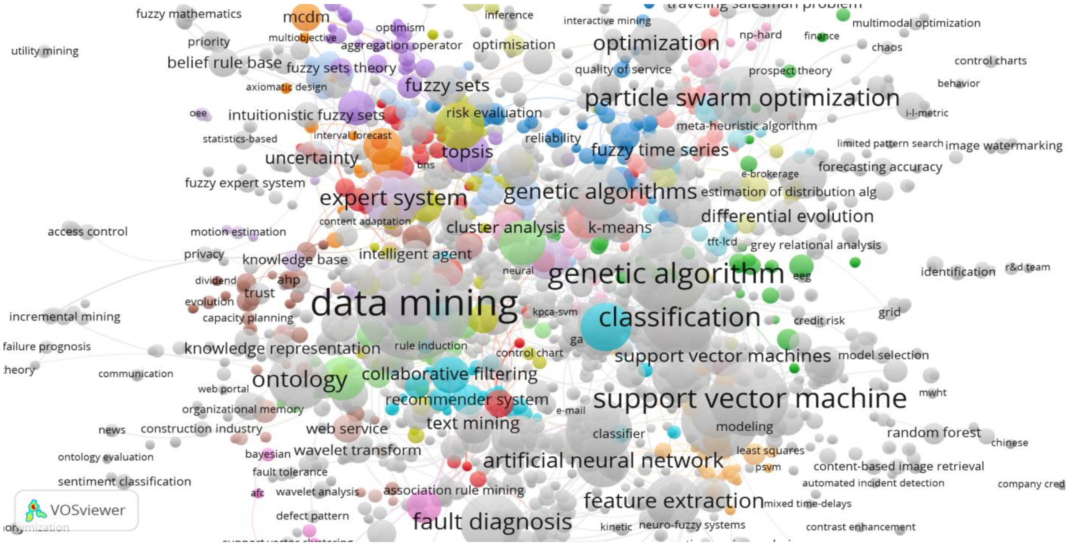

Figure 2. Keyword co-occurrence network

This network consists of 3,435 nodes and 11841 edges. Each node represents a keyword, while the edges demonstrate the co-occurrence of these keywords. The weight of the edges indicates the number of co-occurrences.

Solving the model using the branch and bound algorithm

The judging team is organized using a binary model, which falls under the category of integer programming. This makes the branch and bound algorithm an effective solution for the problem. We have implemented this algorithm using Python software. Initially, we build the root nodes. For each node in the tree, we determine the upper bound of the objective function by solving an integer linear programming problem. We then identify the variable with the largest decimal part (for example, yi) to impose a new constraint on the problem. We add a lower bound constraint at one node, leading to a new problem to solve, while at another node, we introduce an upper bound constraint, creating an additional problem. Both of these new problems are tackled using integer linear programming, and we calculate the upper bound of the objective function for each case. If we find a problem to be infeasible or if the upper bound of the objective function is lower than the best value discovered thus far, we will not expand the tree at that node. Conversely, if it is feasible, we queue the new node for further exploration. This process continues until we have examined all nodes. The results from this algorithm are shown in Table 3.

Table 3. Results ofthe branch and bound algorithm

Cluster	Reviewer's
0	chin, ks ; lee, s ; wu, x ; xu, g ; yang, jb
1	chen, m; chin, ks; lee, s, pedrycz, w; tang, y; wu, w; yang, jb
2	chin, ks; lee, s; li, g; liang, l; song, j; xu, w; yu, s
3	huang, l; jin, z; lee, j; tang, s
4	chen, l; chen, m; gao, x; liao, s; ma, h; wu, q; xie, x; zhao, f; zhou, f; zhou, x
5	chen, l; he, c; li, g; sun, y
6	feng, b; gao, x; wu, w; yang, f; zheng, j
7	chen, m; yang, x
8	chin, ks; lee, s; tang, y; xu, q; yang, l; zheng, j
9	lu, l; wang, f; wu, l
10	wu, x; xu, q
11	li, g
12	huang, l, zheng, j
13	li, g
14	yang, x
15	feng, s
16	lee, j; li, m
17	chen, l; liu, w; yu, c
18	chen, g; lee, s; xie, x; zhou, z
19	lee, j
20	gao, x; tang, j
21	kim, w; zhang, k
22	chen, m; yu, j

The algorithm generates an integrated objective function value of 177/349, and it takes around one hour and ten minutes to solve this problem using the branch-and-bound method. For a more in-depth analysis, the three previously mentioned objective functions were addressed separately. The first objective function model requires about one hour to solve, resulting in a value of 27/322610049528. Unfortunately, the branch-and-bound algorithm did not reach optimal solutions for the second and third objective functions. For the second criterion, the initial integer solution provided is 217/3240764304, which, after roughly seven hours, is revised to 217/32564398632. Regarding the third criterion, the final integer solution suggested after six hours of processing is 345/76554098.

Solving the model using the genetic meta-heuristic algorithm

To address the challenges associated with solving this problem in higher dimensions, we utilized a genetic algorithm, allowing for a comparison of results. The code for this algorithm has been implemented in Python. First, we need to establish the parameters for the genetic algorithm. After conducting trial-and-error experiments and consulting with experts in the field, we aimed to adopt the most effective combination of parameters. For this problem, we set the number of generations to 1,500, the initial population size to 300, the crossover probability to 0.95, and the mutation probability to 0.06. We then randomly generated the initial population and entered an evaluation loop where we calculated the fitness of this population and checked the algorithm's termination condition. If the algorithm had not met the termination criteria, we proceeded to create the next generation population. This process continued until we reached the termination condition of completing 1,500 generations. Below, we describe the operators used in this problem:For the purpose of crossover, a random number is first generated. If this random number is greater than the crossover probability (0.7), recombination is not performed, and the parents are copied directly into the next generation. If the random number is less than or equal to the crossover probability, crossover occurs, and we need to select the parents. A tournament selection method with a size of 2 is used to choose the parents, meaning that the chromosome with the best fitness is selected from two randomly chosen chromosomes. If a child is created from this crossover, the mutation probability (0.4) is considered. If the random number is within this probability, the mutation operator is applied to the child before it is added to the new generation. To calculate fitness, both the objective function and constraints are taken into account. The fitness value is the sum of the objective function and the weighted values of the constraints. For equality constraints, if the resulting solution does not equal the specified value, the penalty is calculated as the square of the violation of that value. For inequality constraints, if the value falls within the desired range, the penalty is zero; if it lies outside this range, the penalty is determined by the distance from the nearest boundary of the acceptable range. Ultimately, competency is defined as follows:Competency = objective function – 50 × 23 × (Penalty for tie restrictions + Penalty for unequal restrictions)

Here, a negative coefficient (50 × 23) is considered for the penalty. This coefficient results in a significant penalty for any violations of the constraints, affecting the entire chromosome. As long as the fitness value is negative, it indicates that the constraints are not satisfied. Once the fitness value becomes positive, it signifies that the constraints have been met, allowing the search to continue in order to maximize the objective function. Typically, this process leads to a solution that is close to the optimal one. The results obtained from this algorithm are presented in Table 4.

Table 4. Results of the genetic algorithm

Cluster	Reviewer's
0	jin, z; song, j; tang, j; xu, g; zheng, j; zhou, f
1	chen, m; feng, b; jin, z; kim, w; yang, f; yang, jb; zhao, f
2	lee, s; sun, y; wu, l; wu, x; yang, f; yang, jb; zheng, j
3	lee, j; li, m; tang, y; wu, w
4	chen, l; chen, m; he, c; kim, w; wu, q; xie, x; xu, w; yang, l; yang, x; zhou, f
5	chen, g; li, g; liu, m; xie, x
6	lee, j; pedrycz, w; sun, y; wang, f; yu, c
7	gao, x ; tang, j
8	zhou, z;huang, l ; sun, y ; xu, m ; zheng, j; zhou, f
9	chen, l ; feng, s ; yu, s
10	xu, q ; yu, c
11	yu, c
12	xu, g ; lu, l
13	ma, h
14	pedrycz, w
15	li, g
16	tang, s ; zhou, x
17	zhang, k ; yu, c; liang, l
18	liu, w ; wu, l ; xu, q ; zhou, f
19	xu, g
20	wu, w ; jin, z
21	liao, s ; yu, j
22	zhou, z ; chin, ks

The best integrated objective function obtained from this algorithm is 120/35, and the time to solve this problem with the algorithm is about 7 minutes.

Analysis and sensitivity of the results

To conduct the sensitivity analysis of the mathematical model, we examined how varying the weights of three objective functions—subject coverage, fairness in judgment, and the reviewer's expertise in the subject area—affects the model's output. These weights are crucial because the overall objective function is formulated as a linear combination of the three criteria, each assigned a specific weight coefficient. For the sensitivity analysis, we selected several combinations of weights, altering the priority of one criterion relative to the others. This approach allowed us to assess how changes in the weights impact the model's output and the values of each objective function, denoted as Z1 (maximum thematic coverage), Z2 (fairness of judgment), and Z3 (reviewer’s skill). After running the model for each combination of weights, we extracted the resulting optimal values for each objective function. These values were then plotted in a graph (Figure 3) to illustrate their variations in response to the weight changes. The combinations included scenarios with a heightened focus on subject coverage, fairness, or skill, as well as a scenario where all weights were equal to examine the equilibrium state of the model.

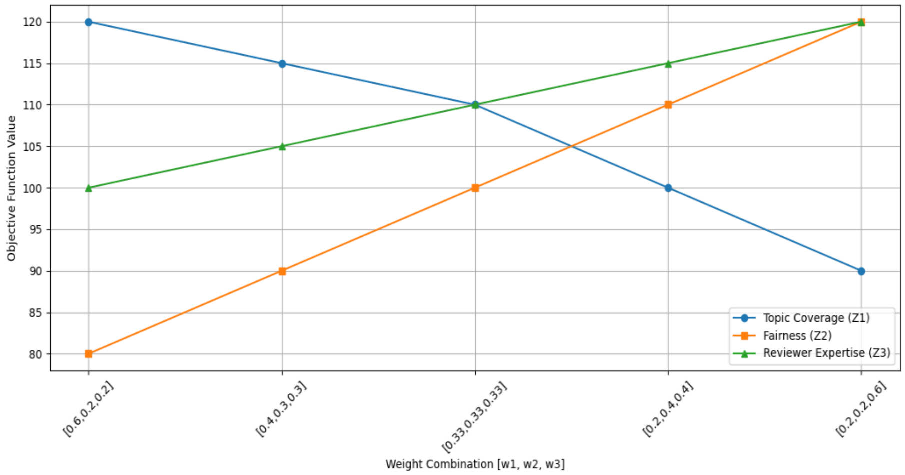

Figure 3. Results of sensitivity analysis of objective function weights

As shown in Figure 1, when the weight of the justice criterion increases, the value of the associated objective function rises significantly. However, this increase also leads to a decrease in the level of subject coverage. When the three weights are considered equally, the values of all three criteria are at a median level, indicating the model's stability and relative balance in this scenario. This behavior suggests that the model effectively responds to changes in weights, allowing users to optimize their use of the model according to their preferences in various situations. Furthermore, to analyze the performance of the assignment model in greater detail, a sensitivity analysis was conducted based on the objective function values resulting from assigning reviewers to article clusters, as presented in Figure 4. In this analysis, the horizontal axis represents the article clusters, while the vertical axis lists the selected reviewers. The value of each cell indicates the degree of fit or the objective function value for each judge-cluster combination. The results displayed in Figure 4 demonstrate that certain judges fit better in specific clusters, and selecting them for those clusters generates more value for the model.

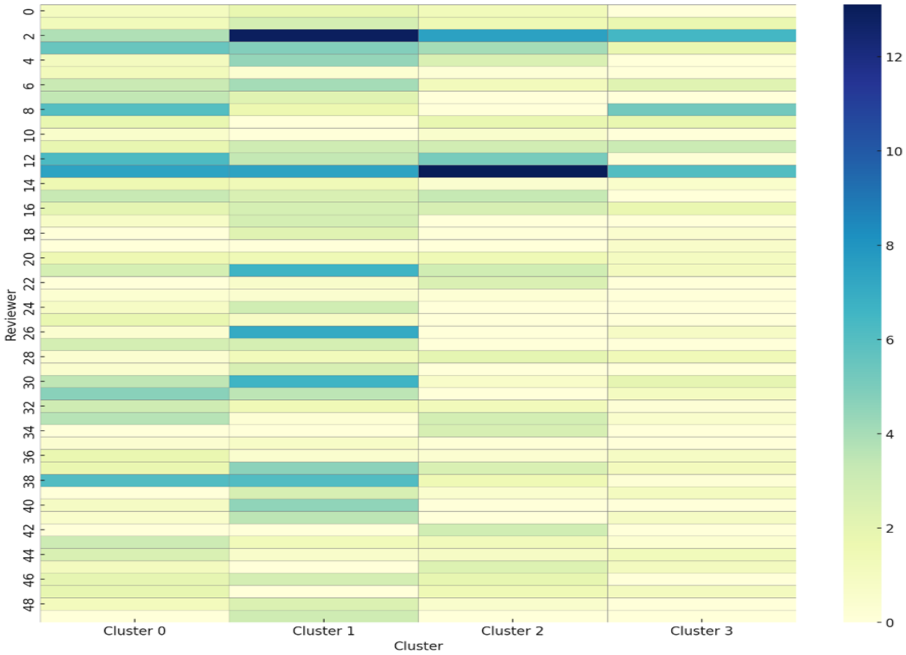

Figure 4. Sensitivity analysis of the objective function value resulting from the assignment of reviewers to paper clusters

Figure 4 illustrates that the assignment of reviewers to different clusters demonstrates varying levels of efficiency and compatibility. The analysis showed that certain reviewers consistently achieve high objective function values across one or more clusters, suggesting a strong alignment with the content of the articles in those clusters. On the other hand, some reviewers tend to perform poorly in most clusters, which may be due to a lack of expertise or experience in the relevant topics. The distribution of colors in the figure indicates that the model is sensitive to the choice of reviewers, and altering their assignments can have a significant effect on the overall quality of the evaluations.

Conclusion

In today's competitive landscape, human resources, including team evaluations, are recognized as vital components for success in scientific and research organizations. Editorial directors of journals and conferences should understand that quick problem-solving is a significant advantage in this field. Utilizing advanced mathematical modeling algorithms can be particularly beneficial in situations where time is critical, such as during large conferences or expedited review processes. Unfortunately, few studies have thoroughly addressed these issues, often focusing only on a limited range of criteria and constraints. Furthermore, the concept of article clustering, which is essential for enhancing the efficiency of reviewer assignment, has not received adequate attention from researchers. This study introduces a mathematical model that offers a new approach to assigning reviewers to articles, aiming to maximize thematic coverage, fairness, and expertise while forming a judging team for submitted articles. This method considers various constraints, such as balancing workloads and limiting the number of members in each group. Our research focused on developing a mathematical model for organizing judging teams by initially clustering articles and incorporating all pertinent criteria and limitations in the reviewer assignment process. The study population consisted of all articles published in expert journals indexed in the Web of Science database, from which 150 articles were selected through purposive sampling to apply the model. The proposed model was implemented using Python software, utilizing two algorithms: branch and bound, and a genetic meta-heuristic algorithm. We compared the results and conducted sensitivity analyses. The objective function generated by the branch and bound algorithm resulted in a value of 177.349, with a solution time of around one hour. In comparison, the genetic algorithm achieved a better value of 120.35 in just seven minutes. While the branch and bound algorithm delivers an optimal solution, its considerably longer solution time compared to the genetic algorithm raises concerns, particularly for larger-scale problems where the time required might substantially increase, potentially hindering the algorithm's ability to provide a viable solution.

In this study, we utilized an implicit method that relies on the reviewers' research backgrounds to measure the evaluation criteria. For future research, it would be advantageous to integrate this method with explicit approaches, which involve directly querying reviewers about their preferences, while still retaining the implicit method based on their research profiles. This combined strategy could draw on actual conference data and improve access to reviewer information. Furthermore, investigating additional meta-heuristic algorithms, such as ant colony optimization or tabu search, may allow for a comparison with the results of this research, potentially enhancing the overall findings. If the article authors are available, it would be beneficial to distribute a questionnaire to ascertain the subject areas of the articles. This information could then be utilized to apply fuzzy methods, incorporating linguistic variables to address the problem. Lastly, it is recommended that multi-objective algorithms be employed to analyze the model, creating a set of Pareto optimal solutions. This would facilitate the selection of the most suitable options for assigning reviewers to clusters through decision-making methods.

Data Availability Statement

Data available on request from the authors.

Acknowledgements

The authors would like to thank all participants of the present study.

Ethical considerations

The authors avoided data fabrication, falsification, plagiarism, and misconduct.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no conflict of interest.

References

Aksoy, M., Yanik, S., & Amasyali, M. F. (2023). Reviewer Assignment Problem: A Systematic Review of the Literature. *Journal of Artificial Intelligence Research*, 76, 761-827. https://doi.org/10.1613/jair.1.14318

Bouanane, K., Medakene, A. N., Benbelghit, A., & Belhaouari, S. B. (2024). FairColor: An efficient algorithm for the balanced and fair reviewer assignment problem. *Information Processing & Management*, *61*(6), 103865. https://doi.org/10.1016/j.ipm.2024.103865

Carpenter, J. M., Corvillón, A., & Shah, N. B. (2025). Enhancing Peer Review in Astronomy: A Machine Learning and Optimization Approach to Reviewer Assignments for ALMA. *Publications of the Astronomical Society of the Pacific*, *137*(3), 034501. https://doi.org/10.1088/1538-3873/adb5c1

Charlin, L., Zemel, R. S., & Boutilier, C. (2011). A Framework for Optimizing Paper Matching. In UAI (Vol. 11, pp. 86-95). https://doi.org/10.48550/arXiv.1202.3706

Cook, W. D., Golany, B., Kress, M., Penn, M., & Raviv, T. (2005). Optimal allocation of proposals to reviewer'sto facilitate effective ranking. *Management Science*, 51(4), 655-661. https://doi.org/10.1287/mnsc.1040.0290

Daraei, F. (2022). Combining Genetic Algorithms And Motor Nest Optimization To Solve The Supplier's Selection Problem, *Journal of Intelligent Marketing Management*, 3(3), 41-81.

Daş, G. S., & Göçken, T. (2014). A fuzzy approach for the reviewer assignment problem. *Computers & industrial engineering*, 72, 50-57. https://doi.org/10.1016/j.cie.2014.02.014

Delavar, A. (2008). Research method in psychology and educational sciences. Tehran.

Fan, Z. P., Chen, Y., Ma, J., & Zhu, Y. (2009). Decision support for proposal grouping: A hybrid approach using knowledge rule and genetic algorithm. *Expert systems with applications,* 36(2), 1004-1013. https://doi.org/10.1016/j.eswa.2007.11.011

Hettich, S., & Pazzani, M. J. (2006). Mining for proposal reviewer's: lessons learned at the national science foundation. *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 862-871)*. https://doi.org/10.1145/1150402.1150521

Hoang, D. T., Nguyen, N. T., Collins, B., & Hwang, D. (2021). Decision support system for solving reviewer assignment problem. *Cybernetics and Systems*, 52(5), 379-397. https://doi.org/10.1080/01969722.2020.1871227

Immanuel, S. D., & Chakraborty, U. K. (2019). Genetic algorithm: An approach on optimization. *In 2019 international conference on communication and electronics systems (ICCES)* (pp. 701-708). https://doi: 10.1109/ICCES45898.2019.9002372.

Kamps, J., Marx, M., Mokken, R. J., & De Rijke, M. (2004). Using WordNet to measure semantic orientations of adjectives. *In Lrec* (Vol. 4, pp. 1115-1118).

Kolasa, T., & Krol, D. (2011). A survey of algorithms for paper-reviewer assignment problem. *IETE Technical Review,* 28(2), 123-134. https://doi.org/10.4103/0256-4602.78092

Kalmukov, Y. (2013). Describing papers and reviewer's' competences by taxonomy of keywords. *arXiv preprint arXiv:1309.6527*. https://doi.org/10.48550/arXiv.1309.6527

Karimzadehgan, M., Zhai, C., & Belford, G. (2008). Multi-aspect expertise matching for review assignment. *In Proceedings of the 17th ACM conference on Information and knowledge management* (pp. 1113-1122). https://doi.org/10.1145/1458082.1458230

Karimzadehgan, M., & Zhai, C. (2012). Integer linear programming for constrained multi-aspect committee review assignment. Information processing & management, 48(4), 725-740. https://doi.org/10.1016/j.ipm.2011.09.004

Land, A. H., & Doig, A. G. (2009). An automatic method for solving discrete programming problems. In *50 Years of Integer Programming 1958-2008:* (pp. 105-132). https://doi.org/10.1007/978-3-540-68279-0_5

Leyton-Brown, K., Nandwani, Y., Zarkoob, H., Cameron, C., Newman, N., & Raghu, D. (2024). Matching papers and reviewer'sat large conferences. *Artificial Intelligence*, *331*, 104119. https://doi.org/10.1016/j.artint.2024.104119

Li, K., Cao, Z., & Qu, D. (2017). Fair reviewer assignment considering academic social network. *In Web and Big Data: First International Joint Conference, APWeb-WAIM 2017, Beijing, China, July 7–9, 2017, Proceedings,* Part I 1 (pp. 362-376). https://doi.org/10.1007/978-3-319-63579-8_28

Li, X., & Watanabe, T. (2013). Automatic Paper-to-reviewer Assignment, Based on the Matching Degree of the Reviewers .Procedia Computer Science, 22, 633-642. https://doi.org/10.1016/j.procs.2013.09.144

Long, C., Wong, R. C. W., Peng, Y., & Ye, L. (2013). On good and fair paper-reviewer assignment. *In 2013 IEEE 13th international conference on data mining* (pp. 1145-1150).

Luo, X. G., Li, H. J., Zhang, Z. L., & Jiang, W. (2024). Multi-objective optimization for assigning reviewer'sto proposals based on social networks. *Journal of Management Science and Engineering*. *9*(3), 419-439. https://doi.org/10.1016/j.jmse.2024.05.001

Mimno, D., & McCallum, A. (2007). Expertise modeling for matching papers with reviewer's. *In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 500-509). https://doi.org/10.1145/1281192.1281247

Monteiro, R. D. (1997). School of Industrial and Systems Engineering Georgia Institute of Technology, Atlanta, GA 30332.

Neshati, M., Beigy, H., & Hiemstra, D. (2014). Expert group formation using facility location analysis. *Information processing & management*, 50(2), 361-383. https://doi.org/10.1016/j.ipm.2013.10.001

Ribeiro, A. C., Sizo, A., & Reis, L. P. (2023). Investigating the reviewer assignment problem: A systematic literature review. *Journal of Information Science*, *0*(0). https://doi.org/10.1177/01655515231176668

Rordorf, D., Käser, J., Crego, A., & Laurenzi, E. (2023). A Hybrid Intelligent Approach Combining Machine Learning and a Knowledge Graph to Support Academic Journal Publishers Addressing the Reviewer Assignment Problem (RAP). In *AAAI Spring Symposium: MAKE*. https://doi.org/10.26041/fhnw-11149

Schirrer, A., Doerner, K. F., & Hartl, R. F. (2007). Reviewer assignment for scientific articles using memetic algorithms. *Metaheuristics: Progress in Complex Systems Optimization*, 113-134. https://doi.org/10.1007/978-0-387-71921-4_6

Simon, H. A., & Newell, A. (1958). Heuristic problem solving: The next advance in operations research. *Operations research*, 6(1), 1-10. https://doi.org/10.1287/opre.6.1.1

Stelmakh, I., Wieting, J., Neubig, G., & Shah, N. B. (2023). A gold standard dataset for the reviewer assignment problem. *arXiv preprint arXiv:2303.16750*. https://doi.org/10.48550/arXiv.2303.16750

Xu, Y., Ma, J., Sun, Y., Hao, G., Xu, W., & Zhao, D. (2010). A decision support approach for assigning reviewer'sto proposals. *Expert Systems with Applications*, 37(10), 6948-6956. https://doi.org/10.1016/j.eswa.2010.03.027

Wang, F., Zhou, S., & Shi, N. (2013). Group-to-group reviewer assignment problem. *Computers & operations research*, 40(5), 1351-1362. https://doi.org/10.1016/j.cor.2012.08.005

Wi, H., Oh, S., Mun, J., & Jung, M. (2012). A team formation model based on knowledge and collaboration. *IEEE Engineering Management Review*, 40(1), 44-57. https://doi.org/10.1016/j.eswa.2008.12.031

Zhao, X., & Zhang, Y. (2022). Reviewer assignment algorithms for peer review automation: A survey. Information Processing & Management, 59(5), 103028. https://doi.org/10.1016/j.ipm.2022.103028