AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(1): 1959–1974.
DOI: 10.3934/math.2023101
Received: 18 July 2022
Revised: 23 September 2022
Accepted: 10 October 2022
Published: 26 October 2022

Research article

Outer space branching search method for solving generalized affine fractional optimization problem

Junqiao Ma[1], **Hongwei Jiao**[1,2,*], **Jingben Yin**[1] **and Youlin Shang**[2]

[1] School of Mathematical Sciences, Henan Institute of Science and Technology, Xinxiang 453003, China

[2] Postdoctoral Research Station of Control Science and Engineering, Henan University of Science and Technology, Luoyang 471023, China

* **Correspondence:** Email: jiaohongwei@126.com.

Abstract: This paper proposes an outer space branching search method, which is used to globally solve the generalized affine fractional optimization problem (GAFOP). First, we will convert the GAFOP into an equivalent problem (EP). Next, we structure the linear relaxation problem (LRP) of the EP by using the linearization technique. By subsequently partitioning the initial outer space rectangle and successively solving a series of LRPs, the proposed algorithm globally converges to the optimum solution of the GAFOP. Finally, comparisons of numerical results are reported to show the superiority and the effectiveness of the presented algorithm.

Keywords: generalized affine fractional optimization; global optimization; linear relaxation problem; outer space branching search method; computational complexity
Mathematics Subject Classification: 90C26, 90C32, 65K05

1. Introduction

The considered generalized affine fractional optimization problem is as follows:

$$
\text{(GAFOP)}:\begin{cases} \min\max & \left\{ \dfrac{\sum\limits_{j=1}^{n} e_{1j}y_j + f_1}{\sum\limits_{j=1}^{n} c_{1j}y_j + h_1}, \dfrac{\sum\limits_{j=1}^{n} e_{2j}y_j + f_2}{\sum\limits_{j=1}^{n} c_{2j}y_j + h_2}, \ldots, \dfrac{\sum\limits_{j=1}^{n} e_{pj}y_j + f_p}{\sum\limits_{j=1}^{n} c_{pj}y_j + h_p} \right\} \\ \text{s.t.} & y \in Y = \{y \in \mathbb{R}^n | Ay \le b\}, \end{cases}
$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $e_i, d_i \in \mathbb{R}^n$, and g_i, f_i are arbitrary real numbers. $p \geq 2$, Y is a nonempty compact set, $\sum_{j=1}^{n} e_{ij}y_j + f_i$ and $\sum_{j=1}^{n} c_{ij}y_j + h_i$ are all bounded linearity functions defined on Y, and for any $y \in Y$, the denominator $\sum_{j=1}^{n} c_{ij}y_j + h_i \neq 0, i = 1, 2, \ldots, p$.

As a class of special fractional optimization problems, the GAFOP has attracted the attention of many researchers and practitioners for decades. It has a variety of applications in many fields, including finance and investment [1–3], transportation planning [4, 5], optimal design [6], estimation of iterative parameters [7], signal processing [8], data envelopment analysis and others [9–17]. Furthermore, since the GAFOP may be not (quasi)convex, there may exist many local optimal solutions, many of which fail to be global solutions. Hence, it is still of great significance to propose an effective global algorithm to solve the GAFOP.

Some algorithms have been presented to globally solve the GAFOP over the past several decades: for instance, the cutting plane algorithm [18], branch-relaxation-bound methods [19, 20], the interior-point algorithm [21], the partial linearization algorithm [22], the monotonic optimization method [23], the method of centers [24] and the prox-regularization method [25]. Recently, based on the Dinkelbach type algorithm, Ghazi and Roubi [26] presented a difference of convex functions (DC) method for globally solving the generalized convex fractional optimization problem. By utilizing the proximal bundle theory, Boualam and Roubi [27] proposed a dual method for the generalized convex fractional optimization problem. Jiao et al. [28] designed an image space branch-and-bound algorithm for solving minimax linear fractional programs. Haffari and Roubi [29] described a prox-dual regularization method for globally solving generalized fractional programs. By utilizing convex hull and concave hull approximation of a bilinear function, Jiao and Li [30] put forward a novel algorithm for globally addressing min-max linear fractional programs. However, the previous reviewed methods can only solve a particular form of the GAFOP, or they are difficult to use to solve large-scale practical problems. Therefore, there remains the necessity to propose a practical algorithm to solve the GAFOP.

In addition to the methods reviewed above, some theoretical progress on the generalized fractional optimization problem (GFOP) has also been made. For example, Ahmad and Husain [31] gave a duality theory for a non-differentiable GFOP with generalized convexity. Schmitendorf [32] presented some optimality conditions for the GFOP. By utilizing the optimality condition, Tanimoto [33] gave a dual problem for a class of non-differentiable GFOP and derived the duality theorems. Yadav and Mukherjee [34] gave a duality theory for GFOP. When the data in the system are uncertain, Jeyakumar et al. [35] put forward a strong duality theorem for the robust GFOP. Based on unconstrained conditions, Lai et al. [36] gave the duality theorem for the GFOP. For a detailed review of the methods and theories for the GFOP, readers can refer to Stancu-Minasian [37, 38].

In this article, an outer space branching search method is designed for globally solving the GAFOP. We first convert the GAFOP into the EP. Next, by utilizing the structural characteristics of the EP, we construct a new linearizing method for establishing the LRP of the EP. Compared with the known existing algorithms, the branching search of the presented algorithm occurs in the outer space $\mathbb{R}^p$ rather than the variable dimension space $\mathbb{R}^n$, which provides the possibility of mitigating the required computational efforts of the algorithm. In addition, the numerical computational results are reported, indicating that the proposed algorithm has higher efficiency and notable superiority compared to the known existing algorithms [19, 39, 40].

The remainder of this article is organized as follows. We derive the EP of the GAFOP and establish the LRP of the EP in Section 2. In Section 3, we give an outer space branching search method for globally solving the GAFOP, and we also analyze the global convergence of the algorithm. Numerical results for some test examples from recent studies are presented in Section 4. Finally, Section 5 gives some conclusions.

2. Linear relaxation programming problem

In the following, to solve the GAFOP, we first transform the GAFOP into the EP. Then, we present a novel linearization technique and construct the LRP of the EP. For this purpose, for each $i = 1, \ldots, p$, we introduce the additional variables $z_i = \sum_{j=1}^{n} c_{ij}y_j + h_i$. By computing the minimum value $\underline{z}_i^0 = \min_{y\in Y} \sum_{j=1}^{n} c_{ij}y_j + h_i$ and the maximum value $\overline{z}_i^0 = \max_{y\in Y} \sum_{j=1}^{n} c_{ij}y_j + h_i$ of the linear function $\sum_{j=1}^{n} c_{ij}y_j + h_i$ over Y, an initial outer space rectangle $Z^0 = \{z \in \mathbb{R}^p \mid \underline{z}_i^0 \leq z_i \leq \overline{z}_i^0,\ i = 1, \ldots, p\}$, can be constructed. By introducing the new variable $r = \max\left\{\frac{\sum_{j=1}^{n} e_{1j}y_j+f_1}{z_1}, \frac{\sum_{j=1}^{n} e_{2j}y_j+f_2}{z_2}, \ldots, \frac{\sum_{j=1}^{n} e_{pj}y_j+f_p}{z_p}\right\}$, we can simplify the objective function of the original problem GAFOP to r, so that we can get the EP of the GAFOP as below:

$$
(\text{EP}): \begin{cases} \min & r \\ \text{s.t.} & \frac{\sum_{j=1}^{n} e_{ij}y_j+f_i}{z_i} \leq r,\ i = 1, 2, \ldots, p \\ & z_i = \sum_{j=1}^{n} c_{ij}y_j + h_i \\ & Ay \leq b,\ z \in Z^0. \end{cases}
$$

Theorem 1. y^* is a global optimum solution of the GAFOP if and only if (y^*, z^*, r^*) is a global optimum solution of the EP, with

$$
z_i^* = \sum_{j=1}^{n} c_{ij}y^* + h_i, i = 1, 2, \ldots, p,
$$

and

$$
r^* = \max\left\{\frac{\sum_{j=1}^{n} e_{1j}y_j^* + f_1}{z_1^*}, \frac{\sum_{j=1}^{n} e_{2j}y_j^* + f_2}{z_2^*}, \ldots, \frac{\sum_{j=1}^{n} e_{pj}y_j^* + f_p}{z_p^*}\right\}.
$$

Additionally, the global optimal values of the GAFOP and EP are equal.

Proof. By the above discussion, the conclusions are obvious, and thus we omit the proof. □

By Theorem 1, to globally solve the GAFOP, we can instead solve the EP. In the following, we only consider solving the EP.

For globally solving the EP, we need to establish its LRP for providing the lower bound in the branch-and-bound process. The detailed derivation process of the LRP is as follows.

For any $Z = \{z \in \mathbb{R}^p \mid \underline{z}_i \leq z_i \leq \overline{z}_i,\ i = 1, \ldots, p\} \subseteq Z^0$, we define

$$\Phi_i(y, z_i) = \frac{\sum_{j=1}^{n} e_{ij} y_j + f_i}{z_i},$$

$$\underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i) = \begin{cases} \sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\overline{z}_i}, & \text{if } f_i > 0, \\ \sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\underline{z}_i}, & \text{if } f_i < 0. \end{cases}$$

Obviously, for each $i = 1, \ldots, p$, we can see that

$$\Phi_i(y, z_i) = \frac{\sum_{j=1}^{n} e_{ij} y_j + f_i}{z_i} \geq \underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i) = \begin{cases} \sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\overline{z}_i}, & \text{if } f_i > 0, \\ \sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\underline{z}_i}, & \text{if } f_i < 0. \end{cases} \tag{1}$$

Based on (1), for any $Z \subseteq Z^0$, we can construct the LRP of the EP as below.

$$(\text{LRP}): \begin{cases} \min & r \\ \text{s.t.} & \underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i) \leq r,\ i = 1, 2, \ldots, p, \\ & z_i = \sum_{j=1}^{n} c_{ij} y_j + h_i \\ & Ay \leq b,\ y \geq 0,\ z \in Z. \end{cases}$$

From the above discussion, it is known that all feasible points of the EP over the sub-rectangle Z are also feasible for the LRP. Let $v(\text{EP})$ and $v(\text{LRP})$ be the global optimal values of the LRP and EP, respectively, and we have $v(\text{LRP}) \leq v(\text{EP})$ over Z^k. Thus, the optimal value of the LRP will provide a valid lower bound for that of the EP over Z.

Next, we will prove that the optimal solution of the LRP will infinitely approximate the optimal solution of the EP over Z as $\|\overline{z} - \underline{z}\| \to 0$, as detailed in Theorem 2.

Theorem 2. For each $i = 1, 2, \ldots, p$, consider the functions $\Phi_i(y, z_i)$ and $\underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i)$. We have the following:

$$\lim_{\|\overline{z} - \underline{z}\| \to 0} \left(\Phi_i(y, z_i) - \underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i) \right) = 0.$$

Proof. By the definitions of the functions $\Phi_i(y, z_i)$ and $\underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i)$, for any $y \in Y$, $z_i \in [\underline{z}_i, \overline{z}_i]$, we have

$$\Phi_i(y, z_i) - \underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i) = \begin{cases} \frac{\sum_{j=1}^{n} e_{ij}y_j + f_i}{z_i} - \left[\sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\overline{z}_i}\right], & \text{if } f_i > 0 \\ \frac{\sum_{j=1}^{n} e_{ij}y_j + f_i}{z_i} - \left[\sum_{j=1, e_{ij}>0}^{n} \frac{e_{ij}}{\overline{z}_i} y_j + \sum_{j=1, e_{ij}<0}^{n} \frac{e_{ij}}{\underline{z}_i} y_j + \frac{f_i}{\underline{z}_i}\right], & \text{if } f_i < 0 \end{cases}$$

$$= \begin{cases} \sum_{j=1, e_{ij}>0}^{n} \left[\frac{e_{ij}y_j}{z_i} - \frac{e_{ij}y_j}{\overline{z}_i}\right] + \sum_{j=1, e_{ij}<0}^{n} \left[\frac{e_{ij}y_j}{z_i} - \frac{e_{ij}y_j}{\underline{z}_i}\right] + \left[\frac{f_i}{z_i} - \frac{f_i}{\overline{z}_i}\right], & \text{if } f_i > 0 \\ \sum_{j=1, e_{ij}>0}^{n} \left[\frac{e_{ij}y_j}{z_i} - \frac{e_{ij}y_j}{\overline{z}_i}\right] + \sum_{j=1, e_{ij}<0}^{n} \left[\frac{e_{ij}y_j}{z_i} - \frac{e_{ij}y_j}{\underline{z}_i}\right] + \left[\frac{f_i}{z_i} - \frac{f_i}{\underline{z}_i}\right], & \text{if } f_i < 0 \end{cases}$$

$$= \begin{cases} \frac{(\overline{z}_i - z_i)}{z_i \overline{z}_i} \sum_{j=1, e_{ij}>0}^{n} e_{ij}y_j + \frac{(\underline{z}_i - z_i)}{z_i \underline{z}_i} \sum_{j=1, e_{ij}<0}^{n} e_{ij}y_j + \frac{f_i(\overline{z}_i - z_i)}{z_i \overline{z}_i}, & \text{if } f_i > 0 \\ \frac{(\overline{z}_i - z_i)}{z_i \overline{z}_i} \sum_{j=1, e_{ij}>0}^{n} e_{ij}y_j + \frac{(\underline{z}_i - z_i)}{z_i \underline{z}_i} \sum_{j=1, e_{ij}<0}^{n} e_{ij}y_j + \frac{f_i(\underline{z}_i - z_i)}{z_i \underline{z}_i}, & \text{if } f_i < 0 \end{cases}$$

$$\leq \frac{(\overline{z}_i - \underline{z}_i)}{\underline{z}_i^2} \left[\sum_{j=1}^{n} |e_{ij}| y_j + |f_i|\right].$$

Since $\sum_{j=1}^{n} |e_{ij}| y_j + |f_i|$ is a bounded linear function, we have

$$\lim_{\|\overline{z} - \underline{z}\| \to 0} \left(\Phi_i(y, z_i) - \underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i)\right) = 0$$

and complete the proof of the Theorem. □

The above Theorem ensures that the function $\Phi_i(y, z_i)$ will be infinitely approximated by the function $\underline{\Phi}_i(y, \underline{z}_i, \overline{z}_i)$ as $\|\overline{z} - \underline{z}\| \to 0$, so the global optimal solution of the LRP will infinitely approximate the global optimal solution of the EP over Z as $\|\overline{z} - \underline{z}\| \to 0$.

3. Global algorithm and its convergence

In this section, we first put forward an outer space rectangle bisection method. Next, by combining the previous LRP and the branch-and-bound framework, an outer space branching search method is designed to globally solve the GAFOP. In addition, we derive the global convergence of the outer space branching search method.

3.1. Outer space rectangle bisection method

The outer space rectangle bisection method iteratively subdivides the currently investigated rectangle into two sub-rectangles. Consider any selected sub-rectangle $Z = \{z \in \mathbb{R}^p | \underline{z}_i \leq z_i \leq \overline{z}_i, i = 1, 2, \ldots, p\} \subseteq Z^0$. The outer space rectangle bisection method is given as follows:

(i) Let $q = \arg\max\{\overline{z}_i - \underline{z}_i | i = 1, 2, \ldots, p\}$;

(ii) Let

$$Z^1 = \{z \in \mathbb{R}^p | \underline{z}_i \leq z_i \leq \overline{z}_i, i = 1, 2, \ldots, p, i \neq q;\ \underline{z}_q \leq z_q \leq (\underline{z}_q + \overline{z}_q)/2\}$$

and

$$Z^2 = \{z \in \mathbb{R}^p | \underline{z}_i \leq z_i \leq \overline{z}_i, i = 1, 2, \ldots, p, i \neq q;\ (\underline{z}_q + \overline{z}_q)/2 \leq z_q \leq \overline{z}_q\}.$$

Through utilizing the proposed outer space rectangle bisection method, the selected sub-rectangle Z can be subdivided into two sub-rectangles Z^1 and Z^2.

3.2. Outer space branching search method

In this subsection, the basic steps of the proposed outer space branching search method are formulated as follows.

Step 0. Let the convergence error $\epsilon \geq 0$, and let the initial outer space rectangle

$$Z^0 = \{z \in \mathbb{R}^p \mid \underline{z}_i^0 \leq z_i \leq \overline{z}_i^0,\ i = 1, 2, \ldots, p\}.$$

Denote $F = \emptyset$ as the set of the initial feasible points, let $k = 0$, and let the set of all active nodes $\Omega_0 = \{Z^0\}$.

Step 1. Solve the LRP over Z^0, and define (y^0, z^0, r^0) and LB_0 as its optimal solution and optimal value. Let

$$UB_0 = \max\left\{\frac{\sum_{j=1}^n e_{1j}y_j^0 + f_1}{\sum_{j=1}^n c_{1j}y_j^0 + h_1}, \frac{\sum_{j=1}^n e_{2j}y_j^0 + f_2}{\sum_{j=1}^n c_{2j}y_j^0 + h_2}, \ldots, \frac{\sum_{j=1}^n e_{pj}y_j^0 + f_p}{\sum_{j=1}^n c_{pj}y_j^0 + h_p}\right\}.$$

If $UB_0 - LB_0 \leq \epsilon$, then the proposed algorithm stops. y^0 and $(y^0, z^0, \hat{r}^0)$ are ϵ-optimal solutions of the GAFOP and EP over (Z^0), respectively. Otherwise, proceed with Step 2.

Step 2. Use the proposed rectangle bisection method to subdivide Z^{k-1} into two sub-rectangles $Z^{k,1}$ and $Z^{k,2}$. Let $Q = \{Z^{k,1}, Z^{k,2}\}$.

Step 3. For each $Z^{k,t}, t = 1, 2$, compute the lower bound $LB(Z^{k,t})$ and $(y(Z^{k,t}), z(Z^{k,t}), r(Z^{k,t}))$ by solving the LRP over $Z^{k,t}$, and let

$$UB(Z^{k,t}) = \max\left\{\frac{\sum_{j=1}^n e_{1j}y_j^0(Z^{k,t}) + f_1}{\sum_{j=1}^n c_{1j}y_j^0(Z^{k,t}) + h_1}, \frac{\sum_{j=1}^n e_{2j}y_j^0(Z^{k,t}) + f_2}{\sum_{j=1}^n c_{2j}y_j^0(Z^{k,t}) + h_2}, \ldots, \frac{\sum_{j=1}^n e_{pj}y_j^0(Z^{k,t}) + f_p}{\sum_{j=1}^n c_{pj}y_j^0(Z^{k,t}) + h_p}\right\}.$$

If $LB(Z^{k,t}) > UB_k$, then set $Q = Q \backslash Z^{k,t}$; else, let

$$F = F \bigcup \{(y(Z), z(Z))\} \text{ and } UB_k = \min\{UB_k, UB(Z^{k,t})\}.$$

If $UB_k = UB(Z^{k,t})$, then let $y^k = y(Z^{k,t})$ and $(y^k, z^k, \hat{r}^k) = (y(Z^{k,t}), z(Z^{k,t}), r(Z^{k,t}))$.

Step 4. Set $\Omega_k = (\Omega_{k-1} \backslash Z^{k-1}) \bigcup Q$.

Step 5. Set $LB_k = \min\{LB(Z) | Z \in \Omega_k\}$, and let Z^k be the sub-rectangle which satisfies $LB_k = LB(Z^k)$.

If $UB_k - LB_k \leq \epsilon$, then the proposed algorithm stops. y^k and (y^k, z^k) are the ϵ-global optimal solutions of the GAFOP and EP, respectively.

Otherwise, set $k = k + 1$, and go back to Step 2.

3.3. Global convergence analysis

In this part, first of all, we define

$$\Lambda(y) = \max\left\{\frac{\sum_{j=1}^{n} e_{1j}y_j + f_1}{\sum_{j=1}^{n} c_{1j}y_j + h_1}, \frac{\sum_{j=1}^{n} e_{2j}y_j + f_2}{\sum_{j=1}^{n} c_{2j}y_j + h_2}, \ldots, \frac{\sum_{j=1}^{n} e_{pj}y_j + f_p}{\sum_{j=1}^{n} c_{pj}y_j + h_p}\right\}.$$

Let v be the global optimal value of the EP over Θ^0, and define $r(y^k, z^k)$ as the objective functional value of the EP corresponding to the feasible solution (y^k, z^k). The global convergence analysis of the proposed algorithm can be given by the following theorem.

Theorem 3. Given any $\epsilon \geq 0$, if the proposed algorithm finitely terminates after k iterations, then y^k is a global ϵ-optimal solution to the GAFOP in the sense that

$$r^k(y^k, z^k) \leq v + \epsilon.$$

Otherwise, the proposed algorithm will generate an infinite sequence $\{y^k\}$, whose accumulation point will be a global optimum solution to the GAFOP.

Proof. If the presented algorithm finitely terminates after k iterations, according to the termination of the algorithm, it follows that

$$UB_k - LB_k \leq \epsilon.$$

By Step 3 of the presented algorithm, we can find a feasible solution (y^k, z^k) to the EP such that

$$r(y^k, z^k) - LB_k \leq \epsilon \text{ and } LB_k \leq v.$$

Since (y^k, z^k) is feasible for the EP, we have

$$r(y^k, z^k) \geq v.$$

By using the above conclusions, we have

$$v \leq r(y^k, z^k) \leq LB_k + \epsilon \leq v + \epsilon.$$

So, (y^k, z^k) is a global ϵ-optimal solution of the EP, with

$$v \leq r(y^k, z^k) \leq v + \epsilon.$$

Thus, y^k is a global ϵ-optimum solution to the GAFOP.

If the presented algorithm does not finitely terminate, then it must produce an infinite feasible solution sequence $\{(y^k, z^k)\}$, and the sequence $\{(y^k, z^k)\}$ has a convergence subsequence. Therefore, we can let

$$\lim_{k\to\infty}(y^k, z^k) = (y^*, z^*).$$

So, we have

$$\lim_{k\to\infty} z_i^k = z_i^* = \sum_{j=1}^{n} c_{ij}y_j^* + h_i, i = 1, 2, \ldots, p.$$

From the branch-and-bound structure of the algorithm, we also get

$$\lim_{k\to\infty} LB_k \leq v.$$

Since y^* is a feasible solution of the GAFOP over Z^0, and due to Theorem 2, we can get

$$v \leq \Lambda(y^*).$$

Combining the above inequalities, we have

$$\Lambda(y^*) \geq v \geq \lim_{k\to\infty} LB_k = \lim_{k\to\infty} r(y^k, z^k) = r(y^*, z^*). \tag{2}$$

Furthermore, by the equivalence of the GAFOP and EP, and the continuity of the function $\Lambda(y)$, we can conclude the following:

$$\lim_{k\to\infty} r(y^k, z^k) = r(y^*, z^*) = \Lambda(y^*) = \lim_{k\to\infty} \Lambda(y^k). \tag{3}$$

Based on the above inequalities (2) and (3), we have

$$v = \Lambda(y^*) = \lim_{k\to\infty} \Lambda(y^k) = r(y^*, z^*) = \lim_{k\to\infty} LB_k.$$

Therefore, this implies that any accumulation point y^* of the sequence $\{y^k\}$ is a globally optimum solution to the GAFOP. The proof is complete. □

4. Numerical comparisons

For verifying the computational superiority of the algorithm, the presented algorithm is implemented in the software MATLAB R2014a and solved on the same microcomputer with an Intel(R) Core(TM) i5-7200U CPU @2.50 GHz processor and 4 GB RAM.

We first tested some randomly generated Problem 1 with small-size variables, numerically compared them with the known existing algorithms [19, 39, 40] and listed these numerical comparison results in Table 1. Next, we tested some randomly generated Problem 1 with large-size variables to verify our algorithm further and listed the numerical results in Table 2. In Table 2, Avg.Iter represents the average iteration times and Avg.Time represents the average execution CPU time in seconds.

Table 1. Numerical comparisons among some algorithms and our algorithm on Problem 1.

(p, m, n)	Algorithms	#iter			Time(s)		
		min.	ave.	max.	min.	ave.	max.
(2,10,2)	Feng et al. [19]	39	361	1188	0.45	4.22	13.11
	Wang et al. [39]	**11**	**19.5**	**35**	**0.13**	**0.24**	**0.42**
	Jiao & Liu [40]	14	24.5	38	0.16	0.31	0.46
	Our algorithm	28	275.5	971	0.33	3.30	11.37
(2,10,4)	Feng et al. [19]	365	6579.3	18164	3.92	77.36	213.11
	Wang et al. [39]	133	340.3	833	1.45	3.79	9.00
	Jiao & Liu [40]	**38**	**190.9**	**498**	**0.42**	**2.18**	**5.55**
	Our algorithm	**14**	**51**	**601**	**0.17**	**0.79**	**7.31**
(2,10,6)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	79	4165.8	24017	0.92	48.07	285.29
	Jiao & Liu [40]	220	661.3	1806	2.41	7.26	19.82
	Our algorithm	**36**	**265.3**	**439**	**0.43**	**3.24**	**5.35**
(2,10,8)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	189	9030.5	44047	2.03	118.66	654.93
	Jiao & Liu [40]	1205	7875	59143	13.03	115.83	940.51
	Our algorithm	**31**	**84**	**520**	**0.40**	**1.04**	**6.11**
(2,10,10)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	–	–	–	–	–	–
	Jiao & Liu [40]	613	4679.4	10880	6.72	52.54	124.93
	Our algorithm	**48**	**168.9**	**452**	**0.56**	**2.02**	**5.32**
(3,10,10)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	–	–	–	–	–	–
	Jiao & Liu [40]	2599	8162.3	12849	28.56	93.13	150.47
	Our algorithm	**183**	**1232.8**	**3860**	**2.17**	**15.2**	**47.8**
(4,10,10)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	–	–	–	–	–	–
	Jiao & Liu [40]	1629	21785.3	83513	17.83	340.12	1510.87
	Our algorithm	**1071**	**8368.7**	**31234**	**12.53**	**120.12**	**537.53**
(5,10,10)	Feng et al. [19]	–	–	–	–	–	–
	Wang et al. [39]	–	–	–	–	–	–
	Jiao & Liu [40]	2894	34659.2	179384	31.37	859.55	6021.98
	Our algorithm	**1943**	**27459**	**59576**	**22.41**	**497.90**	**1259.80**

Table 2. Numerical computational results of our algorithm for Problem 1.

(p,m,n)	Avg.N	Avg.T
(2,100,1000)	40.2	46.0363
(2,100,2000)	45.4	116.6136
(2,100,3000)	44.4	181.6550
(2,100,4000)	35.7	195.7975
(2,100,5000)	34.1	252.4213
(2,100,6000)	31.2	278.1843
(2,100,7000)	29.1	312.0120
(2,100,8000)	18.6	219.5177
(3,100,1000)	302.1	365.2927
(3,100,2000)	499.2	1424.3399
(3,100,3000)	393.3	1792.6109
(3,100,4000)	200.7	1232.3972

The maximum CPU time limit of all algorithms is set at 3600 s, and the approximation error is set as $\epsilon = 10^{-2}$. "–" denotes the situation in which the used algorithm failed to terminate in 3600 s. Since the known existing algorithms [19, 39, 40] failed to solve ten arbitrary randomly generated Problem 1 with large-size variables in 3600 s, we only list the numerical results obtained by our algorithm in Table 2.

We solved ten arbitrary randomly generated examples for all test problems. First of all, we tested the randomly generated Problem 1 with small-size variables. Table 1 shows the best results, worst results and average results among these ten test results, and we highlighted in bold the winners of these average results in their numerical comparison results in Table 1. Second, we solved the randomly generated Problem 1 with large-size variables, and numerical results are reported in Table 2.

From the computational results of Table 1, it can be seen that, when $p \geq 2, m \geq 10$, and $n \geq 6$, the algorithm of Feng et al. [19] failed to solve any one of ten randomly generated Problem 1 in 3600 s. When $p \geq 2, m \geq 10$, and $n \geq 10$, the algorithm of Wang et al. [39] failed to solve any one of ten randomly generated Problem 1 in 3600 s. When $p \geq 3, m \geq 10$, and $n \geq 20$, the algorithm of Jiao & Liu [40] failed to solve any one of ten randomly generated Problem 1 in 3600 s. However, in all cases, our algorithm can globally solve any one of ten randomly generated Problem 1. In addition, when $p \geq 2, m \geq 10$, and $n \geq 6$, compared with the known existing algorithms [19, 39, 40], our algorithm takes less running time and iterations. Thus, our algorithm has better computational superiority than the algorithms of Feng et al. [19], Wang et al. [39] and Jiao & Liu [40].

From the computational results of Table 2, it is obvious that the proposed algorithm can globally solve Problem 1 with large-size variables, and this demonstrates the strong robustness and the reliable stability of our algorithm.

Problem 1.

$$\begin{cases} \min\ \max \left\{ \frac{\sum_{j=1}^{n} d_{1j}y_j + g_1}{\sum_{j=1}^{n} e_{1j}y_j + h_1}, \frac{\sum_{j=1}^{n} d_{2j}y_j + g_2}{\sum_{j=1}^{n} e_{2j}y_j + h_2}, \dots, \frac{\sum_{j=1}^{n} d_{pj}y_j + g_p}{\sum_{j=1}^{n} e_{pj}y_j + h_p} \right\} \\ \text{s. t. } \sum_{j=1}^{n} a_{kj}y_j \leq b_k,\ k = 1, 2, \dots, m, \\ \quad y_j \geq 0,\ j = 1, 2, \dots, n, \end{cases}$$

where $d_{ij}, e_{ij}, b_k, a_{kj}, i = 1, 2, \ldots, p, k = 1, 2, \ldots, m, j = 1, 2, \ldots, n$, are all randomly generated in the interval $[0, 10]$; g_i and $h_i, i = 1, 2, \ldots, p$, are all randomly generated in the unit interval $[0, 1]$. The numerators g_i and denominators h_i of the linear fraction function in test Problem 1 are small constants.

Problem 2. [20]

$$\begin{cases} \min\max\left\{\dfrac{2x_1+2x_2-x_3+0.9}{x_1-x_2+x_3}, \dfrac{3x_1-x_2+x_3}{8x_1+4x_2-x_3}\right\} \\ \text{s.t. } x_1+x_2-x_3 \le 1, \\ \quad -x_1+x_2-x_3 \le -1, \\ \quad 12x_1+5x_2+12x_3 \le 34.8, \\ \quad 12x_1+12x_2+7x_3 \le 29.1, \\ \quad -6x_1+x_2+x_3 \le -4.1, \\ \quad 1.0 \le x_1 \le 1.2, \\ \quad 0.55 \le x_2 \le 0.65, \\ \quad 1.35 \le x_3 \le 1.45. \end{cases}$$

Before executing the algorithm, by calculating the upper and lower bounds of z, we can obtain the initial rectangle $Z^1 = Z = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.9292, 8.8500 \le z_2 \le 9.5500\}$.

We set the approximation error as $\epsilon = 10^{-2}$, and a brief summary of the algorithm's solution steps for this problem is as follows.

Initialization. Solving the problem LRP over Z^1 yields $LB_1 = 1.3076$ and its optimal solution

$$(y^1, z^1, r^1) = (1.0167, 0.5500, 1.4500, 1.9167, 8.8833, 1.3076).$$

Let $F^1 = \{(y^1, z^1, r^1)\}$ and $\Omega_1 = \{Z^1\}$.

According to

$$UB_1 = \max\left\{\frac{\sum_{j=1}^n e_{1j}y_j^1 + f_1}{\sum_{j=1}^n c_{1j}y_j^1 + h_1}, \frac{\sum_{j=1}^n e_{2j}y_j^1 + f_2}{\sum_{j=1}^n c_{2j}y_j^1 + h_2}, \ldots, \frac{\sum_{j=1}^n e_{pj}y_j^1 + f_p}{\sum_{j=1}^n c_{pj}y_j^1 + h_p}\right\}.$$

Following this, the upper bound of the currently known optimal value can be found: $UB_1 = 1.3478$. Since $UB_1 - LB_1 > \epsilon$, the algorithm continues with the following iterations.

Iteration 1. Subdivide Z^1 into two sub-rectangles, compress the range of each sub-rectangle, and denote the remaining two sub-rectangles as follows:

$$Z^{1,1} = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.9292, 8.8500 \le z_2 \le 9.2000\}$$

and

$$Z^{1,2} = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.9292, 9.2000 \le z_2 \le 9.5500\}.$$

Solving the problem LRP over $Z^{1,1}$ yields $LB_{Z^{1,1}} = 1.3076$ and its optimal solution

$$(y^{Z^{1,1}}, z^{Z^{1,1}}, r^{Z^{1,1}}) = (1.0167, 0.5500, 1.4500, 1.9167, 8.8833, 1.3076),$$

and $UB_{Z^{1,1}} = 1.3478$.

Solving the problem *LRP* over $Z^{1,2}$ yields $LB_{Z^{1,2}} = 1.3657$ and its optimal solution

$$(y^{Z^{1,2}}, z^{Z^{1,2}}, r^{Z^{1,2}}) = (1.0515, 0.5500, 1.4118, 1.9132, 9.2000, 1.3657),$$

and $UB_{Z^{1,2}} = 1.3478$.

Let $\Omega_2 = \{Z^{1,1}, Z^{1,2}\}$, $F^2 = F^1 \cup \{(y^{Z^{1,1}}, z^{Z^{1,1}}, r^{Z^{1,1}}), (y^{Z^{1,2}}, z^{Z^{1,2}}, r^{Z^{1,2}})\}$ and the upper bound $UB_2 = \min\{1.3478, 1.3478, 1.3478\} = 1.3478$. The currently best feasible solution $(y^2, z^2, r^2) = (y^{Z^{1,1}}, z^{Z^{1,1}}, r^{Z^{1,1}})$, and the lower bound $LB_2 = \min\{LB(Z)|Z \in \Omega_2\} = \min\{LB_{Z^{1,1}}, LB_{Z^{1,2}}\} = \min\{1.3076, 1.3657\} = 1.3076$.

Since $LB_2 = LB_{Z^{1,1}}$, let $Z^2 = Z^{1,1}$. Since $UB_2 - LB_2 > \epsilon$, continue to iteration 2.

Iteration 2. Consistent with the above, subdivide Z^2 into two sub-rectangles, compress the range of each sub-rectangle, and denote the remaining two sub-rectangles as follows:

$$Z^{2,1} = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.9292, 8.8500 \le z_2 \le 9.0250\}$$

and

$$Z^{2,2} = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.9292, 9.0250 \le z_2 \le 9.2000\}.$$

Solving the problem *LRP* over $Z^{2,1}$ yields $LB_{Z^{2,1}} = 1.3076$ and its optimal solution

$$(y^{Z^{2,1}}, z^{Z^{2,1}}, r^{Z^{2,1}}) = (1.0167, 0.5500, 1.4500, 1.9167, 8.8833, 1.3076),$$

and $UB_{Z^{2,1}} = 1.3478$.

Solving the problem *LRP* over $Z^{2,2}$ yields $LB_{Z^{1,2}} = 1.3293$ and its optimal solution

$$(y^{Z^{2,2}}, z^{Z^{2,2}}, r^{Z^{2,2}}) = (1.0335, 0.5500, 1.4426, 1.9261, 9.0250, 1.3293),$$

and $UB_{Z^{2,2}} = 1.3625$.

Let $\Omega_3 = \{Z^{1,1}, Z^{2,1}, Z^{2,2}\}$, $F^3 = F^2 \cup \{(y^{Z^{2,1}}, z^{Z^{2,1}}, r^{Z^{2,1}}), (y^{Z^{2,2}}, z^{Z^{2,2}}, r^{Z^{2,2}})\}$ and the upper bound $UB_3 = \min\{1.3478, 1.3478, 1.3625\} = 1.3478$. The currently best feasible solution $(y^3, z^3, r^3) = (y^{Z^{2,1}}, z^{Z^{2,1}}, r^{Z^{2,1}})$, and the lower bound $LB_3 = \min\{LB(Z)|Z \in \Omega_3\} = \min\{LB_{Z^{2,1}}, LB_{Z^{2,2}}\} = \min\{1.3076, 1.3293\} = 1.3076$.

Since $LB_3 = LB_{Z^{2,1}}$, let $Z^3 = Z^{2,1}$. Since $UB_3 - LB_3 > \epsilon$, continue to iteration 3.

Iteration 3. During this iteration, subdivide Z^3 into two sub-rectangles, compress the range of each sub-rectangle, and denote the remaining two sub-rectangles as follows:

$$Z^{3,1} = \{z \in \mathbb{R}^2 \mid 1.7315 \le z_1 \le 1.8333, 8.8500 \le z_2 \le 9.0250\}$$

and

$$Z^{3,2} = \{z \in \mathbb{R}^2 \mid 1.8333 \le z_1 \le 1.9292, 9.0250 \le z_2 \le 9.2000\}.$$

Solving the problem *LRP* over $Z^{3,1}$ yields $LB_{Z^{3,1}} = 1.4207$ and its optimal solution

$$(y^{Z^{3,1}}, z^{Z^{3,1}}, r^{Z^{3,1}}) = (1.0048, 0.5500, 1.3786, 1.8333, 8.8595, 1.4207),$$

and $UB_{Z^{3,1}} = 1.3478$.

Solving the problem *LRP* over $Z^{3,2}$ yields $LB_{Z^{3,2}} = 1.3242$ and its optimal solution

$$(y^{Z^{3,2}}, z^{Z^{3,2}}, r^{Z^{3,2}}) = (1.0167, 0.5500, 1.4500, 1.9167, 8.8833, 1.3242),$$

and $UB_{Z^{3,2}} = 1.3478$.

Let $\Omega_4 = \{Z^{3,1}, Z^{3,2}, Z^{2,1}, Z^{2,2}\}$, $F^4 = F^3 \cup \{(y^{Z^{3,1}}, z^{Z^{3,1}}, r^{Z^{3,1}}), (y^{Z^{3,2}}, z^{Z^{3,2}}, r^{Z^{3,2}})\}$, and the upper bound $UB_3 = \min\{1.3625, 1.3478, 1.3478\} = 1.3478$. The currently best feasible solution $(y^4, z^4, r^4) = (y^{Z^{3,2}}, z^{Z^{3,2}}, r^{Z^{3,2}})$, and the lower bound $LB_4 = \min\{LB(Z)|Z \in \Omega_4\} = \min\{LB_{Z^{3,1}}, LB_{Z^{3,2}}\} = \min\{1.4207, 1.3242\} = 1.3242$.

Since $LB_4 = LB_{Z^{3,2}}$, let $Z^4 = Z^{3,2}$. Since $UB_4 - LB_4 > \epsilon$, continue to iteration 4.

Repeat the above iterative process, and the algorithm stops when $UB - LB \leq \epsilon$ is satisfied. The optimal solution $(y_1, y_2, y_3) = (1.0167, 0.5500, 1.4500)$ and optimal value 1.34783 of the problem can be obtained after the algorithm executes 14 iterations.

5. Conclusions

We study the GAFOP. By exploiting equivalent conversion and a new linearizing technique, the initial GAFOP is able to be converted into a series of LRPs. By integrating the outer space branching search method and the LRP, we put forward an efficient global algorithm for the GAFOP. In contrast to the known existing algorithms, our algorithm has the following computational superiority: (i) The branching search occurs in $\mathbb{R}^p$ outer space, which provides the possibility of mitigating the required computational efforts of the algorithm. (ii) Numerical results demonstrate that our algorithm has superior efficiency compared to the known existing algorithms. Future work is to give a further improvement of our algorithm and extend our method to deal with the general nonlinear fractional optimization problem.

Acknowledgments

This paper is supported by the National Natural Science Foundation of China (11871196; 12071133; 12071112), the China Postdoctoral Science Foundation (2017M622340), the Key Scientific and Technological Research Projects in Henan Province (202102210147, 192102210114), the Science and Technology Climbing Program of the Henan Institute of Science and Technology (2018JY01), Henan Institute of Science and Technology Postdoctoral Science Foundation.

Conflict of interest

The authors declare that they have no conflicts of interest.

References

1. C. D. Maranas, I. P. Androulakis, C. A. Floudas, A. J. Bergerb, J. M. Mulvey, Solving long-term financial planning problems via global optimization, *J. Econ. Dyn. Contr.*, **21** (1997), 1405–1425. https://doi.org/10.1016/S0165-1889(97)00032-8
2. H. W. Jiao, J. Q. Ma, P. P. Shen, Y. J. Qiu, Effective algorithm and computational complexity for solving sum of linear ratios problem, *J. Ind. Manag. Optim.*, 2022. http://dx.doi.org/10.3934/jimo.2022135

3. H. W. Jiao, S. Y. Liu, An efficient algorithm for quadratic Sum-of-Ratios fractional programs problem, *Numer. Func. Anal. Opt.*, **38** (2017), 1426–1445. https://doi.org/10.1080/01630563.2017.1327869

4. J. E. Falk, S. W. Palocsay, *Optimizing the sum of linear fractional functions, recent advances in global optimization*, New Jersey: Princeton University Press, 1992.

5. H. W. Jiao, J. Q. Ma, An efficient algorithm and complexity result for solving the sum of general ratios problem, *Chaos Soliton. Fract.*, **164** (2022), 112701. https://doi.org/10.1016/j.chaos.2022.112701

6. C. Bajona-Xandri, J. E. Martinez-Legaz, Lower subdifferentiability in minimax fractional programming, *Optimization*, **45** (1999), 1–12. https://doi.org/10.1080/02331939908844423

7. F. Ding, Two-stage least squares based iterative estimation algorithm for CARARMA system modeling, *Appl. Math.Model.*, **37** (2013), 4798–4808. https://doi.org/10.1016/j.apm.2012.10.014

8. F. Ding, Decomposition based fast least squares algorithm for output error systems, *Signal Process.*, **93** (2013), 1235–1242. https://doi.org/10.1016/j.sigpro.2012.12.013

9. H. W. Jiao, S. Y. Liu, Range division and compression algorithm for quadratically constrained sum of quadratic ratios, *Comp. Appl. Math.*, **36** (2017), 225–247. https://doi.org/10.1007/s40314-015-0224-5

10. H. W. Jiao, S. Y. Liu, W. J. Wang, Solving generalized polynomial problem by using new affine relaxed technique, *Int. J. Comput. Math.*, **99** (2022), 309–331. https://doi.org/10.1080/00207160.2021.1909727

11. H. W. Jiao, W. J. Wang, R. J. Chen, Y. L. Shang, J. B. Yin, An efficient outer space algorithm for generalized linear multiplicative programming problem, *IEEE Access*, **8** (2020), 80629–80637. https://doi.org/10.1109/ACCESS.2020.2990677

12. H. W. Jiao, Y. L. Shang, R. J. Chen, A potential practical algorithm for minimizing the sum of affine fractional functions, *Optimization*, 2022. https://doi.org/10.1080/02331934.2022.2032051

13. Y. Y. Ding, Y. H. Xiao, J. W. Li, A class of conjugate gradient methods for convex constrained monotone equations, *Optimization*, **66** (2017), 2309–2328. https://doi.org/10.1080/02331934.2017.1372438

14. Y. H. Xiao, L. Chen, D. H. Li, A generalized alternating direction method of multipliers with semi-proximal terms for convex composite conic programming, *Math. Program. Comput.*, **10** (2018), 533–555. https://doi.org/10.1007/s12532-018-0134-9

15. H. W. Jiao, W. J. Wang, J. B. Yin, Y. L. Shang, Image space branch-reduction-bound algorithm for globally minimizing a class of multiplicative problems, *Rairo-Oper. Res.*, **56** (2022), 1533–1552. https://doi.org/10.1051/ro/2022061

16. H. W. Jiao, Y. L. Shang, Two-level linear relaxation method for generalized linear fractional programming, *J. Oper. Res. Soc.*, 2022. https://doi.org/10.1007/s40305-021-00375-4

17. H. W. Jiao, W. J. Wang, Y. L. Shang, Outer space branch-reduction-bound algorithm for solving generalized affine multiplicative problem, *J. Comput. Appl. Math.*, **419** (2023), 114784. https://doi.org/10.1016/j.cam.2022.114784

18. A. I. Barros, J. B. G. Frenk, Generalized fractional programming and cutting plane algorithms, *J. Optim. Theory Appl.*, **87** (1995), 103–120. https://doi.org/10.1007/BF02192043
19. Q. G. Feng, H. P. Mao, H. W. Jiao, A feasible method for a class of mathematical problems in manufacturing system, *Key Eng. Mater.*, **460** (2011), 806–809. https://doi.org/10.4028/www.scientific.net/KEM.460-461.806
20. Q. G. Feng, H. W. Jiao, H. P. Mao, Y. Q. Chen, A deterministic algorithm for min-max and max-min linear fractional programming problems, *Int. J. Comput. Int. Sys.*, **4** (2011), 134–141. http://dx.doi.org/10.1080/18756891.2011.9727770
21. R.W. Freund, F. Jarre, An interior-point method for fractional programs with convex constraints, *Math. Program.*, **67** (1994), 407–440. https://doi.org/10.1007/BF01582229
22. Y. Benadada, J. A. Fedand, Partial linearization for generalized fractional programming, *Zeitschrift Für Oper. Res.*, **32** (1988), 101–106. https://doi.org/10.1007/BF01919185
23. N. T. H. Phuong, H. Tuy, A unified monotonic approach to generalized linear fractional programming, *J. Global Optim.*, **26** (2003), 229–259. https://doi.org/10.1023/A:1023274721632
24. A. Roubi, Method of centers for generalized fractional programming, *J. Optim. Theory Appl.*, **107** (2000), 123–143. https://doi.org/10.1023/A:1004660917684
25. M. Gugat, Prox-regularization methods for generalized fractional programming, *J. Optim. Theory Appl.*, **99** (1998), 691–722. https://doi.org/10.1023/A:1021759318653
26. A. Ghazi, A. Roubi, A DC approach for minimax fractional optimization programs with ratios of convex functions, *Optim. Methods Softw.*, 2020. https://doi.org/10.1080/10556788.2020.1818234
27. H. Boualam, A. Roubi, Dual algorithms based on the proximal bundle method for solving convex minimax fractional programs, *J. Ind. Manag. Optim.*, **15** (2019), 1897–1920. https://doi.org/10.3934/jimo.2018128
28. H. W. Jiao, J. Q. Ma, Y. Shang, Image space branch-and-bound algorithm for globally solving minimax linear fractional programming problem, *Pac. J. Optim.*, **18** (2022), 195–212.
29. M. E. Haffari, A. Roubi, Prox-dual regularization algorithm for generalized fractional programs, *J. Ind. Manag. Optim.*, **13** (2017). https://doi.org/10.3934/jimo.2017028
30. H. W. Jiao, B. B. Li, Solving min-max linear fractional programs based on image space branch-and-bound scheme, *Chaos Soliton. Fract.*, **164** (2022), 112682. https://doi.org/10.1016/j.chaos.2022.112682
31. I. Ahmad, Z. Husain, Duality in nondifferentiable minimax fractional programming with generalized convexity, *Appl. Math. Comput.*, **176** (2006), 545–551. https://doi.org/10.1016/j.amc.2005.10.002
32. W. E. Schmitendorf, Necessary conditions and sufficient conditions for static minimax problems, *J. Math. Anal. Appl.*, **57** (1977), 683–693. https://doi.org/10.1016/0022-247X(77)90255-4
33. S. Tanimoto, Duality for a class of nondifferentiable mathematical programming problems, *J. Math. Anal. Appl.*, **79** (1981), 286–294. https://doi.org/10.1016/0022-247X(81)90025-1
34. S. R. Yadav, R. N. Mukherjee, Duality for fractional minimax programming problems, *ANZIAM J.*, **31** (1990), 484–492. https://doi.org/10.1017/S0334270000006809

35. V. Jeyakumar, G. Y. Li, S. Srisatkunarajah, Strong duality for robust minimax fractional programming problems, *Eur. J. Oper. Res.*, **228** (2013), 331–336. https://doi.org/10.1016/j.ejor.2013.02.015

36. H. C. Lai, J. C. Liu, K. Tanaka, Duality without a constraint qualification for minimax fractional programming, *J. Math. Anal. Appl.*, **101** (1999), 109–125. https://doi.org/10.1023/A:1021771011210

37. I. M. Stancu-Minasian, A ninth bibliography of fractional programming, *Optimization*, **68** (2019) 2125–2169. https://doi.org/10.1080/02331934.2019.1632250

38. I. M. Stancu-Minasian, A eighth bibliography of fractional programming, *Optimization*, **66** (2017) 439–470. https://doi.org/10.1080/02331934.2016.1276179

39. C. F. Wang, Y. Jiang, P. P. Shen, A new branch-and-bound algorithm for solving minimax linear fractional programming, *J. Math.*, **38** (2018), 113–123.

40. H. W. Jiao, S. Y. Liu, A new linearization technique for minimax linear fractional programming, *Int. J. Comput. Math.*, **91** (2014), 1730–1743. https://doi.org/10.1080/00207160.2013.860449