Lexicographic Improvement Strategy for TOPSIS

Dariusz Grynia[a], Robert Susmaga[a], Izabela Szczęch [a,*] and Dariusz Brzezinski[a]

[a]Institute of Computing Science, Poznan University of Technology, Poland
ORCID (Dariusz Grynia): https://orcid.org/0000-0001-7196-4665, ORCID (Robert Susmaga): https://orcid.org/0000-0001-6707-0394, ORCID (Izabela Szczęch): https://orcid.org/0000-0002-9655-4109, ORCID (Dariusz Brzezinski): https://orcid.org/0000-0001-9723-525X

Abstract. Multi-criteria decision analysis (MCDA) is extensively used across diverse industries to assess and rank alternatives. Among numerous MCDA methods developed to solve real-world ranking problems, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) remains one of the most popular choices in many application areas. While TOPSIS effectively ranks alternatives based on their proximity to ideal and anti-ideal solutions, its user, the decision maker (DM) often requires the ability to better understand the ranking and identify potential improvements to specific alternatives. Currently, however, such post-factum analysis (PFA) methods for TOPSIS primarily focus on single-criterion modifications, limiting their applicability in complex, multi-criteria scenarios. This paper addresses this gap by extending the existing PFA toolbox by developing a lexicographic method that leverages multi-criteria modifications. Furthermore, the efficacy and applicability of the proposed method are demonstrated through a real-world case study and performance benchmark. Notably, the preliminary findings presented in this paper can be extended to enhance post-factum analysis for a broad class of function-based ranking methods.

1 Introduction

As a sub-discipline of operations research, multi-criteria decision analysis (MCDA) aims to assist the decision maker (DM) in addressing problems that involve evaluating real-world objects (alternatives) based on multiple conflicting criteria. This process often includes selecting the most preferred objects, classifying them into preference groups, or establishing a ranking order [1, 8, 4]. Among the various methods used to rank alternatives from the most to the least preferred, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [10] is a widely adopted approach. This method generates rankings based on predefined ideal and anti-ideal alternatives. Specifically, TOPSIS computes distances from each alternative to these, yielding non-negative real values that establish a linear preorder, which is then used for ranking purposes.

The TOPSIS method has been widely used in many applications, including logistics [2], manufacturing [17], marketing [18], sustainable development [15], and engineering [12]. In many of these applications, DMs are not only interested in the final ranking but also in finding ways of modifying alternatives to improve their ranking positions. This idea of implementing modifications to the alternative evaluations in order to achieve a user-specified performance target is related to the problem of benchmarking [14] and the concept of target setting in data envelopment analysis [3, 6], which is a benchmarking technique for improving the performance of organizations [9]. Similar ideas have also been applied to multi-criteria sorting [11] and additive value ranking models [5]. However, none of the abovementioned methods are applicable to TOPSIS. The only method for targeted alternative modifications in TOPSIS was proposed by Dutta *et al.* as part of their post-factum analysis framework [7]. However, the proposed method for shifting the ranking position of an alternative relies on recommending modifications to a single criterion. Therefore, such an approach shows only one way of attaining the desired ranking position and is only useful when the position change is attainable through a single criterion and, to some extent, ignores the multi-criteria nature of TOPSIS.

In this paper, we introduce a mathematical programming-based formalization of the PFA problem, applicable to both single-objective and multiple-objective improvements. Following the proposed formalization, we extend the PFA toolbox to include a novel multi-criteria method for improving the ranking positions of alternatives. Additionally, we demonstrate the proposed method in a practical case study. The main contributions of this paper are as follows:

- In Section 2, we mathematically formalize the problem of specifying improvement targets and defining performance changes.
- In Section 3, we recall the single-criterion Direct Method and introduce a *Lexicographic Binary Search* method.
- In Section 4, we apply the proposed method to the problem of selecting development areas in Poland based on investment conditions.
- In Section 5, we discuss ongoing work and future extensions to this preliminary study.

2 Formal framework for Post-Factum Analysis

In this section, we first introduce the necessary notation and recall the fundamental principles of the TOPSIS method. Then, we mathematically formalize the concepts of alternative modifications in post-factum analysis. In particular, we define *performance targets* an alternative may be expected to achieve and the *mathematical programming problems* that correspond to optimizing an alternative to reach a specified target. The concepts and notation presented in this section will be used to define the newly proposed post-factum analysis methods in Section 3.

* Corresponding Author. Email: iszczech@cs.put.poznan.pl.

2.1 The TOPSIS procedure

Let $A = \{a_1, a_2, ..., a_m\}$ denote a finite set of *alternatives* evaluated on a family of *criteria* $G = \{g_1, g_2, ..., g_n\}$. Then, the *performance* of the alternative a_i on the criterion g_k is denoted as p_{ik}. The performance vector describing the evaluations of a_i on all n criteria will be $\mathbf{p}_i = [p_{i1}, p_{i2}, ..., p_{in}]$. We assume that the criteria map the alternatives to real-valued intervals. Commonly, two types of criteria are distinguished: gain and cost, for which the preference increases or decreases with the value, respectively. Additionally, vector $\mathbf{w} = [w_1, w_2, ..., w_n]$ contains *weights* that will be used to distinguish the influence of the criteria. As such, the weights constitute subjective preferential information provided on input, as opposed to the descriptions of alternatives in terms of criteria, which constitute objective data provided on input.

The main actions performed by the TOPSIS [10] method can be summarized as:

1. *Normalize and weight the performances of alternatives.* Normalization is meant to equate the influence of potentially different ranges of criteria values. For this purpose, one can employ min-max scaling, as in [16]; as a result, the evaluations of each alternative performance vector $\mathbf{p}_i$ are re-scaled so the values on each criterion fall into the [0, 1] interval. Afterward, weighting by a set of weights $\mathbf{w}$ is meant to control the relative importance of criteria. The resulting normalized and weighted performance vector of alternative a_i will be denoted as $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{in}]$.
2. *Determine the positive and negative ideal solution.* This involves the construction of two hypothetical alternatives: the positive ideal solution (PIS) and the negative ideal solution (NIS), which are characterized by the best and the worst possible performances on *all* criteria, respectively.
3. *Calculate the relative closeness coefficient for each alternative.* The relative closeness of the performance vector $\mathbf{x}_i$ of alternative a_i is calculated as:

$$\mathsf{R}(\mathbf{x}_i) = \frac{d(\mathbf{x}_i, NIS)}{d(\mathbf{x}_i, NIS) + d(\mathbf{x}_i, PIS)}, \tag{1}$$

   where $d(\cdot,\cdot)$ denotes the Euclidean distance between two vectors. By utilizing the two distances from an alternative to the ideal and anti-ideal solutions, the coefficient naturally aggregates the objective performances of the alternatives and the subjective preferential information expressed by weights. As opposed to the performance vector, which conveys the individual (criterion-based) evaluations of alternative $\mathbf{x}_i$, the value of $\mathsf{R}(\mathbf{x}_i)$ conveys its global evaluation (or rating).
4. *Rank alternatives based on the relative closeness.* The final ranking is obtained by ordering the alternatives based on the assigned R values. Because relative closeness is of type gain, the higher the R value, the better rank an alternative may achieve.

2.2 Formalization of performance targets and performance modifications

When an alternative performs poorly on criteria that are important to the DM (on criteria with high weights), then it is likely to obtain a low value of R and occupy an unsatisfactory position in the ranking. In such a scenario, it may be expected to achieve a better position. We will refer to such an expected ranking position as the *performance target*. The simplest example of such a target is the top position. However, in practice, such an action may prove unattainable. To solve this problem, we propose a more general approach that allows one to define and potentially attain any performance target for any alternative. Because the set of all possible modifications that attain the target may be exceedingly numerous, it is important to focus on very specific modifications, especially the minimal ones, which will be easiest to achieve in practical situations. Moreover, the final solution should potentially reflect various limiting conditions imposed on the process, e.g., the fact that certain criteria (e.g., age) cannot or should not be modified.

To specify the performance targets, let us first denote by $\mathbf{x}_c$ the *current alternative's performance* that is to be improved and by $\mathbf{x}_t$ the *performance target* that the current alternative is to surpass. Therefore, $\mathbf{x}_t$ determines a set of *possible solutions* $T(\mathbf{x}_t) = \{\mathbf{v} : \mathsf{R}(\mathbf{v}) > \mathsf{R}(\mathbf{x}_t)\}$, i.e., a set of performance vectors that are better than $\mathbf{x}_t$ according to R. Notice that at the lowest level 'improving the performance' involves 'improving' values of individual criteria, which resolves itself to increasing the values of those of type 'gain' and decreasing the values of those of type 'cost'.

In practice, additional conditions may be introduced, e.g., conditions reflecting limitations to allowed criteria values. These may be simple linear inequalities that impose allowed ranges on the variable values or more intricate conditions on the variable values. As a result of incorporating those conditions, set $T(\mathbf{x}_t)$ is further constrained, producing a set of *feasible solutions*, denoted as $S(\mathbf{x}_t)$. The final selected solution (the modification of $\mathbf{x}_c$ that surpasses $\mathbf{x}_t$) will be denoted by $\mathbf{x}_s$; by definition $\mathbf{x}_s \in S(\mathbf{x}_t)$. Notice that even when $T(\mathbf{x}_t) \neq \varnothing$, the additional constraints may result in $S(\mathbf{x}_t) = \varnothing$, in which case there may be no feasible solution to the problem.

Let us illustrate performance target specification with an example. Consider two alternatives, $\mathbf{x}_i$ and $\mathbf{x}_j$, such that $\mathbf{x}_i$ is ranked lower than $\mathbf{x}_j$. If this is to be changed, $\mathbf{x}_i$ may be assigned target $\mathbf{x}_j$. In the process of searching for the necessary changes, $\mathbf{x}_i$ will be denoted as $\mathbf{x}_c$ (the current version of $\mathbf{x}_i$), $\mathbf{x}_j$ as $\mathbf{x}_t$ (the target performance), while the new version of $\mathbf{x}_i$ as $\mathbf{x}_s$ (the modification to $\mathbf{x}_i$).

As stated above, because there may exist multiple possible modifications of $\mathbf{x}_c$, the search should be focused on particular ones, especially those that are close to $\mathbf{x}_c$. Such versions are of special interest, because they are both easy to express formally and to achieve in practice. As a result, the search for performance modifications of an alternative $\mathbf{x}_c$ may be formulated as an instance of a non-linear programming problem:

$$\begin{array}{ll} min & \mathsf{R}(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S(\mathbf{x}_t) \end{array} \tag{Problem 1}$$

where $S(\mathbf{x}_t)$ is the set of feasible performance vectors that are better than the target $\mathbf{x}_t$ according to R. Because the number of solutions to the above problem can be excessively large, it may be modified to look for such a feasible performance vector that is closest to the current alternative. To achieve this, the problem can be re-formulated as follows:

$$\begin{array}{ll} min & d(\mathbf{x}, \mathbf{x}_c) \\ \text{s.t.} & \mathbf{x} \in S(\mathbf{x}_t) \end{array} \tag{Problem 2a}$$

The solution to this formulation will be a single performance vector $\mathbf{x}_s$ that is characterized by a minimal Euclidean distance $d(\cdot,\cdot)$ from the current alternative's performance. The solution $\mathbf{x}_s$ will be one of the multiple solutions of the more general formulation (Problem 1).

The formulation presented in Problem 2a assumes that the target $\mathbf{x}_t$ has a performance value that is ranked higher than the current alternative $\mathbf{x}_c$. However, if the alternative's evaluations are already sufficiently favorable, one may be interested in determining the allowed deterioration of performances that would achieve a lower, but acceptable rating. For instance, if an expert concludes that it is impossible for a highly-ranked company to maintain all quality indicators at their current level, determining their admissible deterioration may be useful. This means that selected criteria can be sacrificed to some extent while the current overall rank is retained. This leads to a re-specification of the more general Problem 1 to:

$$\begin{array}{ll} min & d(\mathbf{x}, \mathbf{x}_t) \\ \text{s.t.} & \mathbf{x} \in S(\mathbf{x}_t) \end{array} \quad \text{(Problem 2b)}$$

Here, we demand a solution $\mathbf{x}_s$ that is as close as possible, but better, than the specified target $\mathbf{x}_t$. Once again, the solution will be one of the multiple solutions of the more general problem formulation presented in Problem 1.

3 Post-factum analysis methods

In the following section, we discuss different possible implementations of post-factum analysis (PFA) methods that let users solve performance modification problems, taking into account additional constraints.

3.1 Single criterion direct method

In our suite of PFA methods, we include an approach proposed by Dutta et al. [7] that enables precise calculation of single-criterion performance modifications needed to achieve a target performance in TOPSIS. We will refer to this approach as the *Direct Method*. The method is based on formulating and solving a quadratic equation for the desired value of the relative closeness coefficient (R) and can be considered a special case of Problem 1. The primary limitation of the approach of Dutta et al. [7] is its single-criterion focus, which may prove inadequate for achieving performance targets in real-world scenarios. When alternatives underperform across multiple criteria or require significant ranking improvements, single-criterion modifications rarely suffice. For instance, elevating a poorly ranked alternative to a top position typically demands improvements across multiple criteria. In such cases, more comprehensive methods enabling simultaneous multi-criteria analysis become necessary.

3.2 Lexicographic binary search

To address the limitations of the single-criterion Direct Method, we propose a different implementation of Problem 1 in the form of a *Lexicographic Binary Search* algorithm. This approach enables the DM to define a list of criteria eligible for performance modification with constrained performance modification ranges. It is especially well justified when the criteria differ heavily in terms of difficulty associated with improving their values, which is very often the case.

The algorithm operates in two phases. In the first phase, it sequentially optimizes criteria from a user-specified list (maximizes for criteria of type gain and minimizes for criteria of type cost) until either all criteria are exhausted (indicating the target is unachievable) or the performance target is reached. If a feasible solution exists, the second phase employs a binary search to refine the performance modification of the final criterion in the initial solution. As a result, the method finds a minimal modification that exceeds the target performance $\mathsf{R}(\mathbf{x}_t)$ by no more than the pre-defined marginal value.

The algorithm's greedy approach makes results dependent on criteria ordering, which can be an advantage. Users can prioritize easier-to-implement criteria first, relegating more challenging criteria to later positions. Thus, the DM maintains control over the solution's practicality while exploring nuanced improvement pathways. Notably, this heuristic can also handle single-criterion improvements—if a performance target can be achieved by modifying a single criterion, this method will find the solution. Therefore, the lexicographic search approach can potentially replace the direct method, offering greater flexibility.

4 Case study

We applied the proposed PFA methods on a dataset of Special Economic Zones in Poland [13]. The goal of this case study was twofold: to demonstrate the usefulness and usability of the approaches for improving the alternative's ranking position, and to discuss the pros and cons of each method.

4.1 Dataset

The analyzed dataset, as provided in [13], contains ten Special Economic Zones located in Poland. Special Economic Zones (SEZs) are areas that attract investors by offering favorable investment conditions, better infrastructure, and facilitated access to specialized staff. Moreover, running a business in an SEZ usually entails favorable tax regulations and reimbursements for some innovative projects. For a more comprehensive overview of the characteristics of SEZs and the benefits they offer to investors, please refer to websites of the Polish Ministry of Economic Development and Technology.

The considered ten SEZs are characterized by the following five criteria (see Table 1 for details):

- **Total area** (hectares; to be minimized) — the area occupied by each SEZ. Efficient zones should be compact.
- **Capital expenditures** (billions PLN; to be minimized) — a cumulative amount of money invested in a SEZ. Ideally, a thriving SEZ should require minimal capital expenditures.
- **Number of jobs** (to be maximized) — the total number of people employed in enterprises running within a given SEZ with valid business permits. One of the primary goals of establishing a SEZ is to create many new job opportunities.
- **Business permits** (to be maximized) — the number of business permits granted to companies operating within an SEZ.
- **Financial result** (billions PLN; to be maximized) — the total revenue earned by all companies operating within an SEZ.

Following the steps of the TOPSIS method (see Section 2.1) the SEZ performances presented in Table 1 underwent normalization using min-max scaling and weighting using weights $\mathbf{w}$=[6.06, 26.95, 20.02, 16.53, 30.44] outlined in [13]. Then, for each alternative, the relative closeness coefficient R was calculated, which led directly to the final TOPSIS ranking of all SEZs. The computed R and the ranking position of each SEZ are also reported in Table 1.

The economic area called Kostrzyn-Słubice (KOS) is ranked best, whereas the Pomorze region (POM) is the worst of all the considered SEZs. The ranking positions 6–9 share quite similar values of R, and as a result, even moderate changes in R can result in improving the alternative's ranking position. For simplicity, we will treat each SEZ as an alternative with its own ranking position, although we do note that Łódź and Starachowice (ranks 7 and 8) would probably be considered indifferent in an actual decision making scenario.

The solutions proposed by different PFA methods are presented as changes that need to be applied to particular criteria in order to improve the rank of some considered SEZ. Obviously, only feasible changes are discussed, i.e., those that keep the alternative's performance within the allowed interval ranges of criteria.

Table 1. The performances of ten Special Economic Zones (SEZs) located in Poland in terms of five criteria. The criteria to be minimized are marked by ↓, the criteria to be maximized by ↑. The last two columns present the relative closeness coefficient (R) and the final TOPSIS rank under the weight vector $\mathbf{w} = [6.06, 26.95, 20.02, 16.53, 30.44]$.

Full name	SEZ Code	Total Area [ha] ↓	Capital Expenditures [B PLN] ↓	Number of Jobs ↑	Business Permits ↑	Financial Result [B PLN] ↑	R	Rank
Kostrzyn-Słubice	KOS	2201.25	7.13	32400	180	22.98	0.7154	1
Tarnobrzeg	TAR	1868.21	7.47	20740	195	18.22	0.6241	2
Mielec	MIE	1723.97	7.84	34992	268	4.96	0.5073	3
Kraków	KRA	949.66	4.24	29580	189	1.37	0.4848	4
Legnica	LEG	1341.15	5.13	15294	86	7.61	0.4359	5
Słupsk	SLU	910.16	1.59	3478	79	0.76	0.4122	6
Łódź	LOD	1754.64	13.32	33401	209	7.40	0.4114	7
Starachowice	STA	707.98	1.79	6829	56	0.70	0.4113	8
Kamienna Góra	KAM	540.83	2.56	7530	60	0.56	0.3980	9
Pomorze	POM	2246.29	10.48	24893	173	1.48	0.3207	10

Table 2. Overview of possible ranking position improvements for Legnica (LEG) under different criteria changes for the Direct Method. A negative value in the “Change needed" column indicates that the criterion value must be decreased.

Changed criterion	Rank improvement	Change needed
Capital Expenditures	LEG 5 → 4	-2.07
	LEG 5 → 3	-3.21
Number of Jobs	LEG 5 → 4	$+10372.00$
	LEG 5 → 3	$+15433.00$
Business Permits	LEG 5 → 4	$+105.00$
	LEG 5 → 3	$+152.00$
Financial Result	LEG 5 → 4	$+3.25$
	LEG 5 → 3	$+4.70$
	LEG 5 → 2	$+13.55$

Table 3. Overview of possible ranking position improvements for Legnica (LEG) under different criteria changes for the Lexicographic Binary Search method. Each improvement was considered for two criteria orderings: Number of Jobs followed by Financial Result and Financial Result followed by Number of Jobs.

Order of changed criteria	Rank improvement	Δ Number of Jobs	Δ Financial Result [B PLN]
Number of Jobs	LEG 5 → 4	+10372	0.00
Financial Result	LEG 5 → 3	+15433	0.00
	LEG 5 → 2	+19698	+6.86
Financial Result	LEG 5 → 4	0	+3.25
Number of Jobs	LEG 5 → 3	0	+4.70
	LEG 5 → 2	0	+13.55

4.2 Direct method

To illustrate the application of the Direct Method [7] (Section 3.1), let us consider the Special Economic Zone called Legnica (LEG), originally ranked at the fifth spot (Table 1). There are several possibilities for LEG to improve its position when changes are made to one criterion only (Table 2). In particular, LEG could move from fifth to fourth position (LEG $5 \rightarrow 4$) if the Capital Expenditures were reduced by 2.07B PLN or more. LEG can even go up to the third position if the Capital Expenditures get reduced by 3.21B PLN. Similar position changes can be obtained by increasing the Number of Jobs, Business Permits, or Financial Results. It is then up to the DM to determine which criterion is the easiest for them to change. However, of the considered criteria, only Financial Result is capable of promoting LEG to the second rank (in that case, it needs to be increased by over 13.55B PLN). It should be noted that LEG cannot obtain the first rank by introducing changes to one criterion only, and methods that modify larger groups of criteria must be applied.

4.3 Lexicographic binary search

The proposed Lexicographic Binary Search method (Section 3.2) works on a list of criteria that the DM designates as changeable. The approach tries to achieve the expected improvement using only the first criterion on the list, and when that is insufficient, the next criterion is considered, and so on. The DM should, therefore, carefully prepare the set of criteria and put them in order depending on which criteria are easiest to improve.

Possible improvements in LEG's rank when considering the Number of Jobs and Financial Result are reported in Table 3. For each position change, both criteria were considered, however, Lexicographic Binary Search was run twice to analyze both orderings: Number of Jobs followed by Financial Result and Financial Result followed by Number of Jobs.

For moving LEG up to the fourth (LEG $5 \rightarrow 4$) or third (LEG $5 \rightarrow 3$) ranking position, it is enough to change just the first criterion, which corresponds to the results of the Direct Method in Table 2. The change on the second criterion is then equal to 0. An interesting scenario can be observed when LEG is to be put in the second position (LEG $5 \rightarrow 2$). When the criteria are considered in the order Number of Jobs before Financial Result, then the heuristic will propose a maximal change on the first criterion and a small change on the second one. However, if the order is reversed, then the change on the first criterion (Financial Result) is enough, and no change in the Number of Jobs is needed.

The two considered criteria are nonetheless insufficient for moving Legnica up to the first rank. Such an improvement can be obtained by adding a third criterion—the number of Business Permits. Table 4 reports all possible changes to those criteria as produced by the Lexicographic Binary Search aiming to move Legnica to the top rank (LEG $5 \rightarrow 1$). All possible orderings of the three-criterion set are presented. Interestingly, an extreme change to the Number of Jobs and Financial Result must be accompanied by a 3.16 increase in Business Permits (see the first row in Table 4). However, should the DM reverse the ordering of the considered criteria (see the last row in Table 4), the extreme changes to Business Permits and Financial Result would be sufficient to place Legnica on top of the ranking and no change in the Number of Jobs would be needed.

Lexicographic Binary Search always recommends an extreme change in the first criterion on the list, then moves to the second one, etc. This, however, might result in a change that is too drastic

Table 4. Overview of possible criteria changes for moving Legnica from 5th to 1st rank with respect to the Lexicographic Binary Search considering three criteria: Number of Jobs, Financial Result and Business Permits. The improvement was considered for all criteria orderings.

Order of changed criteria	Rank Improvement	Δ Number of Jobs	Δ Financial Result [B PLN]	Δ Business Permits
Number of Jobs, Financial Result, Business Permits	LEG 5 → 1	+19698	+15.37	+3.16
Number of Jobs, Business Permits, Financial Result	LEG 5 → 1	+19698	+6.53	+182.00
Financial Result, Number of Jobs, Business Permits	LEG 5 → 1	+19698	+15.37	+3.16
Financial Result, Business Permits, Number of Jobs	LEG 5 → 1	0	+15.37	+140.74
Business Permits, Number of Jobs, Financial Result	LEG 5 → 1	+19698	+6.53	+182.00
Business Permits, Financial Result, Number of Jobs	LEG 5 → 1	0	+13.73	+182.00

to be implemented in practice and, as such, might not be satisfactory to the DMs. When a more balanced improvement strategy that involves a larger number of criteria is required, one can try to limit the range of the allowed changes on individual criteria in order to find an adequate solution. This, however, would require potentially several attempts.

5 Discussion

This paper has introduced a framework for post-factum analysis (PFA) within the context of TOPSIS, a widely used multi-criteria decision analysis (MCDA) ranking method. We have presented a novel mathematical programming-based formalization of the PFA problem, enabling a systematic approach to identifying modifications to alternative performances that achieve desired ranking targets. Building upon this formalization, we presented Lexicographic Binary Search, a new PFA implementation (of multiple-criterion character) alongside the existing Direct Method (of single-criterion character). A real-world case study on Special Economic Zones in Poland demonstrated the practical usefulness of the proposed method.

The impact of this work lies in providing a robust methodology for targeted alternative improvement. Building upon this general methodology, our ongoing work focuses on extending the PFA toolset by providing genetic algorithms and non-linear programming implementations as alternatives to the Lexicographic Binary Search approach. The additional methods will provide decision makers with the ability to control which criteria are modified and to what extent, empowering them to explore a diverse set of feasible and meaningful improvement strategies.

Apart from the mentioned ongoing work, our study opens several other avenues for future research. Firstly, the underlying mathematical programming formalization may readily be adapted to other MCDA ranking methods, like UTA or SAW. Secondly, to achieve more challenging, multi-stage targets (e.g., improving by several positions in a ranking), long-term multi-step improvement strategies may be designed and implemented. Overall, this research represents a step towards making MCDA methods more explainable, actionable, and, ultimately, more valuable for supporting real-world decision-making processes.

Acknowledgements

This research was partly funded by the National Science Centre, Poland, grants number 2022/47/D/ST6/01770 and 2023/51/B/ST6/00545.

References

[1] R. Bisdorff, L. Dias, P. Meyer, V. Mousseau, and M. Pirlot. *Evaluation and Decision Models with Multiple Criteria*. Springer, 2015.

[2] E. Bottani and A. Rizzi. A fuzzy TOPSIS methodology to support outsourcing of logistics services. *Supply Chain Management*, 11(4):294–308, 2006.

[3] A. Charnes, W. Cooper, and E. Rhodes. Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2 (6):429–444, 1978.

[4] M. Cinelli, M. Kadziński, G. Miebs, M. Gonzalez, and R. Słowiński. Recommending multiple criteria decision analysis methods with a new taxonomy-based decision support system. *European Journal of Operational Research*, 302(2):633–651, 2022.

[5] K. Ciomek, V. Ferretti, and M. Kadziński. Predictive analytics and disused railways requalification: Insights from a post factum analysis perspective. *Decision Support Systems*, 105:34–51, 2018. ISSN 0167-9236.

[6] W. W. Cooper, L. M. Seiford, and K. Tone. *Data envelopment analysis: a comprehensive text with models, applications, references and DEA-solver software*. Springer, New York, 2007.

[7] B. Dutta, T. Singha, M. Goh, M.-T. Lamata, and J.-L. Verdegay. Post factum analysis in topsis based decision making method. *Expert Systems with Applications*, 138:112806, 2019. ISSN 0957-4174.

[8] S. Greco, M. Ehrgott, and J. Figueira. *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer, 2016.

[9] D. Grynia, M. Kadziński, and M. Stamenković. Attaining robust performance targets in data envelopment analysis with application to efficiency evaluation of airports. *Computers & Industrial Engineering*, 190:110065, 2024. ISSN 0360-8352.

[10] C. L. Hwang and K. Yoon. *Multiple Attribute Decision Making: Methods and Applications*. Springer-Verlag, 1981.

[11] M. Kadziński, M. Stamenković, and M. Uniejewski. Stepwise benchmarking for multiple criteria sorting. *Omega*, 108:102579, 2022. ISSN 0305-0483.

[12] S.-S. Lin, A. Zhou, and S.-L. Shen. Safety assessment of excavation system via TOPSIS-based MCDM modelling in fuzzy environment. *Applied Soft Computing*, 138:110206, 2023.

[13] K. Martyn, M. Martyn, M. Kadziński, J. Liu, and X. Liao. PrefRank: a family of multiple criteria decision analysis methods for exploiting a valued preference relation. *Engineering Applications of Artificial Intelligence*, submitted, 2025.

[14] M. Petrović, N. Bojković, I. Anić, M. Stamenković, and S. P. Tarle. An electre-based decision aid tool for stepwise benchmarking: An application over eu digital agenda targets. *Decision Support Systems*, 59: 230–241, 2014. ISSN 0167-9236.

[15] M. Piwowarski, D. Miłaszewicz, M. Łatuszyńska, M. Borawski, and K. Nermend. TOPSIS and VIKOR methods in study of sustainable development in the EU countries. *Procedia Computer Science*, 126: 1683–1692, 2018.

[16] R. Susmaga, I. Szczech, and D. Brzezinski. Towards explainable TOPSIS: Visual insights into the effects of weights and aggregations on rankings. *Applied Soft Computing*, 153:111279, 2024. ISSN 1568-4946.

[17] W.-P. Wang. Toward developing agility evaluation of mass customization systems using 2-tuple linguistic computing. *Expert Systems with Applications*, 36(2):3439–3447, 2009.

[18] X. Yu, S. Guo, J. Guo, and X. Huang. Rank B2C e-commerce websites in e-alliance based on AHP and fuzzy TOPSIS. *Expert Systems with Applications*, 38(4):3550–3557, 2011.