AIMS Mathematics

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

AIMS Mathematics, 8(4): 9247–9264.
DOI: 10.3934/math.2023464
Received: 17 October 2022
Revised: 18 January 2023
Accepted: 01 February 2023
Published: 14 February 2023

*Research article*

# A novel algorithm for solving sum of several affine fractional functions

**Hongwu Li**[1,2,*], **Yuling Feng**[3], **Hongwei Jiao**[3,4,*] **and Youlin Shang**[4]

[1] School of Science, Beijing University of Technology, Beijing 100124, China
[2] School of Mathematics and Statistics, Nanyang Normal University, Nanyang 473061, China
[3] School of Mathematical Sciences, Henan Institute of Science and Technology, Xinxiang 453003, China
[4] Postdoctoral Research Station of Control Science and Engineering, Henan University of Science and Technology, Luoyang 471023, China

* **Correspondence:** Email: jiaohongwei@hist.edu.cn, lihongwu2018@163.com.

**Abstract:** By using the outer space branch-and-reduction scheme, we present a novel algorithm for globally optimizing the sum of several affine fractional functions problem (SAFFP) over a nonempty compact set. For providing the reliable lower bounds in the searching process of iterations, we devise a novel linearizing method to establish the affine relaxation problem (ARP) for the SAFFP. Thus, the main computational work involves solving a series of ARP. For improving the convergence speed of the algorithm, an outer space region reduction technique is proposed by utilizing the objective function characteristics. Through computational complexity analysis, we estimate the algorithmic maximum iteration times. Finally, numerical comparison results are given to reveal the algorithmic computational advantages.

**Keywords:** sum of several affine fractional functions; Branch-and-reduction; affine relaxation problem; outer space region reduction technique; computational complexity
**Mathematics Subject Classification:** 90C26, 90C32

## 1. Introduction

We investigate globally optimizing the sum of several affine fractional functions problem defined by

$$\text{(SAFFP)}: \begin{cases} v = \min \quad F(z) = \sum_{m=1}^{q} \dfrac{\sum_{j=1}^{n} h_{mj} z_j + d_m}{\sum_{j=1}^{n} g_{mj} z_j + f_m} \\ \text{s.t.} \quad z \in \Lambda = \{ z \in R^n \mid Az \leq b, z \geq 0 \}, \end{cases}$$

where $h_{mj}, d_m, g_{mj}, f_m \in R, m = 1, 2, \ldots, q, j = 1, 2, \ldots, n, A \in R^{\mu\times n}, b \in R^m$, $\Lambda$ is a nonempty compact set, and $\sum_{j=1}^{n} g_{mj}z_j + f_m \neq 0$ for any $z \in \Lambda$.

From 1990s, the SAFFP has attracted a lot of attention attentions of many practitioners and researchers. The SAFFP is widely used in computer vision, investment and portfolio optimization, optimal strategy in supply chain, risk-averse and so on, see Refs. [1–6]. Besides, the SAFFP is non-convex optimization problem, which usually contains many locally minimum solutions that are not globally minimum.

In the past 20 years, many scholars have presented a large number of different algorithms to globally solve the SAFFP. Generally, these algorithms may be classified as below, such as, parametric simplex algorithms [7], outer approximation algorithms [8], image space analysis methods [9], monotonic optimization algorithms [10], branch-and-bound algorithms [11–28], polynomial-time approximation algorithms [29, 30], and so on. In addition, for an excellent review, we can refer to Schaible and Shi [31].

Additionally, there are some theoretical progress on the generalized SAFFP, for example, Saxena and Jain [32] presented an dual problem for the linear fractional programming problem under fuzzy environment. Based on the membership function of the target multiplied by the appropriate weights, Borza and Rambely [33] proposed a set of linear inequalities. Goli and Nasseri [34] investigate for linear programming problems with intuitionistic fuzzy variables and proposed its pairwise results with a generalization of the pairwise simplex method.

In this article, by using the outer space branch-and-reduction scheme, we propose a global algorithm to effectively solve the SAFFP. We first convert the SAFFP into an equivalent bilinear optimization problem (EBOP). Next, by utilizing new linearizing method, we establish the ARP of the EBOP. To improve the running speed of the outer space searching algorithm, an outer space region reduction method is proposed. By iteratively subdividing the initial outer space region and computing a sequence of LRP, the presented algorithm is globally convergent to the minimum point of the SAFFP. By analysing the algorithmic complexity, we give an estimation for the maximum number of iterations of the proposed algorithm in this paper. Finally, numerical comparisons are reported to reveal the computational superiority and higher efficiency of the algorithm.

The rests of this article are organized as below. In Section 2, the EBOP and its ARP of the SAFFP are derived. In Section 3, based the outer space branch-and-reduction scheme, we construct a global algorithm for the SAFFP, prove and analyse the algorithmic convergence and complexity, and estimate the algorithmic maximum iteration times. Numerical examples and their computational comparisons are reported in Section 4. Finally, we give some conclusions in Section 5.

## 2. Equivalence problem and its affine relaxation

In this section, we firstly equivalently convert the SAFFP into the EBOP. Since the denominator $\sum_{j=1}^{n} g_{mj}z_j + f_m \neq 0$ for any $z \in \Lambda$, by the continuity of the function $\sum_{j=1}^{n} g_{mj}z_j + f_m$, it follows that $\sum_{j=1}^{n} g_{mj}z_j + f_m > 0$ or $\sum_{j=1}^{n} g_{mj}z_j + f_m < 0$. Since $\frac{\sum_{j=1}^{n} h_{mj}z_j+d_m}{\sum_{j=1}^{n} g_{mj}z_j+f_m} = \frac{-\left(\sum_{j=1}^{n} h_{mj}z_j+d_m\right)}{-\left(\sum_{j=1}^{n} g_{mj}z_j+f_m\right)}$, without losing generality, we can always assume $\sum_{j=1}^{n} g_{mj}z_j + f_m > 0$.

Without losing generality, let $s_m = \frac{1}{\sum_{j=1}^{n} g_{mj}z_j+f_m}$, and define

$$\underline{s}_m^0 = \frac{1}{\max_{z\in\Lambda} \sum_{j=1}^{n} g_{mj}z_j + f_m}, \quad \overline{s}_m^0 = \frac{1}{\min_{z\in\Lambda} \sum_{j=1}^{n} g_{mj}z_j + f_m}, \quad m = 1, 2, \ldots, q,$$

and construct the initial outer space rectangle

$$S^0 = \left\{ s \in R^q \mid \underline{s}_m^0 \le s_m \le \overline{s}_m^0, m = 1, 2, \ldots, q \right\},$$

then the SAFFP may be changed to the following equivalent bilinear optimization problem:

$$\text{EBOP}\left(S^0\right) : \begin{cases} v(S^0) = \max \quad \Psi_0(z, s) = \sum_{m=1}^{q} s_m \left( \sum_{j=1}^{n} h_{mj}z_j + d_m \right) \\ \text{s.t.} \quad \Psi_m(z, s) = s_m \left( \sum_{j=1}^{n} g_{mj}z_j + f_m \right) = 1, \quad m = 1, 2, \ldots, q, \\ \qquad z \in \Lambda, \ s \in S^0. \end{cases}$$

Obviously, $(z^*, s^*)$ is a globally optimum solution to the EBOP$\left(S^0\right)$ if and only if $z^*$ is a globally optimum solution to the SAFFP, where $s_m^* = \frac{1}{\sum_{j=1}^{n} g_{mj}z_j^*+f_m}, m = 1, 2, \ldots, q$.

Therefore, we may consider globally solving the EBOP$\left(S^0\right)$ instead of globally solving the SAFFP. Next, we will give the detailed process for constructing the ARP of the EBOP$\left(S^0\right)$ as below.

For the convenience of expression, for each $m = 1, 2, \ldots, q$, we let

$$H_m^+ = \left\{ j \mid h_{mj} > 0, j = 1, 2, \ldots, n \right\}, H_m^- = \left\{ j \mid h_{mj} < 0, j = 1, 2, \ldots, n \right\},$$
$$G_m^+ = \left\{ j \mid g_{mj} > 0, j = 1, 2, \ldots, n \right\}, G_m^- = \left\{ j \mid g_{mj} < 0, j = 1, 2, \ldots, n \right\}.$$

Firstly, consider the objective function $\Psi_0(z, s)$, we can follow that

$$\begin{aligned} \Psi_0(z, s) &= \sum_{m=1}^{q} s_m (\sum_{j=1}^{n} h_{mj}z_j + d_m) \\ &\ge \sum_{m=1}^{q} (\sum_{j\in H_m^+} h_{mj}\underline{s}_m z_j + \sum_{j\in H_m^-} h_{mj}\overline{s}_m z_j) + \sum_{m=1}^{q} d_m s_m \\ &= \Psi_0^L(z, s). \end{aligned} \tag{1}$$

Secondly, consider the constrained function $\Psi_m(z, s), m = 1, 2, \ldots, q$, we can follow that

$$\Psi_m(z, s) = s_m(\sum_{j=1}^{n} g_{mj}z_j + f_m) \ge \sum_{j\in G_m^+} g_{mj}\underline{s}_m z_j + \sum_{j\in G_m^-} g_{mj}\overline{s}_m z_j + f_m s_m = \Psi_m^L(z, s),$$
$$\Psi_m(z, s) = s_m(\sum_{j=1}^{n} g_{mj}z_j + f_m) \le \sum_{j\in G_m^+} g_{mj}\overline{s}_m z_j + \sum_{j\in G_m^-} g_{mj}\underline{s}_m z_j + f_m s_m = \Psi_m^U(z, s).$$

Therefore, for any $S=\left\{s\in R^q \mid \underline{s}_m\le s_m\le \overline{s}_m, m=1,2,\dots,q\right\}\subseteq S^0$, we can construct the ARP($S$) of the problem (EBOP ($S$)) as below:

$$\begin{cases} LB(S)=\min\ \Psi_0^L(z,s)=\sum\limits_{m=1}^{q}(\sum\limits_{j\in H_m^+}h_{mj}\underline{s}_m z_j+\sum\limits_{j\in H_m^-}h_{mj}\overline{s}_m z_j)+\sum\limits_{m=1}^{q}d_m s_m\\ s.t.\ \ \Psi_m^L(z,s)=\sum\limits_{j\in G_m^+}g_{mj}\underline{s}_m z_j+\sum\limits_{j\in G_m^-}g_{mj}\overline{s}_m z_j+f_m s_m\le 1,\ \ m=1,2,\dots,q,\\ \quad\ \Psi_m^U(z,s)=\sum\limits_{j\in G_m^+}g_{mj}\overline{s}_m z_j+\sum\limits_{j\in G_m^-}g_{mj}\underline{s}_m z_j+f_m s_m\ge 1,\ \ m=1,2,\dots,q,\\ \quad\ z\in\Lambda,\ \ s\in S. \end{cases}$$

**Theorem 1.** For each $m\in\{1,2,\dots,q\}$, we have

$$|\Psi_0(z,s)-\Psi_0^L(z,s)|\to 0\ \text{ as }\ \|\overline{s}_m-\underline{s}_m\|\to 0.$$

*Proof.* From the above conclusion, we have that

$$\begin{aligned}\left|\Psi_0(z,s)-\Psi_0^L(z,s)\right| &=\left|\sum_{m=1}^{q}s_m\left(\sum_{j=1}^{n}h_{mj}z_j+d_m\right)-\sum_{m=1}^{q}(\sum_{j\in H_m^+}h_{mj}\underline{s}_m z_j+\sum_{j\in H_m^-}h_{mj}\overline{s}_m z_j)-\sum_{m=1}^{q}d_m s_m\right|\\ &=\left|\sum_{m=1}^{q}\left[\sum_{j\in H_m^+}h_{mj}z_j(s_m-\underline{s}_m)+\sum_{j\in H_m^-}h_{mj}z_j(s_m-\overline{s}_m)\right]\right|\\ &\le(\overline{s}_m-\underline{s}_m)\times\left|\sum_{m=1}^{q}\sum_{j=1}^{n}h_{mj}z_j\right|.\end{aligned}$$

When $\|\overline{s}_m-\underline{s}_m\|\to 0, |\Psi_0(z,s)-\Psi_0^L(z,s)|\to 0$, the proof of Theorem is finished.

**Remark 1.** Denotes $v$[P] as the globally minimum value of the problem (P), based on the previous discussions, then: for any $S\subseteq S^0$, the global minimum values for the ARP($S$) and EBOP($S$) satisfy $v$[ARP($S$)] $\le v$[EBOP($S$)].

**Remark 2.** Obviously, for any $\hat{S}\subseteq S\subseteq S^0$, it follows that $LB(\hat{S})\ge LB(S)$.

## 3. Global algorithm, convergence, and its complexity

In this part, for globally solving the SAFFP, combining the previous affine relaxation problem, we design an outer space region reduction operation, and based on the branch-and-bound searching framework, a global algorithm is designed.

### 3.1. Outer space region reduction operation

To enhance convergence speed of the presented algorithm, we construct a new outer space region reduction operation as follows.

For any investigated rectangles

$$S=\left\{s\in R^q|\underline{s}_m\le s_m\le\overline{s}_m, m=1,2,\dots,q\right\}\subseteq S^0,$$

denote by

$$RLB=\sum_{m=1}^{q}\min\{l_m^0\underline{s}_m, l_m^0\overline{s}_m\},$$

where

$$l_m^0 = \min_{z\in\Lambda} \sum_{j=1}^{n} h_{mj} z_j + d_m,\ m = 1, 2, \ldots, q.$$

**Theorem 2.** Denote $UB_k$ as the known best upper bound at the $k_{th}$ iteration, for any investigated rectangle $S \subseteq S^0$, we get the following several conclusions:

**(i)** If $RLB > UB_k$, then the rectangle $S$ contains no globally optimum point to the EBOP($S^0$).

**(ii)** If $RLB \leq UB_k$, then, for each $\sigma \in \{1, 2, \ldots, q\}$, the following two cases hold:

(a) If $l_\sigma^0 > 0$, then the rectangle $\hat{S}^1$ contains no globally optimum point to the EBOP($S^0$), where

$$\hat{S}^1 = \{s \in R^q | \underline{s}_m \leq s_m \leq \overline{s}_m, m = 1, \ldots, q, m \neq \sigma; \rho_\sigma^1 < s_\sigma \leq \overline{s}_\sigma\}$$

with

$$\rho_\sigma^1 = \frac{UB_k - RLB + l_\sigma^0 L_\sigma}{l_\sigma^0}.$$

(b) If $l_\sigma^0 < 0$, then the rectangle $\hat{S}^2$ contains no globally optimum point to the EBOP($S^0$), where

$$\hat{S}^2 = \{s \in R^q | \underline{s}_m \leq s_m \leq \overline{s}_m, m = 1, \ldots, q, m \neq \sigma; \underline{s}_\sigma \leq s_\sigma < \rho_\sigma^2\}$$

with

$$\rho_\sigma^2 = \frac{UB_k - RLB + l_\sigma^0 \overline{s}_\sigma}{l_\sigma^0}.$$

*Proof.* (i) If $RLB > UB_k$, then:

$$\begin{aligned}
\min_{z\in\Lambda, s\in S} \Psi_0(z, s) &= \min_{z\in\Lambda, s\in S} \sum_{m=1}^{q} s_m (\sum_{j=1}^{n} h_{mj} z_j + d_m) \\
&= \sum_{m=1}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} \\
&= RLB > UB_k.
\end{aligned}$$

Thus, the rectangle $S$ contains no globally optimum point to the EBOP($S^0$).

(ii) If $RLB \leq UB_k$, for each $\sigma \in \{1, 2, \ldots, q\}$, then, we firstly prove the conclusion.

(a) If $l_\sigma^0 > 0$, $\sigma \in \{1, 2, \ldots, q\}$, then, for ans $z \in \Lambda$ and $s \in \hat{S}^1$, we have

$$\sum_{j=1}^{n} h_{mj} z_j + d_m \geq l_m^0$$

and

$$\underline{s}_m \leq s_m \leq \overline{s}_m, m = 1, 2, \ldots, q, m \neq \sigma;\ l_\sigma^0 > 0, \rho_\sigma^1 < s_\sigma \leq \overline{s}_\sigma.$$

Thus, $\min\limits_{z\in\Lambda, s\in S^1} \Psi_0(z,s)$ satisfies the following inequalities:

$$\begin{aligned}
\min_{z\in\Lambda, s\in S^1} \Psi_0(z,s) &= \min_{z\in\Lambda, s\in S^1} \sum_{m=1}^{q} s_m(\sum_{j=1}^{n} h_{mj}z_j + d_m) \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + \min_{z\in\Lambda, s\in S^1} s_\sigma(\sum_{j=1}^{n} h_{\sigma j}z_j + d_\sigma) \\
&> \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + l_\sigma \rho_\sigma^1 \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + l_\sigma^0 \times \frac{UB_k - RLB + l_\sigma^0 \underline{s}_\sigma}{l_\sigma^0} \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + UB_k - RLB + l_\sigma^0 \underline{s}_\sigma \\
&= RLB + UB_k - RLB = UB_k.
\end{aligned}$$

Therefore, $\hat{S}^1$ contains no globally optimum point to the EBOP($S^0$).

(b) Similarly, if $l_\sigma^0 < 0$, $\sigma \in \{1, 2, \ldots, q\}$, then, for $\forall z \in \Lambda$, $s \in \hat{S}^2$, we have

$$\sum_{j=1}^{n} h_{mj}z_j + d_m \geq l_m^0$$

and

$$\underline{s}_m \leq s_m \leq \overline{s}_m, m = 1, 2, \ldots, q, m \neq \sigma;\ l_\sigma^0 < 0, \underline{s}_\sigma < s_\sigma \leq \rho_\sigma^2.$$

Thus, $\min\limits_{z\in\Lambda, s\in S^2} \Psi_0(z,s)$ satisfies the following inequalities:

$$\begin{aligned}
\min_{z\in\Lambda, s\in S^2} \Psi_0(z,s) &= \min_{z\in\Lambda, s\in S^2} \sum_{m=1}^{q} s_m(\sum_{j=1}^{n} h_{mj}z_j + d_m) \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + \min_{z\in\Lambda, s\in S^2} s_\sigma(\sum_{j=1}^{n} h_{\sigma j}z_j + d_\sigma) \\
&> \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + l_\sigma \rho_\sigma^2 \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + l_\sigma^0 \times \frac{UB_k - RLB + l_\sigma^0 \overline{s}_\sigma}{l_\sigma^0} \\
&= \sum_{m=1, m\neq\sigma}^{q} \min\{l_m^0 \underline{s}_m, l_m^0 \overline{s}_m\} + UB_k - RLB + l_\sigma^0 \overline{s}_\sigma \\
&= RLB + UB_k - RLB = UB_k.
\end{aligned}$$

Therefore, $\hat{S}^2$ contains no globally optimum point to the EBOP($S^0$). □

From Theorem 2, the constructed outer space region reduction technique gives a probability to prune the whole rectangle $S$ or a portion of it which contains no global optimum point of the EBOP($S^0$). Next, we will propose a novel algorithm based on the outer space branch-and-reduction scheme.

### 3.2. Novel algorithm base on the outer space branch-and-reduction scheme

By combining the above affine relaxation problem and outer space region reduction technique, a novel algorithm to globally solve the SAFFP can be described as below:

**Step 0.** Letting

$$S^0 = \left\{ s \in R^q \mid \underline{s}_m^0 \leq s_m \leq \overline{s}_m^0, m = 1, \ldots, q \right\},$$

and setting $\epsilon \in [0, 1)$, solve the ARP$\left(S^0\right)$ to achieve its optimum solution $(z^0, \hat{s}^0)$ and optimum value $LB\left(S^0\right)$, respectively. Simultaneously, let

$$LB_0 = LB\left(S^0\right), z^c = z^0, s_m^c = \frac{1}{\sum_{j=1}^{n} g_{mj} z_j^c + f_m}, m = 1, \ldots, q, UB_0 = \Psi_0 (z^c, s^c).$$

If $UB_0 - LB_0 \leq \epsilon$, then the presented algorithm will finish with obtaining the $\epsilon$-globally optimum solution $(z^c, s^c)$ to the EBOP$(S^0)$ and the $\epsilon$-globally optimum solution $z^c$ to the SAFFP.

Otherwise, set $P_0 = \left\{S^0\right\}, F = \emptyset, k = 1$, and continue to Step 1.

**Step 1.** Let $UB_k = UB_{k-1}$, by using the dichotomy method to segment the largest edge of the selected rectangle, and subdivide $S^{k-1}$ into two $q$-dimensional sub-rectangles $S^{k,1}$ and $S^{k,2}$. Let $F = F \cup \left\{S^{k-1}\right\}$.

**Step 2.** Use the proposed outer space region reduction technique to compress the range of the rectangle $S^{k,\alpha}$, where $\alpha = 1, 2$, solve the ARP$(S^{k,\alpha})$ to obtain $LB\left(S^{k,\alpha}\right)$ and its optimum solution $(z^{k,\alpha}, \hat{s}^{k,\alpha})$. Set $\eta = 0$.

**Step 3.** Let $\eta = \eta + 1$. If $\eta > 2$, then continue with Step 5. Otherwise, continue with Step 4.

**Step 4.** If $LB\left(S^{k,\eta}\right) \geq UB_k$, then let $F = F \cup \left\{S^{k,\eta}\right\}$, and continue with Step 3.

Otherwise, let

$$s_m^{k,\eta} = \frac{1}{\sum_{j=1}^{n} g_{mj} z_j^{k,\eta} + f_m}, m = 1, 2, \ldots, q,$$

renew the upper bound $UB_{k+1} = \min\left\{UB_k, \Psi_0\left(z^{k,\eta}, s^{k,\eta}\right)\right\}$.

If $UB_k < \Psi_0\left(z^{k,\eta}, s^{k,\eta}\right)$, then proceed with Step 3.

If $UB_k = \Psi_0\left(z^{k,\eta}, s^{k,\eta}\right)$, then let $z^c = z^{k,\eta}$, $(z^c, s^c) = \left(z^{k,\eta}, s^{k,\eta}\right)$,

$$F = F \bigcup \{S \in P_{k-1} \mid LB(S) \geq UB_k\},$$

and proceed with Step 5.

**Step 5.** Let

$$P_k = \left\{S \mid S \in \left(P_{k-1} \cup \left\{S^{k,1}, S^{k,2}\right\}\right), S \notin F\right\}$$

and

$$LB_k = \min\{LB(S) \mid S \in P_k\}.$$

**Step 6.** Let

$$P_{k+1} = \{S \mid UB_k - LB(S) > \epsilon, S \in P_k\}.$$

If $P_{k+1} = \emptyset$, then the presented algorithm will finish with obtaining the $\epsilon$-globally optimum solution $(z^c, s^c)$ to the EBOP$(S^0)$ and the $\epsilon$-globally optimum solution $z^c$ to the SAFFP. Otherwise, select $S^{k+1}$ satisfying that $S^{k+1} = \arg\min_{S \in P_{k+1}} LB(S)$, set $k = k + 1$, and go back Step 1.

### 3.3. Convergence analysis

In this sub-section, we will prove the convergence of the proposed algorithm by the following theorem.

**Theorem 3.** For any given $\epsilon \in [0, 1)$. We denote $z^k$ as the obtained best solution $z^c$ of the SAFFP at the $k_{th}$ iteration. If the presented algorithm finitely terminates after $k$ iterations, then we can obtain an $\epsilon$-globally optimum solution $(z^c, s^c)$ to the EBOP$(S^0)$ and an $\epsilon$-globally optimum solution $z^c$ to the SAFFP. Otherwise, the presented algorithm will generate an infinite feasible solution sequence $\{z^k\}$ with that its each gathering point is a globally optimum solution to the SAFFP.

*Proof.* Assume that the presented algorithm finitely finishes at the $k_{th}$ iteration, then: when the algorithm terminates, $(z^c, \hat{s}^c)$ can be obtained by solving the ARP$(S)$ for some $S \subseteq S^0$, and let

$$s_m^c = \frac{1}{\sum\limits_{j=1}^{n} g_{mj} z_j^c + f_m}, m = 1, 2, \ldots, q.$$

Obviously, $z^c$ and $(z^c, s^c)$ are the feasible solutions for the SAFFP and EBOP$\left(S^0\right)$, respectively. Upon termination of the presented algorithm, we have

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

From Steps 0 and 4, this implies that

$$\Psi_0\left(z^c, s^c\right) \leq LB_k + \epsilon.$$

By the bounding method, it can follow that

$$LB_k \leq v.$$

Since $(z^c, s^c)$ is feasible to the EBOP$\left(S^0\right)$, it follows that

$$v \leq \Psi_0\left(z^c, s^c\right).$$

Combine the above several inequalities, we can get that

$$v \leq \Psi_0\left(z^c, s^c\right) \leq LB_k + \epsilon \leq v + \epsilon.$$

Therefore,

$$v \leq \Psi_0\left(z^c, s^c\right) \leq v + \epsilon.$$

Since $s_m^c = \frac{1}{\sum\limits_{j=1}^{n} g_{mj} z_j + f_m}, m = 1, 2, \ldots, q$, we can follow that

$$F(z^c) = \sum_{m=1}^{q} \frac{\sum\limits_{j=1}^{n} h_{mj} z_j^c + d_m}{\sum\limits_{j=1}^{n} g_{mj} z_j^c + f_m} = \sum_{m=1}^{q} s_m^c \left(\sum_{j=1}^{n} h_{mj} z_j^c + d_m\right) = \Psi_0(z^c, s^c).$$

Combine the above several inequalities, we have that

$$v \le F(z^c) \le v + \epsilon.$$

If the presented method does not terminate in finite step, then it will create a best feasible solution sequence $\left\{\left(z^k, s^k\right)\right\}$ to the EBOP$\left(S^0\right)$.

For each $k \ge 1$, for some a rectangle $S^k \subseteq S^0$, suppose that $\left(z^k, \hat{s}^k\right)$ is obtained by solving the problem ARP$\left(S^k\right)$, and let

$$s_m^k = \frac{1}{\sum_{j=1}^{n} g_{mj} z_j^k + f_m}, m = 1, 2, \ldots, q.$$

Obviously, $\left\{\left(z^k, s^k\right)\right\}$ is a feasible solution sequence to the EBOP$\left(S^0\right)$.

Without losing generality, we assume that $\tilde{z}$ is an accumulation point of the sequence $\left\{z^k\right\}$ with that $\lim_{k\to\infty} z^k = \tilde{z}$, then, due to the fact that $z^k$ is always feasible solution to the SAFFP and $\Lambda$ is a nonempty bounded compact set, we must have $\tilde{z} \in \Lambda$.

Furthermore, when the presented algorithm is infinite, without loss of generality, for each $k \ge 1$, assume that $S^{k+1} \subseteq S^k$. For each $k \ge 1$, since the rectangles $S^k$ are generated by rectangular bisection, by Horst and Tuy [35], then there must exist some a point $\tilde{s} \in R^q$ such that

$$\lim_{k\to\infty} S^k = \bigcap_k S^k = \{\tilde{s}\}. \tag{2}$$

Let $\tilde{S} = \{\tilde{s}\}$ and

$$S^k = \left\{s \in R^q \mid \underline{s}_m^k \le s_m \le \overline{s}_m^k, m = 1, 2, \ldots, q\right\}$$

for each $k \ge 1$, since $S^{k+1} \subset S^k \subset S^0$, and from Step 4 of the algorithm and Remark 2, this indicates that $\left\{LB\left(S^k\right)\right\}$ is a nondecreasing bounded sequence satisfying that $LB\left(S^k\right) \le v$. Thus, $\lim_{k\to\infty} LB\left(S^k\right)$ exists and meets that

$$\lim_{k\to\infty} LB\left(S^k\right) \le v. \tag{3}$$

From Step 2 of the algorithm, for each $k \ge 0$, $LB\left(S^k\right)$ is the optimum value to the ARP$\left(S^k\right)$, and $(z^k, \hat{s}^k)$ is the optimal solution for this problem.

From (2), it follows that

$$\lim_{k\to\infty} \underline{s}^k = \lim_{k\to\infty} \overline{s}^k = \{\tilde{s}\} = \tilde{S}.$$

By the continuity of the function $\sum_{j=1}^{n} g_{mj} z_j + f_m$, $\lim_{k\to\infty} z^k = \tilde{z}$, and

$$\underline{s}_m^k \le \frac{1}{\sum_{j=1}^{n} g_{mj} z_j^k + f_m} \le \overline{s}_m^k,$$

it follows that

$$\tilde{s}_m = \frac{1}{\sum_{j=1}^{n} g_{mj} \tilde{z}_j + f_m}, m = 1, 2, \ldots, q.$$

This indicates that $(\tilde{z}, \tilde{s})$ is a feasible solution to the EBOP$\left(S^0\right)$). Thus,

$$\Psi_0(\tilde{z}, \tilde{s}) \geq v. \tag{4}$$

Combine (3) and (4) together, it follows that

$$\lim_{k\to\infty} LB\left(S^k\right) \leq v \leq \Psi_0(\tilde{z}, \tilde{s}).$$

Since $\hat{s}^k \in [\underline{s}^k, \overline{s}^k]$ and $\lim_{k\to\infty} \underline{s}^k = \lim_{k\to\infty} \overline{s}^k = \{\tilde{s}\}$, it follows that

$$\begin{aligned} \lim_{k\to\infty} LB(S^k) &= \lim_{k\to\infty} \sum_{m=1}^{q} (\sum_{j\in H_m^+} h_{mj} z_j \underline{s}_m^k + \sum_{j\in H_m^-} h_{mj} z_j \overline{s}_m^k) + \sum_{m=1}^{q} d_m \hat{s}_m^k \\ &= \sum_{m=1}^{q} \tilde{s}_m (\sum_{j=1}^{n} h_{mj} \tilde{z}_j + d_m) \\ &= \Psi_0(\tilde{z}, \tilde{s}). \end{aligned} \tag{5}$$

From (4), (5), and the former discussions, it can follow that

$$\lim_{k\to\infty} LB\left(S^k\right) = v = \Psi_0(\tilde{z}, \tilde{s}).$$

Hence, $(\tilde{z}, \tilde{s})$ is a globally optimum solution to the EBOP$\left(S^0\right)$. From equivalent conclusions of the EBOP$\left(S^0\right)$ and SAFFP, this indicates that $\tilde{z}$ is also a global optimal solution to the SAFFP.

For each $k \geq 1$, since $z^k$ is the best feasible solution to the SAFFP at the $k_{th}$ iteration, then the upper bound satisfies that

$$UB_k = F\left(z^k\right).$$

By the function continuity of $F(z)$, we can follow that

$$\lim_{k\to\infty} F\left(z^k\right) = F\left(\lim_{k\to\infty} z^k\right) = F(\tilde{z}).$$

Since $\tilde{z}$ is a globally optimum solution to the SAFFP, we have $F(\tilde{z}) = v$. Thus, we have that

$$\lim_{k\to\infty} UB_k = \lim_{k\to\infty} F\left(z^k\right) = F(\tilde{z}) = v = \lim_{k\to\infty} LB_k,$$

and the proof of the theorem is completed. □

By the above theorem, the algorithm is convergent, then, we will analyze the computational efficiency of the algorithm in the worst case.

### 3.4. Complexity results

In this sub-part, by analyzing the algorithmic complexity, we give a maximum estimation of iterations of the outer space algorithm. First of all, for convenience, we denote the maximum size $\Delta(S)$ of the sub-rectangle

$$S = \{s \in R^q | \underline{s}_m \leq s_m \leq \overline{s}_m, m = 1, 2, \ldots, q\}$$

as

$$\Delta(S) := \max\{\overline{s}_m - \underline{s}_m | \ m = 1, 2, \ldots, q\}.$$

In addition, we denote

$$\beta = \max\left\{\sum_{j=1}^{n} |h_{mj}\delta_j^0| + |d_m|| m = 1, 2, \ldots, q\right\},$$

where $\delta_j^0 = \max\{z_j | \ z \in \Lambda\}$.

**Theorem 3.** For any setting convergence error $\epsilon > 0$, at iteration $k$, when the sub-rectangle $S^k$ generated by the outer space branching process satisfies

$$\Delta(S^k) \leq \frac{\epsilon}{q\beta},$$

we can get that

$$UB - LB(S^k) \leq \epsilon,$$

where $LB(S^k)$ is the optimum value to the ARP($S^k$), and $UB$ is the currently known upper bound of the global optimum value to the EBOP$\big(S^0\big)$.

*Proof.* Denote $(z^k, \hat{s}^k)$ as the optimum solution to the ARP($S^k$), and let

$$s_m^k = \frac{1}{\sum\limits_{j=1}^{n} g_{mj} z_j^k + f_m}, m = 1, 2, \ldots, q,$$

then $(z^k, s^k)$ is a feasible point of the EBOP($S^k$).

By the updating and computing methods of $UB$ and $LB(S^k)$, we have that

$$\Psi_0(z^k, s^k) \geq UB \geq LB(S^k) = \Psi_0^L(z^k, \hat{s}^k). \tag{6}$$

Thus, from (1), (6), and the definitions of $\Delta(S^k)$ and $\beta$, it follows that

$$\begin{aligned}
UB - LB(S^k) &\leq \Psi_0(z^k, s^k) - \Psi_0^L(z^k, \hat{s}^k) \\
&= s_m^k(\sum_{j=1}^{n} h_{mj} z_j^k + d_m) - (\sum_{m=1}^{q}(\sum_{j\in H_m^+} h_{mj}\underline{s}_m^k z_j^k + \sum_{j\in H_m^-} h_{mj} U_m^k z_j^k) + \sum_{m=1}^{q} d_m \hat{s}_m^k) \\
&= \sum_{m=1}^{q}(\sum_{j\in H_m^+} h_{mj}(s_m^k - \underline{s}_m^k) z_j^k - \sum_{j\in H_m^-} h_{mj}(\overline{s}_m^k - s_m^k) z_j^k) + \sum_{m=1}^{q}(d_m(s_m^k - \hat{s}_m^k)) \\
&\leq \sum_{m=1}^{q}(\sum_{j\in H_m^+} h_{mj}(\overline{s}_m^k - \underline{s}_m^k) z_j^k - \sum_{j\in H_m^-} h_{mj}(\overline{s}_m^k - \underline{s}_m^k) z_j^k) + \sum_{m=1}^{q}(d_m(\overline{s}_m^k - \underline{s}_m^k)) \\
&= \sum_{m=1}^{q}((\overline{s}_m^k - \underline{s}_m^k)(\sum_{j\in H_m^+} h_{mj} z_j^k - \sum_{j\in H_m^-} h_{mj} z_j^k + d_m)) \\
&\leq \sum_{m=1}^{q}((\overline{s}_m^k - \underline{s}_m^k)(\sum_{j\in H_m^+} h_{mj}\delta_j^0 - \sum_{j\in H_m^-} h_{mj}\delta_j^0 + |d_m|)) \\
&\leq \sum_{m=1}^{q}((\overline{s}_m^k - \underline{s}_m^k)(\sum_{j=1}^{n} |h_{mj}\delta_j^0| + |d_m|)) \\
&\leq \sum_{m=1}^{q}(\Delta(S^k)\beta) \\
&= q\beta\Delta(S^k).
\end{aligned}$$

Further, from the previous inequalities and $\Delta(S^k) \leq \frac{\epsilon}{q\beta}$, we can follow that

$$UB - LB(S^k) \leq \sum_{m=1}^{q}(\Delta(S^k)\beta) \leq \epsilon,$$

and the proof of the theorem is completed. ⋄

By the above Theorem 4 and Step 6 of the presented algorithm, when $\Delta(S^k) \leq \frac{\epsilon}{q\beta}$, $S^k$ will be deleted. Hence, when the sizes of all refined subdivision rectangle $S$ produced by the outer space bisection operation satisfy $\Delta(S) \leq \frac{\epsilon}{q\beta}$, the proposed algorithm will be terminated. According to Theorem 4, we may give a maximum estimation of iteration times for the proposed algorithm in this article, see the following Theorem 4 for details.

**Theorem 4.** For arbitrary $\epsilon > 0$, the presented algorithm can seek out an $\epsilon$-globally optimum solution to the SAFFP in at most

$$K = 2^{\sum_{m=1}^{q}\lceil \log_2 \frac{q\beta(\bar{s}_m^0 - \underline{s}_m^0)}{\epsilon}\rceil} - 1$$

iterations, where $\beta$ is defined in the former, and $S^0 = \prod_{m=1}^{q} S_m^0$ with $S_m^0 = [L_m^0, U_m^0]$.

*Proof.* According to Theorem 4 and the partitioning process of the algorithm, the conclusion of the Theorem can be easily concluded, so it is omitted. ⋄

## 4. Numerical experiments

In this part, we give numerical comparison results among the BARON solver [36], the algorithm proposed in Jiao and Liu [12] which works by globally addressing an equivalent bilinear programming problem, and our algorithm. All algorithms are coded in the software MATLAB R2014a and run on a microcomputer with 2.50 GHz i5-7200U processor and 16 GB RAM. The maximum CPU running time limit for all test problems is set at 3800 $s$. We reported the numerical result statistics for all test Problems 1 and 2. For each randomly generated test problem, we all solved ten randomly generated test examples and recorded their best results, their worst results and their average results, and highlighted the winner of comparisons of their average results in bold. In the following, we firstly present these test problems and then report their numerical comparisons.

**Problem 1:**

$$\begin{cases} \max \ \sum_{i=1}^{p} \frac{\sum_{j=1}^{n} h_{ij}x_j + d_i}{\sum_{j=1}^{n} g_{ij}x_j + f_i} \\ \text{s.t.} \ \sum_{j=1}^{n} a_{kj}x_j \leq b_k, \ k = 1, 2, \ldots, m, \\ \qquad x_j \geq 0.0, \ j = 1, 2, \ldots, n, \end{cases}$$

where $h_{ij}, g_{ij}, a_{kj}, i = 1, 2, \ldots, p, k = 1, 2, \ldots, m, j = 1, 2, \ldots, n$, are all randomly generated in the interval $[0, 10]$; $b_k = 10, k = 1, 2, \ldots, m$, $g_i$ and $h_i, i = 1, 2, \ldots, p$, are all randomly generated in the unit interval $[0, 1]$. What needs to be clearly pointed out is that, Problem 1 has the little constant number $d_i$ and $f_i$ at the numerators and denominators of ratios.

For Problem 1 with the large-size number of variables, with the convergent tolerance $\epsilon = 10^{-2}$, numerical comparisons among algorithm of Jiao and Liu [12], our algorithm and BARON are reported in Table 1. For each random example, we solve ten independently generated instances and record the best, the worst and the average results among these ten tests, and we highlight in bold the winner of average results in comparison.

**Problem 2:**

$$\begin{cases} \min \ \sum_{i=1}^{p} \frac{\sum_{j=1}^{n} h_{ij}x_j+d_i}{\sum_{j=1}^{n} g_{ij}x_j+f_i} \\ \text{s.t.} \ \ \sum_{j=1}^{n} a_{kj}x_{ij} \le b_k, \ \ k = 1, \dots, m, \\ \qquad x_j \ge 0.0, \ \ j = 1, \dots, n. \end{cases}$$

where $h_{ij}, g_{ij} \in [-0.1, 0.1]$, $i = 1, \dots, p$, $j = 1, \dots, n$, $a_{kj} \in [0.01, 1]$, $j = 1, \dots, n$, are all uniform distribution random numbers; $b_k = 10$, $k = 1, \dots, m$; all constant terms $d_i$ and $f_i$ of numerators and denominators of ratios satisfying $\sum_{j=1}^{n} h_{ij}x_j + d_i > 0$ and $\sum_{j=1}^{n} g_{ij}x_j + f_i > 0$.

For Problem 2 with the large-size number $q$, with the convergence tolerance $\epsilon = 10^{-3}$, numerical comparisons between our algorithm and BARON are reported in Table 2. In Tables 1 and 2, "−" stand for the condition that the used algorithm failed to seek out the globally optimum solution to some of ten random examples in $3800s$.

From Table 1, for Problem 1 with large-size number of variables, we firstly can observe that the BARON solver takes more time than our algorithm proposed in this article, despite its number of iterations for the BARON solver is smaller. Secondly, our algorithm is obviously better than the BARON solver and the algorithm of Jiao and Liu [12]. The iteration number of our algorithm proposed in this article is much less than the algorithm of Jiao and Liu [12]. Especially, when $q = 2$ and $n = 8000$, the BARON solver failed to seek out the globally optimum solution to each of ten random examples in $3800s$, but our outer space searching algorithm can achieve the globally optimum solution to all ten random examples of Problem 1 with higher computational efficiency and performance.

From Table 2, for Problem 2 with the large-size number $q$, we observe that, when $q = 10, 15$ and $n = 500, 600$, and $q = 20$ and $n = 400, 500$, the BARON solver failed to terminate in $3800s$ for each one of ten independently generated instances, but our outer space searching algorithm in this paper can seek out the globally optimum solution to all ten independently generated instances within a reasonable time, this demonstrate the strong robustness and reliable stability of our algorithm.

**Table 1.** Comparisons of numerical results among the algorithm of Jiao and Liu [12], the BARON solver and our algorithm in this article on Problem 1 with $q = 2$ and $n = 100$.

| $n$ | Algorithms | Iterations | | | Time(s) | | |
|---|---|---|---|---|---|---|---|
| | | Min | Ave | Max | Min | Ave | Max |
| 1000 | Jiao and Liu [12] | 25 | 81.7 | 142 | 20.05 | 70.78 | 122.97 |
| | BARON | **1** | **1.8** | **3** | **20.36** | **42.02** | **86.48** |
| | Ours | 29 | 98.3 | 178 | 18.89 | 59.67 | 107.69 |
| 2000 | Jiao and Liu [12] | 28 | 108.7 | 222 | 51.92 | 205.71 | 441.71 |
| | BARON | **1** | **1.2** | **3** | 77.42 | 279.01 | 478.45 |
| | Ours | 32 | 105.9 | 199 | **43.16** | **152.34** | **285.78** |
| 3000 | Jiao and Liu [12] | 46 | 82.7 | 153 | 136.07 | 239.74 | 459.27 |
| | BARON | **1** | **1.4** | **5** | 214.25 | 587.91 | 1198.08 |
| | Ours | 47 | 109.6 | 189 | **92.51** | **236.23** | **465.73** |
| 4000 | Jiao and Liu [12] | 56 | 74.6 | 110 | 225.69 | 290.80 | 429.96 |
| | BARON | **1** | **1.8** | **5** | 527.52 | 1408.32 | 2671.62 |
| | Ours | 37 | 80.5 | 146 | **97.79** | **224.18** | **439.39** |
| 5000 | Jiao and Liu [12] | 40 | 104.8 | 244 | 186.21 | 530.14 | 1244.53 |
| | BARON | **1** | **1.2** | **3** | 920.05 | 1083.93 | 1408.27 |
| | Ours | 52 | 80.3 | 121 | **180.41** | **291.78** | **453.67** |
| 6000 | Jiao and Liu [12] | 67 | 93.5 | 146 | 431.38 | 611.27 | 969.71 |
| | BARON | **1** | **1** | **1** | 1392.75 | 1909.50 | 2518.44 |
| | Ours | 27 | 94.4 | 185 | **111.42** | **422.56** | **849.12** |
| 7000 | Jiao and Liu [12] | 31 | 81.7 | 184 | 217.49 | 615.68 | 1290.42 |
| | BARON | **1** | **1** | **1** | 2253.22 | 2778.35 | 3727.55 |
| | Ours | 26 | 74.9 | 160 | **130.01** | **395.59** | **835.53** |
| 8000 | Jiao and Liu [12] | 32 | 84.9 | 139 | 276.25 | 802.90 | 1323.32 |
| | BARON | – | – | – | – | – | – |
| | Ours | **25** | **71.5** | **111** | **145.40** | **452.78** | **712.18** |
| 10000 | Jiao and Liu [12] | 35 | 76.6 | 112 | 405.80 | 933.54 | 1414.22 |
| | BARON | – | – | – | – | – | – |
| | Ours | **42** | **69.5** | **96** | **329.85** | **585.72** | **826.81** |
| 20000 | Jiao and Liu [12] | **41** | **69.4** | **105** | 1239.04 | 2216.69 | 3495.84 |
| | BARON | – | – | – | – | – | – |
| | Ours | 35 | 69.5 | 140 | **691.74** | **1551.82** | **3343.63** |

**Table 2.** Comparisons of numerical results between the BARON solver and our algorithm on Problem 2.

| $(q, \mu, n)$ | Algorithms | Number of iterations | | | Time(s) | | |
|---|---|---|---|---|---|---|---|
| | | Min | Ave | Max | Min | Ave | Max |
| (10,100,300) | BARON | **3** | **9.2** | **13** | **8.28** | **12.66** | **17.64** |
| | Ours | 200 | 220.8 | 269 | 112.49 | 122.92 | 139.39 |
| (10,100,400) | BARON | **9** | **35.8** | **93** | **22.28** | **30.86** | **42.33** |
| | Ours | 203 | 218.2 | 236 | 131.94 | 144.23 | 161.48 |
| (10,100,500) | BARON | – | – | – | – | – | – |
| | Ours | **197** | **221.1** | **273** | **163.62** | **184.40** | **222.16** |
| (10,100,600) | BARON | – | – | – | – | – | – |
| | Ours | **199** | **216.6** | **255** | **191.31** | **202.13** | **230.07** |
| (15,100,300) | BARON | **5** | **10.4** | **17** | **15.66** | **24.24** | **38.45** |
| | Ours | 375 | 559.2 | 1032 | 219.29 | 293.51 | 492.41 |
| (15,100,400) | BARON | **11** | **34** | **157** | **36.14** | **47.92** | **79.81** |
| | Ours | 342 | 453.1 | 734 | 247.01 | 317.91 | 513.57 |
| (15,100,500) | BARON | – | – | – | – | – | – |
| | Ours | **376** | **616.9** | **1075** | **323.77** | **501.02** | **858.10** |
| (15,100,600) | BARON | – | – | – | – | – | – |
| | Ours | **382** | **593.1** | **1004** | **390.22** | **578.89** | **970.74** |
| (20,100,200) | BARON | **15** | **17** | **19** | **17.84** | **20.50** | **26.39** |
| | Ours | 929 | 1634.6 | 4332 | 406.10 | 662.12 | 1611.65 |
| (20,100,300) | BARON | **5** | **14** | **17** | **22.53** | **36.14** | **51.11** |
| | Ours | 787 | 1401.3 | 2215 | 477.65 | 758.77 | 1115.45 |
| (20,100,400) | BARON | – | – | – | – | – | – |
| | Ours | **628** | **1804.8** | **3688** | **464.02** | **1344.29** | **2805.79** |
| (20,100,500) | BARON | – | – | – | – | – | – |
| | Ours | **974** | **1820** | **3246** | **885.26** | **1574.39** | **2739.26** |

## 5. Conclusions

By combining the outer space branch searching scheme, the constructed affine relaxation problem, and the outer space region reduction technique, we design a novel algorithm to efficiently solve the SAFFP. In contrast to the known existing algorithms, by analysing the algorithmic complexity, we can get that the proposed algorithm in this paper can achieve an $\epsilon$-global optimum solution of the SAFFP after at most $2^{\sum_{m=1}^{q} \lceil \log_2 \frac{q\beta(\overline{s}_m^0 - \underline{s}_m^0)}{\epsilon} \rceil} - 1$ iterations. Finally, numerical comparison results are given to demonstrate better computational performance of the proposed algorithm in this paper. In the future work, we will extend our algorithm to globally solve generalized linear fractional programming problem.

**Acknowledgments**

This work was supported by the National Natural Science Foundation of China (11871196; 12071133; 12071112), China Postdoctoral Science Foundation (2017M622340), the Key Scientific and Technological Research Projects in Henan Province (222102110409; 202102210147; 192102210114), the Science and Technology Climbing Program of Henan Institute of Science and Technology (2018JY01).

**Conflict of interest**

The authors declare no conflicts of interest.

**References**

1. E. B. Bajalinov, *Linear-fractional programming theory, methods, applications and software*, Springer Science & Business Media, Vol. 84, 2003. https://doi.org/10.1007/978-1-4419-9174-4
2. I. M. Stancu-Minasian, A eighth bibliography of fractional programming, *Optimization*, **66** (2017), 439–470. https://doi.org/10.1080/02331934.2016.1276179
3. T. Kuno, T. Masaki, A practical but rigorous approach to sum-of-ratios optimization in geometric applications, *Comput. Optim. Appl.*, **54** (2013), 93–109. https://doi.org/10.1007/s10589-012-9488-5
4. I. M. Stancu-Minasian, A ninth bibliography of fractional programming, *Optimization*, **68** (2019), 2125–2169. https://doi.org/10.1080/02331934.2019.1632250
5. Y. Ji, H. Li, H. Zhang, Risk-averse two-stage stochastic minimum cost consensus models with asymmetric adjustment cost, *Group Decis. Negot.*, **31** (2022), 261–291. https://doi.org/10.1007/s10726-021-09752-z
6. S. Qu, L. Shu, J. Yao, Optimal pricing and service level in supply chain considering misreport behavior and fairness concern, *Comput. Ind. Eng.*, **174** (2022), 108759. https://doi.org/10.1016/j.cie.2022.108759
7. H. Konno, Y. Yajima, T. Matsui, Parametric simplex algorithms for solving a special class of nonconvex minimization problems, *J. Glob. Optim.*, **1** (1991), 65–81. https://doi.org/10.1007/BF00120666
8. H. Konno, H. Yamashita, Minimizing sums and products of linear fractional functions over a polytope, *Nav. Res. Log.*, **46** (1999), 583–596.
9. J. E. Falk, S. W. Palocsay, Image space analysis of generalized fractional programs, *J. Glob. Optim.*, **4** (1994), 63–88. https://doi.org/10.1007/BF01096535
10. N. T. H. Phuong, H. Tuy, A unified monotonic approach to generalized linear fractional programming, *J. Glob. Optim.*, **26** (2003), 229–259. https://doi.org/10.1023/A:1023274721632
11. H. Konno, K. Fukaishi, A branch-and-bound algorithm for solving low-rank linear multiplicative and fractional programming problems, *J. Glob. Optim.*, **18** (2000), 283–299. https://doi.org/10.1023/A:1008314922240

12. H. Jiao, S. Liu, A practicable branch and bound algorithm for sum of linear ratios problem, *Eur. J. Oper. Res.*, **243** (2015), 723–730. https://doi.org/10.1016/j.ejor.2015.01.039

13. Y. Ji, K. C. Zhang, S. J. Qu, A deterministic global optimization algorithm, *Appl. Math. Comput.*, **185** (2007), 382–387. https://doi.org/10.1016/j.amc.2006.06.101

14. H. P. Benson, A simplicial branch and bound duality-bounds algorithm for the linear sum-of-ratios problem, *Eur. J. Oper. Res.*, **182** (2007), 597–611. https://doi.org/10.1016/j.ejor.2006.08.036

15. T. Kuno, A revision of the trapezoidal branch-and-bound algorithm for linear sum-of-ratios problems, *J. Glob. Optim.*, **33** (2005), 215–234. https://doi.org/10.1007/s10898-004-1952-z

16. H. Jiao, Y. Shang, W. 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

17. H. W. Jiao, Y. L. Shang, Two-level linear relaxation method for generalized linear fractional programming, *J. Oper. Res. Soc. China*, 2022, 1–26. https://doi.org/10.1007/s40305-021-00375-4

18. H. Jiao, Y. Shang, R. Chen, A potential practical algorithm for minimizing the sum of affine fractional functions, *Optimization*, 2022, 1–31. https://doi.org/10.1080/02331934.2022.2032051

19. H. Jiao, J. Ma, P. Shen, Y. Qiu, Effective algorithm and computational complexity for solving sum of linear ratios problem, *J. Ind. Manag. Optim.*, 2022. https://doi.org/10.3934/jimo.2022135

20. H. Jiao, J. 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

21. H. Jiao, 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

22. H. Jiao, W. Wang, Y. 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

23. H. Jiao, J. Ma, Y. Shang, Image space branch-and-bound algorithm for globally solving minimax linear fractional programming problem, *Pac. J. Optim.*, **18** (2022), 195–212.

24. J. Ma, H. Jiao, J. Yin, Y. Shang, Outer space branching search method for solving generalized affine fractional optimization problem, *AIMS Math.*, **8** (2022), 1959–1974. https://doi.org/10.3934/math.2023101

25. H. Jiao, W. Wang, J. Yin, Y. 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

26. H. Jiao, R. Chen, A parametric linearizing approach for quadratically inequality constrained quadratic programs, *Open Math.*, **16** (2018), 407–419. https://doi.org/10.1515/math-2018-0037

27. H. Jiao, S. Liu, An efficient algorithm for quadratic sum-of-ratios fractional programs problem, *Numer. Func. Anal. Optim.*, **38** (2017), 1426–1445. https://doi.org/10.1080/01630563.2017.1327869

28. H. Jiao, S. Liu, J. Yin, Y. Zhao, Outcome space range reduction method for global optimization of sum of affine ratios problem, *Open Math.*, **14** (2016), 736–746. https://doi.org/10.1515/math-2016-0058
29. P. Shen, B. Huang, L. Wang, Range division and linearization algorithm for a class of linear ratios optimization problems, *J. Comput. Appl. Math.*, **350** (2019), 324–342. https://doi.org/10.1016/j.cam.2018.10.038
30. D. Depetrini, M. Locatelli, Approximation algorithm for linear fractional multiplicative problems, *Math. Program.*, **128** (2011), 437–443. https://doi.org/10.1007/s10107-009-0309-2
31. S. Schaible, J. Shi, Fractional programming: the sum-of-ratios case, *Optim. Method. Softw.*, **18** (2003), 219–229. https://doi.org/10.1080/1055678031000105242
32. P. Saxena, R. Jain, Duality in linear fractional programming under fuzzy environment using hyperbolic membership functions, *Int. J. Fuzzy Syst. Appl. (IJFSA)*, **9** (2020), 1–21. https://doi.org/10.4018/IJFSA.2020070101
33. M. Borza, A. S. Rambely, A linearization to the sum of linear ratios programming problem, *Mathematics*, **9** (2021), 1004. https://doi.org/10.3390/math9091004
34. M. Goli, S. H. Nasseri, Extension of duality results and a dual simplex method for linear programming problems with intuitionistic fuzzy variables, *Fuzzy Inf. Eng.*, **12** (2020), 392–411. https://doi.org/10.1080/16168658.2021.1908818
35. R. Horst, H. Tuy, *Global optimization: deterministic approaches*, Springer Science & Business Media, 2013. https://doi.org/10.1007/978-3-662-02598-7
36. A. Khajavirad, N. V. Sahinidis, A hybrid LP/NLP paradigm for global optimization relaxations, *Math. Program. Comput.*, **10** (2018), 383–421. https://doi.org/10.1007/s12532-018-0138-5