Shahid Bahonar University of Kerman

Journal of Mahani Mathematical Research

Print ISSN: 2251-7952 Online ISSN: 2645-4505

AN ALGORITHM FOR A MULTICRITERIA OPTIMIZATION PROBLEM AND ITS APPLICATION TO A FACILITY LOCATION PROBLEM

N. Ghafari and M.A. Yaghoobi ✉

Dedicated to sincere professor Mashaallah Mashinchi

Article type: Research Article

(Received: 20 June 2022, Received in revised form: 18 September 2022)

(Accepted: 04 November 2022, Published Online: 04 November 2022)

Abstract. In this paper, a new algorithm is proposed for solving a multicriteria optimization problem where the feasible set is an $m-$dimensional cube. In fact, the idea of the multicriteria big cube small cube method is employed to develop the new algorithm. It is proved that, for a given epsilon vector, the output of the suggested algorithm involves all epsilon efficient solutions as well as all efficient solutions. Furthermore, the algorithm is applied to a multicriteria location problem. The results show that the recommended algorithm can obtain more epsilon efficient solutions in comparison with the main multicriteria big cube small cube method.

Keywords: Multicriteria optimization, Efficient solutions, Epsilon efficient solutions, Multicriteria facility location.
2020 MSC: Primary 90C29, 90B85.

1. Introduction

The current research focuses on multicriteria optimization problems where the feasible set is an $m-$dimensional cube. A multicriteria optimization problem optimizes several, often conflicting and incommensurate, objective functions over a feasible set [7, 28]. Many studies and applications of multicriteria problems have been reported in literature [7, 27]. It is due to the fact that most of real world decision making problems involve more than one objective function (see, e.g., [22, 29]). There are many methods and algorithms to solve a multicriteria optimization problem [7, 27]. For instance in [9], Eichfelder and Warnow introduce an approximation with a simple structure respecting the natural ordering. In particular, they compute a box-coverage of the nondominated set. Usually, the algorithms are applicable to only special types of objective functions such as linear, quadratic, convex, smooth, etc. On the other hand, a special structure of the feasible set is another important feature of the existing algorithms. In literature, the feasible set is assumed to

✉ yaghoobi@uk.ac.ir, ORCID: 0000-0003-1320-2552
DOI: 10.22103/jmmr.2022.19734.1282
Publisher: Shahid Bahonar University of Kerman
How to cite: N. Ghafari, M.A. Yaghoobi, *An algorithm for a multicriteria optimization problem and its application to a facility location problem*, J. Mahani Math. Res. 2022; 11(3): 197-213.

be polyhedral, convex, bounded, cube, etc. This paper deals with multicriteria optimization problems with arbitrary objective functions which should be optimized over an $m-$dimensional cube with sides parallel to the axes. These kinds of problems are considered by Schobel and Scholz [24] and Scholz [25]. They proposed an algorithm called the multicriteria big cube small cube (MBCSC). In fact, they have extended the idea of the big square small square (BSSS) technique, suggested by Hansen et al. [15], to the multicritera case. Indeed, the BSSS technique considers only a single objective function. The BSSS and MBCSC algorithms are applied to facility location problems as real world applications [15, 24].

The aim of facility location theory is to find an optimal location for a new facility which might be a desirable or an undesirable one [4]. One of the most obvious areas for application of facility location problems is city and regional planning. For instance, in residental area, a new facility may be either a fire station, a shopping center, a bus station, a hospital, or other public facilities. First, Hansen et al. [15] applied BSSS to some location problems on the plane with two variables. Plastria extended this method to the generalized big square small square method [21]. Drezner and Suzuki [5] suggested the big triangle small triangle method. Indeed, they used triangles instead of squares to propose a global optimization method for solving location problems. It should be noted that all these techniques are branch and bound approaches for single objective facility location problems on the plane with two variables and require lower bounds for each square or triangle [24]. On the other hand, in reality for finding a good location more than one objective function should be considered [14]. For example, consider the problem of locating an airport. Some customers would like the airport to be close to the residental areas so that they do not need to travel a long distance to receive service. On the other hand, they may want the airport to be far away because it generates noise and pollution. Eiselt and Laporte [10] have reviewed different objective functions and their classifications in facility location models.

McGinnis and White [18] have done one of the first investigations in multicriteria location theory. They considered a single facility rectilinear location problem with multiple criteria. Nickel et al. [19] presented the ordered median problems for modeling multicriteria location problems on the plane. Badri et al. [1] proposed a multiple objective model for a set covering problem of locating fire stations. Rakas et al. [23] developed a multiobjective model for determining locations of undesirable facilities. Concretely, the number of multicriteria location problems with different objective functions reported in literature is too large. A complete description of a multicritteria location problem is presented by Hekmatfar and SteadieSeifi [16]. Moreover, a good survey on these problems is given by Current et al. [3]. They classified litrature, based on their types of objectives, into four categories: dealing cost, demand coverage, profit maximization, and environmental issues. Farahani et al. [13] provided another good survey in multicriteria location problems. They discussed multicriteria location literature based on their classic operational research families: Weber, median, covering, constrained, uncapacitated, location-allocation, location-routing, dynamic, competitive, network, and undesirable location problems.

The current research aims to develop an algorithm based on MBCSC method for a multicriteria optimization problem. Moreover, its application to a multicriteria facility location problem is discussed. Rest of the paper is organized as follows. Section 2 reviews some definitions and notations. Section 3 proposes an Algorithm to solve a multicriteria optimization problem. Properties of output set of the proposed algorithm are discussed in Section 4. Section 5 provides an application of the proposed algorithm to a bicriteria location problem. Finally, Section 6 is devoted to concluding remarks.

2. Definitions and notations

In this section a brief description of a multicriteira optimization problem and some basic definitions are given. For more details the reader is referred to [7]. A multicritria optimization problem can be formulated as [7]:

$$\min_{x\in\mathcal{X}} f(x) = (f_1(x), ..., f_p(x)), \tag{1}$$

where $\mathcal{X} \subset \mathbb{R}^m$ is the feasible set, and $f_i : \mathbb{R}^m \to \mathbb{R}$ for $i = 1, ..., p$ are objective functions.

For a given $x \in \mathcal{X}$, the vector $y = f(x)$ is called the outcome vector or the criterion vector. Let $\mathcal{Y} = f(\mathcal{X}) \subset \mathbb{R}^p$ denote the set of all outcomes. To compare two feasible solutions, an order for comparing their criterion vectors is needed. The common orders are as follows [7]:
Let $y^1, y^2 \in \mathbb{R}^p$. Then:

- $y^1 \leqq y^2 \Leftrightarrow y^1_i \leqslant y^2_i\ \forall\ i = 1, \dots, p$, (weak componentwise order);
- $y^1 \leq y^2 \Leftrightarrow y^1_i \leqslant y^2_i\ \forall\ i = 1, \dots, p$ and $y^1 \neq y^2$, (componentwise order);
- $y^1 < y^2 \Leftrightarrow y^1_i < y^2_i\ \forall\ i = 1, \dots, p$, (strict componentwise order).

$\mathbb{R}^p_{\geqq} := \{x \in \mathbb{R}^p : x \geqq 0\}$ denotes the set of all nonnegative vectors in $\mathbb{R}^p$. Note that $\mathbb{R}^p_{\geq}$ and $\mathbb{R}^p_{>}$ can be defined analogously.

Definition 2.1. ([7]) A solution $\hat{x} \in \mathcal{X}$ is called an efficient solution of Problem (1) if there does not exist $x \in \mathcal{X}$ such that $f(x) \leq f(\hat{x})$. $\mathcal{X}_E$ denotes the set of all efficient solutions of Problem (1).

To test the efficiency of a given solution $\hat{x} \in \mathcal{X}$, an efficiency test can be used. Benson's efficiency test [27] is one the popular tests which solves a single objective optimization problem as follows:

$$\begin{aligned} \max\ z \ &= \ l_1 + l_2 + \cdots + l_p \\ s.t.\ & \\ & f_i(x) + l_i = f_i(\hat{x}),\ i = 1, ..., p, \\ & x \in \mathcal{X};\ l_i \geqslant 0,\ i = 1, ..., p. \end{aligned} \tag{2}$$

The following theorem introduces a criterion for efficiency of a feasible solution $\hat{x} \in \mathcal{X}$.

Theorem 2.2. *([7]) A solution $\hat{x} \in \mathcal{X}$ is an efficient solution of Problem (1) if and only if the optimal objective value of Problem (2) is zero.*

Often, attaining the exact efficient solutions is not an easy task. Thus, approximating them is of interest. Next definition introduces the concept of epsilon efficiency.

Definition 2.3. ([11, 17]) For a given $\varepsilon \in \mathbb{R}^p_{\geqq}$, a solution $\hat{x} \in \mathcal{X}$ is called an $\varepsilon-$efficient solution of Problem (1) if there does not exist $x \in \mathcal{X}$ such that $f(x) + \varepsilon \leq f(\hat{x})$. $\mathcal{X}^{\varepsilon}_E$ denotes the set of all $\varepsilon-$efficient solutions of Problem (1).

It is obvious that $\mathcal{X}_E \subseteq \mathcal{X}^{\varepsilon}_E$ for all $\varepsilon \in \mathbb{R}^p_{\geqq}$ and for $0 = \varepsilon \in \mathbb{R}^p_{\geqq}$, $\mathcal{X}_E = \mathcal{X}^0_E$.

Definition 2.4. ([25]) Let $X \subseteq \mathbb{R}^n$ be a nonempty and compact set. Then the length of the Euclidean diameter of X, denoted by $\delta(X)$, defines as:

$$\delta(X) = max\{\|x' - x''\|_2 : x', x'' \in X\},$$

where $\|.\|_2$ is the Euclidean norm.

3. A new algorithm

This section focuses on a multicriteria optimization problem such as Problem (1) in which the feasible set is assumed to be an $m-$dimensional cube with sides parallel to the axes (or, for simplicity, a box). In other words, the multicriteria optimization problem is as follows:

$$\begin{aligned} &\min \quad f(x) = (f_1(x), ..., f_p(x)) \\ &s.t. \\ &\qquad x \in \mathcal{X} = [x^1_{min}, x^1_{max}] \times \cdots \times [x^m_{min}, x^m_{max}]. \end{aligned} \tag{3}$$

In the following, an algorithm for solving Problem (3) is suggested. In multicriteria optimization the common favorable solutions are efficient ones. However, finding the exact efficient set is not an easy task in most of real world problems. Therefore, $\varepsilon-$efficient solutions are of interest. Some methods for generating $\varepsilon-$efficient solutions in multiobjective programming are proposed by Engau and Wiecek [12]. They introduced some scalarizing methods and then used epsilon optimal solutions of the scalarized single objective optimization problems to generate $\varepsilon-$efficient solutions.

The current research does not use the scalarization technique. In fact, the proposed algorithm gets an arbitrary $\varepsilon \in \mathbb{R}^p_{\geqq}$. Then, it divides the feasible set to smaller boxes and tries to discard the boxes which do not involve any $\varepsilon-$efficient solutions. In each iteration of the algorithm some single objective optimization problems are solved for attaining the lower and upper bounds of the objective functions over the new boxes. Further, some comparisons help us to discard some of the existing boxes. In fact, the new algorithm uses a similar idea as the MBCSC algorithm [25]. However, the discarding process in the new algorithm is completely changed. Moreover, since the set of $\varepsilon-$efficient solutions is not closed [12], the MBCSC algorithm tries to find a subset of it. But, the output of the new algorithm contains all $\varepsilon-$efficient solutions.

Let us first introduce some sets which will be used in the new algorithm. Consider Problem (3). Suppose that $B \subseteq \mathcal{X}$ is a box and $\varepsilon \in \mathbb{R}^p_{\geqq}$. Then:

- $L = (L_1, ..., L_p)^t$, where $L_i = \min_{x\in\mathcal{X}} f_i(x)$ for $i = 1, ..., p$;
- $LB(B) = (LB_1(B), ..., LB_p(B))^t$, where $LB_i(B) = \min_{x\in B} f_i(x)$ for $i = 1, ..., p$;
- $UB(B) = (UB_1(B), ..., UB_p(B))^t$, where $UB_i(B) = \max_{x\in B} f_i(x)$ for $i = 1, ..., p$;
- $E_i(B) = \{x \in \mathcal{X} : L_i \leqslant f_i(x) \leqslant UB_i(B) - \varepsilon_i\}$, and $E(B) = \cap_{i=1}^p E_i(B)$;
- $E_0(B) = \{x \in \mathcal{X} : f(x) = UB(B) - \varepsilon\}$;
- $F_i(B) = \{x \in \mathcal{X} : L_i \leqslant f_i(x) \leqslant LB_i(B) - \varepsilon_i\}$, and $F(B) = \cap_{i=1}^p F_i(B)$;
- $F_0(B) = \{x \in \mathcal{X} : f(x) = LB(B) - \varepsilon\}$;

where ^{t} denotes the transpose operator.

Assumption 1: We assume that, for Problem (3), the vectors $L, LB(B)$, and $UB(B)$ exist for all $B \subseteq \mathcal{X}$.

Note that Assumption 1 is necessay for the next results which make the proposed algorithm applicable to Problem (3). Next, we present two propositions which are needed for building the algorithm.

Proposition 3.1. *Suppose that $B \subseteq \mathcal{X}$ is a box. Then, $E(B)\setminus E_0(B) \neq \emptyset$ if and only if*

$$\exists\ x \in \mathcal{X}\ \ni\ L \leqq f(x) \leq UB(B) - \varepsilon.$$

Proof. By the above definitions, the proof is straightforward. □

Proposition 3.2. *Consider Problem (3) and suppose that for a box $B \subseteq \mathcal{X}$, $F(B)\setminus F_0(B) \neq \emptyset$. Then B does not contain any efficient or $\varepsilon-$efficient solutions.*

Proof. Since $F(B)\setminus F_0(B) \neq \emptyset$, we have $\exists\ \hat{x} \in \mathcal{X}\ \ni\ L \leqq f(\hat{x}) \leq LB(B) - \varepsilon$. Now, let $x_0 \in B$ be arbitrary. Then, $f(\hat{x}) + \varepsilon \leq LB(B) \leqq f(x_0)$. Therefore, x_0 is not $\varepsilon-$efficient and, consequently, it is not also an efficient solution. □

Algorithm 3.1.
Input. An instance of Problem (3) as the given multicritteria optimization problem and $\varepsilon,\ \varepsilon^0 \in \mathbb{R}^p_{\geqq}$.
Step 1. Set $S = \{\mathcal{X}\}, T_1 = T_2 = T_3 = T_4 = \emptyset$, and calculate the vector of lower bounds $L = (L_1, ..., L_p)$.
Step 2. Select a box $B \in S$ and split it into $n = 2^m$ subboxes $B_1, ..., B_n$. Then, calculate $LB(B_i)$ and $UB(B_i)$ for $i = 1, ..., n$ and set $S = (S\setminus\{B\}) \cup$

$\{B_1, ..., B_n\}$.
Step 3. For each box $B \in S$, if one of the conditions (4) - (6) holds, then set $T_1 = T_1 \cup \{B\}$ and $S = S \diagdown \{B\}$:

(4) $\exists\ j \in \{1, ..., p\} \ \ni\ (UB_j(B) - L_j) < \varepsilon_j;$

(5) $(UB_j(B) - L_j) = \varepsilon_j, \forall\ j \in \{1, ..., p\};$

(6) $E(B) \diagdown E_0(B) = \emptyset.$

Step 4. For each box $B \in S$, if

$$F(B) \diagdown F_0(B) \neq \emptyset,$$

then set $T_3 = T_3 \cup \{B\}$ and $S = S \diagdown \{B\}$.
Step 5. For each box $B \in S$, if

$$\exists\ j \in \{1, ..., p\} \ \ni\ (UB_j(B) - LB_j(B)) > \varepsilon_j^0,$$

then set $T_4 = T_4 \cup \{B\}$ and $S = S \diagdown \{B\}$. Else $T_2 = T_2 \cup \{B\}$ and $S = S \diagdown \{B\}$.
Step 6. If $T_4 \neq \emptyset$ then:

(a) for each box $B \in T_4$, if

$$\exists\ B' \in T_1 \ \ni\ UB(B) \leqq LB(B'),$$

then set $T_1 = T_1 \cup \{B\}$ and $T_4 = T_4 \diagdown \{B\}$;

(b) for each box $B \in T_4$, if

$$\exists\ B' \in T_3 \ \ni\ UB(B') \leqq LB(B),$$

then set $T_3 = T_3 \cup \{B\}$ and $T_4 = T_4 \diagdown \{B\}$;

(c) set $S = S \cup T_4$ and $T_4 = \emptyset$.

Step 7. If $S \neq \emptyset$, then go to Step 2. Otherwise, stop.
Output. The output is the set:

$$X_A^\varepsilon = T_1 \cup T_2.$$

In the next section, it will be proved that T_1 in the output of Algorithm 3.1 contains $\varepsilon-$efficient solutions and T_2 contains some $(\varepsilon + \varepsilon^0)-$efficient solutions. Moreover, T_3 contains neither $\varepsilon-$efficient nor efficient solutions, and T_4 has the role of a temporary set.

Remark 3.3. Notice that Algorithm 3.1 is also suitable for more general shapes of the feasible region. In fact, if the feasible region can be approximated by a union of boxes $X_1, ..., X_r$, then in Step 1 of Algorithm 3.1 the set S should be set as $S = \{X_1, ..., X_r\}$.

Remark 3.4. Since $B \subset \mathbb{R}^m$, splitting it in Step 2 of Algorithm 3.1 to 2^m subboxes is possible by dividing all its sides to equal segments and then considering all end vertexes for building the subboxes. Following this process a square in $\mathbb{R}^2$ splits to 4 subsquares and a cube in $\mathbb{R}^3$ splits to 8 subcubes, and so on. Moreover, in Algorithm 3.1, each box should be removed or needs further splitting until the stopping criterion in Step 7 is satisfied. The large number of boxes make the algorithm inefficient or instable. The number of boxes is affected by input data ε and ε^0. By our experiments,

$0.01(x^i_{max}-x^i_{min}) \leq \varepsilon_i \leq 0.1(x^i_{max}-x^i_{min}),\ i=1,...,m\ \&\ \varepsilon^0=\alpha\varepsilon,\ (0<\alpha<1)$

are appropriate in many instances. Although in the other cases, the algorithm will work, but it may take considerable time to stop.

Now, we solve some examples by Algorithm 3.1.

Example 3.5. *Consider a multicriteria optimization problem as follows:*

$$\begin{array}{rcl} \min f_1((x_1,x_2)^t) & = & (x_1-1)^2 \\ \min f_2((x_1,x_2)^t) & = & (x_2-1)^2 \\ s.t. & & x\in\mathcal{X}=[0,2]\times[0,2]\subset\mathbb{R}^2. \end{array} \tag{7}$$

Since $(f_1(x),f_2(x))^t>(0,0)^t=(f_1((1,1)^t),f_2((1,1)^t))^t$ *for all* $(1,1)^t\neq(x_1,x_2)^t$ $\in\mathcal{X}$, *the only efficient solution is* $(1,1)^t$. *Consequently, for* $\varepsilon=(0.04,0.04)^t$, *if*

$$(f_1(x),f_2(x))^t\geq(f_1((1,1)^t),f_2((1,1)^t))^t+(0.04,0.04)^t=(0.04,0.04)^t$$

then x *is not an* $\varepsilon-$*efficient solution. Therefore,* $x\in\mathcal{X}$ *is an* $\varepsilon-$*efficient solution if it satisfies to the following relations:*

$$\left\{\begin{array}{l} f_1(x)\geqslant\varepsilon_1 \\ f_2(x)<\varepsilon_2\end{array}\right. \Leftrightarrow \left\{\begin{array}{l}(x_1-1)^2\geqslant 0.04 \\ (x_2-1)^2<0.04\end{array}\right. \Leftrightarrow \left\{\begin{array}{l} x_1\geqslant 1.2\ or\ x_1\leqslant 0.8 \\ 0.8<x_2<1.2\end{array}\right. , \tag{8}$$

$$\left\{\begin{array}{l} f_2(x)\geqslant\varepsilon_2 \\ f_1(x)<\varepsilon_1\end{array}\right. \Leftrightarrow \left\{\begin{array}{l}(x_2-1)^2\geqslant 0.04 \\ (x_1-1)^2<0.04\end{array}\right. \Leftrightarrow \left\{\begin{array}{l} x_2\geqslant 1.2\ or\ x_2\leqslant 0.8 \\ 0.8<x_1<1.2\end{array}\right. , \tag{9}$$

$$\left\{\begin{array}{l} f_1(x)\leqslant\varepsilon_1 \\ f_2(x)\leqslant\varepsilon_2\end{array}\right. \Leftrightarrow \left\{\begin{array}{l}(x_1-1)^2\leqslant 0.04 \\ (x_2-1)^2\leqslant 0.04\end{array}\right. \Leftrightarrow \left\{\begin{array}{l} 0.8\leqslant x_1\leqslant 1.2 \\ 0.8\leqslant x_2\leqslant 1.2\end{array}\right. . \tag{10}$$

Hence, the set of all $\varepsilon-$*efficient solutions is as follows:*

$$\begin{array}{rcl} \mathcal{X}^\varepsilon_E & = & \{(x_1,x_2)^t\in\mathbb{R}^2 \mid 0.8\leqslant x_1\leqslant 1.2,\ 0.8\leqslant x_2\leqslant 1.2\} \\ & \cup & \{(x_1,x_2)^t\in\mathbb{R}^2 \mid 0.8<x_1<1.2,\ (x_2\geqslant 1.2\ or\ x_2\leqslant 0.8)\} \\ & \cup & \{(x_1,x_2)^t\in\mathbb{R}^2 \mid (x_1\geqslant 1.2\ or\ x_1\leqslant 0.8),\ 0.8<x_2<1.2\}. \end{array}$$

Problem (7) is solved by Algorithm 3.1 which is coded by MATLAB. To solve it, $\varepsilon=(0.04,0.04)^t$ *and* $\varepsilon^0=(0.01,0.01)^t$ *are used. Figure 1 shows the output set of Algorithm 3.1. In Figure 1, the small boxes (yellow color) are* $(\varepsilon+\varepsilon^0)-$*efficient and the big ones (cyan color) are* $\varepsilon-$*efficient solutions. It can be seen from Figure 1 that Algorithm 3.1 finds all* $\varepsilon-$*efficient solutions.*

Example 3.6. *We consider a multicriteria optimization problem taken from [28]:*

$$\begin{array}{rcl} \min f_1((x_1,x_2)^t) & = & x_1^2+x_2^2 \\ \min f_2((x_1,x_2)^t) & = & (x_1-5)^2+(x_2-5)^2 \\ s.t. & & x\in\mathcal{X}=[-5,10]\times[-5,10]\subset\mathbb{R}^2. \end{array} \tag{11}$$

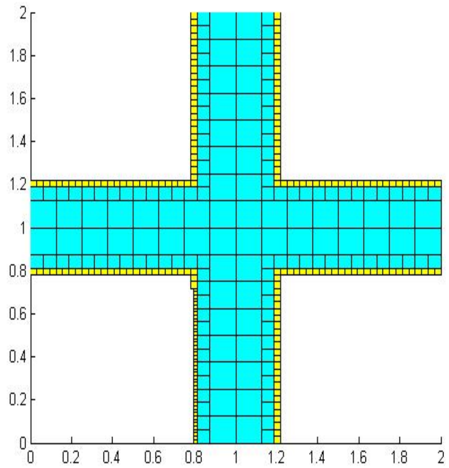

FIGURE 1. The output set obtained by Algorithm 3.1 for Problem (7).

Problem (11) is solved by Algorithm 3.1 with $\varepsilon = (0.5, 0.5)$ *and* $\varepsilon^0 = \frac{2}{3}\varepsilon$. *Figure 2 shows the* ε*-efficients and* $(\varepsilon+\varepsilon^0)-$*efficient solutions of Problem (11). Figure 3 shows the image of the points in Figure 2 in the objective space* ((f_1, f_2) *space). These two sets are compatible with the results of [28] to Problem (11).*

4. Main results

In the sequel, some results related to Algorithm 3.1 are summarized in some theorems.

Theorem 4.1. *Algorithm 3.1 terminates after a finite number of iterations for every* $\varepsilon \in \mathbb{R}^p_{\geqq}$ *and* $\varepsilon^0 \in \mathbb{R}^p{}_{>}$, *if there exist fixed constants* $C_1, ..., C_p > 0$ *such that:*

$$UB_i(B) - LB_i(B) \leqslant C_i\delta(B), \ \forall\ B \subseteq \mathcal{X}, \ \forall\ i = 1, ..., p.$$

Proof. By the split rule in Step 2, there is a finite number of iterations such that:

$$UB_i(B) - LB_i(B) \leqslant C_i\delta(B) \leqslant \varepsilon_i^0, \ \forall\ B \in S, \ \forall\ i = 1, ..., p.$$

Hence, after a finite number of iterations:

$$UB(B) - LB(B) \leqq \varepsilon^0\ \forall\ B \in S.$$

Now, Steps 5 and 7 ensure termination of Algorithm 3.1 after a finite number of iterations. $\square$

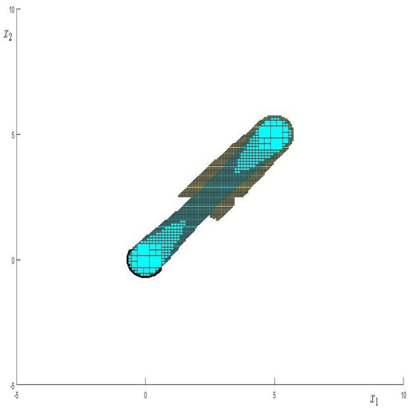

FIGURE 2. The output set obtained by Algorithm 3.1.

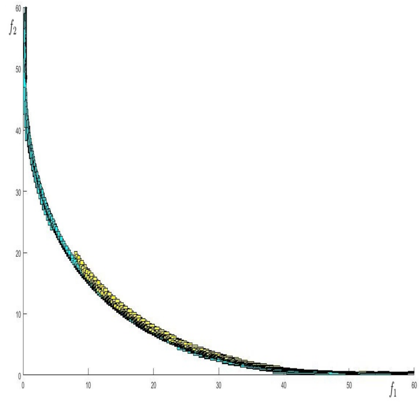

FIGURE 3. The image of the points in Figure 2 in the objective space.

Theorem 4.2. *Let $\mathcal{X}_A^\varepsilon$ be the output set obtained by Algorithm 3.1. Then, $\mathcal{X}_E^\varepsilon \subseteq \mathcal{X}_A^\varepsilon$.*

Proof. It suffices to show that Algorithm 3.1 does not delete any $\varepsilon-$efficient solutions. Since $\mathcal{X}_A^\varepsilon$ does not contain the sets T_3 and T_4, the elements of these two sets should be considered. T_4 is a temporary set. Thus, it is only enough to consider the elements of T_3. In other words, Step 4 and Step 6 should be analyzed. Rest of the proof is by induction on the number of iterations. Initially, $T_3 = \emptyset$ and therefore it has neither efficient nor $\varepsilon-$efficient solutions. Suppose that T_3 has not any $\varepsilon-$efficient solutions before the $k-$th iteration and consider the $k-$th iteration of Algorithm 3.1. Now:

In Step 4: Suppose that $B \in S$ is a box and $\hat{x} \in B$ is an $\varepsilon-$efficient solution. Therefore, $B \in S$ should not be discarded. On the contrary, assume that Step 4 removes B from S and adds it into T_3. It occurs when $F(B)\diagdown F_0(B) \neq \emptyset$. Now, Proposition 3.2 implies that B does not contain any $\varepsilon-$efficient solutions, which is a contradiction. Hence, Step 4 does not insert any $\varepsilon-$efficient solutions into T_3.

Also, on the contrary, suppose that Step 6(b) removes B from S and adds it into T_3, where B is a box which involves at least an $\varepsilon-$efficient solution $\hat{x} \in \mathcal{X}$. It occurs when:

$$\exists\ B' \in T_3 \ \ni\ UB(B') \leqq LB(B).$$

Since $B' \in T_3$ is augmented to T_3 in Step 4 or before the $k-$th iteration, all of the elements of B' are not $\varepsilon-$efficient solutions. Let $x' \in B$. Then:

$$\exists\ x^0 \in \mathcal{X} \ \ni\ f(x^0) + \varepsilon \leq f(x') \leqq UB(B') \leqq LB(B) \leqq f(\hat{x}).$$

Thus, $f(x^0)+\varepsilon \leq f(\hat{x})$. This means that $\hat{x}$ is not an $\varepsilon-$efficient solution, which is a contradiction. Therefore, T_3 in Algorithm 3.1 does not contain any $\varepsilon-$efficient solutions and thus $\mathcal{X}_E^\varepsilon \subseteq \mathcal{X}_A^\varepsilon$. □

Corollary 4.3. *Let $\mathcal{X}_A^\varepsilon$ be the output set obtained by Algorithm 3.1. Then, $\mathcal{X}_E \subseteq \mathcal{X}_A^\varepsilon$.*

Proof. By Theorem 4.2 and the fact that $\mathcal{X}_E \subseteq \mathcal{X}_E^\varepsilon$ for all $\varepsilon \in \mathbb{R}^p_{\geqq}$, the proof is obvious. □

Corollary 4.3 proves that the output set of Algorithm 3.1, $\mathcal{X}_A^\varepsilon = T_1 \cup T_2$, involves all efficient solutions. Moreover, the properties of the sets T_1 and T_2 are given in the next two theorems.

Theorem 4.4. *Let $\mathcal{X}_A^\varepsilon = T_1 \cup T_2$ be the output set obtained by Algorithm 3.1. Then $T_1 \subseteq \mathcal{X}_E^\varepsilon$.*

Proof. It is needed to show that every point of T_1 is an $\varepsilon-$efficient solution of Problem (3). Let $\hat{x} \in B \in T_1$. Then, according to the steps of Algorithm 3.1, four possible cases may be occurred as follows:

(1) $(UB_j(B)-L_j) < \varepsilon_j$ for some $j \in \{1,...,p\}$, (due to Step 3 - condition (4));
(2) $(UB(B)-L)=\varepsilon$, (due to Step 3 - condition (5));
(3) $(UB(B)-L) \geq \varepsilon, E(B)\diagdown E_0(B)=\emptyset$, (due to Step 3 - condition (6));
(4) $\exists\, B' \in T_1 \ni UB(B) \leqq LB(B')$, (due to Step 6(a)).

(1)- Suppose that there exists $j_0 \in \{1,...,p\} \ni (UB_{j_0}(B)-L_{j_0}) < \varepsilon_{j_0}$, and on the contrary $\hat{x} \in B$ is not an $\varepsilon-$efficient solution. Then, there exists $\bar{x} \in \mathcal{X} \ni f(\bar{x})+\varepsilon \leq f(\hat{x})$. Thus:

$$f_{j_0}(\bar{x})+\varepsilon_{j_0} \leqslant f_{j_0}(\hat{x}) \leqslant UB_{j_0}(B) < \varepsilon_{j_0}+L_{j_0}.$$

Therefore, $f_{j_0}(\bar{x}) < L_{j_0}$, which is a contradiction. Hence, in this case $\hat{x} \in \mathcal{X}_E^\varepsilon$.

(2)- On the contrary suppose that $\hat{x} \in B$ is not an $\varepsilon-$efficient solution. Then:

$$\exists\, \bar{x} \in \mathcal{X} \ni f(\bar{x})+\varepsilon \leq f(\hat{x}) \leqq UB(B)=\varepsilon+L,$$

Thus, $f(\bar{x}) \leq L$, which is a contradiction. Hence, $\hat{x} \in \mathcal{X}_E^\varepsilon$.

(3)- Suppose that $\hat{x} \in B$ is not an $\varepsilon-$efficient solution. Then, there exists $\bar{x} \in \mathcal{X} \ni f(\bar{x})+\varepsilon \leq f(\hat{x})$. Thus, $f(\bar{x}) \leq f(\hat{x})-\varepsilon \leqq UB(B)-\varepsilon$. Therefore, there exists $\bar{x} \in \mathcal{X} \ni L \leqq f(\bar{x}) \leq UB(B)-\varepsilon$. Now, Proposition 3.1 implies that $E(B)\diagdown E_0(B) \neq \emptyset$, which is a contradiction. Hence, $\hat{x} \in \mathcal{X}_E^\varepsilon$.

(4)- Since $B' \in T_1$, due to Step 4, $F(B')\diagdown F_0(B')=\emptyset$. Therefore:

$$\nexists\, x \in \mathcal{X} \ni f(x) \leq LB(B')-\varepsilon. \tag{12}$$

Now, on the contrary suppose that $\hat{x} \in B$ is not an $\varepsilon-$efficient solution. Then, there exists $\bar{x} \in \mathcal{X} \ni f(\bar{x})+\varepsilon \leq f(\hat{x})$. Thus, $f(\bar{x})+\varepsilon \leq f(\hat{x}) \leqq UB(B) \leqq$

$LB(B')$. It implies that $f(\bar{x}) + \varepsilon \leq LB(B')$, which is a contradiction to (12). Hence, in all cases, $\hat{x} \in \mathcal{X}_E^{\varepsilon}$ and the proof is complete. □

Theorem 4.5. *Let $\mathcal{X}_A^{\varepsilon} = T_1 \cup T_2$ be the output set obtained by Algorithm 3.1. Then $T_2 \subseteq \mathcal{X}_E^{\varepsilon+\varepsilon^0}$.*

Proof. To show that every point of T_2 is an $(\varepsilon+\varepsilon^0)-$efficient solution of Problem (3), let $\hat{x} \in B \in T_2$. Then by Step 5, $UB(B) - LB(B) \leqq \varepsilon^0$. Also, due to Step 4, $F(B) \setminus F_0(B) = \emptyset$. Thus:

$$\nexists\, x \in \mathcal{X} \ \ni\ f(x) \leq LB(B) - \varepsilon. \tag{13}$$

Now, on the contrary, suppose that $\hat{x}$ is not an $(\varepsilon+\varepsilon^0)-$efficient solution. Then, there exists $\bar{x} \in \mathcal{X} \ \ni\ f(\bar{x}) + \varepsilon + \varepsilon^0 \leq f(\hat{x})$. Therefore:

$$f(\bar{x}) + \varepsilon \leq f(\hat{x}) - \varepsilon^0 \leqq UB(B) - \varepsilon^0 \leqq LB(B).$$

Thus, we conclude that

$$L + \varepsilon \leqq f(\bar{x}) + \varepsilon \leq f(\hat{x}) - \varepsilon^0 \leqq LB(B).$$

It implies that

$$f(\bar{x}) + \varepsilon \leq LB(B). \tag{14}$$

Note that (14) is a contradiction to (13). Hence, $\hat{x} \in \mathcal{X}_E^{\varepsilon+\varepsilon^0}$ and the proof is complete. □

Notice that by Theorems 4.2-4.5, T_1 in the output set of Algorithm 3.1 contains all $\varepsilon-$efficient solutions of Problem (3).

5. An application of Algorithm 3.1 to a facility location problem

As an application, this section applies Algorithm 3.1 to a facility location problem. As discussed in Section 1, facility location is an important branch of operations research which aims to find a suitable location. It has been an active area of research since the early 1960's [25]. Using an appropriate location has many benefits such as saving time and cost. Consider n existing facilities $a_k \in \mathbb{R}^2$ for $k = 1, ..., n$. This paper considers planar 1-facility location problems. In other words, one new facility $x \in \mathbb{R}^2$ is needed to be established. Let w_k and v_k be the weights of facility a_k for $k = 1, ..., n$. Further, consider a distance function d. The most common used objective functions in literature are as follows [4, 10, 25]:

Weber problem. Minimize the sum of service costs from each existing facility to the new location:

$$min_{x \in \mathcal{X}} f_1(x) := \sum_{k=1}^{n} w_k d(a_k, x).$$

Some approaches use squared distance $d(a_k, x)^2$ in instead of $d(a_k, x)$ [8, 20]. By using the Euclidean distance, the sum of squared Euclidean distances meets the Pigou-Dalton condition of transfers [20] and can easily be minimized. Further, in this case, the optimal solution is the well known center of gravity. On the other hand, the squared distance measure is used where it is needed to omit excessive distances [6]. It is due to the fact that since the objective function should be minimized and squaring a large distance number results in larger distance number then we expect excessive distances to be omitted.

Center problem. Minimize the distance from the new location to the furthest existing facility:

$$min_{x\in\mathcal{X}} f_2(x) := max_{1\leq k\leq n} w_k d(a_k, x).$$

Obnoxious problem. Minimize the sum of the reciprocal squared distance from each existing facility to the new location:

$$min_{x\in\mathcal{X}} f_3(x) := \sum_{k=1}^{n} v_k/d(a_k, x)^2.$$

Note that sometimes $d(a_k, x)$ is used instead of $d(a_k, x)^2$ [25].

Maximin problem. Maximize the distance between the new facility and the nearest resident:

$$max_{x\in\mathcal{X}} f_4(x) := min_{1\leq k\leq n} v_k d(a_k, x).$$

In the above problems, the first two objectives represent an attractive new location while the latter two ones describe a repulsive new facility location. The current research assumes d is the Euclidean distance. Moreover, it considers a bicriteria facility location problem with the attractive objective f_1 and the repulsive objective f_3. It is clear that these two objectives are conflicting each other. This problem was discussed in [2, 25, 26]. A numerical example of this kind is given in the sequel.

Example 5.1. *The data of this example are taken from [25], where a bicriteria location problem with the Weber and the obnoxious objectives is considered. In other words, the bicriteria optimization problem is:*

$$\begin{aligned} \min f_1(x) &= \sum_{k=1}^{n} w_k d(a_k, x)^2 \\ \min f_3(x) &= \sum_{k=1}^{n} v_k/d(a_k, x) \\ s.t. \quad & x \in \mathcal{X}, \end{aligned} \tag{15}$$

where $\mathcal{X} = [0, 1] \times [0, 1]$ *and nonnegative weights* $w_k,\ v_k \in \mathbb{R}^1_>$ *as well as* $a_k \in \mathbb{R}^2$ *for* $k = 1, ..., 7$ *are given in Table 1.*

TABLE 1. Data for Example 5.1.

k	a_k^1	a_k^2	w_k	v_k
1	*0.20*	*0.80*	*5.0*	*1.0*
2	*0.72*	*0.32*	*7.0*	*1.0*
3	*0.88*	*0.64*	*2.0*	*1.0*
4	*0.56*	*0.68*	*3.0*	*1.0*
5	*0.28*	*0.08*	*6.0*	*1.0*
6	*0.20*	*0.60*	*1.0*	*1.0*
7	*0.48*	*0.16*	*5.0*	*1.0*

Let us first exam the efficiency of some feasible solutions. For instance, to test the efficiency of $(0, 0.25)^t$ *model (2) can be solved. Since* $f_1((0, 0.25)^t) = 10.72290$ *and* $f_3((0, 25)^t) = 13.12749$*, model (2) related to Problem (15) is as follows:*

$$
\begin{aligned}
\max z \;&=\; l_1 + l_2,\\
s.t.\;& \\
& \sum_{k=1}^{7} w_k d(a_k, x) + l_1 = 10.72290, \\
& \sum_{k=1}^{7} v_k / d(a_k, x)^2 + l_2 = 13.12749, \\
& x \in \mathcal{X}.
\end{aligned} \tag{16}
$$

Problem (16) is solved by LINGO 17. The optimal value of Problem (16) is zero. Therefore, by Theorem 2.2, the solution $(0, 0.25)^t$ *is an efficient solution and also an* $\varepsilon-$*efficient solution for every* $\varepsilon \in \mathbb{R}^p_{\geqq}$*. Moreover, some solutions in the vicinity of* $(0, 0.25)^t$ *should be* $\varepsilon-$*efficient solutions. The same is true for the solutions* $(1, 1)^t$ *and* $(0.4004913, 0.4046507)^t$*.*

To solve Problem (15) by Algorithm 3.1, we set $\varepsilon = (0.05, 0.05)^t$ *and* $\varepsilon^0 = \varepsilon/3$*. Algorithm 3.1 is coded by MATLAB 9.0 (R2016a) software. Figure 4 shows the output set of Algorithm 3.1 for Problem (15). Further,* T_1 *and* T_2 *are depicted separately in Figures 5 and 6, respectively. It can be seen from Figure 5 that Algorithm 3.1 obtains the points* $(0, 0.25)^t$*,* $(1, 1)^t$*, and* $(0.40049, 0.40465)^t$ *in its output set which are efficient solutions. However, these efficient solutions have not been obtained by the main MBCSC method in [25].*

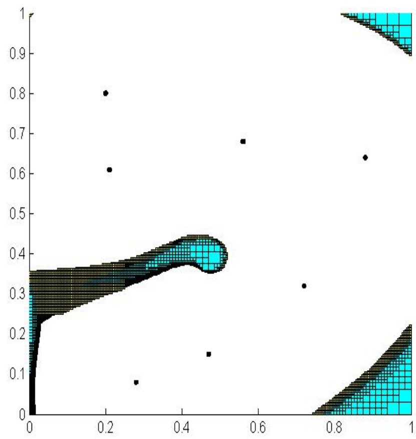

Figure 4. The output set obtained by Algorithm 3.1.

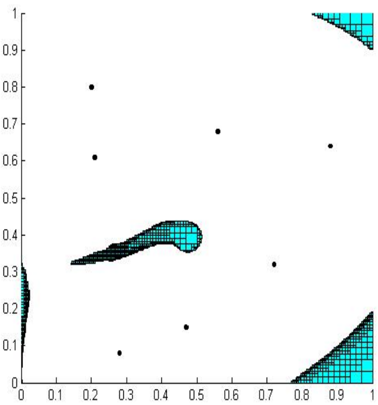

Figure 5. T_1 in the output of Algorithm 3.1.

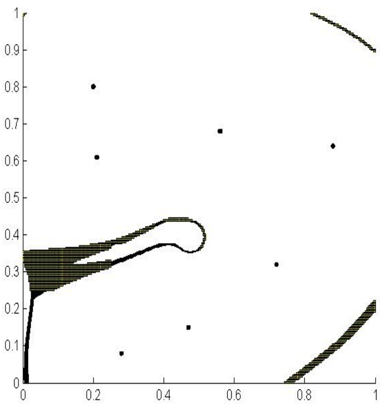

Figure 6. T_2 in the output of Algorithm 3.1.

6. **Concluding Remarks**

The current research developed an algorithm (Algorithm 3.1) to solve a multicriteria optimization problem where the feasible set is an $m-$dimensional cube. The output of Algorithm 3.1 is a set which contains all efficient and all $\varepsilon-$efficient solutions of the problem for a given ε. Algorithm 3.1 splits the feasible set into smaller boxes and then deletes the boxes that have not any efficient or $\varepsilon-$efficient solutions. A similar idea is used by the Multicriteria Big Cube Small Cube (MBCSC) method [25]. The MBCSC only uses the lower and upper bounds of the objectives over the boxes. However, Algorithm 3.1, in addition to those bounds, uses the lower bounds of the objectives over the entire feasible set as well. Moreover, the process of Algorithm 3.1 is different from the MBCSC. As a matter of fact, since the set of $\varepsilon-$efficient solutions is not closed, the MBCSC only attains a proper subset of $\varepsilon-$efficient solutions. In contrast to the MBCSC, Algorithm 3.1 obtains all $\varepsilon-$efficient solutions. Indeed, Algorithm 3.1 obtains some more solutions which are neither efficient nor $\varepsilon-$efficient solutions to overcome the fact that the set of $\varepsilon-$efficient solutions is not closed. Further, properties of Algorithm 3.1 are proved in a few theorems. Furthermore, to see the validity of Algorithm 3.1, it is applied to two numerical examples and a location problem. Extending Algorithm 3.1 for solving other kinds of multicriteria optimization problems and applying it to more location problems or other real world problems can be subjects for further research.

7. Aknowledgement

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions that helped us to considerably improve and clarify the first draft of the manuscript.

References

[1] Badri, M. A., Mortagy, A. K., Alsayed, C. A., A multi-objective model for locating fire stations, European Journal of Operational Research, 110 (1998) 243-260.

[2] Brimberg J., Juel, H., A bicriteria model for locating a semi-desirable facility in the plane, European Journal of Operational Research 106 (1998) 144-151.

[3] Current, J., Min, H., Schilling, D., Mutiobjective analysis of facility location decisions, European Journal of Operational Research, 49 (1990) 295-307.

[4] Drezner Z., Hamacher H. W., Facility Location: Application and Theory, Springer, Berlin, 2002.

[5] Drezner, Z., Suzuki, A., The big trinangle small triangle method for the solution of nonconvex facility location problems, Operations Research, 52 (2004) 128-135.

[6] Dvorak, T., Vlkovsky, M., Supply chain optimization models in the area of operation, Science & Military 1 (2011) 20-24.

[7] Ehrgott, M., Multicriteria Optimization, Springer, Berlin, 2005.

[8] Ehrgott, M., Hamacher, H. W., Nickel, S., Geometric methods to solve max-ordering location problems, Discrete Applied Mathematics, 93(1) (1999) 3-20.

[9] Eichfelder, G., Warnow, L., An approximation algorithm for multi-objective optimization problems using a box-coverage, Journal of Global Optimization 83 (2022) 329-357.

[10] Eiselt, H. A., Laporte, G., Facility Location: A survey of Application and Methods, Springer, Berlin, 1995.

[11] Engau, A., Wiecek, M., Exact generation of epsilon-efficient solutions in multiple objective programming, OR Spectrum, 29 (2007) 335-350.

[12] Engau, A., Wiecek, M., Generation $\varepsilon-$efficient solutions in multiobjective programming, European Journal of Operational Research, 177 (2007) 1566-1579.

[13] Farahani, R. Z., SteadieSeifi, M., Asgari, N., Multiple criteria facility location problems: A survey, Applied Mathematical Modelling, 34 (2010) 1689-1709.

[14] Hamacher, H. W., Nickel, S., Multicriteria planar location problems, European Journal of Operational Research, 94 (1996) 66-86.

[15] Hansen, P., Peeters, D., Richard, D., Thisse, J. F., The minisum and minimax location problems revisited, Operations Research, 33 (1985) 1251-1265.

[16] Hekmatfar, M., SteadieSeifi, M., Multi-criteria location problem, In: R.Z. Farahani and M. Hekmatfar (eds.), Facility Location: Concepts, Models, Algorithms and Case Studies, Contributions to Management Science, Physica-Verlag, Heidelberg, 2009, pp. 373-393.

[17] Loridan, P., $\varepsilon-$solutions in vector minimization problems, Journal of Optimization Theory and Applications, 43 (1984) 265-276.

[18] McGinnis, L. F., White, J. A., A single facility rectilinear location problem with multiple criteria. Transportation Science, 12 (1978) 217-231.

[19] Nickel, S., Puerto, J., Rodriguez-Chia, A. M., Weissler, A., Multicriteria planar ordered median problems, Journal of Optimization Theory and Applications, 126(3) (2005) 657-683.

[20] Ohsawa, Y., Ozaki, N., Plastria, F., Equity-efficiency bicriteria location with squared Euclidean distances, Operations Research, 56(1) (2008) 79-87.

[21] Plastria, F., GBSSS: the generalized big square small square method for planar single-facility location, European Journal of Operational Research, 62 (1992) 163-174.

[22] Pourkarimi, L., Yaghoobi, M. A., Mashinchi, M., Efficient Curve Fitting: An Application of Multiobjective Programming, Applied Mathematical Modelling, 35 (1) (2011) 346-365.

[23] Rakas, J., Teodorovic, D., Kim, T., Multi-objective modeling for determining location of undesirable facilities, Transportation Research Part D, 9 (2004) 125-138.

[24] Schobel, A., Scholz, D., The big cube small cube solution method for multidimensional facility location problems, Computers & Operations Research, 37 (2010) 115-122.

[25] Scholz, D., The multicriteria big cube small cube method, Top, 18 (2010) 286-302.

[26] Skriver A. J. V., Anderson, K. A., The bicriterion semi-obnoxious location (BSL) problem solved by an $\varepsilon-$approximation, European Journal of Operational Research 146 (2003) 517-528.

[27] Steuer, R. E., Multiple Criteria Optimization: Theory, Computation, and Application, John Wiley & Sons, New York, 1986.

[28] Thomann, J., Eichfelder, G., Numerical results for the multiobjective trust region algorithm MHT, Data in Brief, 25 (2019) 104103.

[29] Yaghoobi, M. A., Pourkarimi, L., Mashinchi, M., A multiobjective based approach for mathematical programs with linear flexible constraints, Applied Mathematical Modelling, 36 (2012) 6264-6274.

N. Ghafari
Orcid number: 0000-0002-3179-315X
Department of Applied Mathematics
Faculty of Mathematics and Computer
Shahid Bahonar University of Kerman
Kerman, Iran.
Email address: nemat.ghafari12@gmail.com

M.A. Yaghoobi
Orcid number: 0000-0003-1320-2552
Department of Applied Mathematics
Faculty of Mathematics and Computer
Shahid Bahonar University of Kerman
Kerman, Iran
Email address: yaghoobi@uk.ac.ir