

Article

A Decision-Making Tool for Algorithm Selection Based on a Fuzzy TOPSIS Approach to Solve Replenishment, Production and Distribution Planning Problems

Eduardo Guzman *, Beatriz Andres and Raul Poler

Research Centre on Production Management and Engineering (CIGIP), Universitat Politècnica de València (UPV), Calle Alarcón 1, 03801 Alcoy, Spain; bandres@cigip.upv.es (B.A.); rpoler@cigip.upv.es (R.P.)

* Correspondence: eguzman@cigip.upv.es

Abstract: A wide variety of methods and techniques with multiple characteristics are used in solving replenishment, production and distribution planning problems. Selecting a solution method (either a solver or an algorithm) when attempting to solve an optimization problem involves considerable difficulty. Identifying the best solution method among the many available ones is a complex activity that depends partly on human experts or a random trial-and-error procedure. This paper addresses the challenge of recommending a solution method for replenishment, production and distribution planning problems by proposing a decision-making tool for algorithm selection based on the fuzzy TOPSIS approach. This approach considers a collection of the different most commonly used solution methods in the literature, including distinct types of algorithms and solvers. To evaluate a solution method, 13 criteria were defined that all address several important dimensions when solving a planning problem, such as the computational difficulty, scheduling knowledge, mathematical knowledge, algorithm knowledge, mathematical modeling software knowledge and expected computational performance of the solution methods. An illustrative example is provided to demonstrate how planners apply the approach to select a solution method. A sensitivity analysis is also performed to examine the effect of decision maker biases on criteria ratings and how it may affect the final selection. The outcome of the approach provides planners with an effective and systematic decision support tool to follow the process of selecting a solution method.

Keywords: fuzzy TOPSIS; algorithm selection; production planning; heuristics; metaheuristics; matheuristics

MSC: 68Q25; 65Y20; 03B52

Citation: Guzman, E.; Andres, B.; Poler, R. A Decision-Making Tool for Algorithm Selection Based on a Fuzzy TOPSIS Approach to Solve Replenishment, Production and Distribution Planning Problems *Mathematics* **2022**, *10*, 1544. <https://doi.org/10.3390/math10091544>

Academic Editor: Mariano Luque

Received: 14 April 2022

Accepted: 4 May 2022

Published: 4 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The supply chain comprises different sequential activities, such as replenishment, production and distribution, which must all be planned and optimized. The main management function of companies is planning [1]. Planning activities aim to effectively coordinate and schedule a company's available resources [2]. Planning is accompanied by a set of decisions to be made by the planning manager; for example, a planner must make decisions about the quantity of materials needed for production by taking into account storage capacity and production batches to reduce production and inventory costs, production scheduling and sequencing on machines, and to finally make decisions about the delivery flow of finished products to customers or distribution centers [3].

Many real-world combinatorial optimization problems, such as those in transportation and logistics [4–6] and manufacturing [7–9], pose a huge challenge due to the high complexity of most companies' operations given the type of industry to which they

belong. They are also subject to not only dynamic conditions, such as customer demands, processing times, returns on investment, but also to uncertainties, such as unavailability of items, changes in market conditions and shortages due to changes in demand [10].

Thus, planning problems seek to maximize profit or gain while minimizing costs and meeting market, environmental and societal constraints. For example, in supply planning problems, there is a direct relation between inventory costs and the costs associated with distribution planning, such as transportation costs and on-time delivery to customers [11]. Therefore, the difficulty of such problems is substantial due to the amount of data they handle [12], nonlinearities and discontinuities, complex constraints, possible conflicting objectives and uncertainty [13]. Hence, different types of solvers are used to solve these problems, as are algorithms because of their computational difficulty [14].

Given the large number of algorithms for solving replenishment [15], production [16] and distribution planning problems [17], how to effectively select an algorithm for a given task or a specific problem is an important, but also difficult issue. Peres and Castelli [18] highlight that rules which standardize the formulations of existing combinatorial optimization problems (COP) in planning are lacking, which means that researchers have to start building an algorithm from scratch, which thus limits the interoperability of this field because the algorithms in the literature must be adjusted to solve a specific problem. These authors conclude that the consolidation of combinatorial optimization problems is lacking and note that this is important for the field of COPs to reach a higher degree of maturity.

The algorithm selection problem (ASP) is an active research area in many fields, such as operations research [19–21] and artificial intelligence (AI) [22,23]. For many decades, researchers have developed increasingly sophisticated techniques and algorithms to solve difficult optimization problems [18]. These techniques include mathematical programming approaches, heuristics, metaheuristics, nature-inspired metaheuristics, matheuristics and various hybridizations [24]. Literature reviews such as that presented by Jamalnia et al. [25], who reviewed the aggregate production planning problem under uncertainty between 1970 and 2018, detailed the use of approximately 24 different techniques to solve this type of problem out of 92 reviewed papers. Kumar et al. [26] presented a literature review covering the period from 2000 to 2019 of the quantitative approaches used to solve production and distribution planning problems. They found 13 different techniques and types of solvers, including CPLEX and LINGO, to solve this type of problem out of 74 papers. Pereira et al. [27] analyzed the tactical sales and operations planning problem. To do so, they reviewed 103 papers, where the year was not limited. They detailed about 35 different techniques to solve this type of problem. Hussain et al. [28] conducted a literature review of the applications of metaheuristic algorithms and found 140 different metaheuristic algorithms in 1222 publications over a 33-year search period (1983 to 2016).

Different research papers have conducted experimental studies to determine the performance of an algorithm [29–32] or several algorithms according to a problem type with a collection of datasets available in the literature [33–35]. For example, Pan et al. [36] compared three constructive heuristics and four metaheuristics (discrete artificial bee colony, scatter search, iterated local search, iterated greedy algorithm) for the distributed permutation flowshop problem, for which they made extensive comparative evaluations based on 720 instances. However, these comparisons do not provide any enlightening results because they are generally limited to a set of algorithms and to a specific problem set [24].

In practice, algorithm performance vastly varies from one problem state to another. In many cases, heuristic [37], metaheuristic [28] and matheuristic [38] techniques involve randomization, such as genetic algorithm, particle swarm optimization, bee swarm optimization, bat algorithm, artificial tribe algorithm and firefly algorithm [39–43], which results in performance variability, even across repeated trials in a single problem instance [44]. Risk is an important additional feature of algorithms because the planner or the person in charge of selecting an algorithm for planning must be willing to settle for average or lower performance in exchange for a reasonable answer or may also find a better solution than that expected in the same resolution time. This situation is often encountered in

companies that attempt to maximize their profits because these problems are solved by constructing mixed strategies, i.e., strategies that meet the desired risk and return.

Nowadays, if a study demonstrates the superiority of one algorithm over other algorithms, that algorithm can be expected to be useful for other problem types for which it has not yet been tested. No-free-lunch (NFL) theorems [45] describe that there is no single algorithm that outperforms all algorithms in all the instances of a problem [24].

Therefore, the selection of the most suitable algorithm to solve an optimization problem for replenishment, production and distribution planning is a very difficult task. Algorithm selection requires advanced knowledge of the efficiency of algorithms, the characteristics of the problem, as well as mathematical and statistical knowledge. However, having the necessary knowledge to find a solution with algorithms does not guarantee success [46].

Algorithm selection depends mainly on the expected results and the data that the company has at the time. Therefore, the properties or characteristics of the business problem must be examined. For this purpose, the linearity of the problem, the number of parameters and the characteristics that the solution supports must be analyzed.

Evaluating algorithms to solve a problem usually involves more than one criterion, such as problem type, problem knowledge, performance, computation time, the quality of the expected solution and programming knowledge. Therefore, algorithm selection can be modeled as a multicriteria decision making problem [22].

The objective of multicriteria decision making (MCDM) is to identify the most eligible alternatives from a set of alternatives based on qualitative and/or quantitative criteria with different units of measurement to select or rank them [47]. Different techniques such as AHP, ELECTRE, PROMETHEE, SAW, TOPSIS and VIKOR are used to solve MCDM problems [3]. Several studies have been conducted to compare the performance of these techniques; for example, that presented by Zanakis et al. [48], which compared eight MCDM techniques (four variations of AHP, ELECTRE, TOPSIS and SAW). It concluded that different techniques are affected mainly by the number of alternatives because as alternatives increase, methods tend to generate similar final rankings. Opricovic and Tzeng [49] performed a comparative analysis of the VIKOR and TOPSIS methods. Both these methods are based on an aggregation function that represents the closeness to the ideal. The study revealed that the main differences between the two methods were the employed normalization method types.

Opricovic and Tzeng [50] compared the extended VIKOR method to ELECTRE II, PROMETHEE and TOPSIS. The obtained results showed that ELECTRE II, PROMETHEE and VIKOR gave similar results, while TOPSIS presented different results in some alternatives. Chu et al. [51] made a comparison of the VIKOR, TOPSIS and SAW methods. The study revealed that SAW and TOPSIS presented similar classifications, while VIKOR presented different results. These authors concluded that VIKOR and TOPSIS provided results that were close to reality. Ozcan et al. [52] presented a comparative analysis of the TOPSIS, ELECTRE and Grey Theory techniques for the warehouse location selection problem, where the Grey Theory provided different results to TOPSIS and ELECTRE. Instead, the last two obtained similar results.

In situations in which information is not quantifiable or incomplete, as in real-world problems where data may be incomplete or imprecise, i.e., nondeterministic, data can be represented in a fuzzy way using linguistic variables to represent decision makers' preferences in complex or not well-defined situations. Imprecision in MCDM problems can be modeled using the fuzzy Set Theory, which is used to extend different MCDM techniques. In this background, Ertuğrul and Karakaşoğlu [53] conducted a comparative study of the fuzzy AHP and fuzzy TOPSIS methods for the facility location selection problem. Both methods obtained the same results, i.e., the same rank order of alternatives.

Other studies have used an extension of the classical fuzzy set called the intuitionistic fuzzy set, as proposed by Atanassov [54]. Intuitionistic fuzzy sets have been applied in many fields, such as facility location selection [55], supplier selection [56], evaluation of

project and portfolio management information systems [57,58], and personnel selection [59]. Büyüközkan and Güleriyüz [60] compared the performance of ranked fuzzy TOPSIS and intuitionistic fuzzy TOPSIS by detailing how the alternatives ranking barely differed between the two approaches.

From the above comparisons, it is clear that many techniques are available for multicriteria decision making [61]. These techniques have their advantages and limitations over others depending on the type of problem [62].

Different MCDM techniques have been used for the classification algorithm selection problem, such as the study by Lamba et al. [63] in which TOPSIS and VIKOR were used to evaluate 20 classification algorithms. Both methods obtained similar results. Peng et al. [22] used four different MCDM techniques (TOPSIS, VIKOR, PROMETHEE II and WSM) to select multiclass classification algorithms. The TOPSIS, VIKOR and PROMETHEE II methods achieved similar classifications, while WSM obtained slightly different ones. Peng et al. [64] evaluated ranking algorithms for financial risk prediction purposes. Using TOPSIS, PROMETHEE and VIKOR, they obtained similar results for the three main ranking algorithms. They concluded that the followed techniques were advantageous for choosing a classification algorithm.

Along these lines, TOPSIS stands out as a widely used technique that is efficient for selecting classification algorithms. It has been successful in different areas such as supply chain and logistics management, environment and energy management, health and safety management, business and marketing management, engineering and manufacturing, human resource management and transportation management [47,65–67] and, according to Chu et al. [51], is able to represent reality. It is also useful for companies because it can be run with a spreadsheet [68]. For all these reasons and given the fact that the choice of a solution method is subject to vagueness and uncertainty, we use the fuzzy TOPSIS method.

In this context, the present paper aims to answer this question: which solution method is suitable for a replenishment, production and distribution planning problem given a portfolio of algorithms or solvers?

To answer this question, and by taking into account that no research to date has analyzed the selection of algorithms for planning with a multicriteria decision method, a decision-making tool to select algorithms for a planning problem based on fuzzy TOPSIS is presented. To validate the use of the tool herein proposed, an illustrative example is presented, which has been validated by four different manufacturing companies. This paper is organized as follows. Section 2 deals with the literature review. The adopted methodology is shown in Section 3 and the numerical application of the methodology is presented in Section 4. The sensitivity analysis of the results is provided in Section 5. Finally, Section 6 includes the conclusions and future research lines.

2. Algorithm Selection Problem Literature Review

Algorithm selection has been widely addressed by the scientific community in both the mathematics [69,70] and Artificial Intelligence (AI) [71,72] areas. In the mathematical area, Stützle and Fernandes [73] report how the characteristics of problem instances make the performance of metaheuristics relative to the properties of instances. Therefore, it is necessary to explore the relation between algorithms and instances. In the AI area, different models have been developed to predict which algorithm is the best one for a problem instance, which is conducted by analyzing the relation between the characteristics of an instance and a set of training data used by an algorithm. In this way, with an algorithms portfolio it is possible to predict which algorithm in a new problem instance is most likely to work [74].

Growing interest has been shown in the ASP to put previously developed algorithms to best use to solve a specific problem instead of developing new ones [75]. According to Leyton-Brown et al. [76], some algorithms are better than others on average, and there is rarely a best algorithm for a given problem. Instead “it is often the case that different

algorithms perform well on different problem instances. This phenomenon is most pronounced among algorithms for solving NP-Hard problems, because runtimes for these algorithms are often highly variable from instance to instance". In this context, Rice [77] proposes the first description of methodologies to select algorithms. Kotthoff [75] defines this as the "task of algorithm selection involves choosing an algorithm from a set of algorithms on a per-instance basis in order to exploit the varying performance of algorithms over a set of instances".

In this regard, algorithm selection approaches have been successfully applied in different problem domains [78]. The following table summarizes a literature review of the various papers that have approached ASP from different perspectives (Table 1).

Table 1. Research studies addressing the algorithm selection problem.

Author	Proposal
Lagoudakis and Littman [79]	Algorithm selection using reinforcement learning.
Xu et al. [80]	A scalable and completely automated portfolio construction. The authors improve the ASP methodology by integrating local search solvers as candidate solvers by predicting performance scores instead of runtime, and by using hierarchical hardness models that take into account different types of instances.
Smith-Miles [81]	A unified framework to take the algorithm selection problem as a learning problem and to use this framework to tie together cross-disciplinary developments in tackling the algorithm selection problem. The authors generalize metalearning concepts to algorithms that focus on tasks, including sorting, forecasting, constraint satisfaction and optimization.
Bischi et al. [35]	An algorithm selection problem as a cost-sensitive classification task that is based on an Exploratory Landscape Analysis.
Hoos et al. [82]	A modular open-solver architecture that integrates several different portfolio-based algorithm selection approaches and techniques.
Kotthoff [75]	An algorithm selection for combinatorial search problems.
Tierney and Malitsky [83]	An algorithm selection benchmark based on optimal search algorithms to solve the container premarshalling problem (CPMP), an NP-hard problem from the container terminal optimization field.
Cunha et al. [84]	A metalearning method is used to select the best recommendation algorithms within different scopes to allow to understand the relations between data characteristics and the relative performance of recommendation algorithms, which can be used to select the best algorithm(s) for a new problem. This work analyzes the algorithm selection problem for Recommender Systems by focusing on Collaborative Filtering.
Bożejko et al. [85]	A local and optima network analysis and machine learning is used to select appropriate algorithms on an instance-to-instance basis.
Drozdov et al. [86]	Graph convolutional network-based generative adversarial networks for the algorithm selection problem in classification terms.
Vilas Boas et al. [87]	Integer programming-based approaches to build decision trees for the algorithm selection problem. These techniques allow the automation of three crucial decisions by discerning the most important problem features to determine problem classes by grouping problems into classes, and then selecting the best algorithm configuration for each class.

Marrero et al. [88]	An efficient parallel genetic algorithm (GA) is proposed as a first step to solve the algorithm selection problem. GA is able to attain competitive results in optimal objective value terms and in a short time. The computational results show that the approach is able to efficiently scale and considerably reduce the average elapsed time to solve Knapsack Problem (KNP) instances.
De Carvalho et al. [21]	A cross-domain evaluation for multi-objective optimization. The authors investigate how four state-of-the-art online hyperheuristics with different characteristics perform to find solutions for 18 real-world multi-objective optimization problems. These hyperheuristics were designed in previous studies and tackle the algorithm selection problem from different perspectives: election-based, based on Reinforcement Learning and based on a mathematical function.

In manufacturing environments, formulations are usually very complex [78] because they present a variety of specific constraints related to the company's scope. Generally, these formulations can serve as blocks or subproblems for other formulations of other specific manufacturing environments. In this way, many formulations or algorithms can obtain similar results to the formulations proposed above. When selecting a formulation or algorithm, tuning the parameters of the different techniques is a very demanding task because each algorithm has different characteristics and the number of times that a parameter tuning has to be performed against different instances of a problem when performing a comparison can exponentially grow [33]. Furthermore, to compare algorithms and select one, the feature set of the instances must be taken into account because the characterization of instances determines a solution approach's performance. In practice, the information needed to establish the characteristics is not always available [89], and experimental results may lead to the fact that there is no single best or worst algorithm for all problem instances [46]. In this context, and as shown in Table 1, several approaches have been proposed to address the algorithm selection challenge, including heuristic algorithms, metaheuristics, hybrid metaheuristics, hyperheuristics, and machine-learning techniques. Many of these approaches integrate similarities, such as using a set of instances to learn, measuring or predicting the performance of the best algorithm. The success of algorithm selection approaches for some problem domains has motivated us to develop a decision making tool to support planners of companies to select a solution method (algorithm or a solver) for replenishment, production and distribution planning problems.

3. Solution Methodology

For combinatorial optimization problems with realistic discrete decision variables, such as scheduling, sequencing, distribution and transportation planning problems, performing an exhaustive search space for this problem type is not a realistic option despite having a finite search space. The literature includes several heuristic, metaheuristic and matheuristic algorithms, as well as tests with commercial and non-commercial high-performance solvers to solve such problems. So, this question arises: which algorithm is to be chosen for a combinatorial optimization problem?

Generally one way of finding an algorithm to solve a combinatorial optimization problem is to exhaustively run all the available algorithms and choose the best solution [90]. However, this method requires unlimited computational resources and companies have limited computational, programming and mathematical resources, which makes it impossible to test all the algorithms or to use several solvers to test one instance or several for a specific problem. Weise et al. [12] emphasize that there is a variety of methods to solve different types of problems with acceptable performance, but they can be outperformed by very specialized methods.

Weise et al. [12] consider that there is no optimization method that is better or can outperform others, and the NFL Theorem [45] corroborates this theory. This theorem states that no optimization algorithm is likely to outperform several existing types of methods in different types of problems.

In turn, the same authors mention that the efficiency of an optimization algorithm is based on knowledge of a problem. Radcliffe [91] emphasizes that the algorithm's performance will improve with adequate knowledge of the problem. However, knowledge of one type of problem can be misleading for another type of problem [89] because there is no algorithm that outperforms others in all instances of a problem. Therefore, an algorithm's performance will be based on experience and empirical results.

Algorithm selection schemes are based mainly on approaches that either run a sequence of algorithms in a limited execution time [80,82] or predict the performance of an algorithm for a given instance and select the algorithm with the best predicted performance [92].

Real-world planning problems are subject to inaccuracies and uncertainties, conflicts between constraints and objectives, discontinuities and nonlinearities [13]. Therefore, determining which algorithm is appropriate poses a challenge that can be analyzed using a multicriteria decision technique for ranking and prioritizing algorithms because algorithm selection involves multiple decisions that require the simultaneous assessment of the various advantages and disadvantages.

In most companies, the complexity of operations has several components that must be addressed at the same time. Evaluating an algorithm to solve a problem often involves more than one criterion, such as problem type, problem knowledge, performance, computation time, the quality of the expected solution and programming knowledge.

MCDM techniques integrate different criteria and an order of preference to evaluate and select the optimal option among multiple alternatives based on the expected outcome. The objective of these techniques is to obtain an ideal solution to a problem in which a decision makers' experience does not allow them to decide among the various considered parameters. As a result, a ranking is obtained according to the selected criteria, their respective values and the assigned weights [93].

There are many criteria in real-life problems that can directly or indirectly affect the outcome of different decisions. Decision making often involves inaccuracies and vagueness that can be effectively dealt with using fuzzy sets. This method is especially important for clarifying decisions that are difficult to quantify or compare, especially if decision makers have different perspectives, as in this study. Therefore, we herein adopt the fuzzy TOPSIS methodology to model an algorithm or solver selection given a solution methods portfolio to solve replenishment, production and distribution planning problems.

In decision making problems, the Fuzzy Set Theory was introduced by Zadeh [94] to overcome the ambiguity and uncertainty of human thought and reasoning by using linguistic terms to represent decision makers' choices.

The TOPSIS method was originally proposed by Hwang and Yoon [95]. It is based on choosing an alternative that should have the shortest distance between the positive ideal solution (PIS) and the negative ideal solution (NIS), i.e., the selected alternative is obtained with the closest solution to the PIS and is farthest away from the NIS. The main limitation of this technique is that it cannot capture ambiguity in the decision making process [96]. To overcome this limitation, Chen [97] developed the Fuzzy TOPSIS Method to quantitatively evaluate the score of different alternatives by conferring weight to the different criteria described with linguistic variables. This section briefly describes the employed Fuzzy Set Theory and Fuzzy TOPSIS Method.

3.1. Fuzzy Set Theory and Fuzzy Numbers

The Fuzzy Set Theory [94,98,99] is associated with the TOPSIS method, and are related to another by the degree of membership of the elements in fuzzy sets. A fuzzy set is

characterized by the membership function, which can come in different formats, e.g., triangular, sigmoid or trapezoidal. The membership function assigns a degree of membership to each object according to its relevance $\mu_A(x): x \rightarrow [0.0, 1.0]$. To represent a fuzzy set, a tilde ‘~’ is placed [68].

For our study, we consider a triangular fuzzy number, \tilde{A} , which is denoted by its vertices (l, m, u) , as shown in Figure 1. Triangular fuzzy numbers are used to adapt decision makers’ preference to capture the vagueness of linguistic evaluations, where l, u and m , respectively, denote the lower bound, the upper bound and the crisp central value.

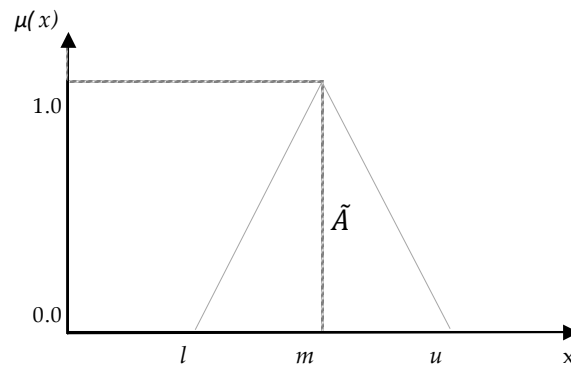


Figure 1. Fuzzy triangular number.

Membership function d of triangular fuzzy number \tilde{A} is defined as:

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-l}{m-l}, & l \leq x \leq m, \\ \frac{u-x}{u-m}, & m \leq x \leq u, \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $\tilde{A} = (l_A, m_A, u_A)$ and $\tilde{B} = (l_B, m_B, u_B)$ are two triangular fuzzy numbers with bases l, m, u . Then, the basic operational laws for triangular numbers are defined as:

$$\tilde{A} (+) \tilde{B} = (l_A, m_A, u_A) (+) (l_B, m_B, u_B) = (l_A + l_B, m_A + m_B, u_A + u_B) \tag{2}$$

$$\tilde{A} (-) \tilde{B} = (l_A, m_A, u_A) (-) (l_B, m_B, u_B) = (l_A - l_B, m_A - m_B, u_A - u_B) \tag{3}$$

$$\tilde{A} (\times) \tilde{B} = (l_A, m_A, u_A) (\times) (l_B, m_B, u_B) = (l_A \times l_B, m_A \times m_B, u_A \times u_B) \text{ for } l_A, l_B > 0; m_A, m_B > 0; u_A, u_B > 0 \tag{4}$$

$$\tilde{A} (\div) \tilde{B} = (l_A, m_A, u_A) (\div) (l_B, m_B, u_B) = \left(\frac{l_A}{u_B}, \frac{m_A}{m_B}, \frac{u_A}{l_B}\right) \text{ for } l_A, l_B > 0; m_A, m_B > 0; u_A, u_B > 0 \tag{5}$$

$$k\tilde{A} = kl_A, km_A, ku_A \tag{6}$$

$$\tilde{A}^{-1} = (l_A, m_A, u_A)^{-1} = \left(\frac{1}{u_A}, \frac{1}{m_A}, \frac{1}{l_A}\right) \text{ for } l_A, l_B > 0; m_A, m_B > 0; u_A, u_B > 0 \tag{7}$$

By assuming that fuzzy numbers \tilde{A} and \tilde{B} are real numbers, then the distance measure is identical to the Euclidean distance. Therefore, the vertex method is defined to calculate the distance between two fuzzy numbers (see Equation (8)). Although there are several ways of measuring distances between fuzzy numbers [100], the vertex method is a simple and efficient method [97,101].

$$d(\tilde{A}, \tilde{B}) = \sqrt{\frac{1}{3} [(l_A - l_B)^2 + (m_A - m_B)^2 + (u_A - u_B)^2]} \tag{8}$$

3.2. The Fuzzy TOPSIS Method

The main fuzzy TOPSIS idea is based on defining the fuzzy positive ideal solution (FPIS) and the fuzzy negative ideal solution (FNIS). The chosen alternative should have the shortest distance to the FPIS and the farthest distance to the FNIS. TOPSIS follows a systematic process and logic that seek to express the logic of human choice [102]. The basic fuzzy TOPSIS method steps are described in the following way (see [97], [103], [104]):

Step 1. Consider a set of k decision makers (D_1, D_2, \dots, D_k) with m alternatives (A_1, A_2, \dots, A_m) and n criteria (C_1, C_2, \dots, C_n) for which the decision matrix is established:

$$C_1 \quad C_2 \quad \dots \quad C_n \tilde{D} = \begin{matrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{matrix} \begin{bmatrix} \tilde{X}_{11} & \tilde{X}_{12} & \dots & \tilde{X}_{1n} \\ \tilde{X}_{21} & \tilde{X}_{22} & \dots & \tilde{X}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{X}_{n1} & \tilde{X}_{n2} & \dots & \tilde{X}_{nm} \end{bmatrix} i = \tag{9}$$

$$1, 2, \dots, m; j = 1, 2, \dots, n$$

$$\tilde{W} = [\tilde{w}_1, \tilde{w}_2, \dots, \tilde{w}_n]$$

Considering that the perception of algorithms and solvers varies according to knowledge and experience with algorithms for planning, the average value method is applied; where \tilde{x}_{ij}^k is the rating or score of the alternative A_i in relation to criterion C_j evaluated by the k -th decision maker (Equation (10)). The weights of criteria are aggregated using Equation (11), where \tilde{w}_j^k describes the weight of each criterion C_j according to decision makers D_k .

$$\tilde{x}_{ij} = \frac{1}{k} (\tilde{x}_{ij}^1 + \tilde{x}_{ij}^2 + \dots + \tilde{x}_{ij}^k) \tag{10}$$

$$\tilde{w}_j = \frac{1}{k} (\tilde{w}_j^1 + \tilde{w}_j^2 + \dots + \tilde{w}_j^k) \tag{11}$$

Step 2. Normalize the fuzzy decision matrix. Decision matrix \tilde{D} with m alternatives and n criteria is normalized to eliminate inconsistencies with the different units of measurement or scales to preserve the ranges of the normalized triangular fuzzy numbers. \tilde{R} represents the normalized decision matrix (Equation (12)):

$$\tilde{R}_j = [\tilde{r}_{ij}]_{m \times n}, i = 1, 2, \dots, m; j = 1, 2, \dots, n \tag{12}$$

The normalization process is performed by Equations (13) and (14), where B and C represent the set of benefit and cost criteria, respectively.

$$\tilde{r}_{ij} = \left(\frac{l_{ij}}{u_j^+}, \frac{ij}{u_j^+}, \frac{u_{ij}}{u_j^+} \right), \text{ and } u_j^+ = \max_i u_{ij} \text{ if } j \in B \tag{13}$$

$$\tilde{r}_{ij} = \left(\frac{l_j^-}{u_{ij}}, \frac{l_j^-}{m_{ij}}, \frac{l_j^-}{l_{ij}} \right), \text{ and } l_j^- = \min_i l_{ij} \text{ if } j \in C \tag{14}$$

Step 3. Construct the weighted normalized fuzzy decision matrix \tilde{V} (Equation (15)). \tilde{v}_{ij} is obtained by multiplying the weights of criteria \tilde{w}_j and the normalized fuzzy decision matrix \tilde{r}_{ij} values:

$$\tilde{V} = [\tilde{v}_{ij}]_{m \times n}, i = 1, 2, \dots, m; j = 1, 2, \dots, n \tag{15}$$

$$\tilde{v}_{ij} = \tilde{r}_{ij} \times \tilde{w}_j \tag{16}$$

Step 4. Obtain the FPIS (FPIS, A^+) and the FNIS (FNIS, A^-), as shown in Equations (17) and (18), respectively. The ideal solutions can be defined according to Chen [97] as: $\tilde{v}_j^+ = (1, 1, 1)$ and $\tilde{v}_j^- = (0, 0, 0)$

$$A^+ = \{ \tilde{v}_1^+, \tilde{v}_2^+, \dots, \tilde{v}_m^+ \} \tag{17}$$

$$A^- = \{\tilde{v}_1^-, \tilde{v}_j^-, \dots, \tilde{v}_m^-\} \quad (18)$$

Step 5. Calculate the distances for each alternative, where D_i^+ indicates the distance between the scores of alternative A_i to the FPIS (Equation (19)), and D_i^- denotes the distances between the values of alternative A_i to the FNIS (Equation (20)), where $d(\tilde{v}_a, \tilde{v}_b)$ represents the distance between two fuzzy numbers.

$$D_i^+ = \sum_{j=1}^n d(\tilde{v}_{ij}, \tilde{v}_j^+) , \quad i = 1, 2, \dots, m; j = 1, 2, \dots, n \quad (19)$$

$$D_i^- = \sum_{j=1}^n d(\tilde{v}_{ij}, \tilde{v}_j^-) , \quad i = 1, 2, \dots, m; j = 1, 2, \dots, n \quad (20)$$

Step 6. Determine proximity coefficient CC_i , which evaluates the rank order of all the alternatives A_i according to their overall performance. The proximity coefficient is calculated as shown in Equation (21).

$$CC_i = \frac{D_i^-}{(D_i^+ + D_i^-)} \quad (21)$$

Step 7. Rank alternatives A_i , using a decreasing order of CC_i values, the shortest distances from the FPIS, i.e., close to 1, to indicate that the overall performance of alternative A_i is better because it is farther away from the FNIS. Having obtained the ranking order, decision makers select the most feasible alternative A_i .

4. The Methodological Approach for the Algorithm Selection Problem

This paper employs a three-stage methodology to select an algorithm or solver to solve a replenishment, production and distribution planning problem (see Figure 2). The objective of this section is to present a numerical analysis to demonstrate the performance of the proposed methodology.

The three stages of the proposed methodology are described in the following subsections.

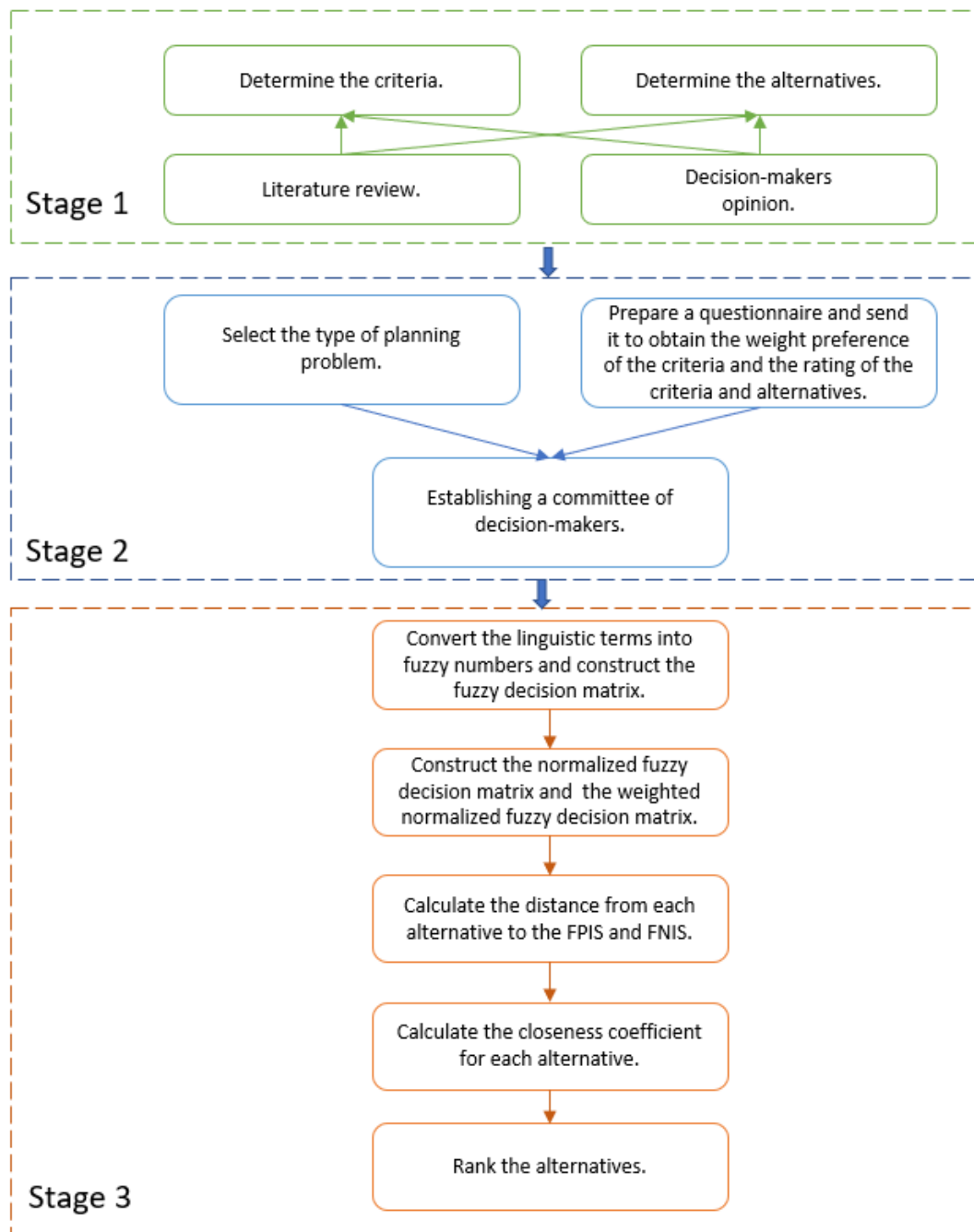


Figure 2. A methodological approach for the algorithm selection problem.

4.1. Stage 1—Define Criteria and Alternatives

We first identify the different criteria that are taken into account when selecting a solution method; these criteria can be identified in the literature and are based on the opinion of experts in the field [105]. According to each identified criterion, the decision maker evaluates the suitability of a solution method for the type of problem; that is, how algorithms or solvers can be suitable and formulated for a given problem.

In this research, 13 criteria are identified based on an exhaustive review of the literature (see [8,18,25,27,106,107]) and the assessments of experts in the field of operations research. These criteria are presented in Table 2.

Table 2. Criteria for algorithm selection.

Id	Criteria	Definition
C1	Problem type	<p>The replenishment (source), production (make) and distribution (deliver) planning problem type is determined by the SCOR (Supply Chain Operation Reference) methodology [106,108] (see Figure 3). Each problem type has its own characteristics and computational difficulty.</p> <p>According to Weise et al. [12], it is very difficult to make accurate estimates of a problems' computational performance because a solution method's performance will almost always depend on experience, the empirical results based on related research areas and the rules of thumb established for these problems. So, a problem's computational performance depends on different factors. Some of the main factors of a problem's complexity are: problem size, linearity, variables and presence of constraints [109]. Based on these considerations, criteria C2–C7 are proposed.</p>
C2	Equation type	It expresses the equations present in the problem. These equations can be linear or nonlinear.
C3	Variable type	It represents the elements to be modeled. Variables can be integer, binary and continuous. Planning problems generally contain a combination of variables: Continuous + Integer, Integer + Binary, Continuous + Binary, Continuous + Integer + Binary. These combinations normally generate greater computational difficulty. Each combination can generate a different behavior for the solution method because algorithm or solver performance is linked with the amount of resources used. These resources can be: amount of memory, processing time to deal with each type of variable [12].
C4	Number of instantiated variables	It determines the number of variables present in a problem, which is a determining factor when establishing the expected response time to obtain an answer.
C5	Type of constraints and solutions	<p>The constraint type determines the computational difficulty that the problem will have because constraints express limitations of resources. Some constraints can be expressed as follows:</p> <ul style="list-style-type: none"> ▪ decision \geq data (e.g., production \geq demand) ▪ decision \leq data (e.g., load \leq capacity) ▪ decision $=$ data (e.g., production \geq demand) ▪ decision \geq decision (e.g., production of A \geq production of B) ▪ decision \leq decision (e.g., load of M \leq load of N) ▪ decision $=$ decision (e.g., inventory A $=$ inventory B) ▪ continuity equations of some variables (e.g., Inventory $=$ Inventory prior period + Production - Demand) <p>One factor that affects a problem's difficulty is when the expected solutions to the problem contain a route or sequence. These routing planning or sequencing problems are generally NP-hard [110,111].</p>
C6	Number of constraints	The number of constraints contained in a problem can be a limiting factor for establishing the problem's difficulty. Therefore, the evaluator analyzes whether the set of constraints can be adapted to an algorithm or to a solver.
C7	Dataset size	When representing the problem input data size, a problem's computation is directly related to the amount of data.

C8	Programming knowledge	Programming knowledge is a determining factor when selecting an algorithm because it determines decision makers' ability to program one algorithm or several algorithms when having to test different algorithms in the hope to obtain a solution that meets the company's needs.
C9	Mathematical knowledge	Mathematical knowledge is important when choosing whether to express the problem as a mathematical model or to directly choose an algorithm. Algorithms generally require certain mathematical knowledge.
C10	Knowledge of algorithms	One aspect to take into account in companies is knowledge of the different algorithms.
C11	Software	This criterion is considered if the company has mathematical modeling software, but is not considered if the company does not. If the company has specific optimization software, the decision maker defines the scope of its performance against each alternative to solve a planning problem.
C12	Quality of solutions	This criterion establishes the quality of the expected solutions to the problem. These solutions can be optimal, near-optimal or good.
C13	Calculation time	The computation time sets the amount of expected time to obtain a solution for the problem.

Second, we identify the portfolio of solution methods (alternatives). This portfolio is composed of a set of nine algorithms and four solvers, identified as the most commonly used ones in the planning problems reported in [107,112]. Alternatives are divided between different algorithm types, which are:

- Heuristic algorithms (HA). They are used when solvers or exact techniques cannot reach solutions in acceptable computation times. These techniques do not provide optimal solutions, but can offer solutions that come very close to the optimum in acceptable computation times [113];
- Metaheuristic algorithms (MA). According to Swan et al. [114], these techniques are: "an iterative master process that guides and modifies the operations of subordinate heuristics to efficiently produce high-quality solutions. At each iteration, it manipulates either a complete (or partial) single solution or a collection of such solutions";
- Matheuristic algorithms (MTA). They combine mathematical programming techniques and heuristic or metaheuristic algorithms [115].

The alternatives in this classification are A1—HA/Benders' decomposition, A2—HA/LP and Fix, A3—HA/LP Relaxation, A4—MA/Tabu Search, A5—MA/Genetic Algorithm, A6—MA/Simulated annealing, A7—MA/Variable Neighborhood Search, A8—MTA/ Genetic Algorithm + Mathematical Model, A9—MTA/Simulated annealing + Mathematical Model. Different solver types used to solve planning problems are also considered. For this purpose, commercial and non-commercial solvers are identified to deal with mathematical models with linear and nonlinear equations. These are: A10—CPLEX (Commercial), A11—CBC (Non-Commercial), A12—BONMIN (Non-Commercial—Nonlinear), A13—LINDO (Commercial—Linear /Nonlinear).



Figure 3. Replenishment, production and distribution planning problem types.

The hierarchical structure that has been defined and constructed to assist in the process of selecting an algorithm or solver is shown in Figure 4. This structure is composed of four layers: the first one corresponds to the objective of this study; the second structure corresponds to the characterization of the different dimensions that have been proposed, which is composed of four dimensions (the problem type and its characteristics, programming knowledge, the software and the expected performance of algorithms or solvers); in the third layer comes the categorization of the 13 identified criteria; in the last one, methods or solution alternatives appear. The correlation between layer 3 and 4 is related to the performance of an algorithm or a solution method.

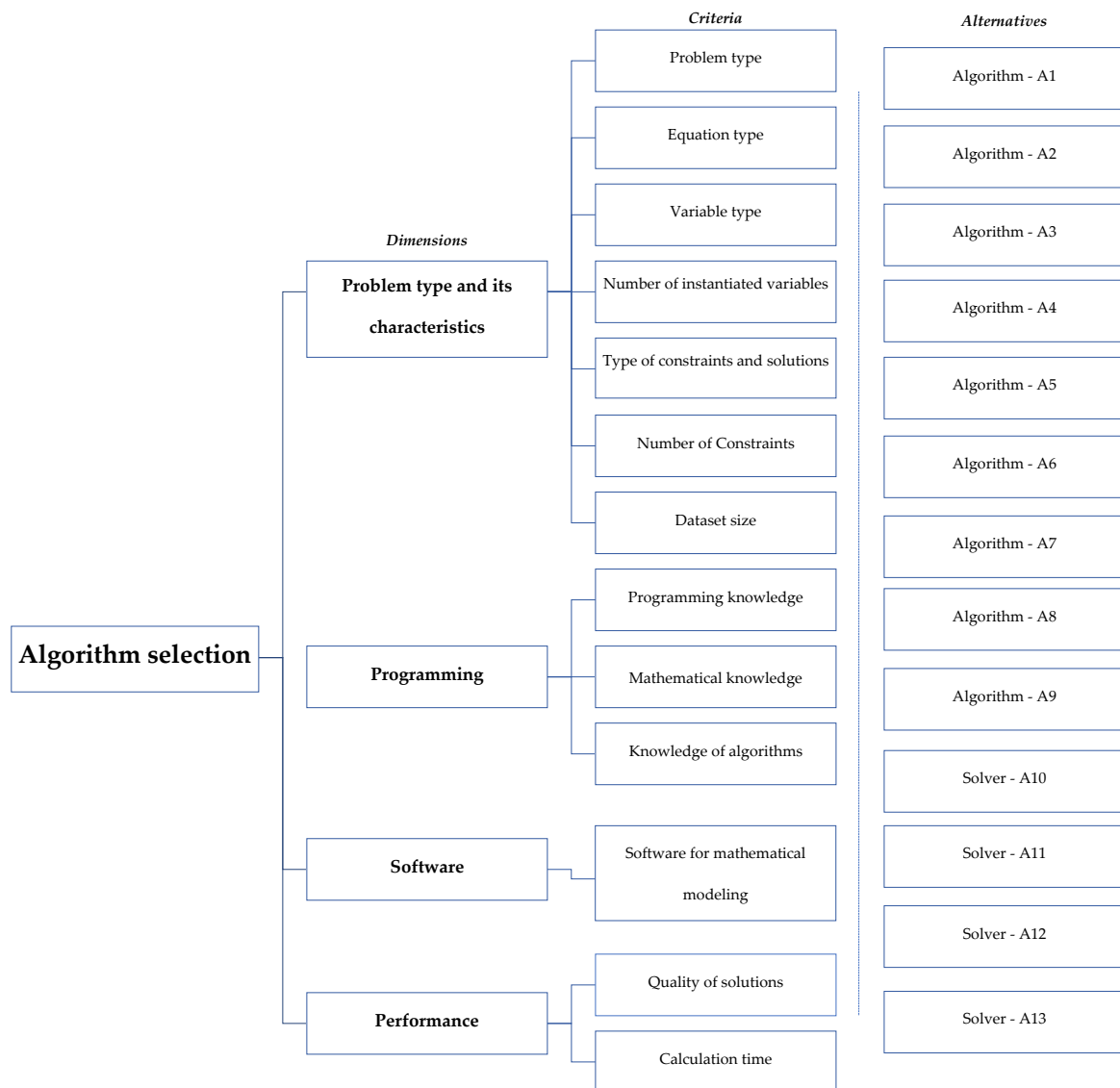


Figure 4. Hierarchical structure for algorithm selection.

4.2. Stage 2—Problem Statement

In this stage, the type of planning problem to be addressed was defined, for which four expert decision makers working in different manufacturing companies in the planning area were invited to propose a planning problem. The decision makers proposed that the problem to be studied should be a production planning problem falling within the make classification, as shown in Figure 3.

Once the problem type has been defined, a questionnaire is developed to obtain the weight of preference of criteria and to thus evaluate alternatives according to the criteria. To devise the questionnaire, it is necessary to construct a fuzzy linguistic scale.

Linguistic scales are used to transform linguistic terms into fuzzy numbers [96]. Linguistic terms are subjective categories of the linguistic variable [116]. Zadeh [117] introduced the linguistic variable concept. A linguistic variable is a variable whose values allow computation with words instead of numbers [118]. Linguistic variables are used to represent decision makers’ assessments, estimates and subjectivity [119].

To evaluate the criteria, we use a scale between 0 and 1. To rate the alternatives, we employ a scale from 0 to 10 [97]. The linguistic scales that evaluate the weights of the criteria and alternatives are shown in Table 3.

Table 3. Linguistic scales to assess the criteria and alternatives (Chen [97]).

Linguistic Expression for Rating Alternatives (Algorithms)				Linguistic Variable for the Relative Importance Weight of Criteria			
Linguistic Expression	<i>l</i>	<i>m</i>	<i>u</i>	Linguistic Expression	<i>l</i>	<i>m</i>	<i>u</i>
Very Low (VL)	0.1	0.1	2.5	Very Low Importance (VLI)	0.01	0.03	0.25
Low (L)	0.1	2.5	5.0	Low Importance (LI)	0.01	0.25	0.50
Moderate (M)	2.5	5.0	7.5	Medium Importance (MI)	0.25	0.50	0.75
High (H)	5.0	7.5	10.0	High Importance (HI)	0.50	0.75	1.00
Very High (VH)	7.5	10.0	10.0	Very High Importance (VHI)	0.75	1.00	1.00

Finally, decision makers were invited to review the questionnaire and to check its content. Based on this review, we were able to adjust the questionnaire.

In this same stage, we invited the four decision makers who worked in the planning area to evaluate the alternatives and to determine the weights of the criteria. For this purpose, we asked the decision makers to use the linguistic scale described in Table 3. An extract of the questionnaires used by the decision makers is shown in Tables A1 and A2.

Table 4 details the fuzzy weights of each criterion based on the linguistic scales selected by the decision makers. The decision makers’ ratings of the alternatives against all criteria are shown in Tables A3–A6.

Table 4. Decision makers’ linguistic assessment of the criteria.

	D ₁	D ₂	D ₃	D ₄
C1	MI	MI	MI	MI
C2	LI	LI	LI	LI
C3	VHI	VHI	VHI	VHI
C4	HI	HI	HI	HI
C5	LI	LI	LI	LI
C6	HI	MI	MI	MI
C7	HI	VHI	VHI	MI
C8	LI	MI	HI	HI
C9	LI	MI	HI	HI
C10	MI	MI	LI	HI
C11	LI	LI	LI	LI
C12	VHI	VHI	VHI	VHI
C13	VHI	VHI	VHI	VHI

4.3. Stage 3—Application of the Fuzzy TOPSIS Method

In this stage, the Fuzzy TOPSIS Method is used to analyze the different alternatives in relation to the identified criteria. The process used to apply the Fuzzy TOPSIS Method consists of five steps, which are detailed below.

Step 1. Based on the linguistic assessments of the alternatives (see Tables A3–A6), the linguistic terms are converted into fuzzy numbers according to Table 3 and the fuzzy decision matrix is constructed. The aggregation of the ratings is performed using the fuzzy arithmetic mean, and the aggregate ratings for each alternative are obtained using Equation (10) (see Table 5).

In order to obtain the aggregate weights of each criterion, the fuzzy weights of each criterion are used, which are extracted by converting the linguistic terms of the four decision makers (see Table 4) into fuzzy numbers according to Table 3; for example, the fuzzy weights of criterion C7 of the four decision makers are D₁ = (0.50, 0.75, 1.00), D₂ = (0.75, 1.00, 1.00), D₃ = (0.75, 1.00, 1.00), D₄ = (0.25, 0.50, 0.75), and applying Equation (11), the aggregate fuzzy weight of C7 = (0.56, 0.81, 0.93) is obtained. The results of the aggregate fuzzy weights of all the criteria are tabulated in Table 6.

Table 5. Decision matrix with the aggregated scores.

	C1		C2		C3		C4		C5		C6		C7		C8		C9		C10		C11		C12		C13														
	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u						
A1	0.1	2.5	5.0	1.3	3.1	5.6	0.7	3.1	5.6	0.7	1.9	4.3	1.3	2.5	5.0	2.5	5.0	7.5	4.3	6.8	9.3	0.1	2.5	5.0	0.1	1.9	4.3	0.1	0.1	2.5	1.3	2.5	5.0	1.9	4.3	6.8			
A2	0	0	0	3	5	3	0	3	3	0	3	8	0	5	0	3	0	0	8	8	8	0	0	0	0	0	0	0	0	0	8	0	0	0	0	5	0	3	8
A3	6.8	9.3	10.	4.3	6.8	9.3	4.3	6.8	9.3	2.5	5.0	7.5	2.5	5.0	7.5	3.1	5.6	8.1	4.3	6.8	9.3	1.9	4.3	6.2	3.1	5.6	8.1	1.9	4.3	7.5	0.1	2.5	5.0	2.5	5.0	7.5	3.1	5.6	8.1
A4	8	8	00	8	8	8	8	8	8	8	3	0	0	0	0	0	3	3	3	8	8	8	5	8	5	3	3	3	3	8	0	0	0	0	0	0	0	3	3
A5	6.8	9.3	10.	6.2	8.7	9.3	4.3	6.8	9.3	5.0	7.5	9.3	4.3	6.8	8.7	3.7	6.2	8.1	4.3	6.8	9.3	5.6	8.1	10.	1.9	4.3	6.2	5.0	7.5	10.	7.5	10.	10.	5.0	7.5	10.	6.8	9.3	10.
A6	8	8	00	5	5	8	8	8	8	0	0	8	8	8	5	5	5	5	8	8	8	0	00	00	3	5	8	8	5	00	0	0	0	00	00	0	00	8	8
A7	2.5	5.0	7.5	2.5	5.0	7.5	1.3	3.7	6.2	1.9	4.3	6.8	1.9	4.3	6.8	1.3	3.7	6.2	4.3	6.8	9.3	0.7	3.1	5.6	2.5	5.0	7.5	0.1	0.7	10.	0.1	2.5	5.0	2.5	5.0	7.5	1.9	4.3	6.8
A8	0	0	0	0	8	5	0	0	0	0	8	8	0	8	5	0	8	8	8	8	8	0	3	3	0	0	0	0	0	00	0	0	0	0	0	0	0	0	8
A9	0.7	3.1	5.6	0.1	1.9	4.3	0.1	2.5	5.0	1.9	4.3	6.8	1.9	4.3	6.2	1.9	4.3	6.8	4.3	6.8	9.3	0.1	2.5	5.0	1.9	4.3	6.8	0.1	2.5	10.	0.1	2.5	5.0	2.5	5.0	7.5	1.9	4.3	6.8
A10	0	3	3	0	0	8	0	0	0	0	8	8	0	8	5	0	8	8	8	8	8	0	0	0	0	8	8	0	00	0	0	0	0	0	0	0	0	0	8
A11	4.3	6.8	9.3	5.6	8.1	9.3	3.7	6.2	8.7	3.1	5.6	8.1	3.7	6.2	8.7	2.5	5.0	7.5	4.3	6.8	9.3	3.1	5.6	8.1	4.3	6.8	9.3	5.0	7.5	10.	4.3	6.8	9.3	5.6	8.1	9.3	5.6	8.1	9.3
A12	0	8	8	3	3	8	5	5	5	3	3	3	5	5	5	0	0	0	8	8	8	3	3	3	3	8	8	8	0	00	8	8	3	3	8	3	8	3	8
A13	1	0	0	0	0	0	8	8	0	0	0	8	0	0	0	3	0	0	8	8	8	0	3	3	0	0	0	0	0	0	0	0	5	5	3	5	5	8	8
A14	0.1	1.9	4.3	0.1	0.7	3.1	0.1	2.5	5.0	0.7	2.5	5.0	0.1	1.9	4.3	0.7	3.1	5.6	0.7	3.1	5.6	0.1	2.5	5.0	0.1	1.9	4.3	0.1	2.5	10.	0.1	2.5	5.0	0.1	2.5	5.0	0.1	2.5	5.0
A15	2	0	0	8	0	0	3	0	0	0	3	0	0	0	8	0	3	3	0	3	3	0	0	0	0	0	8	0	00	0	0	0	0	0	0	0	0	0	0
A16	0.1	1.9	4.3	0.1	1.9	4.3	0.1	2.5	5.0	0.7	2.5	5.0	0.1	1.9	4.3	0.7	3.1	5.6	0.7	3.1	5.6	0.1	2.5	5.0	0.1	1.9	4.3	0.1	2.5	10.	0.1	2.5	5.0	0.1	2.5	5.0	0.1	2.5	5.0
A17	3	0	0	8	0	0	8	0	0	0	3	0	0	0	8	0	3	3	0	3	3	0	0	0	0	0	0	0	00	0	0	0	0	0	0	0	0	0	0

Table 6. Aggregate fuzzy weights for each criterion.

Criteria	Aggregate Fuzzy Weights
C1	(0.25, 0.50, 0.75)
C2	(0.01, 0.25, 0.50)
C3	(0.75, 1.00, 1.00)
C4	(0.50, 0.75, 1.00)
C5	(0.01, 0.25, 0.50)
C6	(0.31, 0.56, 0.81)
C7	(0.56, 0.81, 0.93)
C8	(0.32, 0.56, 0.81)
C9	(0.32, 0.56, 0.81)
C10	(0.25, 0.50, 0.75)
C11	(0.01, 0.25, 0.50)
C12	(0.75, 1.00, 1.00)
C13	(0.75, 1.00, 1.00)

Step 2 and Step 3. Using Equations (13) and (14), the normalized fuzzy decision matrix is obtained. For criteria C1–12, Equation (13) is used because the objective of these criteria is to maximize. For criterion C13, Equation (14) is applied because the aim is to minimize the computation time criterion. Table A7 shows the results of the normalized matrix.

After normalization, the weighted normalized decision matrix is calculated using Equation (16). The results are shown in Table A8.

Step 4. It is followed to calculate the FPIS and the FNIS because the positive triangular fuzzy numbers fall within the range [0, 1], and the FPIS and the FNIS are obtained by

Equations (17) and (18). Then, the relative distance is calculated between the algorithms (alternatives) and is computed with Equations (19) and (20) (see Table 7).

Table 7. Distances between alternatives.

	D^+	D^-
A1	6.051	2.149
A2	5.596	2.724
A3	5.192	3.296
A4	5.136	3.370
A5	4.770	3.802
A6	5.595	2.748
A7	5.677	2.786
A8	5.861	2.467
A9	5.766	2.654
A10	4.918	3.597
A11	5.391	3.004
A12	6.176	2.279
A13	6.144	2.323

Step 5. It is followed to determine the closeness coefficient using Equation (21). CC_i The obtained values represent the total score of each algorithm for a production planning problem. Table 8 shows the obtained results.

Table 8. Closeness quotient and algorithms ranking.

Alternative	Algorithm	CC_i	Rank
A1	HA/Benders decomposition	0.262	13
A2	HA/LP and Fix	0.327	8
A3	HA/LP Relaxation	0.388	4
A4	MA/Tabu Search	0.396	3
A5	MA/Genetic Algorithm	0.444	1
A6	MA/Simulated annealing	0.329	6
A7	MA/Variable Neighborhood Search	0.329	7
A8	MTA Genetic Algorithm + Mathematical Model	0.296	10
A9	MTA Simulated annealing + Mathematical Model	0.315	9
A10	CPLEX (Commercial)	0.422	2
A11	CBC (Non-Commercial)	0.358	5
A12	BONMIN (Non-Commercial—Nonlinear)	0.270	12
A13	LINDO (Commercial—Linear/Nonlinear)	0.274	11

When applying the proposed methodological approach based on the Fuzzy TOPSIS Method for a production planning problem, the GA is the most suitable solution method. This finding is not new because the literature review by Guzman et al. [107] concludes that GAs are the most widely used for this problem. Second in the ranking is CPLEX, which is the most widespread solver [107].

5. Sensitivity Analysis

This section evaluates the effects of the different weightings of the criteria, i.e., we aim to evaluate the answers given by decision makers and how they influence algorithm selection. The aim of the sensitivity analysis is to make minor variations in the weights and to observe the influence of this variation on algorithm choice. The weights for the rating of algorithms range from very low importance (VLI) to very high importance (VHI).

Using this rating, we performed an analysis of 10 combinations, where each combination was expressed as an experiment.

The criteria that were evaluated with the highest weight were the type of variables (C3), the quality of solutions (C12) and computation time (C13) (see Table 4). These parameters in a planning problem are dominant when choosing an algorithm. Therefore, we made a minimal variation and looked for criteria with lower scores, such as: problem type (C1), knowledge of algorithms (C10), dataset size (C7), programming knowledge (C8). The details of the experiments are shown in Table 9, where the second column details the changes in the weights of the criteria, and the third column shows the results of the proximity coefficient, while the last column expresses the alternatives ranking.

Table 9. Quantitative results of the sensitivity analysis.

Experiment No.	Changes in Weights of Criteria	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	Alternatives Ranking
E1	C1 = (0.50, 0.75, 1.00)	0.267	0.337	0.402	0.409	0.457	0.338	0.337	0.302	0.321	0.433	0.366	0.274	0.279	A5 > A10 > A4 > A3 > A11 > A6 > A7 > A2 > A9 > A8 > A13 > A12 > A1
E2	C1 = (0.75, 1.00, 1.00)	0.268	0.341	0.410	0.418	0.465	0.341	0.341	0.304	0.323	0.439	0.370	0.275	0.279	A5 > A10 > A4 > A3 > A11 > A6 > A2 > A7 > A9 > A8 > A13 > A12 > A1
E3	C3 = (0.50, 0.75, 1.00)	0.260	0.322	0.383	0.395	0.438	0.327	0.328	0.295	0.314	0.417	0.355	0.268	0.273	A5 > A10 > A4 > A3 > A11 > A7 > A6 > A2 > A9 > A8 > A13 > A12 > A1
E4	C5 = (0.25, 0.50, 0.75)	0.268	0.335	0.398	0.408	0.455	0.338	0.337	0.304	0.323	0.434	0.367	0.274	0.279	A5 > A10 > A4 > A3 > A11 > A6 > A7 > A2 > A9 > A8 > A13 > A12 > A1
E5	C10 = (0.50, 0.75, 1.00)	0.266	0.336	0.396	0.406	0.455	0.335	0.336	0.301	0.323	0.434	0.366	0.278	0.282	A5 > A10 > A4 > A3 > A11 > A2 > A7 > A6 > A9 > A8 > A13 > A12 > A1
E6	C10 = (0.75, 1.00, 1.00)	0.267	0.339	0.399	0.410	0.461	0.338	0.336	0.302	0.324	0.440	0.370	0.278	0.283	A5 > A10 > A4 > A3 > A11 > A2 > A6 > A7 > A9 > A8 > A13 > A12 > A1
E7	C8 = (0.01, 0.25, 0.50), C12 = (0.50, 0.75, 1.00)	0.254	0.317	0.376	0.370	0.422	0.319	0.319	0.287	0.306	0.404	0.346	0.262	0.267	A5 > A10 > A3 > A4 > A11 > A6 > A7 > A2 > A9 > A8 > A13 > A12 > A1
E8	C10 = (0.75, 1.00, 1.00), C11 = (0.50, 0.75, 1.00)	0.268	0.342	0.402	0.411	0.473	0.342	0.339	0.305	0.326	0.448	0.375	0.280	0.285	A5 > A10 > A4 > A3 > A11 > A6 > A2 > A7 > A9 > A8 > A13 > A12 > A1
E9	C7 = (0.50, 0.75, 1.00)	0.262	0.327	0.388	0.396	0.443	0.329	0.329	0.296	0.315	0.422	0.358	0.270	0.275	A5 > A10 > A4 > A3 > A11 > A6 > A7 > A2 > A9 > A8 > A13 > A12 > A1
E10	C7 = (0.75, 1.00, 1.00)	0.268	0.333	0.394	0.402	0.449	0.335	0.335	0.302	0.321	0.428	0.364	0.272	0.277	A5 > A10 > A4 > A3 > A11 > A6 > A7 > A2 > A9 > A8 > A13 > A12 > A1

The sensitivity analysis shows that alternatives A5 (GA), A10 (CPLEX), A4 (Tabu Search) have the best scores and occupy the first three positions. Hence, the variation in the weights in the chosen criteria minimally affects these alternatives; for example, A5 reaches the first position in all the experiments. The main variations occur in the sixth, seventh and eighth positions with alternatives A2, A6 and A7. However, the last ranking positions remain unchanged in the classification. In this context, decision makers can use these variations or make other modifications to weightings to prioritize a criterion and to thus facilitate the evaluation process in decision making.

6. Conclusions

The complexity of real-world problems should be seen not only as an obstacle, but also as a research challenge for effective solutions for large-scale planning problems. Relatively small companies often face very complex problems.

It is usually very difficult for production planners in companies to determine or choose an algorithm. The algorithm selection process normally involves the experimental evaluation of several algorithms with different dataset sizes. However, these sets of experiments require considerable computational resources and long processing times. This adds to the disadvantage of having fewer resources to invest in commercial solvers. In addition, efforts often have to be duplicated when attempting to replicate the algorithms or models available in the literature.

To overcome these drawbacks, the methodological approach based on the fuzzy TOPSIS proposed herein intends to be a support tool to select a solution method for replenishment, production and distribution planning problems. To this end, 13 different criteria were defined and used to select nine different algorithm types (heuristic, metaheuristic, and matheuristic) and four solvers (commercial and non-commercial) that are often employed in planning problems. All these criteria address several important dimensions when solving a planning problem. These dimensions are related to the computational difficulty of the planning problem, programming skills, mathematical skills, algorithmic skills, mathematical modeling software skills, and also to the expected computational performance of the solution methods. These criteria were analyzed based on the linguistic values given by four planning experts from different manufacturing companies. The problem selected to apply the proposed approach was that of production planning. For this problem, the results of the methodology showed that the GA was the best alternative, while Benders' decomposition was the worst. Given our study results, it can be concluded that it is possible to select a set of suitable candidate algorithms for solving optimization problems with the proposed approach. In this way, not only can one algorithm be selected, but so can other algorithms that provide similar solutions at the same time. The results of this methodology can guide companies to choose whether to use a commercial or non-commercial algorithm or solver. This can help companies to determine whether they should invest in a solver or use mathematical modeling or algorithm programming software and, at the same time, to understand planning staff's training needs.

There are different approaches for algorithm selection [44,70,75,79]. These approaches are heuristic, metaheuristic and AI, and they offer benefits and disadvantages. However, these techniques can be restrictive for companies because they involve a large number of computational resources and experiments that can be affected by accuracy, the number of tested instances, instance generation, consistency, AI techniques, and training time. The proposed approach requires very few resources, is very useful thanks to its simplicity and is easily replicable. The main limitation of this technique is the appropriate selection of criteria and the balance between them, which is a subjective issue that requires experts in the planning problems field, not to mention the personal bias of experts' opinions.

Future research could be conducted to experiment the proposed approach with the portfolio of algorithms and solvers defined in [107], where some 50 algorithms are identified, including optimizing, heuristic, metaheuristic and matheuristic algorithms, as well as different types of commercial solvers. Alternatives and criteria could be evaluated with more decision makers. Other MCDM techniques such as ELECTRE, PROMETHEE, intuitionistic fuzzy TOPSIS, or novel methods such as the performance calculation technique of the integrated multiple multi-attribute decision making (PCIM-MADM) [120], which incorporates four techniques (COPRAS, GRA, SAW and VIKOR) into a single final classification index, could be used.

Author Contributions: Conceptualization, E.G., B.A. and R.P.; methodology, E.G., B.A. and R.P.; software, E.G., B.A. and R.P.; validation, E.G., B.A. and R.P.; writing—review and editing, E.G., B.A. and R.P.; supervision B.A. and R.P. All authors have read and agreed to the published version of the manuscript.

Funding: The research leading to these results received funding from the European Union H2020 Programme with grant agreements No. 825631 “Zero-Defect Manufacturing Platform (ZDMP)” and No. 958205 “Industrial Data Services for Quality Control in Smart Manufacturing (i4Q)” and from the Regional Department of Innovation, Universities, Science and Digital Society of the Generalitat Valenciana with Ref. PROMETEO/2021/065 “Industrial Production and Logistics Optimization in Industry 4.0” (i4OPT).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data are presented in the main text.

Acknowledgments: This work was supported by the Conselleria de Educació, Investigació, Cultura y Deporte—Generalitat Valenciana for hiring predoctoral research staff with Grant (ACIF/2018/170) and the European Social Fund with the Grant Operational Programme of FSE 2014-2020, the Valencian Community (Spain). Funding for open access charge: Universitat Politècnica de València.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1 shows a section of the questionnaire format used by decision makers to evaluate the algorithm selection criteria. Table A2 presents the questionnaire used to score the chosen alternatives, i.e., the selected algorithms and solvers against the 13 identified criteria.

Table A1. Questionnaire used to know decision makers’ preferences for the identified criteria.

	Very Low Importance (VLI)	Low Importance (LI)	Medium Importance (MI)	High Importance (HI)	Very High Importance (VHI)
C1					
C2					
C3					
C4					
C5					
C6					
C7					
C8					
C9					
C10					
C11					
C12					
C13					

Table A2. Questionnaire used to know the decision makers’ preferences for the 13 alternatives according to the criteria.

	C1				
	Very Low (VL)	Low (L)	Moderate (M)	High (H)	Very High (VH)
A1					
A2					
A3					
A4					

A5
A6
A7
A8
A9
A10
A11
A12
A13

Tables A3–A6 show the decision makers’ alternatives ratings against all the criteria

Table A3. Decision maker 1’s linguistic assessment.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A1	L	H	M	M	M	M	M	L	L	L	VL	M	H
A2	M	H	H	M	M	M	M	L	VL	L	L	M	H
A3	VH	H	H	M	M	M	M	L	M	L	L	M	H
A4	VH	H	VL	M	M	M	M	VH	VL	H	VL	VH	H
A5	VH	VH	H	M	M	M	M	H	L	H	VH	H	VH
A6	M	M	L	M	M	M	M	L	M	L	L	M	L
A7	M	M	L	M	M	M	M	L	M	VL	L	M	M
A8	L	L	L	M	M	M	M	L	L	L	L	M	M
A9	L	L	L	M	M	M	M	L	M	L	L	M	M
A10	H	VH	H	M	H	M	M	M	H	H	H	VH	VH
A11	M	M	M	M	M	M	M	L	M	M	M	M	VH
A12	L	L	L	M	L	M	M	L	L	L	L	L	L

Table A4. Decision maker 2’s linguistic assessment.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A1	L	L	L	L	VL	L	H	L	L	VL	VL	VL	M
A2	H	H	H	L	M	L	H	L	VL	M	L	VL	M
A3	VH	H	H	H	M	M	H	L	M	M	L	M	M
A4	VH	H	VL	H	M	M	H	VH	VL	H	VL	VH	H
A5	VH	VH	H	H	M	M	H	H	L	H	VH	H	VH
A6	M	M	L	M	M	L	H	L	M	L	L	M	L
A7	M	M	L	M	VL	L	H	L	M	VL	L	M	M
A8	L	L	L	M	VL	L	H	L	L	L	L	M	M
A9	L	L	L	M	VL	L	H	L	M	L	L	M	M
A10	H	VH	H	M	H	M	H	M	H	VH	H	VH	VH
A11	M	M	M	M	M	L	H	L	M	M	M	M	VH
A12	L	VL	L	VL	VL	L	L	L	L	L	L	L	L

Table A5. Decision maker 3’s linguistic assessment.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A1	L	VL	L	VL	VL	M	H	L	L	L	VL	VL	L
A2	L	M	L	L	VL	M	H	H	H	H	L	VL	M
A3	H	M	M	L	M	M	H	VH	H	H	L	M	M
A4	H	M	M	VL	VH	H	H	VH	H	VL	VL	VH	H
A5	H	M	M	VH	H	VH	H	VH	VH	H	VH	H	VH
A6	M	M	L	L	M	L	H	M	M	H	H	M	H
A7	M	VL	L	L	M	M	H	M	M	L	L	M	M
A8	M	VL	L	L	M	M	H	L	L	L	L	M	M
A9	M	VL	L	L	M	M	H	L	L	L	L	M	M
A10	M	H	M	H	M	M	H	H	M	M	M	M	M
A11	M	M	L	M	M	M	H	M	M	M	M	M	M

Table A8. Weighted normalized fuzzy decision matrix.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13																											
	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u	l	m	u																
A1	0.0	0.1	0.3	0.0	0.0	0.3	0.0	0.3	0.6	0.0	0.1	0.4	0.0	0.0	0.2	0.0	0.3	0.7	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.1	0.4	0.0	0.1	0.3	0.0	0.0	0.1	0.1	0.2	0.5	0.0	0.0	0.0	
	0	3	8	0	8	0	6	3	0	4	5	7	0	7	9	9	2	0	6	0	4	0	4	1	0	5	3	0	0	3	0	0	3	0	6	0	1	2	5	
A2	0.0	0.2	0.6	0.0	0.1	0.5	0.3	0.6	0.9	0.0	0.2	0.5	0.0	0.1	0.3	0.0	0.3	0.7	0.2	0.6	0.9	0.0	0.2	0.5	0.0	0.1	0.3	0.0	0.2	0.7	0.0	0.0	0.2	0.1	0.2	0.5	0.0	0.0	0.0	
	8	8	1	0	8	0	0	7	3	4	0	3	0	1	6	9	2	0	6	0	4	4	1	1	4	2	8	5	2	5	0	6	5	0	6	0	1	2	4	
A3	0.1	0.4	0.7	0.0	0.1	0.5	0.3	0.7	1.0	0.1	0.4	0.8	0.0	0.1	0.4	0.1	0.3	0.7	0.2	0.6	0.9	0.0	0.2	0.5	0.1	0.3	0.7	0.0	0.2	0.5	0.0	0.0	0.2	0.1	0.5	0.7	0.0	0.0	0.0	
	7	7	5	0	8	0	5	3	0	3	0	0	0	4	3	1	6	5	6	0	4	6	5	1	1	4	0	5	2	6	0	6	5	9	0	5	1	2	3	
A4	0.1	0.4	0.7	0.0	0.1	0.5	0.0	0.1	0.4	0.2	0.4	0.8	0.0	0.2	0.5	0.1	0.4	0.8	0.2	0.6	0.9	0.2	0.5	0.8	0.0	0.1	0.3	0.1	0.2	0.7	0.0	0.0	0.1	0.5	1.0	1.0	0.0	0.0	0.0	
	7	7	5	0	8	0	6	4	0	0	5	0	1	0	0	3	0	1	6	0	4	4	6	1	4	2	8	0	8	5	0	0	3	6	0	0	1	1	2	
A5	0.1	0.4	0.7	0.0	0.2	0.5	0.3	0.7	1.0	0.2	0.6	1.0	0.0	0.2	0.5	0.1	0.4	0.7	0.2	0.6	0.9	0.1	0.4	0.8	0.0	0.2	0.5	0.1	0.3	0.7	0.0	0.2	0.5	0.3	0.7	1.0	0.0	0.0	0.0	
	7	7	5	1	3	0	5	3	0	7	0	1	0	0	3	0	5	6	0	4	8	6	1	7	6	4	3	8	5	1	5	0	8	5	0	1	1	1		
A6	0.0	0.2	0.5	0.0	0.1	0.4	0.1	0.4	0.6	0.1	0.3	0.7	0.0	0.1	0.3	0.0	0.2	0.5	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.3	0.6	0.0	0.1	0.3	0.0	0.0	0.3	0.1	0.5	0.7	0.0	0.0	0.0	
	6	5	6	0	3	0	1	0	7	0	5	3	0	3	9	5	4	8	6	0	4	2	8	6	8	0	5	3	9	8	0	9	1	9	0	5	1	2	4	
A7	0.0	0.2	0.5	0.0	0.1	0.3	0.0	0.2	0.5	0.1	0.3	0.7	0.0	0.1	0.3	0.0	0.2	0.6	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.3	0.6	0.0	0.0	0.7	0.0	0.0	0.2	0.1	0.5	0.7	0.0	0.0	0.0	
	6	5	6	0	0	3	1	7	3	0	5	3	0	1	6	7	8	4	6	0	4	2	8	6	8	0	5	0	4	5	0	6	5	9	0	5	1	2	5	
A8	0.0	0.1	0.4	0.0	0.0	0.2	0.0	0.2	0.5	0.1	0.3	0.7	0.0	0.1	0.3	0.0	0.2	0.6	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.1	0.4	0.0	0.1	0.3	0.0	0.0	0.2	0.1	0.5	0.7	0.0	0.0	0.0	
	2	6	2	0	5	3	1	7	3	0	5	3	0	1	6	7	8	4	6	0	4	0	4	1	0	5	3	0	3	8	0	6	5	9	0	5	1	2	5	
A9	0.0	0.1	0.4	0.0	0.0	0.2	0.0	0.2	0.5	0.1	0.3	0.7	0.0	0.1	0.3	0.0	0.2	0.6	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.2	0.6	0.0	0.1	0.7	0.0	0.0	0.2	0.1	0.5	0.7	0.0	0.0	0.0	
	2	6	2	0	5	3	1	7	3	0	5	3	0	1	6	7	8	4	6	0	4	0	4	1	6	6	0	0	3	5	0	6	5	9	0	5	1	2	5	
A10	0.1	0.3	0.7	0.0	0.2	0.5	0.3	0.6	0.9	0.1	0.4	0.8	0.0	0.1	0.5	0.0	0.3	0.7	0.2	0.6	0.9	0.1	0.3	0.6	0.1	0.4	0.8	0.1	0.3	0.7	0.0	0.1	0.4	0.4	0.8	0.9	0.0	0.0	0.0	
	0	1	4	0	1	2	0	0	7	3	7	5	7	0	8	0	9	2	0	6	0	4	0	2	6	5	1	1	3	8	5	0	7	7	2	1	4	1	1	2
A11	0.0	0.2	0.5	0.0	0.1	0.4	0.1	0.4	0.7	0.1	0.4	0.8	0.0	0.1	0.4	0.0	0.3	0.7	0.2	0.6	0.9	0.0	0.1	0.4	0.0	0.3	0.6	0.0	0.2	0.5	0.0	0.1	0.3	0.2	0.6	0.8	0.0	0.0	0.0	
	1	6	5	6	0	3	0	5	7	3	3	0	0	4	3	9	2	0	6	0	4	2	8	6	8	0	5	6	5	6	0	3	8	8	3	1	1	1	2	
A12	0.0	0.1	0.3	0.0	0.0	0.1	0.0	0.2	0.5	0.0	0.2	0.5	0.0	0.0	0.2	0.0	0.2	0.5	0.0	0.2	0.5	0.0	0.1	0.4	0.0	0.1	0.3	0.0	0.1	0.7	0.0	0.0	0.2	0.0	0.2	0.5	0.0	0.0	1.0	
	2	0	0	3	0	2	7	1	7	3	4	0	3	0	5	5	3	0	2	4	7	6	0	4	1	0	1	8	0	3	5	0	6	5	1	5	0	2	4	0
A13	0.0	0.1	0.3	0.0	0.0	0.2	0.0	0.2	0.5	0.0	0.2	0.5	0.0	0.0	0.2	0.0	0.2	0.5	0.0	0.2	0.5	0.0	0.1	0.4	0.0	0.1	0.4	0.0	0.1	0.7	0.0	0.0	0.2	0.0	0.2	0.5	0.0	0.0	1.0	
	3	0	0	3	0	5	3	1	7	3	4	0	3	0	5	5	3	0	2	4	7	6	0	4	1	0	5	3	0	3	5	0	6	5	1	5	0	2	4	0

References

1. Wang, C.; Liu, X.-B. Integrated production planning and control: A multi-objective optimization model. *J. Ind. Eng. Manag.* **2013**, *6*, 815–830. <https://doi.org/10.3926/jiem.771>.
2. Wang, Z.; Zhen, H.-L.; Deng, J.; Zhang, Q.; Li, X.; Yuan, M.; Zeng, J. Multiobjective Optimization-Aided Decision-Making System for Large-Scale Manufacturing Planning. *IEEE Trans. Cybern.* **2021**, *in press*. <https://doi.org/10.1109/tcyb.2021.3049712>.
3. Hartmut, S.; Kilger, C.; Herbert, M. Supply Chain Management and Advanced Planning: Concepts, Models, Software and Case Studies; Springer: Berlin/Heidelberg, Germany, 2015.
4. Crainic, T.G.; Fomeni, F.D.; Rei, W. Multi-period bin packing model and effective constructive heuristics for corridor-based logistics capacity planning. *Comput. Oper. Res.* **2019**, *132*, 105308. <https://doi.org/10.1016/j.cor.2021.105308>.
5. Pratap, S.; Jauhar, S.K.; Paul, S.K.; Zhou, F. Stochastic optimization approach for green routing and planning in perishable food production. *J. Clean. Prod.* **2021**, *333*, 130063. <https://doi.org/10.1016/j.jclepro.2021.130063>.
6. Zarouk, Y.; Mahdavi, I.; Rezaeian, J.; Santos-Arteaga, F.J. A novel multi-objective green vehicle routing and scheduling model with stochastic demand, supply, and variable travel times. *Comput. Oper. Res.* **2022**, *141*, 105698. <https://doi.org/10.1016/j.cor.2022.105698>.
7. Rossi, A.; Lanzetta, M. Integration of hybrid additive/subtractive manufacturing planning and scheduling by metaheuristics. *Comput. Ind. Eng.* **2020**, *144*, 106428. <https://doi.org/10.1016/j.cie.2020.106428>.
8. Mirabelli, G.; Solina, V. Optimization strategies for the integrated management of perishable supply chains: A literature review. *J. Ind. Eng. Manag.* **2022**, *15*, 58–91. <https://doi.org/10.3926/jiem.3603>.
9. Vanajakumari, M.; Sun, H.; Jones, A.; Sriskandarajah, C. Supply chain planning: A case for Hybrid Cross-Docks. *Omega* **2022**, *108*, 102585. <https://doi.org/10.1016/j.omega.2021.102585>.
10. Juan, A.A.; Keenan, P.; Martí, R.; McGarraghy, S.; Panadero, J.; Carroll, P.; Oliva, D. A review of the role of heuristics in stochastic optimisation: From metaheuristics to learnheuristics. *Ann. Oper. Res.* **2021**, Epub ahead of printing. <https://doi.org/10.1007/s10479-021-04142-9>.
11. Stadtler, H. Supply chain management and advanced planning—Basics, overview and challenges. *Eur. J. Oper. Res.* **2005**, *163*, 575–588. <https://doi.org/10.1016/j.ejor.2004.03.001>.
12. Weise, T.; Zapf, M.; Chiong, R.; Nebro, A.J. Why is optimization difficult? In *Nature-Inspired Algorithms for Optimisation*; Chiong, R., Ed.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 1–50.
13. Michalewicz, Z.; Fogel, D.B. *How to Solve It: Modern Heuristics*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2004.
14. Lohmer, J.; Lasch, R. Production planning and scheduling in multi-factory production networks: A systematic literature review. *Int. J. Prod. Res.* **2021**, *59*, 2028–2054. <https://doi.org/10.1080/00207543.2020.1797207>.

15. Adulyasak, Y.; Cordeau, J.-F.; Jans, R. The production routing problem: A review of formulations and solution algorithms. *Comput. Oper. Res.* **2015**, *55*, 141–152. <https://doi.org/10.1016/j.cor.2014.01.011>.
16. Díaz-Madroñero, M.; Mula, J.; Peidro, D. A review of discrete-time optimization models for tactical production planning. *Int. J. Prod. Res.* **2014**, *52*, 5171–5205. <https://doi.org/10.1080/00207543.2014.899721>.
17. Mula, J.; Peidro, D.; Díaz-Madroñero, M.; Vicens, E. Mathematical programming models for supply chain production and transport planning. *Eur. J. Oper. Res.* **2010**, *204*, 377–390. <https://doi.org/10.1016/j.ejor.2009.09.008>.
18. Peres, F.; Castelli, M. Combinatorial Optimization Problems and Metaheuristics: Review, Challenges, Design, and Development. *Appl. Sci.* **2021**, *11*, 6449. <https://doi.org/10.3390/app11146449>.
19. Huerta, I.I.; Neira, D.A.; Ortega, D.A.; Varas, V.; Godoy, J.; Asín-Achá, R. Anytime automatic algorithm selection for knapsack. *Expert Syst. Appl.* **2020**, *158*, 113613. <https://doi.org/10.1016/j.eswa.2020.113613>.
20. Tezel, B.T.; Mert, A. A cooperative system for metaheuristic algorithms. *Expert Syst. Appl.* **2020**, *165*, 113976. <https://doi.org/10.1016/j.eswa.2020.113976>.
21. de Carvalho, V.R.; Özcan, E.; Sichman, J.S. Comparative Analysis of Selection Hyper-Heuristics for Real-World Multi-Objective Optimization Problems. *Appl. Sci.* **2021**, *11*, 9153. <https://doi.org/10.3390/app11199153>.
22. Peng, Y.; Kou, G.; Wang, G.; Shi, Y. FAMCDM: A fusion approach of MCDM methods to rank multiclass classification algorithms. *Omega* **2011**, *39*, 677–689. <https://doi.org/10.1016/j.omega.2011.01.009>.
23. Grąbczewski, K. Using result profiles to drive meta-learning. In Proceedings of the European, Mediterranean, and Middle Eastern Conference on Information Systems, Online, 8–9 December 2021; pp. 69–83. https://doi.org/10.1007/978-3-030-95947-0_6.
24. Smith-Miles, K.; Lopes, L. Measuring instance difficulty for combinatorial optimization problems. *Comput. Oper. Res.* **2012**, *39*, 875–889. <https://doi.org/10.1016/j.cor.2011.07.006>.
25. Jamalnia, A.; Yang, J.-B.; Feili, A.; Xu, D.-L.; Jamali, G. Aggregate production planning under uncertainty: A comprehensive literature survey and future research directions. *Int. J. Adv. Manuf. Technol.* **2019**, *102*, 159–181. <https://doi.org/10.1007/s00170-018-3151-y>.
26. Kumar, R.; Ganapathy, L.; Gokhale, R.; Tiwari, M.K. Quantitative approaches for the integration of production and distribution planning in the supply chain: A systematic literature review. *Int. J. Prod. Res.* **2020**, *58*, 3527–3553. <https://doi.org/10.1080/00207543.2020.1762019>.
27. Pereira, D.F.; Oliveira, J.F.; Carravilla, M.A. Tactical sales and operations planning: A holistic framework and a literature review of decision-making models. *Int. J. Prod. Econ.* **2019**, *228*, 107695. <https://doi.org/10.1016/j.ijpe.2020.107695>.
28. Hussain, K.; Salleh, M.N.M.; Cheng, S.; Shi, Y. Metaheuristic research: A comprehensive survey. *Artif. Intell. Rev.* **2019**, *52*, 2191–2233. <https://doi.org/10.1007/s10462-017-9605-z>.
29. Hooker, J.N. Testing heuristics: We have it all wrong. *J. Heuristics* **1995**, *1*, 33–42. <https://doi.org/10.1007/bf02430364>.
30. Angel, E.; Zissimopoulos, V. On the Hardness of the Quadratic Assignment Problem with Metaheuristics. *J. Heuristics* **2002**, *8*, 399–414. <https://doi.org/10.1023/a:1015454612213>.
31. Seo, D.-I.; Moon, B.-R. An Information-Theoretic Analysis on the Interactions of Variables in Combinatorial Optimization Problems. *Evol. Comput.* **2007**, *15*, 169–198. <https://doi.org/10.1162/evco.2007.15.2.169>.
32. Bansal, S. Performance comparison of five metaheuristic nature-inspired algorithms to find near-OGRs for WDM systems. *Artif. Intell. Rev.* **2020**, *53*, 5589–5635. <https://doi.org/10.1007/s10462-020-09829-2>.
33. Silberholz, J.; Golden, B.; Gupta, S.; Wang, X. Computational Comparison of Metaheuristics. In *Handbook of Metaheuristics*; Gendreau, M., Potvin, J.-Y., Eds.; Springer: Cham, Switzerland, 2019; pp. 581–604.
34. Beasley, J. OR-Library. Available online: <http://people.brunel.ac.uk/~mastjb/jeb/info.html> (accessed 5 March 2022).
35. Bischl, B.; Kerschke, P.; Kotthoff, L.; Lindauer, M.; Malitsky, Y.; Fréchet, A.; Hoos, H.; Hutter, F.; Leyton-Brown, K.; Tierney, K.; et al. ASlib: A benchmark library for algorithm selection. *Artif. Intell.* **2016**, *237*, 41–58. <https://doi.org/10.1016/j.artint.2016.04.003>.
36. Pan, Q.-K.; Gao, L.; Wang, L.; Liang, J.; Li, X.-Y. Effective heuristics and metaheuristics to minimize total flowtime for the distributed permutation flowshop problem. *Expert Syst. Appl.* **2019**, *124*, 309–324. <https://doi.org/10.1016/j.eswa.2019.01.062>.
37. Johnson, D.S. Approximation algorithms for combinatorial problems. *J. Comput. Syst. Sci.* **1974**, *9*, 256–278. [https://doi.org/10.1016/s0022-0000\(74\)80044-9](https://doi.org/10.1016/s0022-0000(74)80044-9).
38. Maniezzo, V.; Boschetti, M.A.; Stützle, T. *Mathheuristics*; Springer: Cham, Switzerland, 2014.
39. de Armas, J.; Lalla-Ruiz, E.; Tilahun, S.L.; Voß, S. Similarity in metaheuristics: A gentle step towards a comparison methodology. *Nat. Comput.* **2021**, *9*, 1–23. <https://doi.org/10.1007/s11047-020-09837-9>.
40. Rahman, A.; Sokkalingam, R.; Othman, M.; Biswas, K.; Abdullah, L.; Kadir, E.A. Nature-Inspired Metaheuristic Techniques for Combinatorial Optimization Problems: Overview and Recent Advances. *Mathematics* **2021**, *9*, 2633. <https://doi.org/10.3390/math9202633>.
41. Douek-Pinkovich, Y.; Ben-Gal, I.; Raviv, T. The stochastic test collection problem: Models, exact and heuristic solution approaches. *Eur. J. Oper. Res.* **2022**, *299*, 945–959. <https://doi.org/10.1016/j.ejor.2021.12.043>.
42. Silva, A.F.; Valente, J.M.; Schaller, J.E. Metaheuristics for the permutation flowshop problem with a weighted quadratic tardiness objective. *Comput. Oper. Res.* **2022**, *140*, 105691. <https://doi.org/10.1016/j.cor.2021.105691>.
43. Tarhan, I.; Oğuz, C. A metaheuristic for the generalized order acceptance and scheduling problem. *Eur. J. Oper. Res.* **2022**, *299*, 87–103. <https://doi.org/10.1016/j.ejor.2021.08.024>.

44. Kerschke, P.; Hoos, H.H.; Neumann, F.; Trautmann, H. Automated Algorithm Selection: Survey and Perspectives. *Evol. Comput.* **2019**, *27*, 3–45. https://doi.org/10.1162/evco_a_00242.
45. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. <https://doi.org/10.1109/4235.585893>.
46. Mostert, W.; Malan, K.; Engelbrecht, A. A Feature Selection Algorithm Performance Metric for Comparative Analysis. *Algorithms* **2021**, *14*, 100. <https://doi.org/10.3390/a14030100>.
47. Salih, M.; Zaidan, B.; Zaidan, A.; Ahmed, M. Survey on fuzzy TOPSIS state-of-the-art between 2007 and 2017. *Comput. Oper. Res.* **2018**, *104*, 207–227. <https://doi.org/10.1016/j.cor.2018.12.019>.
48. Zanakis, S.H.; Solomon, A.; Wishart, N.; Dublisch, S. Multi-attribute decision making: A simulation comparison of select methods. *Eur. J. Oper. Res.* **1998**, *107*, 507–529. [https://doi.org/10.1016/s0377-2217\(97\)00147-1](https://doi.org/10.1016/s0377-2217(97)00147-1).
49. Opricovic, S.; Tzeng, G.-H. Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. *Eur. J. Oper. Res.* **2004**, *156*, 445–455. [https://doi.org/10.1016/s0377-2217\(03\)00020-1](https://doi.org/10.1016/s0377-2217(03)00020-1).
50. Opricovic, S.; Tzeng, G.-H. Extended VIKOR method in comparison with outranking methods. *Eur. J. Oper. Res.* **2007**, *178*, 514–529. <https://doi.org/10.1016/j.ejor.2006.01.020>.
51. Chu, M.-T.; Shyu, J.; Tzeng, G.-H.; Khosla, R. Comparison among three analytical methods for knowledge communities group decision analysis. *Expert Syst. Appl.* **2007**, *33*, 1011–1024. <https://doi.org/10.1016/j.eswa.2006.08.026>.
52. Özcan, T.; Çelebi, N.; Esnaf, Ş. Comparative analysis of multi-criteria decision making methodologies and implementation of a warehouse location selection problem. *Expert Syst. Appl.* **2011**, *38*, 9773–9779. <https://doi.org/10.1016/j.eswa.2011.02.022>.
53. Ertuğrul, I.; Karakaşoğlu, N. Comparison of fuzzy AHP and fuzzy TOPSIS methods for facility location selection. *Int. J. Adv. Manuf. Technol.* **2008**, *39*, 783–795. <https://doi.org/10.1007/s00170-007-1249-8>.
54. Atanassov, K. Intuitionistic fuzzy sets. *Fuzzy Sets Syst.* **1986**, *20*, 87–96. [https://doi.org/10.1016/s0165-0114\(86\)80034-3](https://doi.org/10.1016/s0165-0114(86)80034-3).
55. Boran, F.E. An Integrated Intuitionistic Fuzzy Multi Criteria Decision Making Method for Facility Location Selection. *Math. Comput. Appl.* **2011**, *16*, 487–496. <https://doi.org/10.3390/mca16020487>.
56. Boran, F.E.; Genç, S.; Kurt, M.; Akay, D. A multi-criteria intuitionistic fuzzy group decision making for supplier selection with TOPSIS method. *Expert Syst. Appl.* **2009**, *36*, 11363–11368. <https://doi.org/10.1016/j.eswa.2009.03.039>.
57. Gerogiannis, V.C.; Fitsilis, P.; Kameas, A.D. Using a combined intuitionistic fuzzy set-TOPSIS method for evaluating project and portfolio management information systems. In *Artificial Intelligence Applications and Innovations*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 67–81.
58. Gerogiannis, V.C.; Fitsilis, P.; Kameas, A.D. Evaluation of project and portfolio Management Information Systems with the use of a hybrid IFS-TOPSIS method. *Intell. Decis. Technol.* **2013**, *7*, 91–105. <https://doi.org/10.3233/IDT-120153>.
59. Shan, L. Research on vendor selection based on intuitionistic fuzzy sets. *Adv. Intell. Soft Comput.* **2011**, *110*, 645–652. https://doi.org/10.1007/978-3-642-25185-6_82.
60. Büyüközkan, G.; Güteryüz, S. Multi Criteria Group Decision Making Approach for Smart Phone Selection Using Intuitionistic Fuzzy TOPSIS. *Int. J. Comput. Intell. Syst.* **2016**, *9*, 709–725. <https://doi.org/10.1080/18756891.2016.1204119>.
61. Jato-Espino, D.; Castillo-Lopez, E.; Rodriguez-Hernandez, J.; Canteras-Jordana, J.C. A review of application of multi-criteria decision making methods in construction. *Autom. Constr.* **2014**, *45*, 151–162. <https://doi.org/10.1016/j.autcon.2014.05.013>.
62. Velasquez, M.; Hester, P. An analysis of multi-criteria decision making methods. *Int. J. Oper. Res.* **2013**, *10*, 56–66.
63. Lamba, M.; Munjal, G.; Gigras, Y. ECABC: Evaluation of classification algorithms in breast cancer for imbalanced datasets. In *Data Driven Approach Towards Disruptive Technologies*; Springer: Singapore, 2021; pp. 379–388.
64. Peng, Y.; Wang, G.; Kou, G.; Shi, Y. An empirical study of classification algorithm evaluation for financial risk prediction. *Appl. Soft Comput.* **2011**, *11*, 2906–2915. <https://doi.org/10.1016/j.asoc.2010.11.028>.
65. Behzadian, M.; Otaghsara, S.K.; Yazdani, M.; Ignatius, J. A state-of-the-art survey of TOPSIS applications. *Expert Syst. Appl.* **2012**, *39*, 13051–13069. <https://doi.org/10.1016/j.eswa.2012.05.056>.
66. Jigeesh, N.; Joseph, D.; Yadav, S.K. A review on industrial applications of TOPSIS approach. *Int. J. Serv. Oper. Manag.* **2018**, *30*, 23. <https://doi.org/10.1504/ijsum.2018.10012402>.
67. Palczewski, K.; Sałabun, W. The fuzzy TOPSIS applications in the last decade. *Procedia Comput. Sci.* **2019**, *159*, 2294–2303. <https://doi.org/10.1016/j.procs.2019.09.404>.
68. Choudhary, D.; Shankar, R. An STEEP-fuzzy AHP-TOPSIS framework for evaluation and selection of thermal power plant location: A case study from India. *Energy* **2012**, *42*, 510–521. <https://doi.org/10.1016/j.energy.2012.03.010>.
69. Saleem, S.; Gallagher, M. Using regression models for characterizing and comparing black box optimization problems. *Swarm Evol. Comput.* **2021**, *68*, 100981. <https://doi.org/10.1016/j.swevo.2021.100981>.
70. Muñoz, M.A.; Sun, Y.; Kirley, M.; Halgamuge, S. Algorithm selection for black-box continuous optimization problems: A survey on methods and challenges. *Inf. Sci.* **2015**, *317*, 224–245. <https://doi.org/10.1016/j.ins.2015.05.010>.
71. Raschka, S. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. *arXiv* **2018**. <https://arxiv.org/abs/1811.12808>.
72. Parmezan, A.R.S.; Lee, H.D.; Spolaôr, N.; Wu, F.C. Automatic recommendation of feature selection algorithms based on dataset characteristics. *Expert Syst. Appl.* **2021**, *185*, 115589. <https://doi.org/10.1016/j.eswa.2021.115589>.
73. Stützle, T.; Fernandes, S. New Benchmark Instances for the QAP and the Experimental Analysis of Algorithms. In *Evolutionary Computation in Combinatorial Optimization*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 199–209.

74. Gomes, C.P.; Selman, B. Algorithm Portfolio Design: Theory vs. Practice. *arXiv* **2013**, arXiv:1302.1541. <https://arxiv.org/abs/1302.1541>.
75. Kotthoff, L. Algorithm Selection for Combinatorial Search Problems: A Survey. In *Data Mining and Constraint Programming: Foundations of a Cross-Disciplinary Approach*; Bessiere, C., de Raedt, L., Kotthoff, L., Nijssen, S., O'Sullivan, B., Pedreschi, D., Eds. Springer: Cham, Switzerland, 2016; pp. 149–190.
76. Leyton-Brown, K.; Nudelman, E.; Andrew, G.; McFadden, J.; Shoham, Y. A portfolio approach to algorithm select. In Proceedings of the 18th International Joint Conference on Artificial Intelligence, Acapulco, Mexico, 9–15 August 2003; pp. 1542–1543.
77. Rice, J.R. The Algorithm Selection Problem. *Comput. Sci. Tech. Rep. Pap.* **1975**, *99*, 75–152. Available online: <http://docs.lib.purdue.edu/cstech/99> (accessed on 2 March 2022).
78. Strassl, S.; Musliu, N. Instance space analysis and algorithm selection for the job shop scheduling problem. *Comput. Oper. Res.* **2022**, *141*, 105661. <https://doi.org/10.1016/j.cor.2021.105661>.
79. Lagoudakis, M.G.; Littman, M.L. Algorithm Selection using Reinforcement Learning. In Proceedings of the 17th International Conference on Machine Learning, San Francisco, CA, USA, 29 June–2 July 2000; pp. 511–518.
80. Xu, L.; Hutter, F.; Hoos, H.; Leyton-Brown, K. SATzilla: Portfolio-based Algorithm Selection for SAT. *J. Artif. Intell. Res.* **2008**, *32*, 565–606. <https://doi.org/10.1613/jair.2490>.
81. Smith-Miles, K. Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Comput. Surv.* **2008**, *41*, 1–25. <https://doi.org/10.1145/1456650.1456656>.
82. Hoos, H.; Lindauer, M.; Schaub, T. claspfolio2: Advances in Algorithm Selection for Answer Set Programming. *Theory Pr. Log. Program.* **2014**, *14*, 569–585. <https://doi.org/10.1017/s1471068414000210>.
83. Tierney, K.; Malitsky, Y. An algorithm selection benchmark of the container pre-marshalling problem. In *Learning and Intelligent Optimization*; Springer: Cham, Switzerland, 2015; pp. 17–22.
84. Cunha, T.; Soares, C.; de Carvalho, A. Metalearning and Recommender Systems: A literature review and empirical study on the algorithm selection problem for Collaborative Filtering. *Inf. Sci.* **2018**, *423*, 128–144. <https://doi.org/10.1016/j.ins.2017.09.050>.
85. Bożejko, W.; Gnatowski, A.; Niżyński, T.; Affenzeller, M.; Beham, A. Local optima networks in solving algorithm selection problem for TSP. *Adv. Intell. Syst. Comput.* **2019**, *761*, 83–93. https://doi.org/10.1007/978-3-319-91446-6_9.
86. Drozdov, G.; Zabashta, A.; Filchenkov, A. Graph convolutional network based generative adversarial networks for the algorithm selection problem in classification. In Proceedings of the International Conference on Control, Robotics and Intelligent System, Xiamen, China, 27–29 October 2020; pp. 88–92. <https://doi.org/10.1145/3437802.3437818>.
87. Boas, M.G.V.; Santos, H.; Merschmann, L.H.D.C.; Bergehe, G.V. Optimal decision trees for the algorithm selection problem: Integer programming based approaches. *Int. Trans. Oper. Res.* **2021**, *28*, 2759–2781. <https://doi.org/10.1111/itor.12724>.
88. Marrero, A.; Segredo, E.; Leon, C. A parallel genetic algorithm to speed up the resolution of the algorithm selection problem. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Lille, France, 10–14 July 2021; pp. 1978–1981.
89. Müller, D.; Müller, M.G.; Kress, D.; Pesch, E. An algorithm selection approach for the flexible job shop scheduling problem: Choosing constraint programming solvers through machine learning. *Eur. J. Oper. Res.* **2022**, in press. <https://doi.org/10.1016/j.ejor.2022.01.034>.
90. Karimi-Mamaghan, M.; Mohammadi, M.; Meyer, P.; Karimi-Mamaghan, A.M.; Talbi, E.-G. Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art. *Eur. J. Oper. Res.* **2022**, *296*, 393–422. <https://doi.org/10.1016/j.ejor.2021.04.032>.
91. Radcliffe, N.J. The algebra of genetic algorithms. *Ann. Math. Artif. Intell.* **1994**, *10*, 339–384. <https://doi.org/10.1007/bf01531276>.
92. Hutter, F.; Xu, L.; Hoos, H.; Leyton-Brown, K. Algorithm runtime prediction: Methods & evaluation. *Artif. Intell.* **2014**, *206*, 79–111. <https://doi.org/10.1016/j.artint.2013.10.003>.
93. Ozsahin, I.; Ozsahin, D.U.; Uzun, B.; Mustapha, M.T. Chapter 1—Introduction. In *Applications of Multi-Criteria Decision-Making Theories in Healthcare and Biomedical Engineering*; Ozsahin, I., Ozsahin, D.U., Uzun, B., Eds.; Academic Press: Cambridge, MA, USA, 2021; pp. 1–2.
94. Zadeh, L.A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [https://doi.org/10.1016/s0019-9958\(65\)90241-x](https://doi.org/10.1016/s0019-9958(65)90241-x).
95. Hwang, C.-L.; Yoon, K. Methods for multiple attribute decision making. In *Multiple Attribute Decision Making*; Springer: Berlin/Heidelberg, Germany, 1981; pp. 58–191.
96. Kannan, D.; De Sousa Jabbour, A.B.L.; Jabbour, C.J.C. Selecting green suppliers based on GSCM practices: Using fuzzy TOPSIS applied to a Brazilian electronics company. *Eur. J. Oper. Res.* **2014**, *233*, 432–447. <https://doi.org/10.1016/j.ejor.2013.07.023>.
97. Chen, C.-T. Extensions of the TOPSIS for group decision-making under fuzzy environment. *Fuzzy Sets Syst.* **2000**, *114*, 1–9. [https://doi.org/10.1016/s0165-0114\(97\)00377-1](https://doi.org/10.1016/s0165-0114(97)00377-1).
98. Bellman, E.; Zadeh, L.A. A fuzzy environment. *Manage. Sci.* **1970**, *17*, 141–164.
99. Dubois, D.; Prade, H. *Fuzzy Sets and Systems: Theory and Applications*; Cambridge, MA, USA, 1980.
100. Chakraborty, C.; Chakraborty, D. A theoretical development on a fuzzy distance measure for fuzzy numbers. *Math. Comput. Model.* **2006**, *43*, 254–261. <https://doi.org/10.1016/j.mcm.2005.09.025>.
101. Ploskas, N.; Papathanasiou, J. A decision support system for multiple criteria alternative ranking using TOPSIS and VIKOR in fuzzy and nonfuzzy environments. *Fuzzy Sets Syst.* **2019**, *377*, 1–30. <https://doi.org/10.1016/j.fss.2019.01.012>.

102. Shen, L.; Olfat, L.; Govindan, K.; Khodaverdi, R.; Diabat, A. A fuzzy multi criteria approach for evaluating green supplier's performance in green supply chain with linguistic preferences. *Resour. Conserv. Recycl.* **2013**, *74*, 170–179. <https://doi.org/10.1016/j.resconrec.2012.09.006>.
103. Wang, T.-C.; Chang, T.-H. Application of TOPSIS in evaluating initial training aircraft under a fuzzy environment. *Expert Syst. Appl.* **2007**, *33*, 870–880. <https://doi.org/10.1016/j.eswa.2006.07.003>.
104. Afful-Dadzie, E.; Nabareseh, S.; Afful-Dadzie, A.; Oplatková, Z.K. A fuzzy TOPSIS framework for selecting fragile states for support facility. *Qual. Quant.* **2015**, *49*, 1835–1855. <https://doi.org/10.1007/s11135-014-0062-3>.
105. Piya, S.; Shamsuzzoha, A.; Khadem, M. An approach for analysing supply chain complexity drivers through interpretive structural modelling. *Int. J. Logist. Res. Appl.* **2020**, *23*, 311–336. <https://doi.org/10.1080/13675567.2019.1691514>.
106. Guzmán, E.; Poler, R.; Andrés, B. Un análisis de revisiones de modelos y algoritmos para la optimización de planes de aprovisionamiento, producción y distribución de la cadena de suministro. *Dir. Organ.* **2020**, *70*, 28–52. <https://doi.org/10.37610/dyo.v0i70.567>.
107. Guzman, E.; Andres, B.; Poler, R. Models and algorithms for production planning, scheduling and sequencing problems: A holistic framework and a systematic review. *J. Ind. Inf. Integr.* **2021**, *27*, 100287. <https://doi.org/10.1016/j.jii.2021.100287>.
108. G. Stewart. Supply-chain operations reference model (SCOR): the first cross-industry framework for integrated supply-chain management. *Logist. Inf. Manag.* **1997**, *10*, 62–67. <https://doi.org/10.1108/09576059710815716>.
109. Michalewicz, Z.; Fogel, D.B. *How to Solve It: Modern Heuristics*; Springer Science & Business Media: Berlin, Germany, 2013.
110. Tasan, A.S.; Gen, M. A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Comput. Ind. Eng.* **2012**, *62*, 755–761. <https://doi.org/10.1016/j.cie.2011.11.025>.
111. Ku, W.-Y.; Beck, J.C. Mixed Integer Programming models for job shop scheduling: A computational analysis. *Comput. Oper. Res.* **2016**, *73*, 165–173. <https://doi.org/10.1016/j.cor.2016.04.006>.
112. Fahimnia, B.; Farahani, R.Z.; Marian, R.; Luong, L. A review and critique on integrated production–distribution planning models and techniques. *J. Manuf. Syst.* **2013**, *32*, 1–19. <https://doi.org/10.1016/j.jmsy.2012.07.005>.
113. Gavrilas, M. Heuristic and metaheuristic optimization techniques with application to power systems. In Proceedings of the International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering, Timisoara, Romania, 21–23 October 2010; pp. 95–103.
114. Swan, J.; Adriaensen, S.; Brownlee, A.E.; Hammond, K.; Johnson, C.G.; Kheiri, A.; Krawiec, F.; Merelo, J.; Minku, L.L.; Özcan, E.; et al. Metaheuristics “In the Large”. *Eur. J. Oper. Res.* **2021**, *297*, 393–406. <https://doi.org/10.1016/j.ejor.2021.05.042>.
115. Boschetti, M.A.; Maniezzo, V.; Roffilli, M.; Röhlér, A.B. Matheuristics: Optimization, simulation and control. In *International Workshop on Hybrid Metaheuristics*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5818, pp. 171–177. https://doi.org/10.1007/978-3-642-04918-7_13.
116. Sun, C.-C. A performance evaluation model by integrating fuzzy AHP and fuzzy TOPSIS methods. *Expert Syst. Appl.* **2010**, *37*, 7745–7754. <https://doi.org/10.1016/j.eswa.2010.04.066>.
117. Zadeh, L.A. The concept of a linguistic variable and its application to approximate reasoning—I. *Inf. Sci.* **1975**, *8*, 199–249. [https://doi.org/10.1016/0020-0255\(75\)90036-5](https://doi.org/10.1016/0020-0255(75)90036-5).
118. Nădăban, S.; Dzitac, S.; Dzitac, I. Fuzzy TOPSIS: A General View. *Procedia Comput. Sci.* **2016**, *91*, 823–831. <https://doi.org/10.1016/j.procs.2016.07.088>.
119. Torlak, G.; Sevkli, M.; Sanal, M.; Zaim, S. Analyzing business competition by using fuzzy TOPSIS method: An example of Turkish domestic airline industry. *Expert Syst. Appl.* **2011**, *38*, 3396–3406. <https://doi.org/10.1016/j.eswa.2010.08.125>.
120. Lo, H.-W.; Liaw, C.-F.; Gul, M.; Lin, K.-Y. Sustainable supplier evaluation and transportation planning in multi-level supply chain networks using multi-attribute- and multi-objective decision making. *Comput. Ind. Eng.* **2021**, *162*, 107756. <https://doi.org/10.1016/j.cie.2021.107756>.