

Research Article

A Global Optimization Algorithm for Sum of Linear Ratios Problem

Yuelin Gao and Siqiao Jin

Institute of Information & System Science, Beifang University of Nationalities, Yinchuan 750021, China

Correspondence should be addressed to Yuelin Gao; gaoyuelin@263.net

Received 31 January 2013; Accepted 8 May 2013

Academic Editor: Farhad Hosseinzadeh Lotfi

Copyright © 2013 Y. Gao and S. Jin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We equivalently transform the sum of linear ratios programming problem into bilinear programming problem, then by using the linear characteristics of convex envelope and concave envelope of double variables product function, linear relaxation programming of the bilinear programming problem is given, which can determine the lower bound of the optimal value of original problem. Therefore, a branch and bound algorithm for solving sum of linear ratios programming problem is put forward, and the convergence of the algorithm is proved. Numerical experiments are reported to show the effectiveness of the proposed algorithm.

1. Introduction

We consider the sum of linear ratios programming problem as the following form:

$$(\text{GFP}) : \begin{cases} \min f(x) = \sum_{i=1}^p f_i(x) = \sum_{i=1}^p \frac{n_i(x)}{d_i(x)}, \\ \text{s.t. } Ax \leq b, \quad x \geq 0, \end{cases} \quad (1)$$

where the feasible domain $D \triangleq \{x \in R^n \mid Ax \leq b, x \geq 0\}$ is n -dimensional, nonempty, and bound, $A \in R^{m \times n}$, $b \in R^m$. Assume that $n_i(x) = c_i^T x + d_i \geq 0$ and $d_i(x) = e_i^T x + r_i > 0$ in some rectangle X which contains D , where $c_i, e_i \in R^n$, $d_i, r_i \in R$, $i = 1, 2, \dots, p$, and $2 \leq p \ll n$.

Fractional programming is an important branch of nonlinear optimization and it has attracted many researchers' concern for several decades. Sum of linear ratios problem is a special class of fractional programming problem; it has wide applications, such as investment, transportation scheme, and economic benefits [1–3]. From a research point view, sum of ratios problems challenge theoretical analysis and computation because these problems possess multiple local optima that are not globally optimal solutions; it is difficult to solve the global solution.

At present there exist a number of algorithms for globally solving sum of linear ratios problems. When $p = 2$, Konno et al. [4] constructed a similar parametric simplex algorithm

which can solve large-scale optimization problems; when $p = 3$, Konno and Abe [5] developed parametric simplex algorithm and constructed an effected heuristic algorithm; when $p > 3$, the literature [6] is a sum of linear ratios problem with coefficients; by using an equivalent transformation and linearization technique, the original nonconvex programming problem reduces to a series of linear programming problems to achieve the purpose of solving it. To minimize the problem, Yanjun et al. [7] use the linearization technique twice by the nature of exponential and logarithmic functions to achieve a linear relaxation programming of the original problem. Benson [8] put forward a new branch and bound algorithm to solve the equivalent concave minimum problem of the original problem. Jiao and Feng [9] present a new pruning technique. In the literature [10], the numerator and denominator of the ratios are not necessarily positive. In this paper, we present a new branch and bound algorithm for solving the sum of linear ratios problems, and the convergence of the algorithm is proved. At last, the numerical experiments are carried out.

This paper is organized as follows. In Section 2, we show how to convert the problem (GFP) into an equivalent problem (EP) by a transformed technique. In Section 3, the linear relaxation programming problem of (EP) is constructed. The branching process of the rectangle is given in Section 4. In Section 5, the branch and bound algorithm for globally solving (EP) is presented and the convergence of

the algorithm is proved. In Section 6, some numerical results are given to show the effectiveness of the present algorithm. Finally, the conclusion is given.

2. Equivalent Transformation

Because the set D is nonempty and bound, we can construct the rectangle $X = [l, u]$, which contains the feasible region of the problem (GFP), $l = (l_1, l_2, \dots, l_n)^T, u = (u_1, u_2, \dots, u_n)^T, l_j$ and u_j is the optimal value of the linear programming problem (2) and (3), respectively.

$$\min \quad l(x_j) = x_j, \quad (2)$$

$$\text{s.t.} \quad Ax \leq b, \quad x \geq 0,$$

$$\max \quad u(x_j) = x_j, \quad (3)$$

$$\text{s.t.} \quad Ax \leq b, \quad x \geq 0.$$

Firstly, we solve the following $2p$ linear programming problems:

$$\min \quad d_i(x),$$

$$\text{s.t.} \quad x \in D,$$

$$\max \quad \bar{d}_i(x), \quad (4)$$

$$\text{s.t.} \quad x \in D,$$

$$i = 1, 2, \dots, p.$$

The optimal solutions of (4) are x_i^1 and x_i^2 ($i = 1, 2, \dots, p$), and the optimal value is denoted by \bar{l}_i and \bar{u}_i ($i = 1, 2, \dots, p$) respectively. Obviously, x_i^1 and x_i^2 are feasible to (GFP). Set $W = W \cup \{x_i^1, x_i^2 : i = 1, 2, \dots, p\}$, where W represent the set of the current feasible solution of the problem (GFP). Set

$$H^0 = \{y \in R^p \mid l_i^0 \leq y_i \leq u_i^0, i = 1, 2, \dots, p\}, \quad (5)$$

$$y = (y_1, y_2, \dots, y_p)^T,$$

where $l_i^0 = 1/\bar{u}_i, u_i^0 = 1/\bar{l}_i$. Then the problem (GFP) is converted into an equivalent nonconvex programming problem:

$$\text{EP}(H^0) : \left\{ \begin{array}{l} \min \quad \varphi_0(x, y) = \sum_{i=1}^p y_i n_i(x) \\ \quad \quad \quad = \sum_{i=1}^p y_i \left(\sum_{j=1}^n c_{ij} x_j + d_i \right), \\ \text{s.t.} \quad \varphi_i(x, y) = y_i d_i(x) \\ \quad \quad \quad = y_i \left(\sum_{j=1}^n e_{ij} x_j + r_i \right) \\ \quad \quad \quad \geq 1, \quad i = 1, 2, \dots, p, \\ \quad \quad \quad x \in D \cap X, y \in H^0. \end{array} \right. \quad (6)$$

Theorem 1 (see [10]). *If (x^*, y^*) is a global optimal solution of the problem $\text{EP}(H^0)$, then x^* is a global optimal solution of the problem (GFP), and for every $i = 1, 2, \dots, p$, when $n_i(x^*) \geq 0, y_i^* = 1/d_i(x^*)$; conversely, if x^* is a global optimal solution of the problem (GFP), then (x^*, y^*) is a global optimal solution of the problem $\text{EP}(H^0)$, where $y_i^* = 1/d_i(x^*), i = 1, 2, \dots, p$.*

From Theorem 1, the problems (GFP) and $\text{EP}(H^0)$ are equivalent; their global optimal values are equal. Therefore, in order to solve (GFP), we only need to solve $\text{EP}(H^0)$ instead.

3. Linear Relaxation Technique

From Section 2, $X = [l, u]$ and $H^0 = [l^0, u^0]$ are rectangles; set

$$\begin{aligned} \Omega_i &= \{(x, y_i) \mid l \leq x \leq u, l_i^0 \leq y_i \leq u_i^0\} \\ &= \Omega_{i1} \times \Omega_{i2} \times \dots \times \Omega_{in}, \end{aligned} \quad (7)$$

where

$$\Omega_{ji} = \{(x_j, y_i) \mid l_j \leq x_j \leq u_j, l_i^0 \leq y_i \leq u_i^0\}, \quad (8)$$

$$j = 1, 2, \dots, n.$$

Because in Ω_{ji} we have $x_j - l_j \geq 0, y_i - l_i^0 \geq 0$, so

$$(x_j - l_j)(y_i - l_i^0) \geq 0, \quad j = 1, 2, \dots, n, \quad (9)$$

expanding it, then we have $x_j y_i \geq l_i^0 x_j + l_j y_i - l_j l_i^0, j = 1, 2, \dots, n$.

Similarly, we can obtain that $x_j - u_j \leq 0, y_i - u_i^0 \leq 0$, so

$$(x_j - u_j)(y_i - u_i^0) \geq 0, \quad j = 1, 2, \dots, n, \quad (10)$$

expanding it, then we have $x_j y_i \geq u_i^0 x_j + u_j y_i - u_j u_i^0, j = 1, 2, \dots, n$. Let

$$\theta_{ji}^{11}(x_j, y_i) = l_i^0 x_j + l_j y_i - l_j l_i^0, \quad j = 1, 2, \dots, n, \quad (11)$$

$$\theta_{ji}^{12}(x_j, y_i) = u_i^0 x_j + u_j y_i - u_j u_i^0, \quad j = 1, 2, \dots, n.$$

Because $x_j y_i \geq \theta_{ji}^{11}(x_j, y_i), x_j y_i \geq \theta_{ji}^{12}(x_j, y_i), j = 1, 2, \dots, n$, we have the following result:

$$x_j y_i \geq \max \{ \theta_{ji}^{11}(x_j, y_i), \theta_{ji}^{12}(x_j, y_i) \}, \quad j = 1, 2, \dots, n. \quad (12)$$

Similarly, we have $(x_j - l_j)(y_i - u_i^0) \leq 0, (x_j - u_j)(y_i - l_i^0) \leq 0, j = 1, 2, \dots, n$, expanding them, then we have $x_j y_i \leq u_i^0 x_j + l_j y_i - l_j u_i^0, x_j y_i \leq l_i^0 x_j + u_j y_i - u_j l_i^0$; let

$$\theta_{ji}^{21}(x_j, y_i) = u_i^0 x_j + l_j y_i - l_j u_i^0, \quad j = 1, 2, \dots, n, \quad (13)$$

$$\theta_{ji}^{22}(x_j, y_i) = l_i^0 x_j + u_j y_i - u_j l_i^0, \quad j = 1, 2, \dots, n.$$

Consequently,

$$x_j y_i \leq \min \{ \theta_{ji}^{21}(x_j, y_i), \theta_{ji}^{22}(x_j, y_i) \}, \quad j = 1, 2, \dots, n. \quad (14)$$

From formulae (12) and (14), the following formula is obtained:

$$\begin{aligned} & \max \{ \theta_{ji}^{11}(x_j, y_i), \theta_{ji}^{12}(x_j, y_i) \} \\ & \leq x_j y_i \\ & \leq \min \{ \theta_{ji}^{21}(x_j, y_i), \theta_{ji}^{22}(x_j, y_i) \}. \end{aligned} \tag{15}$$

In the problem $EP(H^0)$, let $LB(x)$ and $UB(x)$, respectively, represent the lower bound and upper bound of x ; then

$$\begin{aligned} & LB(c_{ij}x_j y_i) \\ & = \begin{cases} c_{ij} \cdot \max \{ \theta_{ji}^{11}(x_j, y_i), \theta_{ji}^{12}(x_j, y_i) \}, & c_{ij} \geq 0, \\ c_{ij} \cdot \min \{ \theta_{ji}^{21}(x_j, y_i), \theta_{ji}^{22}(x_j, y_i) \}, & c_{ij} < 0, \end{cases} \\ & UB(e_{ij}x_j y_i) \\ & = \begin{cases} e_{ij} \cdot \min \{ \theta_{ji}^{21}(x_j, y_i), \theta_{ji}^{22}(x_j, y_i) \}, & e_{ij} \geq 0, \\ e_{ij} \cdot \max \{ \theta_{ji}^{11}(x_j, y_i), \theta_{ji}^{12}(x_j, y_i) \}, & e_{ij} < 0. \end{cases} \end{aligned} \tag{16}$$

From formula (16), we can obtain the linear relaxation programming problem $REP(H^0)$ of the problem $EP(H^0)$:

$$REP(H^0) : \begin{cases} \min & \varphi_0^l(x, y) = \sum_{i=1}^p \left(\sum_{j=1}^n LB(c_{ij}x_j y_i) + d_i y_i \right), \\ \text{s.t.} & \varphi_i^u(x, y) = \sum_{j=1}^n UB(e_{ij}x_j y_i) + r_i y_i \geq 1, \\ & i = 1, 2, \dots, p, \\ & x \in D \cap X, y \in H^0. \end{cases} \tag{17}$$

The optimal value of the problem $REP(H^0)$ is a lower bound of the optimal value of the problem $EP(H^0)$ in the feasible region D .

Obviously, the problem $REP(H^0)$ can equivalently be converted into the following linear programming problem $LRP(H^0)$:

$$LRP(H^0) : \begin{cases} \min & f(x, y, t, s) = \sum_{i=1}^p \left(\sum_{j=1}^n t_{ji} + d_i y_i \right), \\ \text{s.t.} & \sum_{j=1}^n s_{ji} + r_i y_i \geq 1, \quad i = 1, 2, \dots, p, \\ & t_{ji} \geq c_{ij} \cdot \theta_{ji}^{11}(x_j, y_i), \quad c_{ij} \geq 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & t_{ji} \geq c_{ij} \cdot \theta_{ji}^{12}(x_j, y_i), \quad c_{ij} \geq 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & t_{ji} \geq c_{ij} \cdot \theta_{ji}^{21}(x_j, y_i), \quad c_{ij} < 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & t_{ji} \geq c_{ij} \cdot \theta_{ji}^{22}(x_j, y_i), \quad c_{ij} < 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & s_{ji} \leq e_{ij} \cdot \theta_{ji}^{21}(x_j, y_i), \quad e_{ij} \geq 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & s_{ji} \leq e_{ij} \cdot \theta_{ji}^{22}(x_j, y_i), \quad e_{ij} \geq 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & s_{ji} \leq e_{ij} \cdot \theta_{ji}^{11}(x_j, y_i), \quad e_{ij} < 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & s_{ji} \leq e_{ij} \cdot \theta_{ji}^{12}(x_j, y_i), \quad e_{ij} < 0, \quad j = 1, 2, \dots, n, \quad i = 1, 2, \dots, p, \\ & x \in D \cap X, y \in H^0. \end{cases} \tag{18}$$

The optimal value of the problem $LRP(H^0)$ can be obtained by solving the linear programming problem $LRP(H^0)$, which is a lower bound of the problem $EP(H^0)$ in feasible region D .

The Determination of Upper Bound. From the process of the determination of lower bound, by solving $LRP(H^0)$, we can obtain a global optimal solution \bar{x}^* ; let

$$\bar{y}_i^* = \left(\sum_{j=1}^n e_{ij} \bar{x}_j^* + r_i \right)^{-1}. \tag{19}$$

It is obvious that (\bar{x}^*, \bar{y}^*) is a feasible solution of $EP(H^0)$. Therefore, $\varphi_0(\bar{x}^*, \bar{y}^*)$ provide an upper bound for the global optimal value $\nu(H^0)$ of the problem $EP(H^0)$.

4. Branching

In this algorithm, the branching process is executed in the space of R^p other than in R^n . In general, when $p \ll n$, the amount of computation will decrease so that the efficiency of computation will improve. Therefore, we choose the rectangle

H^0 which contains y to branch, and the subrectangle after branching is also p -dimensional. Set

$$H = \{y \in R^p \mid L_i \leq y_i \leq U_i, i = 1, 2, \dots, p\}. \quad (20)$$

Denote the initial rectangle H^0 or subrectangle of it. The branching rule is as follows:

- (i) choose the longest side of H , that is, $U_s - L_s = \max\{U_i - L_i : i = 1, 2, \dots, p\}$;
- (ii) let $V_s = (U_s + L_s)/2$ and

$$\begin{aligned} H^1 &= \prod_{i=1}^{s-1} [L_i, U_i] \times [L_s, V_s] \times \prod_{i=s+1}^p [L_i, U_i], \\ H^2 &= \prod_{i=1}^{s-1} [L_i, U_i] \times [V_s, U_s] \times \prod_{i=s+1}^p [L_i, U_i]. \end{aligned} \quad (21)$$

5. Algorithm and Its Convergence

The branch and bound algorithm of the problem (GFP) is stated as follows:

Step 1. Choose $\varepsilon \geq 0$, the initial rectangle $H^0 = \{y \in R^p \mid l_i^0 \leq y_i \leq u_i^0, i = 1, 2, \dots, p\}$; we can find an optimal solution x^0 and the optimal value $LB(H^0)$ by solving the problem $LRP(H^0)$. Set $LB_0 = LB(H^0)$, $x^c = x^0$. Set $y_i^c = (\sum_{j=1}^n e_{ij}x_j^c + r_i)^{-1}$, $i \in \{1, 2, \dots, p\}$, $UB_0 = \varphi_0(x^c, y^c)$.

If $UB_0 - LB_0 \leq \varepsilon$, stop. (x^c, y^c) and x^c are global ε -optimal solutions of problems $EP(H^0)$ and (GFP), respectively. Otherwise, set $P_0 = \{H^0\}$, $F = \emptyset$, $k = 1$, and go to Step 2.

Step 2. Set $UB_k = UB_{k-1}$. Subdivide H^{k-1} into two p -dimensional rectangles $H^{k,1}, H^{k,2} \subseteq R^p$ via the branching rule. Set $F = F \cup \{H^{k-1}\}$.

Step 3. For $j = 1, 2$, compute $LB(H^{j,k})$. If $LB(H^{j,k}) \neq +\infty$, find an optimal solution $x^{k,j}$ of problem $LRP(\bar{H})$ with $\bar{H} = H^{j,k}$; set $t = 0$.

Step 4. Set $t = t+1$. If $t > 2$, go to Step 6. Otherwise, continue.

Step 5. If $UB_k \leq LB(H^{k,t})$, set $F = F \cup \{H^{k,t}\}$; go to Step 4. Otherwise, set

$$y_i^{k,t} = \left(\sum_{j=1}^n e_{ij}x_j^{k,t} + r_i \right)^{-1}, \quad i \in \{1, 2, \dots, p\}. \quad (22)$$

Let $UB_k = \min\{UB_k, \varphi_0(x^{k,t}, y^{k,t})\}$. If $UB_k < \varphi_0(x^{k,t}, y^{k,t})$, go to Step 4. If $UB_k = \varphi_0(x^{k,t}, y^{k,t})$, set $x^c = x^{k,t}$, $(x^c, y^c) = (x^{k,t}, y^{k,t})$. Let

$$F = F \cup \{H \in P_{k-1} \mid UB_k \leq LB(H)\}. \quad (23)$$

Step 6. Set $P_k = \{H \mid H \in (P_{k-1} \cup \{H^{k,1}, H^{k,2}\}), H \notin F\}$.

Step 7. Set $LB_k = \min\{LB(H) \mid H \in P_k\}$. Let $H^k \in P_k$ satisfy $LB_k = LB(H^k)$.

If $UB_0 - LB_0 \leq \varepsilon$, stop. (x^c, y^c) and x^c are global ε -optimal solutions of the problems $EP(H^0)$ and (GFP), respectively. Otherwise, set $k = k + 1$ and go to Step 2.

Next, the convergence of the algorithm is stated in the following theorem.

Theorem 2. (a) If the algorithm is finite, (x^c, y^c) and x^c are global ε -optimal solutions of the problems $EP(H^0)$ and (GFP), respectively.

(b) For $k \geq 0$, let x^k denote the incumbent solution x^c at the end of step k . If the algorithm is infinite, then $\{x^k\}$ is a feasible solution sequence, whose every accumulation point is a global optimal solution of the problem (GFP), and

$$\lim_{k \rightarrow \infty} UB_k = \lim_{k \rightarrow \infty} LB_k = \nu. \quad (24)$$

Proof. (a) If the algorithm is finite, without loss of generality, it terminates in step k ($k \geq 0$), since (x^c, y^c) is obtained by solving problem $LRP(H)$, for some $H \subseteq H^0$ and optimal solution x^c , set

$$y_i^c = \frac{1}{\sum_{j=1}^n e_{ij}x_j^c + r_i}, \quad i \in \{1, 2, \dots, p\}, \quad (25)$$

where x^c is a feasible solution of the problem (GFP) and (x^c, y^c) is a feasible solution of problem $EP(H^0)$. When $UB_k - LB_k \leq \varepsilon$, the algorithm terminates. From Steps 1, 2, and 5, it is implied that $\varphi_0(x^c, y^c) - LB_k \leq \varepsilon$; by the algorithm, it shows that $LB_k \leq \nu$. Since (x^c, y^c) is a feasible solution of the problem $EP(H^0)$, therefore, $\varphi_0(x^c, y^c) \geq \nu$.

Taken together, it is implied that

$$\nu \leq \varphi_0(x^c, y^c) \leq LB_k + \varepsilon \leq \nu + \varepsilon. \quad (26)$$

Therefore,

$$\nu \leq \varphi_0(x^c, y^c) \leq \nu + \varepsilon. \quad (27)$$

From the formula $y_i^c = 1/(\sum_{j=1}^n e_{ij}x_j^c + r_i)$, $i = 1, 2, \dots, p$, we have

$$f(x^c) = \varphi_0(x^c, y^c). \quad (28)$$

From (27), this implies that

$$\nu \leq f(x^c) \leq \nu + \varepsilon. \quad (29)$$

The proof of (a) is complete.

(b) If the algorithm is infinite, then it generates a sequence of incumbent solutions of the problem $EP(H^0)$, denoted by $\{(x^k, y^k)\}$, for each $k \geq 1$, (x^k, y^k) is obtained by solving the problem $LRP(H)$. For some $H^k \subseteq H^0$ and optimal solution $x^k \in D$, set

$$y_i^k = \frac{1}{\sum_{j=1}^n e_{ij}x_j^k + r_i}, \quad i \in \{1, 2, \dots, p\}. \quad (30)$$

Then the sequence $\{x^k\}$ consists of feasible solutions of the problem (GFP).

Suppose that \bar{x} is an accumulation point of $\{x^k\}$. Assume without loss of generality that $\lim_{k \rightarrow \infty} x^k = \bar{x}$. Since D is a compact set, $\bar{x} \in D$. Furthermore, because $\{x^k\}$ is infinite, we assume without loss of generality that, for each k , $H^{k+1} \subseteq H^k$, for some point $\bar{y} \in R^p$,

$$\lim_{k \rightarrow \infty} H^k = \bigcap_k H^k = \{\bar{y}\}. \quad (31)$$

Set $\bar{H} = \{\bar{y}\}$, for each k ; let $H^k = \{y \in R^p \mid L_i^k \leq y_i \leq U_i^k, i = 1, 2, \dots, p\}$. Since $H^{k+1} \subseteq H^k \subseteq H^0$, from Step 5, we know that $\{LB(H^k)\}$ is a nonincreasing sequence, and $\lim_{k \rightarrow \infty} LB(H^k)$ is a finite number and satisfies

$$\lim_{k \rightarrow \infty} LB(H^k) \leq \nu. \quad (32)$$

For each k , from Step 3, we know that $LB(H^k)$ is equal to the optimal value of the problem $LRP(H^k)$ and that x^k is an optimal solution of this problem. From (31), we have

$$\lim_{k \rightarrow \infty} L^k = \lim_{k \rightarrow \infty} U^k = \{\bar{y}\} = \bar{H}. \quad (33)$$

Since $\lim_{k \rightarrow \infty} x^k = \bar{x}$, $L_i^k \leq 1/(\sum_{j=1}^n e_{ij}x_j^k + r_i) \leq U_i^k$, and the continuity of $\sum_{i=1}^p e_{ij}x_j^k + r_i$,

$$\frac{1}{\sum_{j=1}^n e_{ij}x_j^k + r_i} = \bar{y}_i, \quad i = 1, 2, \dots, p. \quad (34)$$

This implies that (\bar{x}, \bar{y}) is a feasible solution of the problem $EP(H^0)$. Therefore,

$$\varphi_0(\bar{x}, \bar{y}) \geq \nu. \quad (35)$$

Together with (32), we have

$$\varphi_0(\bar{x}, \bar{y}) \geq \nu \geq \lim_{k \rightarrow \infty} LB(H^k). \quad (36)$$

Since the branching process is bisection and the branching process of rectangle is exhaustive, we have

$$\lim_{k \rightarrow \infty} LB(H^k) = \nu = \varphi_0(\bar{x}, \bar{y}). \quad (37)$$

Therefore, (\bar{x}, \bar{y}) is a global optimal solution of the problem $EP(H^0)$. By Theorem 1, this implies that \bar{x} is a global optimal solution of the problem (GFP). For each k , since x^k is the incumbent solution of the problem (GFP) at the end of step k , $UB_k = f(x^k)$; by the continuity of f , we obtain that

$$\lim_{k \rightarrow \infty} f(x^k) = f(\bar{x}). \quad (38)$$

Since \bar{x} is a global optimal solution of the problem (GFP),

$$f(\bar{x}) = \nu. \quad (39)$$

Therefore, $\lim_{k \rightarrow \infty} UB_k = \nu$. The proof is complete. \square

6. Numerical Experiment

The proposed algorithm is programmed in MATLAB 7.8 and is run in Pentium(R) 4 CPU 3.20 GHz. In order to compare with the algorithm of the literature [10], we perform three experiments to the literature [10].

Example 1 (see [10]). We choose $p = n = 2$; for each $(x_1, x_2) \in R^2$, the numerator and denominator are

$$\begin{aligned} n_1(x_1, x_2) &= 37x_1 + 73x_2 + 13, \\ n_2(x_1, x_2) &= 63x_1 - 18x_2 + 39, \\ d_1(x_1, x_2) &= 13x_1 + 13x_2 + 13, \\ d_2(x_1, x_2) &= 13x_1 + 26x_2 + 13, \end{aligned} \quad (40)$$

and all $(x_1, x_2) \in D$ satisfy

$$5x_1 - 3x_2 = 3, \quad 1.5 \leq x_1 \leq 3. \quad (41)$$

From our algorithm, we firstly should solve the following linear programming problems:

$$\begin{aligned} \min \quad & d_i(x), \\ \text{s.t.} \quad & Ax \leq b, \\ \max \quad & d_i(x), \\ \text{s.t.} \quad & Ax \leq b, \\ & i = 1, 2, \dots, p, \end{aligned} \quad (42)$$

of which the optimal solutions denote by x_i^1, x_i^2 ($i = 1, 2$); then

$$W = W \cup \{x_i^1, x_i^2 : i = 1, 2, \dots, p\}, \quad (43)$$

where W represent the set of the current feasible solution of the problem $EP(H^0)$, and the optimal value is denoted by \bar{l}_i and \bar{u}_i ($i = 1, 2$); then the initial rectangle is

$$H^0 = \begin{bmatrix} 0.0096 & 0.0192 \\ 0.0064 & 0.0140 \end{bmatrix}. \quad (44)$$

By solving the linear relaxation programming problem $LRP(H^0)$, we obtain the optimal solution $x^0 = [2.0016; 2.3360]$ and the optimal value $LB(H^0) = 3.9743$; then a lower bound of the original problem is $LB(H^0) = 3.9743$. Set

$$y_i^0 = \left(\sum_{j=1}^n e_{ij}x_j^0 + r_i \right)^{-1}. \quad (45)$$

Then (x^0, y^0) is a feasible solution of $EP(H^0)$, $\min \{\varphi_0(x^0, y^0), f(x) : x \in W\} = 4.9126$, then it provides an upper bound for the global optimal value of the problem $EP(H^0)$. Next, we choose the rectangle H^0 corresponding with the lower bound to branch; we obtain the following rectangles via our algorithm:

$$H^{0,1} = \begin{bmatrix} 0.0096 & 0.0144 \\ 0.0064 & 0.0140 \end{bmatrix}, \quad H^{0,2} = \begin{bmatrix} 0.0144 & 0.0192 \\ 0.0064 & 0.0140 \end{bmatrix}. \quad (46)$$

TABLE 1

	ε	Approximate optimal value	Optimal value
	0.01	4.9027	4.9126
Example 1	$1.0e - 3$	4.9116	4.9126
	$1.0e - 4$	4.9125	4.9126

We solve the linear relaxation programming problem LRP in rectangles $H^{0,1}$ and $H^{0,2}$, respectively. In $LRP(H^{0,1})$, the optimal solution and the optimal value are [2.2524; 2.7540] and $v = 4.2345$; then in rectangle $H^{0,1}$, the lower bound of the original problem is $LB(H^{0,1}) = 4.2345$, and the upper bound corresponding with the optimal solution is 4.9617 (>4.9126), so the upper bound is unchanged. In $LRP(H^{0,2})$, the optimal solution and the optimal value are [1.8019; 2.0032]; and $v = 4.5548$; then in rectangle $H^{0,2}$, the lower bound of the original problem is $LB(H^{0,2}) = 4.5548$, and the upper bound corresponding with the optimal solution is 4.9323 (>4.9126), so the upper bound is also unchanged. Then we choose the rectangle corresponding with the lower bound to branch until the 55th iteration, and we can obtain that

$$H^{55,1} = \begin{bmatrix} 0.0186 & 0.0189 \\ 0.0135 & 0.0137 \end{bmatrix}, \tag{47}$$

we solve the linear programming problem LRP in $H^{55,1}$; the lower bound is 4.9125; it satisfies the terminated rule. Therefore, the optimal value and the optimal solution of the original problem are 4.9126 and $x = [1.5000; 1.5000]$; the lower bound of the optimal value is 4.9125, which is approximate optimal value. The accuracy is $\varepsilon = 0.0001$.

The above example satisfies $(n, p) = (2, 2)$, where n denote the number of variables; our algorithm can have a good approach within accuracy. In Example 2, $(n, p) = (3, 3)$; in Example 3, $(n, p) = (3, 4)$ we still get good results. Along with the increase of n and p , the computation complexity is increasing. For example, in Example 3, $(n, p) = (3, 4)$, we can quickly obtain the approximate optimal value and the optimal value by using this paper's algorithm, but its effect is poorer than the former example. The result of Example 1 is shown in Table 1.

Example 2 (see [10]).

$$\begin{aligned} \min \quad & \frac{3x_1 + 5x_2 + 3x_3 + 50}{3x_1 + 4x_2 + 5x_3 + 50} + \frac{3x_1 + 4x_2 + 50}{4x_1 + 3x_2 + 2x_3 + 50} \\ & + \frac{4x_1 + 2x_2 + 4x_3 + 50}{5x_1 + 4x_2 + 3x_3 + 50}, \\ \text{s.t.} \quad & 2x_1 + x_2 + 5x_3 \leq 10, \\ & x_1 + 6x_2 + 2x_3 \leq 10, \\ & 9x_1 + 7x_2 + 3x_3 \geq 10, \\ & x_1, x_2, x_3 \geq 0. \end{aligned} \tag{48}$$

The optimal value is 2.8619.

Example 3 (see [10]).

$$\begin{aligned} \min \quad & \frac{4x_1 + 3x_2 + 3x_3 + 50}{3x_2 + 3x_3 + 50} + \frac{3x_1 + 4x_3 + 50}{4x_1 + 4x_2 + 5x_3 + 50} \\ & + \frac{x_1 + 2x_2 + 4x_3 + 50}{x_1 + 5x_2 + 5x_3 + 50} + \frac{x_1 + 2x_2 + 4x_3 + 50}{5x_2 + 4x_3 + 50}, \\ \text{s.t.} \quad & 2x_1 + x_2 + 5x_3 \leq 10, \\ & x_1 + 6x_2 + 3x_3 \leq 10, \\ & 9x_1 + 7x_2 + 3x_3 \geq 10, \\ & x_1, x_2, x_3 \geq 0. \end{aligned} \tag{49}$$

The optimal value is 3.7109.

We choose $\varepsilon = 1.0e - 4$; then the approximate optimal solution satisfying accuracy and the iteration times and CPU running time are obtained. The results of our algorithm are shown in Table 2. But the results of the literature [10] are shown in Table 3.

According to Tables 2 and 3, in Example 1, although the optimal solution $(3, 4)^T$ of the literature [10] is feasible, its optimal value 5 is bigger than 4.9126 of our algorithm; in Example 2, the optimal solution $(0, 3.3333, 0)^T$ of the literature [10] turns out to be infeasible; in Example 2, the optimal value 4.0000 which corresponds to the optimal solution $(0, 0.625, 1.875)^T$ of the literature [10] is actually 3.8384, but it is still bigger than 3.7109 of our algorithm.

From the above comparison we know that the optimal values of our algorithm are much lesser than in the literature [10], and except for Example 1, the iterations of Examples 2 and 3 are much lesser than in the literature [10]. Although our running time is longer than the literature [10], if we can solve the more accurate optimal solution, the price we pay is acceptable.

In conclusion, our algorithm is feasible and effective, and to some degree, it is better than in the literature [10].

7. Conclusion

In this paper, the solving of the sum of linear ratios programming problem is discussed. The problem is equivalently transformed into bilinear programming problem, then by using the linear characteristics of convex envelope and concave envelope of double variables product, the linear relaxation programming of the bilinear programming problem is given, which can determine the lower bound of the optimal value of original problem. Therefore, a branch and bound algorithm for solving sum of linear ratios programming problem is proposed and the convergence of the algorithm is proved. Numerical results show the effectiveness of the algorithm, and our algorithm is better than the calculation results of the literature [10].

TABLE 2

Example	The optimal solution within accuracy or one solution among solutions		
	x_1	x_2	x_3
1	1.5000	1.5000	
2	5.0000	0.0000	0.0000
3	0.0000	1.6667	0.0000
Example	Approximate optimal value	The number of iterations	CPU (s)
1	4.9125	113	201.626020
2	2.8619	12	28.294344
3	3.7087	5	4.190375

TABLE 3

Example	The optimal solution within accuracy or one solution among solutions		
	x_1	x_2	x_3
1	3	4	
2	0	3.3333	0
3	0	0.625	1.875
Example	Approximate optimal value	The number of iterations	CPU (s)
1	5	32	1.089285
2	3.0029	80	8.566259
3	4.0000	58	2.968694

Acknowledgments

The work is supported by the Foundation of National Natural Science, China (11161001), and by the research project of Beifang University of Nationalities (2013XYZ025).

References

- [1] H. Konno and H. Watanabe, "Bond portfolio optimization problems and their applications to index tracking: a partial optimization approach," *Journal of the Operations Research Society of Japan*, vol. 39, no. 3, pp. 295–306, 1996.
- [2] J. E. Falk and S. W. Palocsay, "Optimizing the sum of linear fractional functions," in *Recent Advances in Global Optimization (Princeton, NJ, 1991)*, Princeton Series in Computer Science, pp. 221–258, Princeton University Press, Princeton, NJ, USA, 1992.
- [3] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to Global Optimization*, vol. 48 of *Nonconvex Optimization and Its Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2nd edition, 2000.
- [4] H. Konno, Y. Yajima, and T. Matsui, "Parametric simplex algorithms for solving a special class of nonconvex minimization problems," *Journal of Global Optimization*, vol. 1, no. 1, pp. 65–81, 1991.
- [5] H. Konno and N. Abe, "Minimization of the sum of three linear fractional functions," *Journal of Global Optimization*, vol. 15, no. 4, pp. 419–432, 1999.
- [6] P.-P. Shen and C.-F. Wang, "Global optimization for sum of linear ratios problem with coefficients," *Applied Mathematics and Computation*, vol. 176, no. 1, pp. 219–229, 2006.
- [7] W. Yanjun, S. Peiping, and L. Zhian, "A branch-and-bound algorithm to globally solve the sum of several linear ratios," *Applied Mathematics and Computation*, vol. 168, no. 1, pp. 89–101, 2005.
- [8] H. P. Benson, "Solving sum of ratios fractional programs via concave minimization," *Journal of Optimization Theory and Applications*, vol. 135, no. 1, pp. 1–17, 2007.
- [9] H. W. Jiao and Q. G. Feng, "Global optimization for sum of linear ratios problem using new pruning technique," *Mathematical Problem in Engineering*, vol. 2008, Article ID 646205, 12 pages, 2008.
- [10] C.-F. Wang and P.-P. Shen, "A global optimization algorithm for linear fractional programming," *Applied Mathematics and Computation*, vol. 204, no. 1, pp. 281–287, 2008.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

