

## Research article

# A comparison among optimization software to solve bi-objective sectorization problem

Aydin Teymourifar

Universidade Católica Portuguesa, Católica Porto Business School, Centro de Estudos em Gestão e Economia Porto, Portugal

## ARTICLE INFO

### Keywords:

Multi-objective optimization  
Sectorization  
Mixed integer non-linear programming  
GAMS  
CPLEX  
Lingo  
Python  
Pulp  
Pymoo  
GA  
NSGA-II

## ABSTRACT

In this study, we compare the performance of optimization software to solve the bi-objective sectorization problem. The used solution method is based on an approach that has not been used before in the literature on sectorization, in which, the bi-objective model is transformed into single-objective ones, whose results are regarded as ideal points for the objective functions in the bi-objective model. Anti-ideal points are also searched similarly. Then, using the ideal and anti-ideal points, the bi-objective model is redefined as a single-objective one and solved. The difficulties of solving the models, which are basically non-linear, are discussed. Furthermore, the models are linearized, in which case how the number of variables and constraints changes is discussed. Mathematical models are implemented in Python's Pulp library, Lingo, IBM ILOG CPLEX Optimization Studio, and GAMS software, and the obtained results are presented. Furthermore, metaheuristics available in Python's Pymoo library are utilized to solve the models' single- and bi-objective versions. In the experimental results section, benchmarks of different sizes are derived for the problem, and the results are presented. It is observed that the solvers do not perform satisfactorily in solving models; of all of them, GAMS achieves the best results. The utilized metaheuristics from the Pymoo library gain feasible results in reasonable times. In the conclusion section, suggestions are given for solving similar problems. Furthermore, this article summarizes the managerial applications of the sectorization problems.

## 1. Introduction

Almost all optimization software and solvers have evolved considerably over time, but they still have significant differences [1]. Therefore, in the literature, there are studies comparing solvers [2–7]. Likewise, some studies compare the evolutionary algorithm (EA) with solvers and discuss the advantages and disadvantages [8–10]. In this study, a comparison is made between optimization software to solve a sectorization problem (SP). The goal of SPs is to split a large region into smaller ones for administrative purposes [11–14]. However the main contribution of the study to the literature is the comparison among optimization software, it has other novelties, which can be summarized as follows:

- A new bi-objective (BO) model is defined for SP, which is nonlinear (NL).
- A new approach is applied to solve the model, in which, at first, the BO model is divided into two single-objective (SO) ones, which are also NL.
- The goal of the division is to gain ideal points (IDPs) and anti-ideal points (ADPs) for the objective functions in the BO model.

E-mail addresses: [aydinteymourifar@gmail.com](mailto:aydinteymourifar@gmail.com), [ateymourifar@ucp.pt](mailto:ateymourifar@ucp.pt).

<https://doi.org/10.1016/j.heliyon.2023.e18602>

Received 15 November 2022; Received in revised form 21 July 2023; Accepted 22 July 2023

Available online 29 July 2023

2405-8440/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

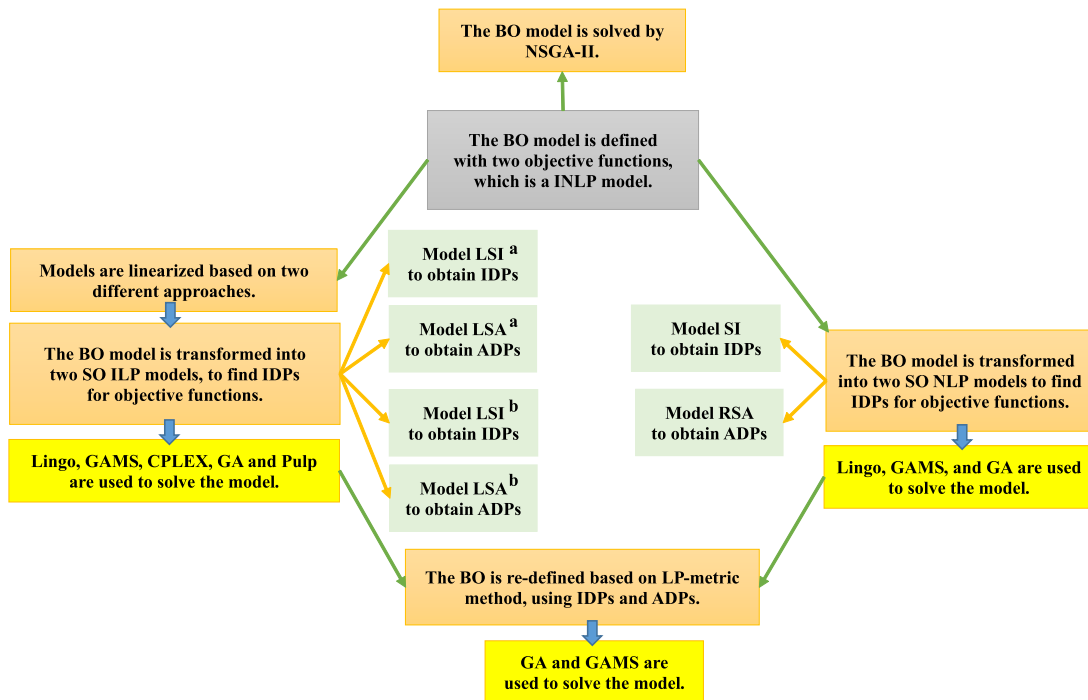


Fig. 1. A summary of applied methods to the BO model.

- The difficulties and complexities of solving the model such as the nonlinearity of objective functions and of some constraints are discussed.
- Moreover, models are linearized for possible easier solutions.
- The outputs of the linearized models are considered as IDPs and ADPs of the objective functions in the BO model.
- The results of the linear (L) and NL models are supposed to be the same if they are optimal. However, this is also a way to evaluate the computational performance of the solvers.
- The BO model is re-defined based on the LP-metric method, using IDPs and ADPs for the objective functions.

Though the employed solution approach is well-known in optimization topics, it is employed for the first time in the sectorization literature.

As a contribution to the literature, a comparison is made between different software for solving MO SP. For mathematical models, different solvers like Python's Pulp library, Lingo, IBM ILOG CPLEX Optimization Studio, and GAMS are used. There is a similar study in the literature comparing optimization tools for solving SPs [15]. But it doesn't report the results of GAMS, IBM ILOG CPLEX Optimization Studio, and Lingo. Furthermore, the solution approach in this study is different. From SO metaheuristics available in the Pymoo library of Python, a Genetic Algorithm (GA), and from MO ones the Non-dominated Sorting Algorithm (NSGA-II) are employed [16].

A summary of the methods that are applied in this paper is presented in Fig. 1, whose details are described in Section 4. The used abbreviations can be found in the appendix.

The other parts of the paper are organized as follows: Section 2 summarizes the related literature. The problem definition and solution method are respectively described in Sections 3 and 4. In the experimental results section, based on generated benchmarks, the operated software and algorithms are compared. Conclusion and future works form the last part of the article, where comments and suggestions for solving similar models are offered as well as several research directions.

## 2. Literature review

Koch et al. [1] report that from 2001 to 2020 linear programming (LP) and mixed integer linear programming (MILP) solvers became about 180 and 1000 times faster, respectively. Furthermore, benchmarks that couldn't be solved years ago are now solved in seconds [1]. Despite all these advancements, occasionally unexpected variability affects the performance of mixed integer programming (MIP) solvers. This issue has been observed for decades. Lodi and Tramontani [2] discuss the causes of this phenomenon. Therefore, the performance of solvers for various problems is of interest to researchers. Neumaier et al. [3] compare the performance of constrained optimization solvers over 1000 benchmarks from literature, which contain 1000 variables. Kronqvist et al. [4] make a review and comparison between mixed integer non-linear programming (MINLP) solvers based on a set of 335 benchmarks, which are divided into subsets based on the continuous relaxation gap, the degree of nonlinearity, and also the relative number of discrete.

**Table 1**

Some of the problems modeled based on sectorization and employed approaches to solve them.

Problem	Solution Approach	MO model
Political districting [11]	ILP	
Facility location [12]	GP	✓
Location allocation [13]	Metaheuristics	
Location routing [17]	Heuristic and metaheuristic	
Location routing [18]	Heuristic	
Supply chain network [19]	Metaheuristic	
Elderly care service districting [20]	MINLP	✓
Emergency medical service districting [21]	Metaheuristic	
Power distribution networks [22]	Metaheuristic	✓
Police districting [23]	Literature review	
Commercial districting [24]	Metaheuristic	
School districting [25]	Heuristic	
Air traffic control [26]	Heuristic and metaheuristic	
Air traffic control [27]	Human factors issues	
Air traffic control [28]	Constraint programming	
Air traffic control [29]	Metaheuristic	✓
Air traffic control [30]	Simulation	
Air traffic control [31]	Metaheuristic	✓
Telecommunications [32]	Simulation	
Telecommunications [33]	IQP, MIP	
Water distribution networks [34]	MINLP and metaheuristic	✓

**Table 2**

Managerial applications of SPs.

Application	Filed
Decreasing wastes [35]	Water management
Managing capacity [36]	Air traffic management
Improving utility [37]	Healthcare management

The acquired results provide information about which solver or method is suitable for which type of benchmarks. Meindl and Tempel [5] compare various solvers on LP benchmarks. Luppold et al. [6] investigate the effects of different types of variables, constraints, and logical expressions on the performance of solvers. Describing the complexity of hard real-time systems with large numbers of constraints, Guasque and Balbastre [7] emphasize the importance of heuristic methods to obtain feasible solutions for them. They also make a comparison between CPLEX and Gurobi.

In this study, the comparison between optimization software is done over SPs. As stated in the previous paragraph, the type of problem and the solution method affect the performance of employed solvers. Therefore, this section summarizes the literature on different types of SPs and the utilized methods for their solution. The main objective of SPs is to divide a vast district into smaller and balanced areas. SPs are mostly formulated as multi-objective (MO) models and they have applications in numerous fields. Some of the problems modelled based on sectorization, the used solution approaches for them, and whether the model is MO or not appear in Table 1. The used abbreviations can be found in the appendix.

SPs have managerial implications, some of which are shown in Table 2.

MO models fall into multi-criteria decision-making (MCDM) techniques. As seen in Fig. 2, MCDM techniques can be categorized as multi-attribute decision-making (MADM) and multi-objective decision-making (MODM) subcategories that respectively have discrete and continuous decision spaces. MODMs are divided into four categories based on the role of the decision maker (DM). In the first class, which is a neutral approach, there is no preference information from the decision-maker. The LP-metric method is in this category, which trade-offs between desired and undesired solutions and in this way converts an MO model to an SO one. In other classes, preference information is received from the DM before, after, or interactively during the resolution process. MADM techniques can be divided into three categories: compensatory, non-compensatory, and partial compensatory. In compensatory approaches, some negative attributes can be compensated by positive ones [38–41].

In the literature, the LP-metric or the methods designed based on IDPs and ADPs have not been used to solve SPs, however, they have been applied for the solution of several problems, some of which are summarized in Table 3 [42].

Some of the problems in the literature solved based on other techniques of MCDM are summarized in Table 4.

There are also other MCDM techniques in the literature that have not been used for SPs. For example, from MODM, a priori methods: lexicographic max-min [53]  $\epsilon$ -constraint, goal programming [54], surrogate worth trade-off method (SWTM) [55], from interactive methods [56], step method (STEM) [57], from MADM, compensatory methods: the technique for order preference by similarity to ideal solution (TOPSIS), analytical network process (ANP) [58–61] and from MADM, non-compensatory methods: Max-Max, Min-Max [62] have not been employed for SPs solution.

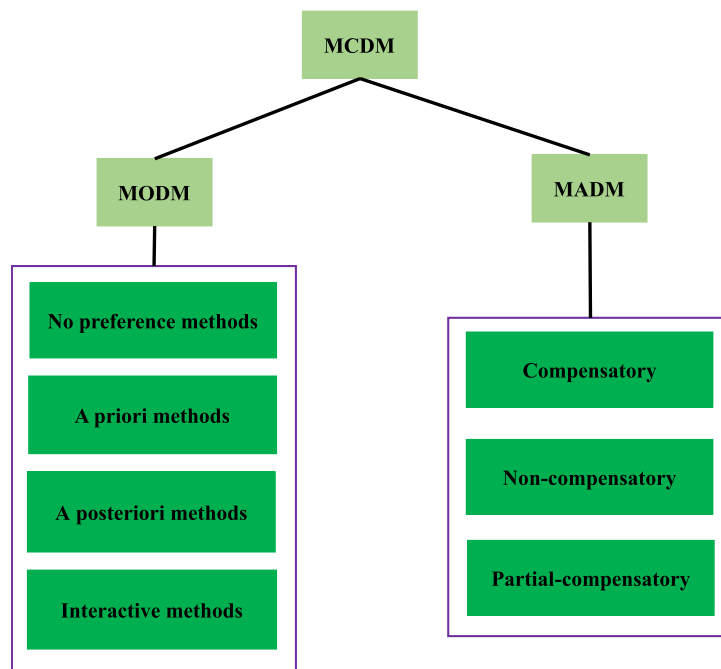


Fig. 2. Classification of MCDMs [38–41].

Table 3

Problems solved based on IDPs and ADPs in the literature.

Problem	Combined Method with IDPs and ADPs
Location selection [43]	Fuzzy decision making
Efficiency evaluation [44]	Data envelopment analysis
Supplier selection [45,46]	MCDM
Wind site selection [47]	Fuzzy MCDM
Portfolio asset management [48]	MODM

Table 4

Some of the problems solved based on MCDM techniques.

Problem	Method	Category of the Method
Air traffic control [49]	Weighted sum	MODM, a priori methods
Water distribution network [50]	Lexicographic	MODM, a priori methods
Air traffic control [51]	Scalarization	MODM, a priori methods
Water distribution network [52]	AHP	MADM, compensatory

### 3. Problem definition

In the problem, it is assumed that there are clients, to be assigned to some service centers, whose numbers and coordinates are predetermined. Each service center and its assigned clients form a sector. Each client is assigned to only one sector. Each of the formed sectors has at least one client. The number, coordinates, and service centers are known beforehand. Each client has a predefined demand.

Some of the used notations are summarized in Table 5.

The binary decision variable of the model is as in Eq. (1).

$$Z_{ij} = \begin{cases} 1, & \text{if client } i \text{ is assigned to sector } j \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

$$i = 1, \dots, n, j = 1, \dots, k.$$

**Table 5**  
Used notations.

Notation	Description
$n$	Total number of clients
$k$	Total number of sectors (and service centers)
$Q$	Target number of clients per sector
$m$	Number of objective function
$Z_{ij}$	Decision variable about assignment of client $i$ to sector $j$
$d_{ij}$	Euclidean distance between client $i$ and the service center of sector $j$
$Dc_j$	Total distance of clients of sector $j$ from the related service center
$\bar{Dc}$	Average of the distance of clients from the service center of sectors
$de_i$	Demand of client $i$
$De_j$	Total demand of clients in sector $j$
$\bar{De}$	Average of the demand of clients in sectors
$\tau_{eqds}$	Parameter for the equilibrium of distances in sectors
$\tau_{eqde}$	Parameter for the equilibrium of demand in sectors
$f_1$	Objective function related to compactness
$f_2$	Objective function related to equilibrium of demands in sectors
$l_m$	Linearized form of $f_m$
$f_m^*$	Ideal point for $f_m$
$f_m^{**}$	Anti-ideal point for $f_m$
$l_m^*$	Ideal point for the linearized form of $f_m$
$l_m^{**}$	Anti-ideal point for the linearized form of $f_m$
$f$	Bi-objective function related to compactness and equilibrium
$V$	Positive variables for linearizing the objective functions

### 3.1. BO model

To be minimized, the objective function related to the equilibrium of distances in sectors is defined as in Eq. (2).

$$f_1 = \sum_{j=1}^k |Dc_j - \bar{Dc}| \quad (2)$$

where  $Dc_j = \sum_{i=1}^n d_{ij} \times Z_{ij}$ ,  $\bar{Dc} = \frac{\sum_{j=1}^k Dc_j}{k}$ .

To be minimized, the objective function related to equilibrium of demands in sectors is defined as in Eq. (3).

$$f_2 = \sum_{j=1}^k |De_j - \bar{De}| \quad (3)$$

where  $De_j = \sum_{i=1}^n de_i \times Z_{ij}$ ,  $\bar{De} = \frac{\sum_{j=1}^k De_j}{k}$ .

The purpose of  $f_1$  and  $f_2$  is to create more balanced sectors in terms of distance and demand.

In this way, to be minimized, the BO function of the model is defined as in Eq. (4).

$$f = (f_1, f_2) \quad (4)$$

Each client is assigned to just one sector, which is satisfied by Constraint (5).

$$\sum_{j=1}^k Z_{ij} = 1, \quad i = 1, \dots, n \quad (5)$$

As specified in Constraint (6), at least one client is assigned per sector.

$$\sum_{i=1}^n Z_{ij} \geq 1, \quad j = 1, \dots, k \quad (6)$$

Furthermore, as defined in Eq. (1),  $Z$  is a binary variable, which is considered as a constraint in the model.

It is expected that the equilibrium is yielded in terms of distances and demands between the formed sectors, which is provided by Constraints (7)-(8).

$$|Dc_j - \bar{Dc}| \leq \bar{Dc}(1 - \tau_{eqds}), \quad j = 1, \dots, k \quad (7)$$

$$|De_j - \bar{De}| \leq \bar{De}(1 - \tau_{eqde}), \quad j = 1, \dots, k \quad (8)$$

$\tau_{eqds}$ , and  $\tau_{eqde}$  are parameters for the equilibrium of distances and demand in sectors. The reason for using Constraints (7)-(8) can be explained as follows: The total deviations of  $Dc_j$  and  $De_j$ ,  $j = 1, \dots, k$ , from their averages are minimized by applying the objective functions  $f_1$ , and  $f_2$ . Whereas using Constraints (7)-(8), more desired solutions are provided by defining upper limits to the deviation

value for each sector. If the objective functions were defined based on variance or standard deviation instead of absolute value, they would have the same influence, but upper limits of deviations could not be guaranteed.

The complete form of the basic NL model can be expressed with Statements (9)-(15).

$$\min \sum_{j=1}^k \left| \sum_{i=1}^n (d_{ij} Z_{ij} - \frac{1}{k} \sum_{p=1}^k d_{ip} Z_{ip}) \right| \quad (9)$$

subject to

$$\min \sum_{j=1}^k \left| \sum_{i=1}^n d_{ij} (Z_{ij} - \frac{1}{k} \sum_{p=1}^k Z_{ip}) \right| \quad (10)$$

$$\sum_{j=1}^k Z_{ij} = 1, \quad i = 1, \dots, n \quad (11)$$

$$\sum_{i=1}^n Z_{ij} \geq 1, \quad j = 1, \dots, k \quad (12)$$

$$\left| \sum_{i=1}^n (d_{ij} Z_{ij} - \frac{1}{k} \sum_{p=1}^k d_{ip} Z_{ip}) \right| \leq \frac{1}{k} (1 - \tau_{eqds}) \sum_{i=1}^n \sum_{p=1}^k d_{ip} Z_{ip}, \quad j = 1, \dots, k \quad (13)$$

$$\left| \sum_{i=1}^n d_{ij} (Z_{ij} - \frac{1}{k} \sum_{p=1}^k Z_{ip}) \right| \leq \frac{1}{k} (1 - \tau_{eqde}) \sum_{i=1}^n \sum_{p=1}^k Z_{ip}, \quad j = 1, \dots, k \quad (14)$$

$$Z_{ij} \in \{0, 1\}, \quad i = 1, \dots, n, j = 1, \dots, k \quad (15)$$

#### 4. Solution method

Before describing the solution method, some concepts are explained. Vectors of nadir [63] and ideal objectives are used in the literature on MO optimization [64], which are not the same as IDPs and ADPs in this work. The next definitions need to be made to clarify the differences:

**Definition 4.1.** In a minimization problem the following conditions must be valid for the solution  $z^1$  to dominate  $z^2$ :  
 $\forall m \in \{1, 2\} \quad f_m(z^1) \leq f_m(z^2) \ \& \ \exists m \in \{1, 2\} \quad f_m(z^1) < f_m(z^2)$ .

**Definition 4.2.**  $z^{Pa}$  is a Pareto optimal or an efficient solution if it is not dominated by any other solution.

In the field of MO optimization, two distinct terms, “weak efficiency” and “strict efficiency,” are distinguished. A solution is considered weakly efficient if it is not dominated by another one in the solution space. This means that, given a certain objective, a weakly efficient solution may be dominated by another one. Yet, a solution is considered strictly efficient if it is not dominated by any other solution according to any objective. In this research, weakly efficient solutions are referred to as efficient solutions.

**Definition 4.3.** The set of efficient solutions forms the Pareto front, which is denoted as  $Z^{Pa}$ .

**Definition 4.4.** The Pareto front is bounded by the vector of nadir and ideal objectives, which are defined as  $Sup_{z^{Pa} \in Z^{Pa}} f_m(z^{Pa})$ , and  $Inf_{z^{Pa} \in Z^{Pa}} f_m(z^{Pa})$ ,  $\forall m \in \{1, 2\}$ , respectively.

The IDPs and ADPs in this study are the outputs obtained from SO models, which are operated in the LP-metric method to transform the MO model into an SO one.

It can be stated that the LP-metric method sets weights for objectives and hence is similar to the weighted sum method. The weighted sum approach is in the category of a priori methods in Fig. 2, because weights are usually defined by the DM before the solution process. Whilst in the LP-metric method, the weights are determined during the solution procedure.

According to Ref. [38], optimal solutions of the weighted sum problem with positive (non-negative) weights are always (weakly) efficient. Furthermore, all (weakly) efficient solutions are optimal solutions of scalarized problems with positive (non-negative) weights under convexity assumptions.

It should be noted that integer programming models are considered non-convex, thus convexity assumptions are not valid for them.

In the next subsection, the solution method is described, which has not been used in previous studies about SP. In this method, at first, the BO model is converted into two SO models, from which IDPs and ADPs are found for the objective functions. Then the BO model is redefined and transformed to an SO model by using IDPs and ADPs. L versions of the models are also utilized.

#### 4.1. Dividing the BO model into two NL SO models

The goal of the division is to determine IDPs and ADPs for objective functions.

##### 4.1.1. $SI_1$ , SO model to find an IDP for $f_1$ in the BO model

To be minimized, the objective function of this model is  $f_1$ , as defined in Eq. (2). The optimal solution of this model,  $f_1^*$ , is an IDP for  $f_1$  in the BO model.

##### 4.1.2. $SI_2$ , SO model to find an IDP for $f_2$ in the BO model

To be minimized, the objective function of this model is  $f_2$ , as defined in Eq. (3). The optimal solution of this model,  $f_2^*$ , is an IDP for  $f_2$  in the BO model.

Constraints (5)-(8) are valid in the  $SI_1$  and  $SI_2$  models.

##### 4.1.3. $SA_1$ , SO model to find an ADP for $f_1$ in the BO model

When the objective function,  $f_1$ , is to be maximized instead of be minimized, the optimal solution,  $f_1^{**}$ , is an ADP for  $f_1$  in the BO model.

##### 4.1.4. $SA_2$ , SO model to find an ADP for $f_2$ in the BO model

To be maximized, the objective function of this model is  $f_2$ , defined as in Eq. (3). The optimal solution of this model,  $f_2^{**}$ , is an ADP for  $f_2$  in the BO model.

Models  $SA_1$  and  $SA_2$  comprise Constraints (5)-(6). However, they don't contain Constraints (7)-(8), since they define an upper limit, which is not meaningful for maximization.

#### 4.2. Linearizing SO models

In this subsection, the SO models are linearized according to two different approaches. The number of variables in these two approaches is different. They are used to demonstrate the influence of the number of variables.

##### 4.2.1. $LSI_1^a$ , linearized SO model based on the first approach to find an IDP for $f_1$ in the BO model

Using non-negative variable  $V_{1j}$  objective function  $f_1$  is linearized.

$$I_1 = \sum_{j=1}^k V_{1j} \quad (16)$$

$$Dc_j - \bar{D}c \leq V_{1j}, \quad j = 1, \dots, k \quad (17)$$

$$\bar{D}c - Dc_j \leq V_{1j}, \quad j = 1, \dots, k \quad (18)$$

$$V_{1j} \geq 0, \quad j = 1, \dots, k \quad (19)$$

$$Dc_j - \bar{D}c \leq \bar{D}c(1 - \tau_{eqds}), \quad j = 1, \dots, k \quad (20)$$

$$\bar{D}c - Dc_j \leq \bar{D}c(1 - \tau_{eqds}), \quad j = 1, \dots, k \quad (21)$$

$$De_j - \bar{D}e \leq \bar{D}e(1 - \tau_{eqde}), \quad j = 1, \dots, k \quad (22)$$

$$\bar{D}e - De_j \leq \bar{D}e(1 - \tau_{eqde}), \quad j = 1, \dots, k \quad (23)$$

Constraints (20)-(23) are the linear versions of Constraints (7)-(8). Constraints (5)-(6) are also included in this model. The optimal solution of this model, indicated as  $I_1^*$ , is an IDP for  $f_1$  in the BO model.

##### 4.2.2. $LSI_1^b$ , linearized SO model with the second approach to find an IDP for $f_1$ in the BO model

Using non-negative variables  $V_{1j}$  and  $V_{2j}$ ,  $j = 1, \dots, k$ , objective function  $f_1$  is linearized.

$$I_1 = \sum_{j=1}^k V_{1j} + \sum_{j=1}^k V_{2j} \quad (24)$$

$$Dc_j - \bar{D}c - V_{1j} + V_{2j} = 0, \quad j = 1, \dots, k \quad (25)$$

$$V_{1j} \text{ and } V_{2j} \geq 0, \quad j = 1, \dots, k \quad (26)$$

To be minimized, the objective function of this model is determined as in Eq. (24). The optimal solution of this model, denoted as  $I_1^*$ , is an IDP for  $f_1$  in the BO model. Constraints (5)-(6) and (20)-(23) are also contained in this model, as well as Constraints (25)-(26).

In the  $LSI_1^a$  model, there are additionally  $2k$  variables due to  $V_{1j}$  and  $V_{2j}$ , while there are further  $k$  variables in the  $LSI_1^b$  model because of  $V_{1j}$  or  $V_{2j}$ . Moreover, in both models,  $7k$  constraints are added.

#### 4.2.3. $LSI_2^a$ , linearized SO model based on the first approach to find an IDP for $f_2$ in the BO model

Using non-negative variable  $V_{2j}$  objective function  $f_2$  is linearized.

$$I_2 = \sum_{j=1}^k V_{2j} \quad (27)$$

$$De_j - \bar{D}e \leq V_{2j}, \quad j = 1, \dots, k \quad (28)$$

$$\bar{D}e - De_j \leq V_{2j}, \quad j = 1, \dots, k \quad (29)$$

$$V_{2j} \geq 0, \quad j = 1, \dots, k \quad (30)$$

Constraints (5)-(6), and (20)-(23) are also included in this model. The optimal solution of this model indicated as  $I_2^*$ , is an IDP for  $f_2$  in the BO model.

#### 4.2.4. $LSI_2^b$ , linearized SO model based on the second approach to find an IDP for $f_2$ in the BO model

Using non-negative variables  $V_{3j}$  and  $V_{4j}$ ,  $j = 1, \dots, k$ , objective function  $f_2$  is linearized. To be minimized, the objective function of this model is determined as in Eq. (31), taking Constraints (32)-(33) into consideration, as well as Constraints (5)-(6), (20)-(23). The optimal solution of this model denoted as  $I_2^*$ , is an IDP for  $f_2$  in the BO model.

$$I_2 = \sum_{j=1}^k V_{3j} + \sum_{j=1}^k V_{4j} \quad (31)$$

$$De_j - \bar{D}e - V_{3j} + V_{4j} = 0, \quad j = 1, \dots, k \quad (32)$$

$$V_{3j} \text{ and } V_{4j} \geq 0, \quad j = 1, \dots, k \quad (33)$$

The number of added variables and constraints in the  $LSI_2^a$  and  $LSI_2^b$  models is the same as described at the end of Subsection 4.2.2.

#### 4.2.5. $LSA_1^a$ , SO model with linearized objective function based on the first approach to find an ADP for $f_1$ in the BO model

The objective function of this model is defined as in Eq. (16), to be maximized, considering Constraints (17)-(19), in addition to Constraints (5)-(6). Specified as  $I_1^{**}$ , the optimal solution of this model is an ADP for  $f_1$  in the BO model.

#### 4.2.6. $LSA_1^b$ , SO model with linearized objective function based on the second approach to find an ADP for $f_1$ in the BO model

To be maximized, the objective function of this model is determined as in Eq. (24), considering Constraints (25)-(26), in addition to Constraints (5)-(6). Denoted as  $I_1^{**}$ , the optimal solution of this model is an ADP for  $f_1$  in the BO model.

#### 4.2.7. $LSA_2^a$ , SO model with linearized objective function based on the first approach to find an ADP for $f_2$ in the BO model

The objective function of this model is as in Eq. (27), to be maximized, regarding Constraints (5)-(6), (28)-(30). Shown as  $I_2^{**}$ , the optimal solution of this model is an ADP for  $f_2$  in the BO model.

#### 4.2.8. $LSA_2^b$ , SO model with linearized objective function based on the second approach to find an ADP for $f_2$ in the BO model

To be maximized, the objective function of this model is specified as in Eq. (31), regarding Constraints (32)-(33), in addition to Constraints (5)-(6). Indicated as  $I_2^{**}$ , the optimal solution of this model is an ADP for  $f_2$  in the BO model.

As mentioned before, since Constraints (7)-(8) identify upper limits, are not applicable for maximization. Therefore, the  $LSA^a$  and  $LSA^b$  models don't include them.

### 4.3. Redefining the BO model

Since it is not easy to deal with multiple objective functions simultaneously, in this section the BO model is redefined as an SO model using IDPs and ADPs.

The LP-metric method is used to convert an MO problem to an SO one, in which the total distance to the IDPs is minimized, along with maximizing the distance from the ADPs.

In the next subsection, three models are defined based on the LP-metric method.

#### 4.3.1. LP-metric-NL, an NL model using the outputs of the SI and SA models and LP-metric method to find efficient solutions for the BO model

Using the results obtained from the NL models, to be minimized, the objective function can be defined as in Eq. (34).

$$f = \left[ \sum_{m=1}^2 \left| \frac{f_m - f_m^*}{f_m^{**} - f_m^*} \right|^p \right]^{\frac{1}{p}} \quad (34)$$

where  $f_1$  and  $f_2$  are as in Eqs. (2) and (3) and  $f_m^*$ ,  $f_m^{**}$  are the outputs of the models  $SI_m$  and  $SA_m$ ,  $m = 1, 2$ , respectively. In this model Constraints (5)-(8) are regarded.



**Table 6**  
Number of variables and constraints  
in model  $SA_1$  to find IDPs for  $f_1$ .

Benchmark	TV	NC
200×10	2000	230
500×31	15500	593
1000×76	76000	1228
2000×200	400000	2600

#### 4.3.2. $LP$ -metric- $L^a$ , a $L$ model using the outputs of the $LSI^a$ and $LSA^a$ models and $LP$ -metric method to find efficient solutions for the $BO$ model

Applying the outcomes achieved from the  $L$  models based on the first approach of linearization, to be minimized, the objective function can be determined as in Eq. (35).

$$f = \left[ \sum_{m=1}^2 \left| \frac{l_m - l_m^*}{l_m^{**} - l_m^*} \right|^p \right]^{\frac{1}{p}} \quad (35)$$

where  $l_1$  and  $l_2$  are as in Eqs. (16) and (27), and  $l_m^*$ ,  $l_m^{**}$  are the outputs of the models  $LSI_m^a$  and  $LSA_m^a$ ,  $m = 1, 2$ , respectively. Constraints (5)-(6), (17)-(23), (28)-(30) are considered in this model.

#### 4.3.3. $LP$ -metric- $L^b$ , a $L$ model using the outputs of the $LSI^b$ and $LSA^b$ models and $LP$ -metric method to find efficient solutions for the $BO$ model

Using the results acquired from the  $L$  models based on the second approach of linearization, to be minimized, the objective function can be determined as in Eq. (35), in which  $l_1$  and  $l_2$  are as in Eqs. (24) and (31), and  $l_m^*$ ,  $l_m^{**}$  are the outputs of the models  $LSI_m^b$  and  $LSA_m^b$ ,  $m = 1, 2$ , respectively. Constraints (5)-(6), (20)-(23), (25)-(26), (32)-(33) are considered in this model.

For  $p = 1$ ,  $p = 2$ , and  $p = \infty$ , the distances are the street-block (or Manhattan), Euclidean, and Chebyshev. A Pareto optimal solution is obtained for each  $p$ . But the higher  $p$ , the higher the complexity. In this study, only  $p = 1$  is used to simplify.

## 5. Experimental results

We generate four benchmarks as 200×10, 500×31, 1000×76 and 2000×200, which are pointed out as the *Number of clients* × *Number of sectors*.  $Q = \lfloor \frac{n}{k} \rfloor$  is the target number of points per sector. In order to have diversity in the benchmarks,  $Q$  is equal to 20, 16, 13, and 10, respectively. Parameters  $\tau$  ranges inside the interval  $[0, 1]$ . Two-dimensional coordinates of clients and center of sectors and likewise clients' demands, are created according to  $N(50;10)$  and  $U(10;100)$ , which are normal and discrete uniform distributions, respectively [13,14,17,18].

The proposed models are not very complex, but the solution becomes difficult because the models are basically NL and also the number of variables and constraints in the benchmarks can be high, especially in the  $L$  models. For example, in the NL models, the number of the binary variables  $Z_{ij}$  is  $n \times k$ . Because of the Constraints (5), (6), (7), (8) respectively,  $n$  linear,  $k$  linear,  $k$  non-linear and  $k$  non-linear constraints are included in the model. As an example, the total number of variables and constraints for the model  $SA_1$  are given in Table 6, where  $TV$  and  $NC$  stand for the total number of variables and constraints, respectively.

Subsections 4.2.2 and 4.2.4 specify how the number of variables and constraints varies in the linear models.

A system with an Intel Core i5 processor, 2.4 GHz with 12 GB of RAM is utilized. IBM ILOG CPLEX Optimization Studio 12.9, GAMS 25.1.2, Lingo 19.0, Pymoo 4.2, and Pulp 2.4 are operated. From GAMS solvers BARON is used for the NL models, while CPLEX is used for  $L$  ones. Even if BARON is used for  $L$  models it is shifted to CPLEX automatically. In the tables of results, CPLEX refers to IBM ILOG CPLEX Optimization Studio, not the solver in GAMS. For the GA and NSGA-II utilized from the Pymoo library, we set the size of the population and offspring equal to 1000. We use a random sampling method and uniform crossover and mutation [16].

We suppose  $0 \leq \tau_{eqds} \leq 1$ , and  $0 \leq \tau_{eqde} \leq 1$ . Using Constraints (7)-(8), the values of  $\tau_{eqds}$  and  $\tau_{eqde}$  can be calculated as in Statements (36)-(37):

$$0 \leq \tau_{eqds} \leq 1 - \max\left(\frac{|Dc_j - \bar{D}c|}{\bar{D}c}\right), \quad j = 1, \dots, k \quad (36)$$

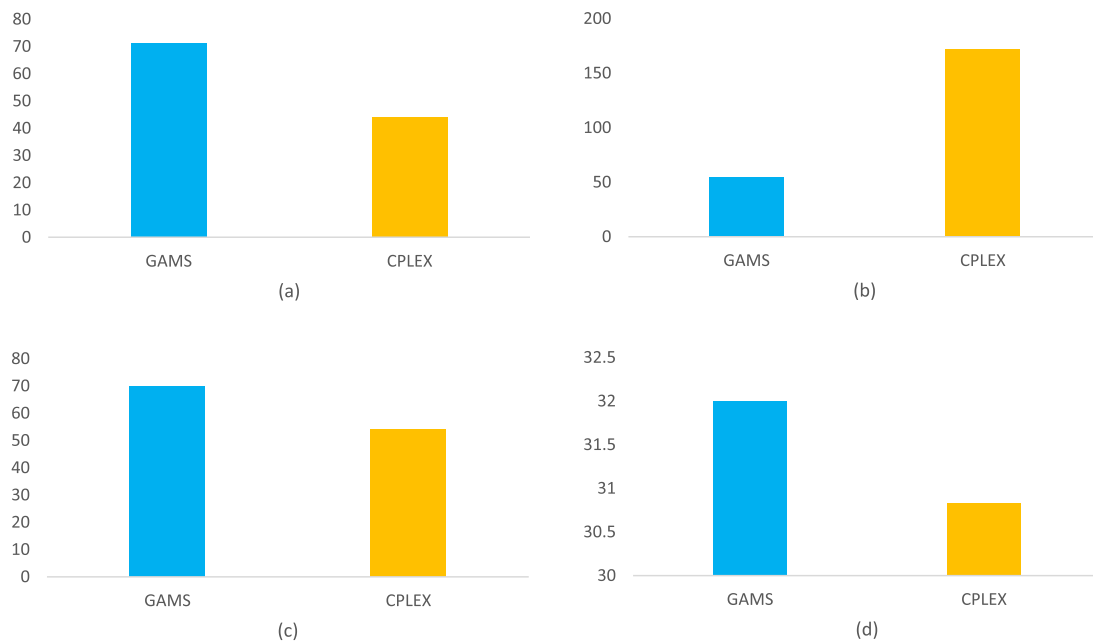
$$0 \leq \tau_{eqde} \leq 1 - \max\left(\frac{|De_j - \bar{D}e|}{\bar{D}e}\right), \quad j = 1, \dots, k \quad (37)$$

We employ GA to define the values of these parameters. Such that, running the GA for the model SI, starting from 1 and decreasing by 0.05 each time, the highest value that gives a feasible result is selected. We assign the values of these two parameters equally to manage them easily. Thus, the acquired values are as in Table 7. The reason for using GA for this purpose is that it provides results faster.

The results are presented in Tables 8-14. Using the same distributions we generate alternative instances to verify whether the acquired results depend on the employed benchmarks. Obtained outputs for the alternative benchmarks are given in Table 10. But the results in Tables 8-9 and 12-14 are for the main benchmarks.

**Table 7**  
Values of the tightness parameters in benchmarks.

Benchmark	$\tau_{eqde}$	$\tau_{eqde}$
200×10	0.75	0.75
500×31	0.5	0.5
1000×76	0.4	0.4
2000×200	0.1	0.1



**Fig. 3.** Visualization of the results of benchmark  $500 \times 31$  for Models (a)  $LSI_1^a$ , (b)  $LSI_1^b$ , (c)  $LSI_2^a$  and (d)  $LSI_2^b$ .

Although the models are not complex, it is not easy to find their optimal solutions. This is because they are basically NL and when they are linearized the number of variables and constraints grows. Therefore, the tables of results mostly report a feasible solution.

With GA and NSGA-II metaheuristics, only NL models are solved, because the linearization is done to reduce the complexity of the problem, which is not a handicap for these algorithms and they can easily solve the NL models. In the tables, *ST* stands for solution time in a format as minutes:seconds (mm:ss). Optimal results are displayed as *Op*. The maximum working time of the solvers is defined as 10 minutes. If no solution is obtained within this time, it is shown in the tables as NaN. If only a feasible solution is found in 10 minutes, this is signed by *F*.

As seen in Table 8, only GA is able to find solutions for all benchmarks of the model SI within the defined time, though their quality is not satisfactory. Although Pulp and Lingo are capable of solving L models, they only find results for the benchmark with the smallest size. Hence, their outcomes are not presented in Table 9. In the L models, although GAMS uses the CPLEX solver, as seen in Tables 9–10, there is a difference between its results and the IBM ILOG CPLEX Optimization Studio. It is observed that GAMS achieves better results for more benchmarks. Moreover, according to the results, linearization based on the first and second approaches does not seem to predominate each other. In Tables 8–10 the best achieved solution for each model is bold.

To understand if there are significant differences between the results of the solvers, we perform an analysis of variance (ANOVA) with a confidence level of 95%. We use the outputs of Table 9. We define factors as the type of solver, size of the benchmark, and model. So, the analysis is a three-way ANOVA. The null hypothesis is that the results of groups of factors are equal, against the alternative hypothesis that at least one of them is different.

As seen in Table 11, since the p-values for the factors of solver and size are less than 0.05, they are significant. However, the p-value for the factor of the model is greater than 0.05, and consequently, it is not significant. This is also valid for the interactions of the factor of the model with other ones, which are shown as Solver:Model and Size:Model.

While the ANOVA indicates a significant overall difference, it does not determine which specific groups differ from each other. To compare groups, we perform a multiple comparison test, which is a post hoc analysis. The p-value for the test is equal to zero. Two groups are formed based on the results of GAMS and IBM ILOG CPLEX Optimization Studio solvers, whose marginal means of groups are significantly different. Overall, based on the results of the ANOVA and multiple comparison tests, we can conclude that GAMS and IBM ILOG CPLEX Optimization Studio perform differently.

As seen in Fig. 3a–d, except for Model  $LSI_1^b$  of benchmark  $500 \times 31$ , IBM ILOG CPLEX has better performance than GAMS. But as it appears in Table 10, GAMS outperforms IBM ILOG CPLEX for other benchmarks.

**Table 8**

Results obtained for the model SI.

Benchmark	Model $SI_1$						Model $SI_2$					
	Lingo		GAMS		GA		Lingo		GAMS		GA	
	$f_1^*$	ST	$f_1^*$	ST	$f_1^*$	ST	$f_2^*$	ST	$f_2^*$	ST	$f_2^*$	ST
$200 \times 10$	NaN	10:00	0 (Op)	09:00	292 (F)	00:09	NaN	10:00	7 (F)	10:00	927 (F)	00:07
$500 \times 31$	NaN	10:00	NaN	10:00	1398 (F)	00:10	NaN	10:00	NaN	10:00	3265.55 (F)	00:08
$1000 \times 76$	NaN	10:00	NaN	10:00	3872.63 (F)	00:33	NaN	10:00	NaN	10:00	11081.1 (F)	00:34
$2000 \times 200$	NaN	10:00	NaN	10:00	11933.44 (F)	01:29	NaN	10:00	NaN	10:00	28450.25 (F)	02:17

**Table 9**

Results obtained for the model LSI.

Benchmark	Model $LSI_1^a$				Model $LSI_1^b$				Model $LSI_2^a$				Model $LSI_2^b$			
	GAMS		CPLEX		GAMS		CPLEX		GAMS		CPLEX		GAMS		CPLEX	
	$l_1^*$	ST	$l_1^*$	ST	$l_1^*$	ST	$l_1^*$	ST	$l_2^*$	ST	$l_2^*$	ST	$l_2^*$	ST	$l_2^*$	ST
$200 \times 10$	0 (Op)	00:13	0 (Op)	00:30	0 (Op)	01:02	0 (Op)	00:30	5 (F)	10:00	5 (F)	10:00	5 (F)	10:00	5 (F)	10:00
$500 \times 31$	20 (F)	10:00	26.06 (F)	10:00	17 (F)	10:00	61.10 (F)	10:00	71 (F)	10:00	55.54 (F)	10:00	55 (F)	10:00	40.45 (F)	10:00
$1000 \times 76$	222 (F)	10:00	306.63 (F)	10:00	158 (F)	10:00	593 (F)	10:00	202 (F)	10:00	268 (F)	10:00	214 (F)	10:00	366 (F)	10:00
$2000 \times 200$	857 (F)	10:00	NaN	10:00	979 (F)	10:00	NaN	10:00	2669 (F)	10:00	NaN	10:00	2791 (F)	10:00	NaN	10:00

**Table 10**

Results of the model LSI for the alternative benchmarks.

Benchmark	Model $LSI_1^a$				Model $LSI_1^b$				Model $LSI_2^a$				Model $LSI_2^b$			
	GAMS		CPLEX		GAMS		CPLEX		GAMS		CPLEX		GAMS		CPLEX	
	$l_1^*$	ST	$l_1^*$	ST	$l_1^*$	ST	$l_1^*$	ST	$l_2^*$	ST	$l_2^*$	ST	$l_2^*$	ST	$l_2^*$	ST
$200 \times 10$	0 (Op)	05:55	0 (Op)	01:47	1 (F)	10:00	2 (F)	10:00	7 (F)	10:00	6 (F)	10:00	5 (F)	10:00	5 (F)	10:00
$500 \times 31$	71 (F)	10:00	43.87 (F)	10:00	55 (F)	10:00	172 (F)	10:00	70 (F)	10:00	54.25 (F)	10:00	32 (F)	10:00	30.83 (F)	10:00
$1000 \times 76$	369 (F)	10:00	656 (F)	10:00	360 (F)	10:00	NaN	10:00	293 (F)	10:00	835 (F)	10:00	255 (F)	10:00	895.55 (F)	10:00
$2000 \times 200$	2333 (F)	10:00	NaN	10:00	2347 (F)	10:00	NaN	10:00	2663 (F)	10:00	NaN	10:00	3393 (F)	10:00	NaN	10:00

**Table 11**

The results of the three-way ANOVA for the outputs of Table 9.

Source	Sum of Square	Degree of Freedom	Mean Square	F-Value	P-Value
Solver	4837880385	1	4837880385	36643.61	0
Size	15484033011.9	3	5161344337.3	39093.63	0
Model	396864	3	132288	1	0.44
Solver:Size	14439241630	3	4813080543.3	36455.77	0
Solver:Model	505963.9	3	168654.6	1.28	0.34
Size:Model	1275973	9	141774.8	1.07	0.46
Error	1188226.8	9	132025.2		
Total	34764522054.6	31			

**Table 12**

Acquired results for the model SA by GA.

Benchmark	Model $SA_1$		Model $SA_1$	
	$f_1^*$	ST	$f_2^*$	ST
$200 \times 10$	482	00:07	406	00:06
$500 \times 31$	2116.32	00:06	7167.54	00:08
$1000 \times 76$	3872.63	00:33	11081.1	00:33
$2000 \times 200$	12071.99	01:34	11933.44	01:29

**Table 13**

Non-dominated solutions obtained for the benchmarks of the model BO by NSGA-II from Pymoo library.

Benchmark	$f_1$	$f_2$	$ND$	$ST$
$200 \times 10$	328.4 423.6	940 927	2	00:05
$500 \times 31$	1753.74 1468.06 1883.8	4020.64 4274.64 3265.59	3	00:07
$1000 \times 76$	3872.63	11081.1	1	00:32
$2000 \times 200$	11933.44 12071.99	29793 29535.25	2	01:24

**Table 14**Obtained solutions for the benchmarks of the MO model using LP-metric-NL, LP-metric- $L^a$  and LP-metric- $L^b$  methods.

Benchmark	LP-metric-NL				LP-metric- $L^a$				LP-metric- $L^b$			
	$f_1$	$f_2$	$f$	ST	$f_1$	$f_2$	$f$	ST	$f_1$	$f_2$	$f$	ST
$200 \times 10$	328.4	940	3.01	00:05	26	27	0.11	10:00	<b>23</b>	<b>25</b>	<b>0.10</b>	10:00
$500 \times 31$	1468.04	4247.64	1.28	00:06	<b>95</b>	<b>217</b>	<b>0.05</b>	10:00	<b>75</b>	<b>220</b>	<b>0.05</b>	10:00
$1000 \times 76$	3872.63	11081.1	2	00:32	<b>286</b>	<b>618</b>	<b>0.06</b>	10:00	355	1074	0.13	10:00
$2000 \times 200$	11140.4	28450.25	3.67	02:17	<b>1442</b>	<b>4367</b>	<b>0.25</b>	10:00	1619	5025	0.32	10:00

It should be noted that for the LP relaxation of the LSI model, in all benchmarks, both GAMS and IBM ILOG CPLEX Optimization Studio find the optimal result in less than one minute. What is meant by LP relaxation is just that the binary decision variable in Eq. (1) is defined continuously in the interval [0, 1].

GAMS, IBM ILOG CPLEX Optimization Studio, and Lingo could not provide solutions for the SA and LSA models, reporting that the models are unbound. It may be useful to apply additional constraints to avoid this, but this is not the subject of this study. Because the aim of this work is to solve the presented model with existing constraints. Nevertheless, the employed GA gives results for SA as in Table 12, which are supposed as ADPs.

As mentioned before, the NL types of the BO model are also solved with NSGA-II available in the Pymoo library of Python. The results are given in Table 13, where  $ND$  shows the number of obtained non-dominated solutions.

It has been mentioned earlier that the optimal solutions of the SO models will be utilized as IDPs and ADPs for the BO model, while according to the acquired results, most of them are from the feasible ones. This can be regarded as a limit for the study. Using the IDPs ve ADPs in the LP-metric method, defining  $p = 1$ , outputs are gained as in Table 14.

In addition to  $f_1$  and  $f_2$ , the values of  $f$  are also given, which is defined as in Eqs. (34) and (35). As it has been mentioned earlier, the results in Table 12 are regarded as ADPs in all models and  $f_m^{**} = f_m^{**}$ ,  $m = 1, 2$ . The LP-metric-NL is solved with GA while the LP-metric- $L^a$  and LP-metric- $L^b$  are handled with GAMS. Related outputs are presented in Table 14 and Fig. 4, where non-dominated solutions are bold. As seen, some of the non-dominated solutions achieved by the LP-metric method for benchmarks  $200 \times 10$ ,  $500 \times 31$ ,  $1000 \times 76$  are the same as those in Table 12. Even for benchmarks  $2000 \times 200$ , the solution provided by the LP-metric-NL method dominates the ones in Table 12. Though, the results of the LP-metric- $L^a$  and LP-metric- $L^b$  are significantly better than the LP-metric-NL, which is also due to the performance of the CPLEX solver in GAMS.

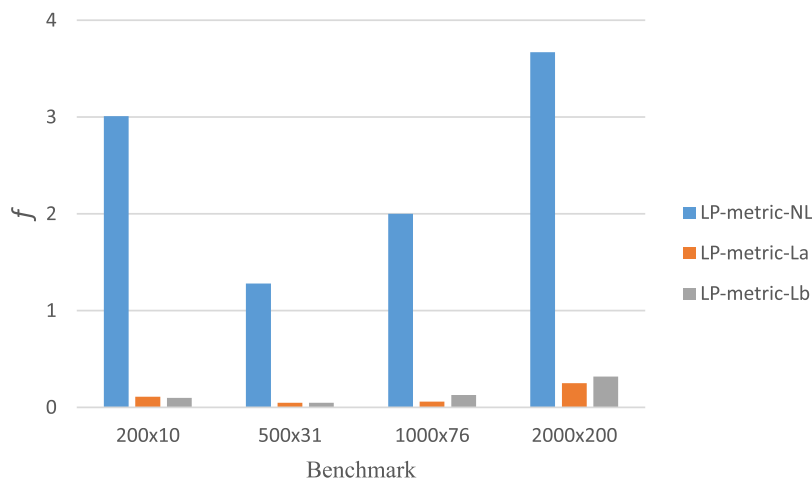


Fig. 4. Visualization of the results presented in Table 14.

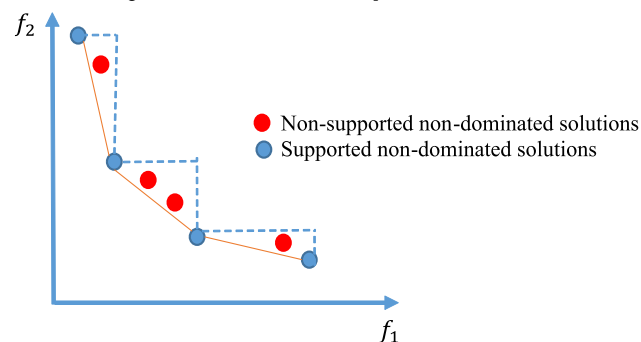


Fig. 5. Supported and non-supported non-dominated solutions.

As mentioned earlier, the reason for the definition as  $p = 1$  is to don't ascend the complexity of the problem. But consequently, some of the solutions on the Pareto front are overlooked. It is possible to find other non-dominated solutions with different values of  $p$  [65]. The models designed based on the LP-metric method can only achieve a few of the supported non-dominated solutions. Such solutions are located on the smallest convex polyhedron that includes the feasible region, which is called the convex hull. Whereas the unsupported ones are inside the convex hull. Examples of these solutions are depicted in Fig. 5. Unlike the unsupported non-dominated solutions, the supported one can be uncovered by applying weighted sum or similar methods like LP-metric. Discovering unsupported non-dominated solutions is a challenge for mathematical programming techniques [66–68]. Based on the similarity between the LP-metric approach and the weighted sum method, it can be emphasized that it is not always possible to identify all non-dominated solutions using this approach.

## 6. Conclusion and future research directions

In this study, for the first time in the literature, a comparison is made between various solvers like Lingo, IBM ILOG CPLEX Optimization Studio, GAMS, and also Pulp, and Pymoo libraries from Python for solving SP. The comparison is made based on a new BO model for an SP. The difficulties of solving the model, which is basically NL, are discussed. The effects of objective functions and the non-linearity of some constraints on this difficulty are mentioned. An approach that had not been used in the literature before is used to solve the model, in which, at first, the BO model is divided into two NL SO models. The results of the SO models, when the objective function is to be minimized and maximized, are assumed to form IDPs and ADPs for the respective objective function in the BO model. Also, the models, which are basically NL, are linearized to solve the BO model easier. Both mathematical programming and metaheuristics are used for the solution. Although the models in the study are not complex, solutions are difficult due to non-linearity and also because of the large number of variables in some benchmarks when they are linearized. For these reasons, solvers are not very efficient in solving models, and in general, during the defined run time, they could not find optimal solutions. Solvers performed better on the L models than the NL ones. Usually, GAMS is known to be one of the best software for operational research. In this study, GAMS provides better results than Lingo and Pulp. Even though GAMS operates the same solver as IBM ILOG CPLEX Optimization Studio for L models, it is slightly better. Among the solvers, the weakest performance is for Pulp.

The applied SO and MO metaheuristics are able to yield feasible solutions in reasonable times though they couldn't find optimal results. Their run times are much less than the other solvers but the quality of the results is not satisfactory. One of the reasons for this matter can be that the used metaheuristics are from a generic open-source library and they are not designed problem-oriented.

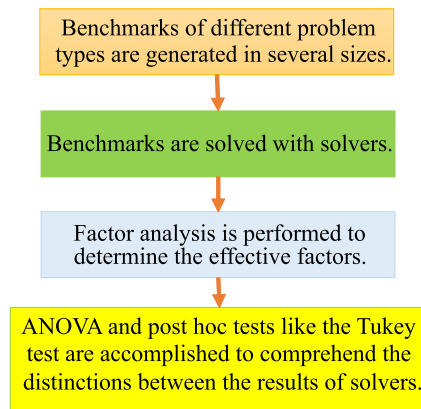


Fig. 6. A framework for comparison between the performance of solvers.

Nevertheless, it may be useful to use the algorithms of such libraries for solving large-scale problems, where it is not easy to find optimal solutions.

The models defined based on the LP-metric method achieve good results in terms of providing a non-dominated solution for each benchmark. Therefore, it would be said that the approach used for the first time in the literature of MO SP can be useful in finding non-dominant solutions. If more types of distance are used in the formulation, it could be possible to acquire more non-dominated solutions, but in this case, the complexity of the models would increase. This can be seen as a limit for this method. Notwithstanding, it can be concluded that the approach suggested in this study for solving MO SP is practical. The approach can be summarized as follows: dividing the BO model into some SOs ones, finding IDP and ADP for each objective function, reformulating and solving the BO model, and using the IDPs and ADPs in the LP-metric approach.

A limitation of this study is that IDPs and ADPs are got from different approaches. It is reported by the solvers that the maximization of the models is unbound. Although this matter is not an impediment to the completion of this work, it showed that it is better to reformulate the models, in future studies.

Figuring out the deficiencies of the metaheuristics employed in this study, in future studies, it is planned to propose a metaheuristic specific to solve MO SPs, in order to cope with the difficulties arising from a large number of variables and non-linearity in the models.

Performance metrics are used for quantifying performance and comparison of MO methods. It is crucial to use metrics to do comparisons between MO EAs designed based on the concept of Pareto optimality. In most MO problems, the Pareto optimal set is unattainable, correspondingly the measure of performance is challenging. Pareto front generation in MO problems needs high computational effort, especially in large-size problems. So its formation in a short time and with a small memory represents the efficiency of the employed algorithm. Convergence, coverage, and success rate measures are used to compare MO algorithms designed based on the Pareto approach. There are also other measures based on accuracy and variance. Accuracy shows generated non-dominated and the best-known solutions closeness. Coverage and variance have alike definitions. Coverage indicates solutions diversity and distribution. Variance represents the non-dominated front maximum range, which is covered by the generated solutions, for each objective. Diversity-based criteria measure the distances of the solutions in the first front of the final population. This front solution set is compared with a uniform distribution and its deviation is calculated. If an algorithm achieves more uniformly distributed solutions with less average deviation from the first front, it has good performance. Furthermore, criteria are categorized as quality, spacing, and diversification metrics. The obtained non-dominated solutions of algorithms are put together and their ratio is calculated as the quality metric. Spacing metric measures the uniformity of the solutions, while the diversification metric measures the spread of the solution set. There are also metrics such as net front contribution ratio and mean relative percentage increase to measure non-dominated solution set quality [69–76].

A significant limitation of the study is that the comparisons are over limited benchmarks and without a comprehensive statistical design of experiments [13]. In particular, this matter gains importance as the performance variability of the solvers is evident in the results. Likewise, in addition to SPs, benchmarks of similar problems such as p-median [77] should also be contained in the comparison. Statistical analyzes should be performed within a certain confidence interval, taking into account the different levels of various factors which can be problem type, benchmark size, solution, and linearization method [78]. The impacts of factors can be evaluated by factor analysis. The discrepancy between solvers' performance should be explored by designing hypotheses, performing ANOVA, and also post hoc methods such as Tukey's test [79,80]. A possible framework is summarized in Fig. 6.

Currently, there are not enough studies about the performance of solvers in the literature. In case of existing enough studies on this subject, a meta-analysis can be done, which is a research approach that can explore publications' bias and can draw a broader conclusion. It employs statistical methods to quantitatively compare and also combines the results of multiple studies [81,82].

In the literature, not enough attention has been given to the MADM techniques outlined in Fig. 2 to solve SPs. In real-life problems, DMs usually assign dissimilar priorities to criteria, in which case both compensatory and non-compensatory MADM can be beneficial.

Integrating MIP techniques with neural networks can provide an intelligence search on the feasible space [83,84], which will be considered in future works.

## CRediT authorship contribution statement

Aydin Teymourifar: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability statement

Data will be made available on request.

## Acknowledgements

This work was supported by The Portuguese Foundation for Science and Technology (FCT) in the framework of the project CEECINST/00137/2018.

The financial support from FCT (through the project UIDB/00731/2020) is also gratefully acknowledged.

## Appendix A

Table of abbreviations

Abbreviations	Definition
ADP	Anti-ideal point for an objective function
AHP	Analytical hierarchy process
ANOVA	Analysis of variance
ANP	Analytic network process
BO	Bi-objective
DM	Decision maker
EA	Evolutionary algorithm
F	Feasible solution
GA	Genetic algorithm
GP	Goal programming
IDP	Ideal point for an objective function
ILP	Integer linear programming
Inf	Infimum
INLP	Integer non-linear programming
IQP	Integer quadratic programming
L	Linear
LP	Linear programming
LSA	Single-objective model with linearized objective function to find anti-ideal points of the bi-objective model
LSI	Single-objective model with linearized objective function to find ideal points of the bi-objective model
MADM	Multi-attribute decision making
MCDM	Multi-criteria decision making
MILP	Mixed integer linear programming
MINLP	Mixed integer non-linear programming
MIP	Mixed integer programming
MO	Multi-objective
MODM	Multi-objective decision making
NaN	No solution obtained
NC	Number of constraints
ND	Number of obtained non-dominated solutions
NL	Non-linear
NLP	Non-linear programming
NSGA-II	Non-dominated sorting genetic algorithm
Op	Optimal solution
p	Emphasis level in the LP-metric method
Pa	Pareto solution
SA	Single-objective model to find anti-ideal points of the bi-objective model
SI	Single-objective model to find ideal points of the bi-objective model
SO	Single-objective
SP	Sectorization problem
ST	Solution time
STEM	Step method
Sup	Supremum
SWTM	Surrogate worth trade-off method
TOPSIS	Technique for Order Preference by Similarity to Ideal Solution
TV	Total number of variables

## References

- [1] T. Koch, T. Berthold, J. Pedersen, C. Vanaret, Progress in mathematical programming solvers from 2001 to 2020, *EURO J. Comput. Optim.* 10 (2022) 100031.
- [2] A. Lodi, A. Tramontani, Performance variability in mixed-integer programming, in: *Theory Driven by Influential Applications*, INFORMS, 2013, pp. 1–12.
- [3] A. Neumaier, O. Shcherbina, W. Huyer, T. Vinkó, A comparison of complete global optimization solvers, *Math. Program.* 103 (2005) 335–356.
- [4] J. Kronqvist, D.E. Bernal, A. Lundell, I.E. Grossmann, A review and comparison of solvers for convex MINLP, *Optim. Eng.* 20 (2019) 397–455.
- [5] B. Meindl, M. Templ, Analysis of commercial and free and open source solvers for linear optimization problems, in: *Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS*, 2012, p. 20.
- [6] A. Luppold, D. Oehlert, H. Falk, Evaluating the performance of solvers for integer-linear programming, 2018.
- [7] A. Guasque, P. Balbastre, Evaluation and comparison of integer programming solvers for hard real-time scheduling, *IEICE Trans. Inf. Syst.* 105 (10) (2022) 1726–1733.
- [8] R. Schuster, J.O. Hanson, M. Strimas-Mackey, J.R. Bennett, Exact integer linear programming solvers outperform simulated annealing for solving conservation planning problems, *PeerJ* 8 (2020) e9258.
- [9] E.B. Tirkolaee, A. Goli, G.W. Weber, Fuzzy mathematical programming and self-adaptive artificial fish swarm algorithm for just-in-time energy-aware flow shop scheduling problem with outsourcing option, *IEEE Trans. Fuzzy Syst.* 28 (11) (2020) 2772–2783.
- [10] A. Salhi, G. Alsoufi, X. Yang, An evolutionary approach to a combined mixed integer programming model of seaside operations as arise in container ports, *Ann. Oper. Res.* 272 (2019) 69–98.
- [11] H. Liu, A. Erdogan, R. Lin, H.S.J. Tsao, Mathematical models of political districting for more representative governments, *Comput. Ind. Eng.* 140 (2020) 62–65.
- [12] N. Wichapa, P. Khokhajaikiat, Solving multi-objective facility location problem using the fuzzy analytical hierarchy process and goal programming: a case study on infectious waste disposal centers, *Oper. Res. Perspect.* 4 (2017) 39–48.
- [13] A. Teymourifar, A.M. Rodrigues, J.S. Ferreira, A comparison between NSGA-II and NSGA-III to solve multi-objective sectorization problems based on statistical parameter tuning, in: *2020 24th International Conference on Circuits, Systems, Communications and Computers (CSCC)*, IEEE, 2020, pp. 64–74.
- [14] V. Romanciuc, C. Lopes, A. Teymourifar, A.M. Rodrigues, J.S. Ferreira, C. Oliveira, E.G. Öztürk, An integer programming approach to sectorization with compactness and equilibrium constraints, in: *Innovations in Industrial Engineering*, Springer International Publishing, Cham, 2021, pp. 185–196.
- [15] A. Teymourifar, A.M. Rodrigues, J.S. Ferreira, C. Lopes, A comparison between optimization tools to solve sectorization problem, in: *International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences*, Springer, Cham, 2021, pp. 40–50.
- [16] J. Blank, K. Deb, Pymoo: multi-objective optimization in python, *IEEE Access* 8 (2020) 89497–89509.
- [17] A. Teymourifar, A.M. Rodrigues, J.S. Ferreira, A comparison between simultaneous and hierarchical approaches to solve a multi-objective location-routing problem, in: *Graphs and Combinatorial Optimization: from Theory to Applications*, Springer, 2021, pp. 251–263.
- [18] A. Teymourifar, A.M. Rodrigues, J.S. Ferreira, C. Lopes, C. Oliveira, V. Romanciuc, A two-stage method to solve location-routing problems based on sectorization, in: *International Conference Innovation in Engineering*, Springer, 2021, pp. 148–159.
- [19] G. Zhou, H. Min, M. Gen, The balanced allocation of customers to multiple distribution centers in the supply chain network: a genetic algorithm approach, *Comput. Ind. Eng.* 43 (2002) 251–261.
- [20] M. Lin, K.S. Chin, L. Ma, K.L. Tsui, A comprehensive multiobjective mixed integer nonlinear programming model for an integrated elderly care service districting problem, *Ann. Oper. Res.* 291 (2020) 499–529.
- [21] K. Sudtachat, M.E. Mayorga, S. Chanta, L.A. Albert, Joint reallocation and districting using a nested compliance model for EMS systems, *Comput. Ind. Eng.* 142 (2020) 106327.
- [22] L.S. De Assis, P.M. Franca, F.L. Usberti, A redistricting problem applied to meter reading in power distribution networks, *Comput. Oper. Res.* 41 (2014) 65–75.
- [23] F. Liberatore, M. Camacho-Collados, B. Vitoriano, Police districting problem: literature review and annotated bibliography, in: *Optimal Districting and Territory Design*, 2020, pp. 9–29.
- [24] R.Z. Rios-Mercado, H.J. Escalante, Grasp with path relinking for commercial districting, *Expert Syst. Appl.* 44 (2016) 102–113.
- [25] J.A. Ferland, G. Guénette, Decision support system for the school districting problem, *Oper. Res.* 38 (1990) 15–21.
- [26] W. Pawlak, V. Goel, D. Rothenberg, C. Brinton, Comparison of algorithms for the dynamic resectorization of airspace, in: *Guidance, Navigation, and Control Conference and Exhibit*, 1998, p. 4106.
- [27] E.S. Stein, P.S. Della Rocco, R.L. Sollenberger, Dynamic resectorization in air traffic control: a human factors perspective, 2005.
- [28] H. Trandac, P. Baptiste, V. Duong, Airspace sectorization with constraints, *RAIRO. Rech. Opér.* 39 (2) (2005) 105–122.
- [29] J. Tang, S. Alam, C. Lokan, H.A. Abbass, A multi-objective evolutionary method for dynamic airspace re-sectorization using sectors clipping and similarities, in: *2012 IEEE Congress on Evolutionary Computation*, IEEE, 2012, pp. 1–8.
- [30] R.J. Teixeira, I. Romani, P.S. Cugnasca, Analysis of the impact of the use of the dynamic resectorization in air traffic control workload by computational simulations, in: *Air Transport Research, Society World Conference*, 2007.
- [31] G. Mohammed, M. El Bekkaye, Fuzzy dynamic airspace sectorization problem, in: *Machine Intelligence and Data Analytics for Sustainable Future Smart Cities*, Springer, Cham, 2021, pp. 229–250.
- [32] P. Ambasth, N.M. Vali Mohamad, R. Jain, H. Subramaniyam, L. Muthukaruppan, S. Gautam, Dynamic sectorization and parallel processing for device-to-device (D2D) resource allocation in 5G and B5G cellular network, 2021.
- [33] A. Antunovic, E. Amiri, C. Biefel, G. Steinbock, J. Aulin, F. Liers, W. Gerstacker, Dynamic sectorization in multi-panel massive MIMO systems, in: *2021 IEEE 4th 5G World Forum (5GWF)*, IEEE, 2021, pp. 70–75.
- [34] A.J. Ulusoy, H.A. Mahmoud, F. Pecci, E.C. Keedwell, I. Stoianov, Bi-objective design-for-control for improving the pressure management and resilience of water distribution networks, *Water Res.* 118914 (2022).
- [35] K. Zhang, H. Yan, H. Zeng, K. Xin, T. Tao, A practical multi-objective optimization sectorization method for water distribution network, *Sci. Total Environ.* 656 (2019) 1401–1412.
- [36] S. Kulkarni, R. Ganesan, L. Sherry, Static sectorization approach to dynamic airspace configuration using approximate dynamic programming, in: *2011 Integrated Communications, Navigation, and Surveillance Conference Proceedings*, IEEE, (J2-1), 2011.
- [37] A. Teymourifar, Simulation-based optimization for resectorization in healthcare systems, *J. Simul.* (2022) 1–20.
- [38] M. Ehrgott, *Multicriteria Optimization*, 2nd edition, Springer Science and Business Media, ISBN 3-540-21398-8, 2005, 323 pp.
- [39] K. Miettinen, *Nonlinear Multiobjective Optimization*, Springer Science & Business Media, 2012, p. 12.
- [40] R.E. Steuer, *Multiple criteria optimization. Theory, computation and applications*, 1986.
- [41] H. Taherdoost, M. Madanchian, Multi-criteria decision making (MCDM) methods and concepts, *Encyclopedia* 3 (1) (2023) 77–87.
- [42] G.S. Liang, Fuzzy MCDM based on ideal and anti-ideal concepts, *Eur. J. Oper. Res.* 112 (3) (1999) 682–691.
- [43] M.S. Kuo, G.H. Tzeng, W.C. Huang, Group decision-making based on concepts of ideal and anti-ideal points in a fuzzy environment, *Math. Comput. Model.* 45 (3–4) (2007) 324–339.
- [44] Y.M. Wang, K.S. Chin, Y. Luo, Cross-efficiency evaluation based on ideal and anti-ideal decision making units, *Expert Syst. Appl.* 38 (8) (2011) 10312–10319.
- [45] Ž. Stević, D. Pamućar, A. Puška, P. Chatterjee, Sustainable supplier selection in healthcare industries using a new MCDM method: measurement of alternatives and ranking according to Compromise solution (MARCOS), *Comput. Ind. Eng.* 140 (2020) 106231.
- [46] Z. Shang, X. Yang, D. Barnes, C. Wu, Supplier selection in sustainable supply chains: using the integrated BWM, fuzzy Shannon entropy, and fuzzy MULTIMOORA methods, *Expert Syst. Appl.* 195 (2022) 116567.



- [47] I.C. Gil-García, A. Ramos-Escudero, M.S. García-Cascales, H. Dagher, A. Molina-García, Fuzzy GIS-based MCDM solution for the optimal offshore wind site selection: the Gulf of Maine case, *Renew. Energy* 183 (2022) 130–147.
- [48] S. Petchrompo, A. Wannakirait, A.K. Parlikad, Pruning Pareto optimal solutions for multi-objective portfolio asset management, *Eur. J. Oper. Res.* 297 (1) (2022) 203–220.
- [49] P. Wei, C. Surakitbanharn, S. Landry, D. Sun, Workload evaluation of sectorized air traffic control and stream management, in: 2012 Integrated Communications, Navigation and Surveillance Conference, IEEE, 2012, p. N2-1.
- [50] S. Hajebi, E. Roshani, N. Cardozo, S. Barrett, A. Clarke, S. Clarke, Water distribution network sectorisation using graph theory and many-objective optimisation, *J. Hydroinform.* 18 (1) (2015) 77–95.
- [51] H. Oktal, K. Yaman, R. Kasimbeyli, A mathematical programming approach to optimum airspace sectorisation problem, *J. Navig.* 73 (3) (2020) 599–612.
- [52] J. Benítez, S. Carpitella, A. Certa, A.E. Ilaya-Ayza, J. Izquierdo, Consistent clustering of entries in large pairwise comparison matrices, *J. Comput. Appl. Math.* 343 (2018) 98–112.
- [53] W. Ogryczak, T. Sliwinski, Lexicographic max-min optimization for efficient and fair bandwidth allocation, in: International Network Optimization Conference (INOC), 2007, pp. 22–25.
- [54] P. Praneethpholkrang, S. Kanjanawattana, A multi-objective optimization model for shelter location-allocation in response to humanitarian relief logistics, *Asian J. Shipp. Logist.* 37 (2) (2021) 149–156.
- [55] Y.Y. Haimes, W.A. Hall, Multiobjectives in water resource systems analysis: the surrogate worth trade off method, *Water Resour. Res.* 10 (4) (1974) 615–624.
- [56] N.J. Singh, J.S. Dhillon, D.P. Kothari, Surrogate worth trade-off method for multi-objective thermal power load dispatch, *Energy* 138 (2017) 1112–1123.
- [57] R. Benayoun, J. De Montgolfier, J. Tergny, O. Larichev, Linear programming with multiple objective functions: step method (STEM), *Math. Program.* 1 (1) (1971) 366–375.
- [58] S. Bid, G. Siddique, Human risk assessment of Panchet dam in India using TOPSIS and WASPAS multi-criteria decision-making (MCDM) methods, *Heliyon* 5 (6) (2019) e01956.
- [59] G.M. Magableh, M.Z. Mistarihi, Applications of MCDM approach (ANP-TOPSIS) to evaluate supply chain solutions in the context of COVID-19, *Heliyon* 8 (3) (2022) e09062.
- [60] J. Wu, H. Wang, L. Yao, Z. Kang, Q. Zhang, Comprehensive evaluation of voltage stability based on EW-AHP and fuzzy-TOPSIS, *Heliyon* 5 (10) (2019) e02410.
- [61] D.S. Costa, H.S. Mamede, M.M. da Silva, A method for selecting processes for automation with AHP and TOPSIS, *Heliyon* 9 (3) (2023).
- [62] N. Vafaei, V. Delgado-Gomes, C. Agostinho, R. Jardim-Goncalves, Analysis of data normalization in decision making process for ICU's patients during the pandemic, *Proc. Comput. Sci.* 214 (2022) 809–816.
- [63] M.J. Alves, J.P. Costa, An exact method for computing the nadir values in multiple objective linear programming, *Eur. J. Oper. Res.* 198 (2) (2009) 637–646.
- [64] R. Benetto, J.H. van Vuuren, Multi-objective evolutionary search strategies in constraint programming, *Oper. Res. Perspect.* 8 (2021) 100177.
- [65] J.C. Ferreira, C.M. Fonseca, A. Gaspar-Cunha, Methodology to select solutions from the Pareto-optimal set: a comparative study, in: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, 2007, pp. 789–796.
- [66] B. Ghasemishabankareh, M. Ozlen, X. Li, NSGA-II for solving multiobjective integer minimum cost flow problem with probabilistic tree-based representation, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, Cham, 2019, pp. 541–552.
- [67] A. Eusébio, J.R. Figueira, Finding non-dominated solutions in bi-objective integer network flow problems, *Comput. Oper. Res.* 36 (9) (2009) 2554–2564.
- [68] A. Eusébio, J.R. Figueira, On the computation of all supported efficient solutions in multi-objective integer network flow problems, *Eur. J. Oper. Res.* 199 (1) (2009) 68–76.
- [69] R. Suresh, K. Mohanasundaram, Pareto archived simulated annealing for job shop scheduling with multiple objectives, *Int. J. Adv. Manuf. Technol.* 29 (1–2) (2006) 184–196.
- [70] S. Mirjalili, A. Lewis, Novel performance metrics for robust multi-objective optimization algorithms, *Swarm Evol. Comput.* 21 (2015) 1–23.
- [71] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, in: International Conference on Parallel Problem Solving from Nature, Springer, 2000, pp. 849–858.
- [72] E.S. Nicoară, Performance measures for multi-objective optimization algorithms, *Bul. Univ. Petrol-Gaze Ploiești, Ser. Mat. Inform. Fiz.* 59 (1) (2007) 19–28.
- [73] R. Tavakkoli-Moghaddam, M. Azarkish, A. Sadeghnejad-Barkousaraie, A new hybrid multi-objective Pareto archive pso algorithm for a bi-objective job shop scheduling problem, *Expert Syst. Appl.* 38 (9) (2011) 10812–10821.
- [74] D. Knowles, Local-search and hybrid evolutionary algorithms for Pareto optimization, PhD thesis, Department of Computer Science, University of Reading, 2002, Submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computer Science.
- [75] H. Ishibuchi, T. Yoshida, T. Murata, Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling, *IEEE Trans. Evol. Comput.* 7 (2) (2003) 204–223.
- [76] E. Zitzler, M. Laumanns, L. Thiele, et al., SPEA2: improving the strength Pareto evolutionary algorithm, in: Eurogen, vol. 3242, 2001, pp. 95–100.
- [77] M. Aider, A.I. Azzi, M. Hifi, The  $\epsilon$ -constraint as a learning strategy in the population-based algorithm: the case of bi-objective obnoxious p-median problems, *Knowl.-Based Syst.* 265 (2023) 110363.
- [78] M. Asghari, A.M. Fathollahi-Fard, S.M.J. Mirzapour Al-e-hashem, M.A. Dulebenets, Transformation and linearization techniques in optimization: a state-of-the-art survey, *Mathematics* 10 (2) (2022) 283.
- [79] L.F. Gutiérrez-Mosquera, S. Arias-Giraldo, A. Zuluaga-Meza, Landfill leachate treatment using hydrodynamic cavitation: exploratory evaluation, *Heliyon* 8 (3) (2022) e09019.
- [80] G. Ozturk, O. Bahadir, A. Teymourifar, Extracting priority rules for dynamic multi-objective flexible job shop scheduling problems using gene expression programming, *Int. J. Prod. Res.* 57 (10) (2019) 3121–3137.
- [81] R.A. Villano, C.D.T. Tran, Survey on technical efficiency in higher education: a meta-fractional regression analysis, *Pac. Econ. Rev.* 26 (1) (2021) 110–135.
- [82] D. Juandi, Y.S. Kusumah, M. Tamur, K.S. Perbowo, T.T. Wijaya, A meta-analysis of Geogebra software decade of assisted mathematics learning: what to learn and where to go?, *Heliyon* 7 (5) (2021) e06953.
- [83] V. Nair, S. Bartunov, F. Gimeno, I. Von Glehn, P. Lichocki, I. Lobov, Y. Zwols, Solving mixed integer programs using neural networks, arXiv preprint, arXiv: 2012.13349, 2020.
- [84] A. Teymourifar, G. Ozturk, A neural network-based hybrid method to generate feasible neighbors for flexible job shop scheduling problem, *Univers. J. Appl. Math.* 6 (1) (2018) 1–16.