

Tilahun, Surafel Lulseged

Article

Feasibility reduction approach for hierarchical decision making with multiple objectives

Operations Research Perspectives

Provided in Cooperation with:

Elsevier

Suggested Citation: Tilahun, Surafel Lulseged (2019) : Feasibility reduction approach for hierarchical decision making with multiple objectives, Operations Research Perspectives, ISSN 2214-7160, Elsevier, Amsterdam, Vol. 6, pp. 1-8, <https://doi.org/10.1016/j.orp.2018.100093>

This Version is available at:

<https://hdl.handle.net/10419/246375>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by-nc-nd/4.0>



Feasibility reduction approach for hierarchical decision making with multiple objectives

Surafel Lulseged Tilahun

Department of Mathematical Sciences, University of Zululand Private Bag X1001, KwaDlangezwa 3886, South Africa



ARTICLE INFO

Keywords:

Hierarchical decision making
Multilevel programming
Multiobjective optimization
Fuzzy theory
Cooperative decision making

ABSTRACT

Several decision making problems with multiple decision makers in a hierarchical decision making scenario can be best described using multilevel optimization problem. Different organizational structure adapt a hierarchical decision making system where each decision maker controls parts of the decision variables and is affected by the decision of other decision makers. Recently, inspired by natural adaptation, a metaheuristic based algorithm is proposed for these problems. The algorithm works by using an initial solution generated by solving the leader's problem, where the leader is the decision maker in the upper level. That solution will go through each level by adjusting and evolving its components. Even though the approach is tested to be promising, the final solution can be very different from the initial one given by the leader. Furthermore, no cooperation mechanism between the decision makers is given. In addition, the decision makers may have multiple and conflicting objectives. In this paper a cooperation mechanism where fuzzy membership function is used to link the cooperation between the decision makers is used. That means once the solution received by the lower level decision makers, a cooperative feasible region will be determined which is a subset of the relaxed feasible region. To deal with the multi-objective optimization problem, preference free method called ideal point method will be used. Bi-level numerical examples are also given to demonstrate how the algorithm works.

1. Introduction

Several organizational structures and applications involves a hierarchical decision making scenario where multiple decision makers try to optimize their own objectives by controlling part of the decision variables. These problems can be formulated as nested optimization problems where part of the constraint set is determined by the solution of other optimization problems. Hence, the decision makers can be arranged in a hierarchical structure with their own, usually conflicting, objectives. That is, a decision maker at one level of the hierarchy may have its own objective function determined partly by variables controlled by decision makers in the other levels. Such type of decision making problems are often modelled as Stackelberg games, with some finite number of hierarchical levels. These types of problems are usually referred to as multilevel programming problems [4] and have got the attention of researchers in the last four decades.

Due to the hierarchical nature of the problems, the decision taken at higher levels in the system will affect the choice of the followers in their objectives and constraints by the values that come from the top level. The constraints at upper levels will also have impact on the choice of values for lower level players. Moreover, the payoffs of the upper level players at their respective objective values and their constraints are also

affected by the choice made at lower levels. Hence, the leader first fixes his/her set of variables, and the next level fixes his/her ones and proceeding this way, eventually the lowest level (the last follower) optimizes his/her objective with the variables from the higher levels are considered to be constants. Since the response of the lower level optimizer may not fit to the best objective values of the higher levels, they optimize their own objectives knowing that lower level will optimize later. Therefore, the higher levels might incorporate these anticipated rational reactions or responses from the lower levels while fixing their variables. Here, the lower levels are not directly dictated by the higher levels but are influenced by the set of variables fixed at higher level decision makers.

Several researches on Multilevel programming problems (MLPP) have been conducted over the decades, the vast majority of which concentrate on Bilevel (two level) programming problems. Many algorithms are also proposed to solve variety of MLPP. Vertex enumeration Methods including the Kth-best algorithm and the grid search algorithm is one of the methods proposed [6,7,10,35]. By transforming the lower level problems using methods like KKT condition and penalty function, solution approaches are proposed [2,5,17,31,32]. Using these approaches some descent algorithms have also been proposed [14,18,29]. Multi-parametric Programming Methods which make use of

E-mail address: TilahunS@unizulu.ac.za.

<https://doi.org/10.1016/j.orp.2018.100093>

Received 10 August 2018; Received in revised form 27 October 2018; Accepted 14 December 2018

Available online 15 December 2018

2214-7160/© 2018 The Author. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

the parametric nature of the upper level variables when one solves the lower level problem is also another approach proposed and used [12,13,23]. Fuzzy Methods, which apply the fuzzy goal programming approach to solve linear MLPP is also proposed and used in [24,25].

MLPP is a very complex and difficult problem. It has been shown that even for the case of what might be considered as the simplest version of it, the linear bilevel programming problem, is strongly NP-Hard [16,29]. Moreover, the complexity of the problem increases as significantly as the number of levels gets larger [8]. Because of this, getting exact solution is very hard even for two level problems cases. There is a possibility that the processing times of many of these solution approaches may exponentially increase at worst as the size of the problem increases ([3], Chapter 5). Hence methods that can give approximate solutions with practically reasonable time are becoming important. The development of metaheuristic algorithms aid these development of new solution approaches [15,20,26,30]. Metaheuristic algorithms have the advantage of not to be affected much by the behaviour of the problem. However, they don't guarantee producing the exact optimal solution but a sound approximation. The performance of these algorithms depends on the parametric setting during implementation [28]. More discussion can be obtained from the following survey papers [19,35].

Most of the proposed algorithms and methods are sensitive to the behaviour of the problem and hence, researchers are still looking for out of the box solution approach of these problems. Furthermore, generalizing and fine tuning existing algorithms to be suitable for real problem solving as well as making them code-able is an important research dimension in the field. In [26], an approach based on natural adaptation and using evolutionary strategy is proposed. In the approach, the leader problem will be solved under the relaxed constraint set where the solution then passed through each of the levels by adapting relevant components of the solution in each level. Even though the approach seems promising, the solution varies highly as no additional controlling or cooperation mechanism is used. Hence, the final solution at the end of the algorithm implementation may be very different from the initial solution proposed by the leader [33]. In this paper, a cooperative decision making approach will be used to overcome this limitation. In the approach, the decision makers will cooperate by considering the objective goal of the upper level leaders and using fuzzy membership function to modify the relaxed constraint set. In addition to that, since the decision makers may have multiple and conflicting objectives a proper solution approach needs to be used. Giving preference is not always easy for the decision maker and a preference free approach based on the ideal point called ideal point method will be used [11].

Hence, the main objectives of this paper are

- to improve the previously proposed algorithm so that a proper guideline is put by each of the decision makers in the levels to force decision makers in the other levels to consider their solution and reduce the variation as much as possible by reducing the feasible region.
- optimization with multiple objective is more common almost in all real applications, hence, cases when the decision makers in each levels with having multiple objectives with a case of unavailable decision makers's preference is discussed

The paper is organized as follows. In the next section basic concepts on fuzzy theory, general formulation of multiobjective and multilevel programming problem will be presented followed by a discussion on the proposed cooperative hierarchical decision making approach in Section 3. A numerical example will be used to demonstrate the approach in Section 4 followed by a conclusion in Section 5.

2. Preliminaries

2.1. Fuzzy numbers

The concept of fuzzy numbers and fuzzy system was introduced in mid-1960s [34]. It extends the number system by introducing what is called a membership function. In the case of crisp number, a number is equivalent to a single value, whereas a fuzzy number is an interval with a membership function to measure the degree of membership of the numbers in the interval. Many real scenarios can better be expressed using fuzzy numbers. For example consider the word tall, represented by a variable say x . Suppose a person with 2 m and more can be classified as tall. What about 1.99 m? 1.98 m? Hence, if fuzzy expression is used, those with 2m and above are tall with membership function value of 1 and the membership function decreases as we go down from 2 m until a defined lower boundary, say 1.6 m. Hence, if we use a linear decreasing function for the membership function, it can be expressed as follows:

$$\mu(x) = \begin{cases} 1, & x \geq 2; \\ 2.5x - 4, & 2 > x > 1.6; \\ 0, & x \leq 1.6. \end{cases} \tag{1}$$

The graph representation of the above membership function is given in Fig. 1(a)

Different shapes of membership functions are used and discuss in literature. Triangular and trapezoidal shaped membership functions are the common membership functions used.

The α - cut of a fuzzy number with $0 < \alpha < 1$ is the member of the interval with membership value of at least α , i.e. $\{x: \mu(x) \geq \alpha\}$, as demonstrated in Fig. 1(b). If the inequality is strict then it is called strong alpha cut.

Fuzzy expressions are more realistic for different situations with unclear boundaries. Some words can perturb the shape of the membership function. These words are called hedges [21,27]. For example the words 'very' and 'somewhat' are hedges. The membership function related to the hedge 'very' can be expressed by squaring the original membership function and the hedge 'somewhat' will have the opposite effect (by taking the radical of the original membership function).

2.2. Multiobjective optimization

2.2.1. Basics

A multiobjective optimization problem is an optimization problem with multiple and usually conflicting objectives [11,22]. Consider an optimization problem with k objectives given as in Eq. (2).

$$\min_{x \in S} F(x) \tag{2}$$

where $F: \mathfrak{R}^n \rightarrow \mathfrak{R}^k$ for $F(x) = (f_1(x), f_2(x), \dots, f_k(x))$ and S is the feasible region.

Since the objective function space is partially ordered, it is not always possible to compare and order the outcome of two solutions.

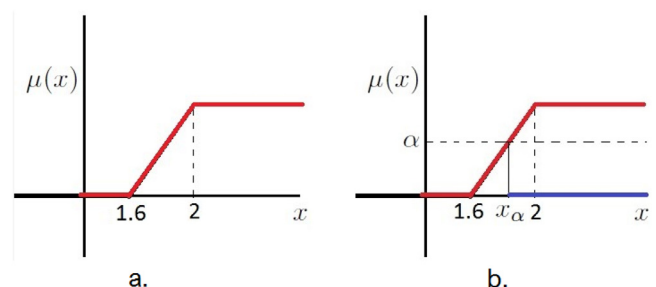


Fig. 1. a. Membership function for the fuzzy expression 'tall' b. α - cut of a fuzzy number 'tall'.

Hence, the optimality is defined based on dominance relation. A solution $x_1 \in S$ is said to dominate another solution $x_2 \in S$, if and only if $f_i(x_1) \leq f_i(x_2)$ for all i and strict inequality for at least one i . A feasible solution is called Pareto optimal solution if there is no other solution which dominates it. Due to the conflict of objectives, there usually will be multiple Pareto solutions. Choosing one solution from the Pareto optimal solutions depends on the preference of the decision maker.

Different solution methods are proposed to deal with multiobjective optimization problems. Basically, these solution methods can be classified into two broad categories, namely classical and heuristic-based approaches. Classical solution approaches are methods which uses a mathematical arguments to arrive to a single Pareto optimal solution. Most of these methods deals with the conversion of the problem to a single objective optimization problem or a series of single objective optimization problems and use existing solution methods designed for single objective optimization problems [11]. Classical solution methods may involve decision maker's preference, like weighting method, ϵ -constraint method and lexicographic method; or may not involve decision maker's preference, like in Benson's method and ideal point method.

Heuristic-based methods are methods based on metaheuristic algorithms like evolutionary algorithms and swarm intelligence. The methods have the advantage of producing multiple near Pareto optimal solution with appropriate implementation over classical methods. These solutions can be preference guided based on the decision maker's preference or produced with the aim of archiving a good representative number of solutions with a good distribution in the near Pareto front. Even though these approaches are very promising by giving the decision maker multiple decision choices, they are sensitive to parameter tuning.

2.2.2. Ideal point method

Definition 1. The ideal point of a multiobjective optimization problem is the collection of best objective function values over the constraint set. That is for a multiobjective optimization problem as given in Eq. (2), a point $y^I = (y_1^I, y_2^I, \dots, y_k^I)$ is called an ideal point if each of the components can be computed using Eq. (3).

$$y_i^I = \min_{x \in S} f_i(x) \tag{3}$$

Another point of interest in multiobjective optimization is the Nadir point.

Definition 2. A Nadir point for a multiobjective optimization problem given in Eq. (2), $y^N = (y_1^N, y_2^N, \dots, y_n^N)$, is computed using

$$y_i^N = \max_{x \in P} f_i(x) \tag{4}$$

where P is the set of all Pareto optimal solutions.

The ideal point method (ideal point approximation method) works towards finding the nearest Pareto optimal solution to the ideal point. This method is usable based on the distance function $d(x, y)$ between two points in the outcome space of the problem. Hence, the problem will be converted to a single objective optimization problem as given in Eq. (5).

$$\min_{x \in S} d(y^I, F(x)) \tag{5}$$

One of the advantages of this method is it doesn't need the decision maker's preference towards solving the problem rather it approximates and finds the compromised solution nearest to the ideal point. Any L_p norm can be used for the distance function. Detailed discussion and analysis on this method is given in [11]. L_2 norm is used in this paper.

In implementing the ideal point method, it should be noted that, the method is sensitive to the scaling of the objective functions. That is some of the objectives can be very much larger than the others, for

example one of the objective can be 10,000s and the other in between 0 and 1. To deal with it a normalization of the objectives can be used. One way to do so is to reformulate the multiobjective optimization problem as follows:

$$\min_{x \in S} \overline{F}(x) = \left(\frac{f_1(x)}{Z_1}, \frac{f_2(x)}{Z_2}, \dots, \frac{f_k(x)}{Z_k} \right) \tag{6}$$

where Z_i solves

$$\max_{x \in S} |f_i(x)| \tag{7}$$

2.3. Multilevel optimization

2.3.1. Basics

Multilevel optimization problems are a finite series of nested optimization problems where one is set in the constraint set of another. Different decision makers will deal with each of the optimization problems in each level. The decision space can be partitioned based on the decision makers or levels, say $X \subseteq \mathbb{R}^n$ (the Euclidean n -space) into r sub-regions, where r indicates the number of levels in the hierarchy of the MLPP, as $X^1 \times X^2 \times \dots \times X^r \subseteq \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \dots \times \mathbb{R}^{n_r}$, where $n_1 + n_2 + \dots + n_r = n$. i.e. for any $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, the vector x is partitioned as (x^1, x^2, \dots, x^r) , with $x^k = (x_1^k, x_2^k, \dots, x_{n_k}^k) \in \mathbb{R}^{n_k}$.

The decision maker in level one is called the leader and the others as followers. We denote a vector $x^i \in \mathbb{R}^{n_i}$ to represent a decision vector controlled by decision maker at level i . At the k th level, if $f_k(x)$ represent the objective function of the decision maker at the sub-level, for given values of x^1, x^2, \dots, x^{k-1} the decision maker chooses values for the vector x^k in an attempt to optimize his/her objective function $f_k(x)$. Note here that an optimal solution \bar{x}^k of the k th level problem is a function of the vectors $x^{k+1}, x^{k+2}, \dots, x^r$ (which are considered as parameters) since the remaining vectors, x^1, \dots, x^{k-1} are assumed to be already fixed at the upper levels. Then a general r -level MLPP is mathematically formulated as:

$$\begin{aligned} & \min_{x^1 \in X^1} f_1(x^1, x^2, \dots, x^r) \\ & \text{s.t. } (x^1, \dots, x^r) \in S^1, \quad \text{where } [x^2, x^3, \dots, x^r] \text{ solves} \\ & \min_{x^2 \in X^2} f_2(x^1, x^2, \dots, x^r) \\ & \text{s.t. } (x^1, \dots, x^r) \in S^2, \quad \text{where } [x^3, x^4, \dots, x^r] \text{ solves} \\ & \min_{x^3 \in X^3} f_3(x^1, x^2, \dots, x^r) \\ & \text{s.t. } (x^1, \dots, x^r) \in S^3, \quad \text{where } [x^4, \dots, x^r] \text{ solves} \\ & \vdots \\ & \min_{x^r \in X^r} f_r(x^1, x^2, \dots, x^r) \\ & \text{s.t. } (x^1, \dots, x^r) \in S^r, \end{aligned} \tag{8}$$

where $X^i \subseteq \mathbb{R}^{n_i}$, $\sum_{i=1}^r n_i = n$. The set $S = S^1 \cap S^2 \cap \dots \cap S^r$ is called the relaxed constraint set that every decision maker should select its decision variables from this set. We assume that S is non-empty and bounded.

To formulate a definition of optimality for MLPP, we need the following terms [33].

1. A constraint region of the MLPP is

$$\Omega = \{(x^1, \dots, x^r) \in X^1 \times \dots \times X^r : (x^1, \dots, x^r) \in S^1 \cap S^2 \cap \dots \cap S^r\}$$

2. For each given vector $(\bar{x}^1, \dots, \bar{x}^k) \in X^1 \times \dots \times X^k$, $1 \leq k < r$, the feasible region of the $(k + 1)$ and lower levels is given by

$$\Omega(\bar{x}^1, \dots, \bar{x}^k) = \{(x^{k+1}, \dots, x^r) \in X^{k+1} \times \dots \times X^r : (\bar{x}^1, \dots, \bar{x}^k, x^{k+1}, \dots, x^r) \in S\}$$

Then, clearly $\Omega(\bar{x}^1, \dots, \bar{x}^k) \subseteq S^{k+1} \cap \dots \cap S^r$.

3. Projection of Ω onto the k^{th} , $1 \leq k < r$, levels decision space is given by

$$\Omega(X^1, \dots, X^k) = \{(x^1, \dots, x^k): \exists (x^{k+1}, \dots, x^r) \in \Omega(x^1, \dots, x^k), \text{ with } (x^1, \dots, x^r) \in \Omega\}$$

4. For each $(\bar{x}^1, \dots, \bar{x}^k) \in \Omega(X^1, \dots, X^k)$, $1 \leq q < r$, $k = r - q$, the rational reaction set for the $(k + 1)$ and lower order levels is given by

$$M(\bar{x}^1, \dots, \bar{x}^k) = \{(\bar{x}^{k+1}, \dots, \bar{x}^r): (\bar{x}^{k+1}, \dots, \bar{x}^r) \in \text{argmin}\{f_{k+1}(\bar{x}^1, \dots, \bar{x}^k, x^{k+1}, \dots, x^r): (x^{k+1}, \dots, x^r) \in \Omega(\bar{x}^1, \dots, \bar{x}^k), (x^{k+2}, \dots, x^r) \in M(\bar{x}^1, \dots, \bar{x}^k, x^{k+1})\}\}$$

5. The Induced Region (\mathcal{IR}) at level one is

$$\mathcal{IR} = \{(x^1, \dots, x^r) \in S^1: x^1 \in X^1, (x^2, \dots, x^r) \in M(x^1)\}$$

Now using the definition of the *induced region* \mathcal{IR} , one can describe the MLPP at the first level as

$$\min_{(x^1, \dots, x^r) \in \mathcal{IR}} f_1(x^1, \dots, x^r) \tag{9}$$

We shall identify any optimal solution of problem (9) (if it exists) as an *optimal solution* of problem (8).

Definition 3. A given point $(\bar{x}^1, \dots, \bar{x}^r) \in \Omega$ is said to be an (ε, δ) -approximate solution to the MLPP in (8), where $\delta, \varepsilon \geq 0$, if

1. $f_1(\bar{x}^1, \dots, \bar{x}^r) \leq f_1(\bar{x}^1, \dots, \bar{x}^r) + \varepsilon(1 + |f_1(\bar{x}^1, \dots, \bar{x}^r)|)$, $\forall (\bar{x}^1, \dots, \bar{x}^r) \in \mathcal{IR}$
2. $f_j(\bar{x}^1, \dots, \bar{x}^r) \leq f_j(\bar{x}^1, \dots, \bar{x}^{j-1}, x^j, \dots, x^r) + \delta(1 + |f_j(\bar{x}^1, \dots, \bar{x}^r)|)$, $\forall j \in \{2, 3, \dots, r\}$, $(x^j, \dots, x^r) \in M(\bar{x}^1, \dots, \bar{x}^{j-1})$

A point $(\bar{x}^1, \dots, \bar{x}^r)$ satisfying the second condition is called *in a δ -reaction to \bar{x}^1* .

Definition 4. A point $(\bar{x}^1, \dots, \bar{x}^r) \in \Omega$ is said to be *first-rank better* (or *simply better*) than $(\bar{x}^1, \dots, \bar{x}^r)$ if either

1. $(\bar{x}^1, \dots, \bar{x}^r) \notin \Omega$, or
2. $\forall (\varepsilon_1, \delta_1)$ such that $(\bar{x}^1, \dots, \bar{x}^r)$ is an $(\varepsilon_1, \delta_1)$ -approximate solution of MLPP, $\exists (\varepsilon, \delta)$ such that $\varepsilon \leq \varepsilon_1$, $\delta < \delta_1$ or $\varepsilon < \varepsilon_1$, $\delta \leq \delta_1$ and $(\bar{x}^1, \dots, \bar{x}^r)$ is an (ε, δ) -approximate solution of MLPP.

Definition 5. A point $(\bar{x}^1, \dots, \bar{x}^r) \in \Omega$ is said to be *second-rank better* than $(\bar{x}^1, \dots, \bar{x}^r)$ if either

1. $(\bar{x}^1, \dots, \bar{x}^r) \notin \Omega$, or
2. $\forall (\varepsilon_1, \delta_1)$ such that $(\bar{x}^1, \dots, \bar{x}^r)$ is an $(\varepsilon_1, \delta_1)$ -approximate solution of MLPP, $\exists (\varepsilon, \delta)$ such that $\varepsilon + \delta < \varepsilon_1 + \delta_1$ and $(\bar{x}^1, \dots, \bar{x}^r)$ is an (ε, δ) -approximate solution of MLPP.

Hence we have the following theorem [33].

Theorem 1.

- (a) Any solution of MLPP is first-rank better than a non-optimal point and any point can never be second-rank better than a solution of MLPP.
- (b) (Transitivity) If a point $(\bar{x}^1, \dots, \bar{x}^r)$ is first-rank (or second-rank) better than $(\bar{x}^1, \dots, \bar{x}^r)$ and $(\bar{x}^1, \dots, \bar{x}^r)$ is first-rank (or second-rank) better than $(\bar{x}^1, \dots, \bar{x}^r)$, then $(\bar{x}^1, \dots, \bar{x}^r)$ is first-rank (or second-rank) better than $(\bar{x}^1, \dots, \bar{x}^r)$.
- (c) A first-rank better solution is second-rank better, but not the converse.

Based on the decision maker's motive towards other decision

makers, a multilevel optimization problem can be categorized into three categories.

- Category one: The first category is cooperative category. It is when each decision maker in each level tries to optimize its objective function and at the same time look at the suggestion of the other level objectives positively. Hence, if there are two optimum solutions it will take the one which is favorable for the other decision makers as well.
- Category two: The second category is non-cooperative category. It is when each decision maker in each level tries to optimize its objective function and at the same time look at the suggestion of the other level objectives negatively. Hence, if there are two optimum solutions it will take the one which is not favorable for the other decision makers.
- Category three: The third category is when the problem is neither cooperative nor non-cooperative. That includes partial cooperative or partial non cooperative problems. That is if a decision maker is cooperative for some of the decision makers and non-cooperative for the others.

2.3.2. Adaptation inspired algorithm for multilevel programming

Inspired by natural adaptation a solution procedure using a meta-heuristic algorithm is proposed [26]. The solution methods works based on an initial solution found by optimizing the leader's problem under the constraint set. That solution will go through each level by evolving and adapting its components relevant to each level's decision variable. It is inspired by natural adaptation of animals, that is when animals passes through or experience different environments it will update to the new environment to get along. In the natural adaptation inspired algorithm, a solution will be proposed by the leader, and each of the decision makers will update that solution while it goes from higher level to the last level to come back to the leader. In each level, the decision maker in the level will update the relevant variable (to that level) based on its own objective function only.

Hence, first the leader problem under the relaxed constraint set for all the variables will be solved, which is given in Eq. (10), using evolutionary strategy.

$$\min_{(x^1, x^2, \dots, x^r)} f_1(x) \tag{10}$$

s. t. $x \in S$

Suppose the solution, for Eq. (10), found is $(x^{1(0)}, x^{2(0)}, \dots, x^{r(0)})$. Then $(x^{1(1)}, x^{2(0)}, \dots, x^{r-1(0)}, x^{r(0)}) = (x^{1(0)}, x^{2(0)}, \dots, x^{r(0)})$. The next step will be this solution will go down in each level by updating components of this solution. That is for $i = 2, 3, \dots, r$, we will have to use evolutionary strategy to solve Eq. (11)

$$\min_{x^i} f_i(x^{1(1)}, x^{2(1)}, \dots, x^{i-1(1)}, x^i, x^{i+1(0)}, \dots, x^{r(0)}) \tag{11}$$

s. t. $x \in S$

to update the solution to $(x^{1(1)}, x^{2(1)}, \dots, x^{i-1(1)}, x^{i(1)}, x^{i+1(0)}, \dots, x^{r(0)})$. After going all the levels down a solution from the first iteration which is given by $(x^{1(1)}, x^{2(1)}, \dots, x^{r(1)})$ will be generated. This solution will be passed to the leader. The leader will then take this updated solution and solve Eq. (12) and generate $(x^{1(2)}, x^{2(1)}, \dots, x^{r(1)})$

$$\min_{x^1} f_1(x_1, x^{2(1)}, \dots, x^{r(1)}) \tag{12}$$

s. t. $x \in S$

This iteration will continue until a termination criterion is met. Suppose if maximum number of iteration is the termination criterion with the number of iterations being set to be T , then the final solution to be solved will be Eq. (13).

$$\min_{x^r} f_r(x^{1(T)}, x^{2(T)}, \dots, x^{r-1(T)}, x_r) \tag{13}$$

s. t. $x \in S$

The final solution which is $(x^{1(T)}, x^{2(T)}, \dots, x^{r(T)})$ will be taken as the outcome of the algorithm.

As mentioned in [26], the approach is not cooperative. Hence, the final solution can be very far from the initial solution proposed by the leader. In addition, it is also vulnerable for the cyclic oscillation between same solutions without converging. However, this can be improved by adding the cooperations parameter in between the decision maker. A fuzzy membership function will be used, in the next section, to extend and overcome the limitations of the approach.

3. Cooperative hierarchical decision making

We consider the cooperative multilevel optimization. Each decision maker will deal with multiobjective optimization, as given in Eq. (14).

$$\begin{aligned}
 & \min_{x^1 \in X^1} F_1(x^1, x^2, \dots, x^r) \\
 & \text{s.t. } (x^1, \dots, x^r) \in S^1, \quad \text{where } [x^2, x^3, \dots, x^r] \text{ solves} \\
 & \quad \min_{x^2 \in X^2} F_2(x^1, x^2, \dots, x^r) \\
 & \quad \text{s.t. } (x^1, \dots, x^r) \in S^2, \quad \text{where } [x^3, x^4, \dots, x^r] \text{ solves} \\
 & \quad \min_{x^3 \in X^3} F_3(x^1, x^2, \dots, x^r) \\
 & \quad \text{s.t. } (x^1, \dots, x^r) \in S^3, \quad \text{where } [x^4, \dots, x^r] \text{ solves} \\
 & \quad \vdots \\
 & \quad \min_{x^r \in X^r} F_r(x^1, x^2, \dots, x^r) \\
 & \quad \text{s.t. } (x^1, \dots, x^r) \in S^r, \tag{14}
 \end{aligned}$$

where $F_i: \mathfrak{R}^n \rightarrow \mathfrak{R}^{k_i}$ given by $F(x) = (f_{i,1}(x), f_{i,2}(x), \dots, f_{i,k_i}(x))$.

The adaptation inspired algorithm will be used however the in solving each of the algorithms in different levels the solution found by the other solution will be considered and a fuzzy membership function is suitable to put these in the objective functions. The main steps of the proposed approach is summarized as follows:

1. Step 1

Solve of the components of the leader's problem under the relaxed constraint set, S , as given in Eq. (15)

$$\begin{aligned}
 f_{1,i}^I &= \min_x f_{1,i}(x) \\
 \text{s.t. } & x \in S \tag{15}
 \end{aligned}$$

After computing the ideal point $y_1^I = (f_{1,1}^I, f_{1,2}^I, \dots, f_{1,k_1}^I)$, compute the solution $x^* = (x^{1*}, x^{2*}, \dots, x^{k*})$ by solving the problem in Eq. (16).

$$\begin{aligned}
 & \min_x d(y_1^I, F_1(x)) \\
 \text{s.t. } & x \in S \tag{16}
 \end{aligned}$$

Set $\bar{S} = \bar{S} \cap \{x: \mu_1(x) \geq \alpha_1\}$ where $\mu_1(x)$ is given by:

$$\mu_1(x) = \begin{cases} 1, & \frac{d(F_1(x), y_1^I) - d(y_1^N, y_1^I)}{d(F_1(x^*), y_1^I) - d(y_1^N, y_1^I)} \leq d(F_1(x^*), y_1^I); \\ 0, & \frac{d(F_1(x), y_1^I) - d(y_1^N, y_1^I)}{d(F_1(x^*), y_1^I) - d(y_1^N, y_1^I)} \geq d(y_1^N, y_1^I); \\ \frac{d(F_1(x), y_1^I) - d(y_1^N, y_1^I)}{d(F_1(x^*), y_1^I) - d(y_1^N, y_1^I)}, & \text{otherwise.} \end{cases} \tag{17}$$

where y_1^N is the Nadir point and y_1^I the ideal point.

Note that, rather than using the Nadir point a decision maker can give a tolerance compromise from his/her ideal point. In that case it will reduce the computational cost of computing the Nadir point.

2. Step 2

For each i from 2 to r by fixing the value of x^r by x^* except x_i ,

minimize

$$\begin{aligned}
 & \min_{x^i} f_{ij}(x^{*/i}) \\
 \text{s.t. } & x \in S \tag{18}
 \end{aligned}$$

for $j = 1, 2, \dots, k_i$ and where $x^{*/i} = (x^{1*}, x^{2*}, \dots, x^i, \dots, x^{r*})$.

Based on the result update y_i^I . Once the ideal point y_i^I and the corresponding Nadir point y_i^N (or a tolerance level) are computed, solve Eq. (19).

$$\begin{aligned}
 & \min_{x^i} d(y_i^I, F_i(x^{*/i})) \\
 \text{s.t. } & x \in S \tag{19}
 \end{aligned}$$

Update the component of x^* using the result. Set $S = S \cap \{x: \mu_i(x^i) \geq \alpha_i\}$ (use a similar formula for level i as given in Eq. (17) for level 1).

3. Step 3

With fixed value for x^i 's with x^* for all $i \in \{2, 3, \dots, n\}$, solve the leader problem for a new y_1^I .

$$\begin{aligned}
 & \min_{x^1} f_{1j}(x^{*/1}) \\
 \text{s.t. } & x \in S \tag{20}
 \end{aligned}$$

for $j = 1, 2, \dots, k_1$

After updating the ideal point for the leader solve:

$$\begin{aligned}
 & \min_{x^1} d(y_1^I, F_1(x^{*/1})) \\
 \text{s.t. } & x \in S \tag{21}
 \end{aligned}$$

Update x^* based on the result and update the relaxed constraint set.

4. Step 4

If termination criterion is met terminate else go back to step 2.

The termination criteria can be satisfaction of the decision makers with the result, maximum number of iteration, no improvement is archived (including cyclic rotation of solutions) and if \bar{S} become empty.

4. Numerical example

To demonstrate the proposed approach the following examples are used

1. The first example: The first example is taken from [1]. It is a bilevel multiobjective optimization problem of the form:

$$\begin{aligned}
 & \min_{x^1 \in \mathfrak{R}} F_1 \\
 & \text{where } x^2 \text{ solves} \\
 & \quad \min_{x^2 \in \mathfrak{R}} F_2 \\
 & \quad \text{s.t. } x \in S
 \end{aligned}$$

where $F_1 = (\frac{x_1 + x_2 - 1}{x_1 - 2x_2 + 1}, \frac{2 - 2x_1 - x_2}{x_2 + 4})$, $F_2 = (\frac{-x_1 + 4}{-x_2 + 3}, \frac{x_1 - 4}{x_2 + 1}, x_1 - x_2)$ and $S = \{(x_1, x_2) \in \mathfrak{R}^2: -x_1 + 4x_2 \leq 0, x_1 - \frac{1}{2}x_2 \leq 4, x_1 \geq 0, x_2 \geq 0\}$. The feasible region S is given in Fig. 2(a). According to the first step of the solution procedure the ideal point for the leader's problem is $(-1, -1.6111)$. Hence, the leader will search for a solution which is near to the found ideal point, by minimizing the following equation:

$$\begin{aligned}
 & \min_x \|y_1^I - F_1(x)\| = \left\| -1 - \frac{x_1 + x_2 - 1}{x_1 - 2x_2 + 1}, -1.6111 - \frac{2 - 2x_1 - x_2}{x_2 + 4} \right\| \\
 \text{s.t. } & x \in S \tag{22}
 \end{aligned}$$

The solution, x^* , is $(3.6393, 0)$ with objective function value of 1.5957. The leader then pass this solution to the follower. The follower fix the first component and determine ideal value for the

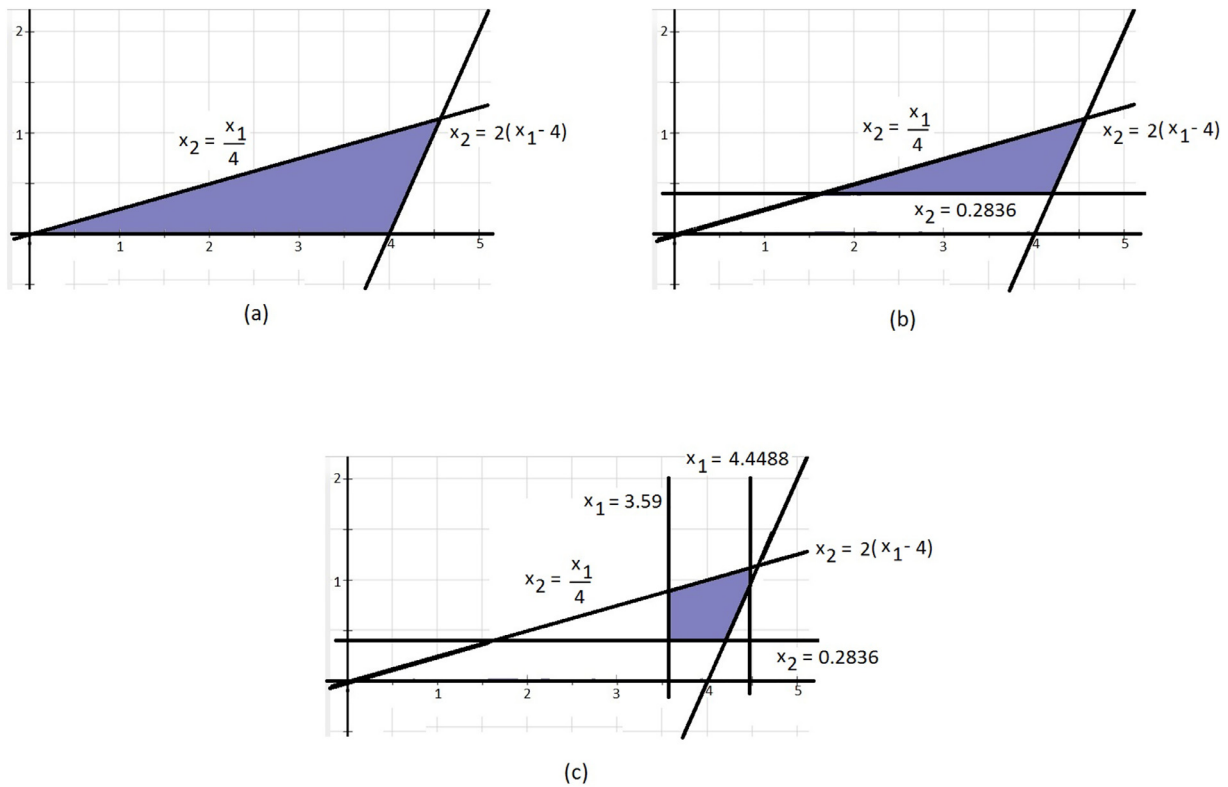


Fig. 2. Diminishing of the feasible region with the iterations of the algorithm.

value it controls. The Nadir point is taken to be (0, 0). The fuzzy membership with value at least 0.5, $\alpha_1 = 0.5$, results $0.2836 \leq x_2$. Hence $S = S \cap \{(x_1, x_2) \in \mathfrak{R}^2: 0.2836 \leq x_2\}$, as shown in Fig. 2(b). By fixing x_1^* , the ideal point, y_2^I , for the follower will be (0.1328, -0.2810, 2.7295). Then the follower optimizes the following problem:

$$\min_{x_2} \|y_2^I - F_2(x_1^*, x_2)\| = \left\| \frac{0.0377 - 0.1328x_2}{-x_2 + 3}, \frac{5.992x_2 - 1.6471}{x_2 + 1}, x_2 \right\|$$

$$s. t. (3.6393, x_2) \in S \tag{23}$$

The optimum value is found at 0.8975. Hence, x^* become (3.6389, 0.8975). With $\alpha_2 = 0.5$ and by contracting the membership function restriction the new feasible region will be $S = S \cap \{(x_1, x_2) \in \mathfrak{R}^2: 3.5900 \leq x_1 \leq 4.4488\}$, as given in Fig. 2(c). By fixing x_2^* , the ideal solution for the leader will be (1.1895, -1.5916) and the nearest point which will minimize the distance of the values in the solution space from the ideal solution is found at $x_1^* = 4.4488$ with objective function value of 2.4351, i.e $x^* = (4.4488, 0.8975)$. In computing the feasible region using the appropriate fuzzy membership function it becomes empty set, i.e. $S = \emptyset$. Hence, the final resulting solution is $x^* = (4.4488, 0.8975)$.

2. The second example:

The second example is taken from [9]. The is given by:

$$\min_{x_1 \in \mathfrak{R}^1} F_1 = x_1 + 2x_2 + x_3$$

$$\min_{(x_2, x_3) \in \mathfrak{R}^2} [F_2, F_3] = [x_2 - 2x_3, -x_2 + x_3]$$

$$s. t. x \in S$$

where

$$S = \{(x_1, x_2, x_3) \in \mathfrak{R}^3: x_1 + x_2 \geq 1, x_1 + x_2 \leq 3, x_1 + x_2 + 2x_3 \leq 5, x_1 \geq 0, x_2 \geq 0, x_3 \geq 0\}$$

The ideal point which is also the optimal objective function for the leader under the feasible region S is 1 at $x = (1, 0, 0)$. The maximum possible objective function value in the feasible region is 7, hence let $y^N = 7$. (Fig. 3)

By applying the feasibility reduction with $\mu \geq 0.5$ gives the following (at $x_1 = 1$):

$$2x_2 + x_3 \leq 3 \tag{24}$$

Solving the follower's bi-objective problem with the feasible region being $S = S \cap \{(x_1, x_2, x_3) \in \mathfrak{R}^3: 2x_2 + x_3 \leq 3\}$ gives the ideal point to be $y^I = (-3, -2)$ with $y^N = (2, 1.5)$. By applying the ideal point method and solving Eq. (25), the solution becomes $x = (1, 0.52, 1.24)$ with objective function values $(-1.96, 0.72)$.

$$\min F_2 = [x_2 - 2x_3, -x_2 + x_3]$$

$$s. t. x_2 + 2x_3 \leq 30 \leq x_2 \leq 2x_3 \geq 0 \tag{25}$$

By applying the feasibility reduction with $\mu \geq 0.8$, the feasible region becomes $S = S \cap \{(x_1, x_2, x_3) \in \mathfrak{R}^3: (x_2 - 2x_3 + 3)^2 + (x_3 - x_2 + 2)^2 \leq 12.7108\}$

Solving the leader's problem again gives $y_1^I = y^* = 2.76$ at $x = (0.48, 0.52, 1.24)$ and set $y_1^N = 4.5$.

The reduce membership function is given by $S = S \cap \{(x_1, x_2, x_3): x_1 \leq 0.8280\}$, which gives an empty set. Hence the termination of the algorithm.

The final results of the examples and the results reported in the literature is presented in Table 1

As can be seen from the table the solution found is not dominated by any of the reported solutions in the literature. The result dominates the leader's objective function in the second example but in the follower's it either dominated nor dominate the solutions.

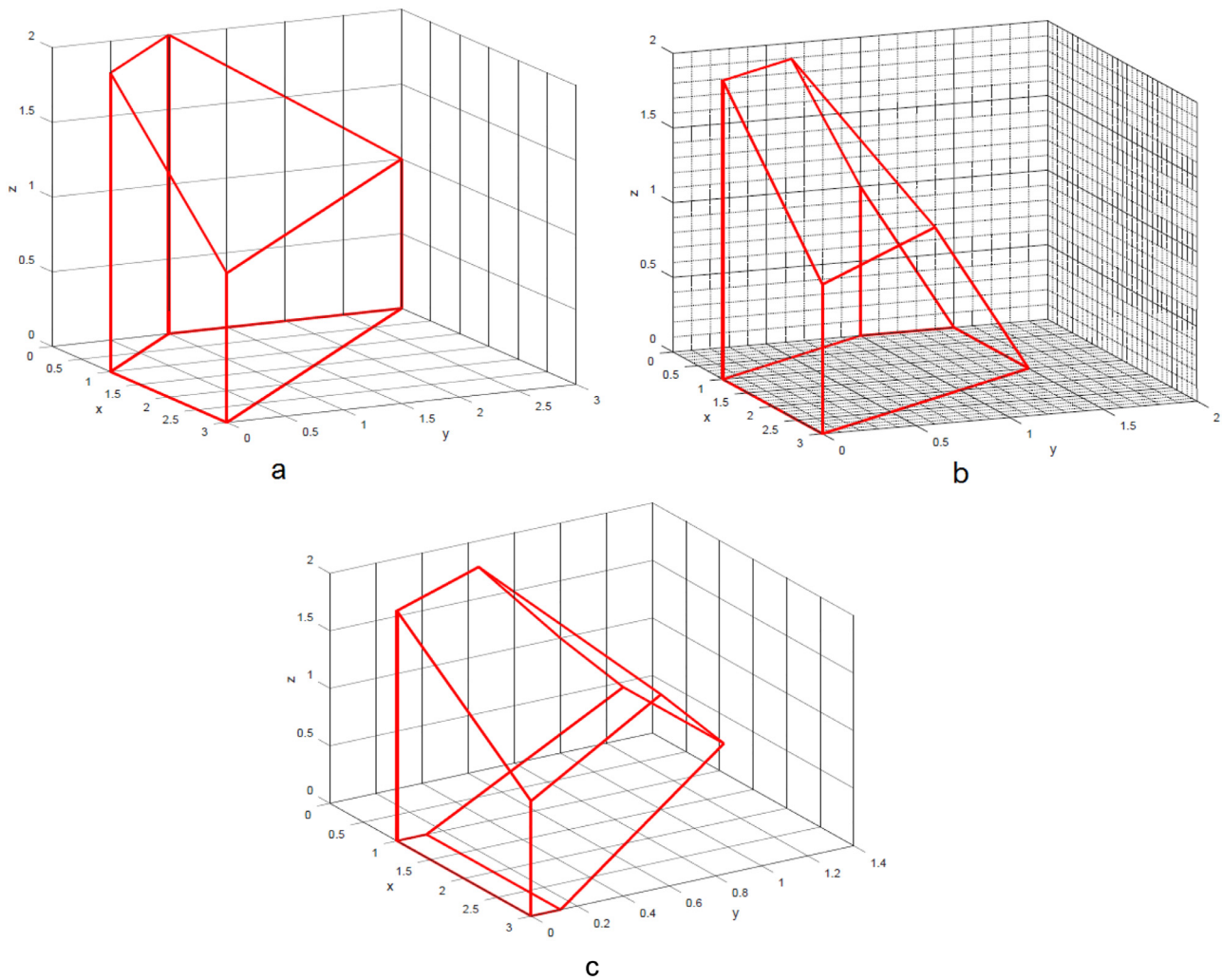


Fig. 3. Feasibility reduction for the second example.

5. Discussion

Hierarchical decision making, possible with multiple objective function for the decision makers, is common in different application especially in organizational structures where there are multiple decision makers with their own objective sharing the same resource. The decision maker may be competing cooperatively, non-cooperatively or neither. In this paper, a cooperative scenario discussed, where adaptation inspired algorithm is modified to suit the multiobjective case, by using ideal point methods, and by incorporating cooperativeness, using fuzzy membership function. According to the proposed approach, the leader will propose an initial solution based on the ideal point of his/her problem. The solution will evolve by going through each lower

levels. The decision maker in a lower level takes the proposed solution and also considers the gain of the upper level decision makers. Hence, the feasible region will keep on diminishing based on these additional criteria.

A numerical example, taken from a literature, is given to demonstrate the idea. Several future works can be proposed. One possible future work is to consider cases where the cooperation is partial. That is when some of the decision makers are cooperative and some others are not. In addition, if a decision maker is cooperative to some of the decision makers and not cooperative some other can also be studied further. A comparative study with other algorithms using a number of benchmark problems needs also to be explored.

One of the possible future work is to test the proposed approach on

Table 1 Results.

Example 1						
Our results		Results from literature				
x^* (4.4488, 0.8975)	F_1^* (1.19, -1.5917)	F_2^* (-0.2135, 4.3785, 3.5513)	x^* (0.8, 0.2)	F_1^* (0, 0.048)	F_2^* (1.143, -2.667, 0.6)	Citation [1]
Our results			Example 2			
x^* (0.48, 0.52, 1.24)	F_1^* 2.76	F_2^* (-1.96, 0.72)	x^* (1, 0, 2) (3, 0, 0)	F_1^* 3 3	F_2^* (-4, 2) (0, 0)	Citation [9] [9]

more complex and real problems. Perhaps, a detailed comparative analysis can also be conducted to see in which type of problems the algorithm perform better. Theoretical analysis on the convergent condition of the algorithm is another interesting issue to study further. Hybridizing multiple methods to produce an efficient method is common in different problem solving approaches, hence, to boost the performance of the algorithm, an hybridization with other approaches can be studied further.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:[10.1016/j.orp.2018.100093](https://doi.org/10.1016/j.orp.2018.100093)

References

- [1] Abo-Sinna MA, Baky IA. Fuzzy goal programming procedure to bilevel multi-objective linear fractional programming problems. *Int J Math Math Sci* 2010;2010:15. Article ID 148975
- [2] Aiyoshi E, Shimizu K. Hierarchical decentralized systems and its new solution by a barrier method. *IEEE Trans Syst, Man, Cybern* 1981;11:444–9.
- [3] Alkhateeb F, Maghayreh EA, Doush IA. Multiagent systems modelling, control, programming, simulations and applications. InTech, Croatia 2011.
- [4] Bracken J, McGill J. Mathematical programs with optimization problems in the constraints. *Oper Res* 1973;21:37–44.
- [5] Bard JF, Falk JE. An explicit solution to the multilevel programming problem. *Comput Oper Res* 1982;9(1):77–100.
- [6] Bard JF. An efficient point algorithm for a linear two-stage optimization problem. *Oper Res* 1983;31:670–84.
- [7] Bard JF. An investigation of the linear three level programming problem. *IEEE Trans Syst, Man, Cybern*, Vol 1984;14:711–7.
- [8] Blair C. The computational complexity of multi-level linear programs. *Ann Oper Res* 1992;34:13–9.
- [9] Calvete HI, Gal C. On linear bilevel problems with multiple objectives at the lower level. *Omega* 2011;39(1):33–40.
- [10] Dempe S. A simple algorithm for the linear bilevel programming problem. *Optimization* 1987;18:373–85.
- [11] Faísca NP, Dua V, Rustem B, Saraiva PM, Pisticopoulos EN. Parametric global optimisation for bilevel programming. *J Global Optim* 2007;38:609–23.
- [12] Faísca NP, Saraiva PM, Rustem B, Pisticopoulos EN. A multi-parametric programming approach for multilevel hierarchical and decentralised optimisation problems. *Comput Manag Sci* 2009;6:377–97.
- [13] Florian M, Chen Y. A bilevel programming approach to estimating o-d matrix by traffic counts. Technical Report CRT-750. Centre de Recherche sur les Transports; 1991.
- [14] Han J, Zhang G, Hu Y, Lu J. A solution to bi/tri-level programming problems using particle swarm optimization. *Inf Sci* 2016;370–371:519–37.
- [15] Hansen P, Jaumard B, Savard G. New branch-and-bound rules for linear bilevel programming. *SIAM J Scient Stat Comput* 1992;13:1194–217.
- [16] Júdice J, Faustino A. The linear-quadratic bilevel programming problem. *INFOR* 1994;32:87–98.
- [17] Kolstad C, Lasdon L. Derivative evaluation and computational experience with large bilevel mathematical programs. *J Optim Theory Appl* 1990;65:485–99.
- [18] Lu J, Han J, Hu Y, Zhang G. Multilevel decision-making: a survey. *Inf Sci* 2016;346–347:463–87.
- [19] Li H, Wang Y. An evolutionary algorithm based on a new decomposition scheme for nonlinear bilevel programming problems. *Int J Commun, Network Syst Sci* 2009;2:87–93.
- [20] Michael, N. (2005). *Artificial intelligence a guide to intelligent systems*.
- [21] Ong CH, Tilahun SL. Integration fuzzy preference in genetic algorithm to solve multiobjective optimization problems. *Far East Math Sci* 2011;55:165–79.
- [22] Pisticopoulos EN, Georgiadis MC, Dua V. *Multiparametric programming*. Editors, volume 1: theory, algorithms, and applications. Wiley-VCH Verlag, Weinheim; 2007.
- [23] Pramanik S, Roy TK. Fuzzy goal programming approach to multilevel programming problems. *Eur J Oper Res* 2007;176:1151–66.
- [24] Sinha S. Fuzzy goal programming approach to multi-level programming problems. *Fuzzy Sets Syst* 2003;14:71–87.
- [25] Tilahun SL, Kassa SM, Ong HC. A new algorithm for multilevel optimization problems using evolutionary strategy, inspired by natural adaptation. *Pacific Rim Int Conf Artif Intell* 2012:577–88.
- [26] Tilahun SL, Ong HC. Fuzzy preference of multiple decision-makers in solving multi-objective optimisation problems using genetic algorithm. *Maejo Int J Sci Technol* 2012;6(2).
- [27] Tilahun SL. Prey predator hyperheuristic. *Appl Soft Comput* 2017;59:104–14.
- [28] Vicente L, Savard G, Júdice J. Descent approaches for quadratic bilevel programming. *J Optim Theory Appl* 1994;81:379–99.
- [29] Wang Y, Jiao YC, Li H. An evolutionary algorithm for solving nonlinear bilevel programming based on a new constraint-handling scheme. *IEEE Trans Syst, Man, Cybern* 2005;35:221–32.
- [30] White DJ, Anandalingam G. A penalty function approach for solving bi-level linear programs. *J Global Optim* 1993;3:397–419.
- [31] White DJ. Penalty function approach to linear trilevel programming. *J Optim Theory Appl* 1997;93(1):183–97.
- [32] Woldemariam AT, Kassa SM. Systematic evolutionary algorithm for general multilevel stackelberg problems with bounded decision variables (SEAMSP). *Ann Oper Res* 2015;229(1):771–90.
- [33] Zadeh LA. Fuzzy sets. *Inf Control* 1965;8(3):338–53.
- [34] Zhang G, Lu J, Montero J, Zeng Y. Model, solution concept, and kth-best algorithm for linear trilevel programming. *Inf Sci* 2010;180:481–92.
- [35] Zhang G, Han J, Lu J. Fuzzy bi-level decision-making techniques: a survey. *Int J Comput Intell Syst* 2016;9:25–34.