Math. Meth. Oper. Res. (2007) 65: 445–459
DOI 10.1007/s00186-006-0130-0

ORIGINAL ARTICLE

Shen Pei-Ping · Yuan Gui-Xia

Global optimization for the sum of generalized polynomial fractional functions

Received: 22 September 2005 / Accepted: 4 September 2006 / Published online: 11 January 2007

Abstract In this paper, a branch and bound approach is proposed for global optimization problem (P) of the sum of generalized polynomial fractional functions under generalized polynomial constraints, which arises in various practical problems. Due to its intrinsic difficulty, less work has been devoted to globally solving this problem. By utilizing an equivalent problem and some linear underestimating approximations, a linear relaxation programming problem of the equivalent form is obtained. Consequently, the initial non-convex nonlinear problem (P) is reduced to a sequence of linear programming problems through successively refining the feasible region of linear relaxation problem. The proposed algorithm is convergent to the global minimum of the primal problem by means of the solutions to a series of linear programming problems. Numerical results show that the proposed algorithm is feasible and can successfully be used to solve the present problem (P).

Keywords Global optimization · Generalized polynomial · Fractional function · Generalized polynomial constraint · Linear relaxation · Branch and bound

S. Pei-Ping (✉) · Y. Gui-Xia
College of Mathematics and Information Science,
Henan Normal University, Xinxiang 453007, China
E-mail: shenpeiping@163.com

1 Introduction

Consider the following global optimization problem of the sum of generalized polynomial fractional functions:

$$
(\mathrm{P}) \quad \begin{cases} \min & h(x) = \sum_{j=1}^{p} a_j \dfrac{n_j(x)}{d_j(x)} \\ \text{s.t.} & g_k(x) \le 0, \quad k = 1, \ldots, M, \\ & x \in X = \{x \mid 0 < \underline{x}_i \le x_i \le \overline{x}_i < \infty, \; i = 1, \ldots, N\}, \end{cases}
$$

where for each $j = 1, \ldots, p$ and each $k = 1, \ldots, M$,

$$
n_j(x) = \sum_{t=1}^{T_j^1} b_{jt}^1 \prod_{i=1}^{N} x_i^{\gamma_{jti}^1}, \quad d_j(x) = \sum_{t=1}^{T_j^2} b_{jt}^2 \prod_{i=1}^{N} x_i^{\gamma_{jti}^2}, \quad g_k(x) = \sum_{t=1}^{T_k^3} b_{kt}^3 \prod_{i=1}^{N} x_i^{\gamma_{kti}^3},
$$

and p, T_j^1, T_j^2, T_k^3 are all natural numbers; a_j, b_{jt}^1, b_{jt}^2 and b_{kt}^3 are all nonzero real constant coefficients; γ_{jti}^1, γ_{jti}^2 and γ_{kti}^3 are all nonzero real constant exponents.

So far, the global optimization problems of linear (or quadratic, polynomial) fractional functions, as a special case of problem (P), have attracted the interest of researchers and practitioners, and various specialized algorithms have been reported for solving these special types (for example, see Konno and Fukaisi 2000; Gotoh and Konno 2001). This is because, from a practical point of view, these problems have spawn a wide variety of applications, specially in transportation planning, government contracting, and finance and investment etc. (see Tuy et al. 2004; Phuong and Tuy 2003; Konno et al. 1997; Tuy 1998). In addition, from a research point of view, these problems pose significant theoretical and computational challenges. This is mainly due to the fact that these problems are global optimization problems, i.e., they are known to generally possess multiple local optima that are not global optima.

Several algorithms have been proposed for globally solving optimization problems of the nonlinear sum of fractional functions (Bensen 2002a,b). In the most considered problems, the feasible regions are polyhedrons or convex sets, or the considered problems have been limited to non-general problems (Phuong and Tuy 2003; Konno et al. 1997; Tuy 1998; Bensen 2002a,b; Konno and Abe 1995; Konno and Yamshita 1997). To our knowledge, there exist few algorithms for globally solving problem (P), where the feasible region is nonconvex set, and the objective function is the sum of ratios with real coefficients, and the expressions of the objective and constrained functions involve in generalized multivariable polynomials.

In this paper, we present a branch and bound algorithm for finding globally optimal solution of problem (P). The proposed algorithm works by solving problem (P2), which is equivalent to problem (P). In the algorithm, we use a convenient linearization techniques to systematically convert problem (P2) into a series of linear programming problems. The solutions of these converted problems can be as close as possible to the global optimum of problem (P) by a successive refinement process. The main computation involves in solving a series of linear programming problems over the partitioned subsets, for which some efficient and known methods are available. Numerical results show that the proposed algorithm is feasible.

This paper is organized as follows. In Sect. 2, by using Bernstein Algorithm (Nataray and Kotecha 2004; Berz and Hoffstatter 1998) and an exponential transformation, problem (P2) is derived that is equivalent to problem (P). Then a linear relaxation programming (LRP) is given by a linearization technique. In Sect. 3, we present a branch and bound algorithm for globally solving Problem (P2) and prove its convergence property. In Sect. 4, we show that the algorithm is feasible via several numerical examples. And finally, the summary of this paper is given.

2 Equivalent problem and its linear relaxation

2.1 Equivalent nonconvex program

In this subsection, we show how to convert problem (P) into an equivalent nonconvex programming problem. For convenience of the following discussions, without loss of generality, we let $a_j > 0\,(j = 1, \ldots, K)$ and $a_j < 0\,(j = K+1, \ldots, p)$ in (P).

In order to introduce an equivalent form of problem (P), we require that problem (P) satisfies the following additional assumption.

Assumption 1 For each $j = 1, \ldots, p$, suppose that

$$n_j(x) > 0,\;\; d_j(x) > 0, \quad \forall\, x \in X.$$

Based on Assumption 1, by using Bernstein Algorithm (Nataray and Kotecha 2004; Berz and Hoffstatter 1998) for each $j = 1, \ldots, p$, we can obtain positive scalars l_j, u_j, L_j and U_j such that

$$0 < l_j \le n_j(x) \le u_j, \quad 0 < L_j \le d_j(x) \le U_j, \quad \forall\, x \in X.$$

Next, let $H = \{(t, s) \in R^{2p} \mid l_j \le t_j \le u_j,\; L_j \le s_j \le U_j,\; j = 1, 2, \ldots, p\}$. Then we can get the following equivalent problem (P1):

$$\text{(P1):}\quad \begin{cases} \min & \displaystyle\sum_{j=1}^{K} a_j t_j s_j^{-1} + \sum_{j=K+1}^{p} a_j t_j s_j^{-1} \\ \text{s.t.} & n_j(x) - t_j \le 0,\;\; s_j - d_j(x) \le 0, \quad j = 1, \ldots, K, \\ & t_j - n_j(x) \le 0,\;\; d_j(x) - s_j \le 0, \quad j = K+1, \ldots, p, \\ & g_k(x) \le 0, \quad k = 1, \ldots, M, \\ & x \in X,\; (t, s) \in H. \end{cases}$$

The key equivalence result for problems (P) and (P1) is given by the following theorem.

Theorem 1 *x^* is a global optimal solution of problem* (P) *if and only if* (x^*, t^*, s^*) *is a global optimal solution of problem* (P1), *where $t_j^* = n_j(x^*)$ and $s_j^* = d_j(x^*)$ for each $j = 1, \ldots, p$.*

Proof Let x^* be a global optimal solution of problem (P), and let $t_j^* = n_j(x^*)$ and $s_j^* = d_j(x^*),\ j = 1, \ldots, p$. Then (x^*, t^*, s^*) is a feasible solution of problem (P1) with objective function value $h(x^*)$. Let (x, t, s) be any feasible solution of problem (P1), it is obvious that

$$a_j \frac{n_j(x)}{d_j(x)} \le a_j \frac{t_j}{s_j}, \quad j = 1, \ldots, K, K+1, \ldots, p.$$

This means that

$$h(x) = \sum_{j=1}^{K} a_j \frac{n_j(x)}{d_j(x)} + \sum_{j=K+1}^{p} a_j \frac{n_j(x)}{d_j(x)} \le \sum_{j=1}^{K} a_j \frac{t_j}{s_j} + \sum_{j=K+1}^{p} a_j \frac{t_j}{s_j}. \qquad (1)$$

By the optimality of x^*, we must have

$$\sum_{j=1}^{p} a_j \frac{t_j^*}{s_j^*} = h(x^*) \le h(x) \le \sum_{j=1}^{p} a_j \frac{t_j}{s_j}. \qquad (2)$$

Therefore, by $x^* \in X$ and $(t^*, s^*) \in H$, it follows that (x^*, t^*, s^*) is a global solution of problem (P1).

Conversely, suppose that (x^*, t^*, s^*) is a global optimal solution for problem (P1), then we have

$$0 < n_j(x^*) \le t_j^* \le u_j,\ 0 < s_j^* \le d_j(x^*) \le U_j, \quad j = 1, \ldots, K;$$
$$0 < l_j \le t_j^* \le n_j(x^*),\ 0 < L_j \le d_j(x^*) \le s_j^*, \quad j = K+1, \ldots, p.$$

This implies that

$$\frac{n_j(x^*)}{d_j(x^*)} \le \frac{t_j^*}{s_j^*} \quad \text{for } j = 1, \ldots, K, \quad \text{and} \quad \frac{n_j(x^*)}{d_j(x^*)} \ge \frac{t_j^*}{s_j^*} \quad \text{for } j = K+1, \ldots, p,$$

and so,

$$h(x^*) = a_j \frac{n_j(x^*)}{d_j(x^*)} \le a_j \frac{t_j^*}{s_j^*} \quad \text{for } j = 1, \ldots, p. \qquad (3)$$

For each $j = 1, \ldots, p$, let $\bar{t}_j = n_j(x^*),\ \bar{s}_j = d_j(x^*)$, then $(x^*, \bar{t}, \bar{s})$ is a feasible solution for problem (P1) with objective function value $h(x^*)$. Since (x^*, t^*, s^*) is a global optimal solution for problem (P1), we get that

$$h(x^*) = \sum_{j=1}^{p} a_j \frac{\bar{t}_j}{\bar{s}_j} \ge \sum_{j=1}^{p} a_j \frac{t_j^*}{s_j^*}\ . \qquad (4)$$

Combining (3) with (4), we have

$$h(x^*) = \sum_{j=1}^{p} a_j \frac{t_j^*}{s_j^*}. \qquad (5)$$

By definition of $h(x)$, since

$$\frac{n_j(x^*)}{d_j(x^*)} \le \frac{t_j^*}{s_j^*},\ 0 < n_j(x^*) \le t_j^*,\ 0 < s_j^* \le d_j(x^*),\quad j = 1, \ldots, K;$$
$$\frac{n_j(x^*)}{d_j(x^*)} \ge \frac{t_j^*}{s_j^*},\ 0 < t_j^* \le n_j(x^*),\ 0 < d_j(x^*) \le s_j^*,\quad j = K+1, \ldots, p,$$

which implies that $t_j^* = n_j(x^*)$ and $s_j^* = d_j(x^*)$ for each $j = 1, \ldots, p$, then for any feasible solution x for problem (P), if we set that $t_j = n_j(x)$, and $s_j = d_j(x)$, $j = 1, \ldots, p$, then (x, t, s) is a feasible solution for problem (P1) with objective function value $h(x)$. By the optimality (x^*, t^*, s^*) and feasibility of x, we have

$$h(x) \ge \sum_{j=1}^{p} a_j \frac{t_j^*}{s_j^*}.$$

Since $x^* \in X$, it follows from the above inequality that x^* is a global optimal solution for problem (P), and the proof is complete. □

For the problem (P1), since it possesses some particular structure, we can utilize an exponential variable transformation to obtain an equivalent problem (P2) of (P1). Let $y = (y^1, y^2, y^3) \in R^{2p+N}$ with $y^1 \in R^N$ and $y^2, y^3 \in R^p$, and let $x_i = \exp(y_i^1)$ for each $i = 1, \ldots, N$ and $s_j = \exp(y_j^2)$, $t_j = \exp(y_j^3)$ for each $j = 1, \ldots, p$. Then, without loss of generalization, we can rewrite the problem (P1) to the following problem:

$$(\mathrm{P2}): \begin{cases} \min & \Phi_0(y) \\ \text{s.t.} & \Phi_m(y) \le 0, \quad m = 1, \ldots, 2p+M, \\ & y \in \Omega = \{y \mid \underline{y}_n \le y_n \le \overline{y}_n,\ n = 1, \ldots, 2p+N\} \\ & \qquad = \{\ln \underline{x}_i \le y_i^1 \le \ln \overline{x}_i,\ i = 1, \ldots, N, \\ & \qquad\quad \ln l_j \le y_j^2 \le \ln u_j,\ \ln L_j \le y_j^3 \le \ln U_j,\ j = 1, \ldots, p\}, \end{cases}$$

where

$$\Phi_m(y) = \sum_{t=1}^{\Gamma_m} c_{mt} \exp\left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n\right) \triangleq \sum_{t=1}^{\Gamma_m} c_{mt} f_{mt}(y), \quad m = 0, 1, \ldots, 2p+M, \tag{6}$$

and for each m, t and n, corresponding with problem (P1), c_{mt} is a real constant coefficient; γ_{mtn} is a real constant exponent; Γ_m is a index set.

From Theorem 1, notice that, to search for a global solution for problem (P), we may globally solving problem (P1) instead, and it is easy to see that the global optimal values of problems (P1) and (P2) are equal. Hence, the branch and bound algorithm to be presented can be applied to problem (P2).

2.2 Linear relaxation programming

In the development of a solution procedure for solving problem (P2), the principal structure is how to construct the linear relaxation programming of problem (P2), which can provide the lower bounds of optimal value of problem (P2) in the branch and bound method to be presented. This linear relaxation programming can be obtained by using a linear underestimating function $L_m^{\Omega}(y)$ of the nonlinear function $\Phi_m(y)$ for each $m = 0, 1, \ldots, 2p + M$. Henceforth, for convenience, for any $y \in \Omega$ some notations are first introduced as follows:

$$Y_{mt}^{\Omega} = \sum_{n=1}^{2p+N} \gamma_{mtn} y_n;$$

$$\underline{Y}_{mt}^{\Omega} = \sum_{n=1}^{2p+N} \min(\gamma_{mtn} \underline{y}_n, \gamma_{mtn} \overline{y}_n);$$

$$\overline{Y}_{mt}^{\Omega} = \sum_{n=1}^{2p+N} \max(\gamma_{mtn} \underline{y}_n, \gamma_{mtn} \overline{y}_n),$$

where $m = 0, 1, \ldots, 2p + M$, $t = 1, \ldots, \Gamma_m$.

To illustrate how the function $L_m^{\Omega}(y)$ can be derived to obtain the linear relaxation problem, we consider the function $f_{mt}(y) = \exp(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n) = \exp(Y_{mt}^{\Omega})$ as defined in (6), for any $y \in \Omega$.

It is known that exponential function $\exp(Y)$ is a monotone increasing, convex function about the single variable Y. Let $F_{mt}^{\Omega}(y)$ ($H_{mt}^{\Omega}(y)$) denote a linear overestimating (underestimating) function of $f_{mt}(y)$ over Ω. Then, by the convexity of $\exp(Y_{mt}^{\Omega})$ on the interval $[\underline{Y}_{mt}^{\Omega}, \overline{Y}_{mt}^{\Omega}]$, we can formulate $F_{mt}^{\Omega}(y)$ as follows:

$$F_{mt}^{\Omega}(y) = A_{mt}^{\Omega} + B_{mt}^{\Omega} \left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n \right), \tag{7a}$$

where

$$A_{mt}^{\Omega} = (\overline{Y}_{mt}^{\Omega} \exp(\underline{Y}_{mt}^{\Omega}) - \underline{Y}_{mt}^{\Omega} \exp(\overline{Y}_{mt}^{\Omega}))/(\overline{Y}_{mt}^{\Omega} - \underline{Y}_{mt}^{\Omega}),$$

$$B_{mt}^{\Omega} = (\exp(\overline{Y}_{mt}^{\Omega}) - \exp(\underline{Y}_{mt}^{\Omega}))/(\overline{Y}_{mt}^{\Omega} - \underline{Y}_{mt}^{\Omega}).$$

Moreover, let $H_{mt}^{\Omega}(y)$ be a an affine function corresponding to a tangent hyperplane of the graph of $f_{mt}(y)$, which is parallel to $F_{mt}^{\Omega}(y)$. Then the point of tangential support will occur at $\tilde{Y}_{mt} = \ln B_{mt}^{\Omega}$, thus, by computing, $H_{mt}^{\Omega}(y)$ can be given as follows:

$$H_{mt}^{\Omega}(y) = B_{mt}^{\Omega}(1 - \ln B_{mt}^{\Omega}) + B_{mt}^{\Omega} \left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n \right). \tag{7b}$$

Consequently, combining (7a) with (7b), it follows that

$$H_{mt}^{\Omega}(y) \le f_{mt}(y) \le F_{mt}^{\Omega}(y), \quad \forall\, y \in \Omega.$$

Next, by using the above results, we can give the following linear relaxation programming of problem (P2). Letting $\Omega_q = [\underline{y}^q, \overline{y}^q] \subseteq \Omega$, and denoting the linear underestimating function of each nonlinear function $c_{mt} \exp(\sum_{n=1}^{2p+N} \gamma_{mtn}\, y_n)$ as $L_{mt}^{\Omega_q}(y)$, for any $y \in \Omega_q$ we have

$$c_{mt} \exp\left(\sum_{n=1}^{2p+N} \gamma_{mtn}\, y_n\right) \geq L_{mt}^{\Omega_q}(y) = \begin{cases} c_{mt} H_{mt}^{\Omega_q}(y), & \text{if } c_{mt} > 0, \\ c_{mt} F_{mt}^{\Omega_q}(y), & \text{if } c_{mt} < 0. \end{cases} \quad \forall\, m,\ t.$$

Thus, summing it over all the terms t ($t = 1, \ldots, \Gamma_m$) and denoting the resulting right-hand side $\sum_{t=1}^{\Gamma_m} L_{mt}^{\Omega_q}(y)$ in this sum as $L_m^{\Omega_q}(y)$, we easily see that

$$\Phi_m(y) \geq L_m^{\Omega_q}(y) = \sum_{t=1}^{\Gamma_m} L_{mt}^{\Omega_q}(y), \quad \forall\, y \in \Omega_q,\ m = 0, 1, \ldots, 2p + M. \tag{8}$$

Consequently, We construct the corresponding linear relaxation programming (LRP) of the problem (P2) on Ω_q as follows:

$$(\text{LRP}(\Omega_q)): \begin{cases} \min & L_0^{\Omega_q}(y) \\ \text{s.t.} & L_m^{\Omega_q}(y) \leq 0, \quad m = 1, \ldots, 2p + M \\ & y \in \Omega_q. \end{cases}$$

Notice that it follows immediately from (8) that every feasible point of problem (P2) in subdomain Ω_q is feasible for problem LPR(Ω_q), and the objective function value of (LPR) is less than or equal to that of problem (P2) for all points in Ω_q. Moreover, it should be noted that problem (LPR) contains only the necessary constraints to guarantee convergence of the algorithm.

Lemma 1 *The minimum of problem* LPR(Ω_q) *provides a lower bound of the global optimal value of problem* (P2) *over the partition set* Ω_q.

Proof This is obvious by the above construction method. □

Lemma 2 *Let* $\{\Omega_q\}$ *be any subsequence of rectangles such that* $\Omega_{q+1} \subset \Omega_q \subseteq R^{2p+N}$ *for any* q *and* $\bigcap_{q=1}^{\infty} \Omega_q = \{y^*\}$. *Consider the functions* $f_{mt}(y)$, $F_{mt}^{\Omega_q}(y)$ *and* $H_{mt}^{\Omega_q}(y)$ *for any* $y \in \Omega_q$. *Then it holds that*

$$\max_{y \in \Omega_q} |F_{mt}^{\Omega_q}(y) - f_{mt}(y)| = \max_{y \in \Omega_q} |H_{mt}^{\Omega_q}(y) - f_{mt}(y)| \to 0, \quad \text{as } q \to \infty,$$

where $m = 0, 1, \ldots, 2p + M$ *and* $t = 1, \ldots, \Gamma_m$.

Proof Let $\Omega_q = [\underline{y}^q, \overline{y}^q] \subseteq R^{2p+N}$. Then, by assumption of this Lemma, there exists a corresponding sequence $\{(\underline{y}^q, \overline{y}^q)\}$ with $\underline{y}^q = (\underline{y}_n^q)_{(2p+N)\times 1}$ and $\overline{y}^q = (\overline{y}_n^q)_{(2p+N)\times 1}$ such that

$$(\underline{y}^q, \overline{y}^q) \to (y^*, y^*), \quad \text{as } q \to \infty. \tag{9}$$

For any m, t, let $\omega_{mt}^q = \overline{Y}_{mt}^{\Omega_q} - \underline{Y}_{mt}^{\Omega_q}$ for each Ω_q. Then, by the definitions of $\overline{Y}_{mt}^{\Omega_q}$ and $\underline{Y}_{mt}^{\Omega_q}$, we get that from (9)

$$\omega_{mt}^q = \sum_{n=1}^{2p+N} |\gamma_{mtn}|(\overline{y}_n^q - \underline{y}_n^q) \to 0 \quad \text{as } q \to \infty. \tag{10}$$

Now, we consider two differences $F_{mt}^{\Omega_q}(y) - f_{mt}(y) \triangleq \Delta_{mt}^{\Omega_q}(y)$ and $f_{mt}(y) - H_{mt}^{\Omega_q}(y) \triangleq \Lambda_{mt}^{\Omega_q}(y)$ for any $y \in \Omega_q$. Then it follows from (6), (7a) and (7b) that

$$\begin{aligned}
\Delta_{mt}^{\Omega_q}(y) &= A_{mt}^{\Omega_q} + B_{mt}^{\Omega_q}\left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n\right) - \exp\left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n\right) \\
&= A_{mt}^{\Omega_q} + B_{mt}^{\Omega_q} Y_{mt}^{\Omega_q} - \exp(Y_{mt}^{\Omega_q}), \\
\Lambda_{mt}^{\Omega_q}(y) &= \exp\left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n\right) - B_{mt}^{\Omega_q}(1 - \ln B_{mt}^{\Omega_q}) - B_{mt}^{\Omega_q}\left(\sum_{n=1}^{2p+N} \gamma_{mtn} y_n\right) \\
&= \exp(Y_{mt}^{\Omega_q}) - B_{mt}^{\Omega_q}(1 - \ln B_{mt}^{\Omega_q} + Y_{mt}^{\Omega_q}).
\end{aligned}$$

Since $\Delta_{mt}^{\Omega_q}(Y_{mt}^{\Omega_q})$ is a concave function of the single variable $Y_{mt}^{\Omega_q}$ over $[\overline{Y}_{mt}^{\Omega_q}, \underline{Y}_{mt}^{\Omega_q}]$, $\Delta_{mt}^{\Omega_q}(Y_{mt}^{\Omega_q})$ can attain its maximum $\Delta_{\max}^q$ at the point $\ln B_{mt}^{\Omega_q}$, and through computing we can obtain that

$$\begin{aligned}
\Delta_{\max}^q &= A_{mt}^{\Omega_q} - B_{mt}^{\Omega_q} + B_{mt}^{\Omega_q} \ln B_{mt}^{\Omega_q} \\
&= \exp(\underline{Y}_{mt}^q) - B_{mt}^{\Omega_q}(\underline{Y}_{mt}^q + 1 - \ln B_{mt}^{\Omega_q}) = \exp(\underline{Y}_{mt}^q)(1 - u_{mt}^q + u_{mt}^q \ln u_{mt}^q),
\end{aligned}$$

where $u_{mt}^q = (\exp(\omega_{mt}^q) - 1)/\omega_{mt}^q$.

On the other hand, since $\Lambda_{mt}^{\Omega_q}(Y_{mt}^{\Omega_q})$ is a convex function of $Y_{mt}^{\Omega_q}$, for any $Y_{mt}^{\Omega_q} \in [\overline{Y}_{mt}^{\Omega_q}, \underline{Y}_{mt}^{\Omega_q}]$, it follows that its maximum, denoted by $\Lambda_{\max}^q$, will occur at the point $\overline{Y}_{mt}^{\Omega_q}$ or $\underline{Y}_{mt}^{\Omega_q}$. Note that

$$\begin{aligned}
\Lambda_{mt}^{\Omega_q}(\underline{Y}_{mt}^{\Omega_q}) &= \exp(\underline{Y}_{mt}^{\Omega_q}) - B_{mt}^{\Omega_q}(1 - \ln B_{mt}^{\Omega_q} + \underline{Y}_{mt}^{\Omega_q}) = \Delta_{\max}^q, \\
\Lambda_{mt}^{\Omega_q}(\overline{Y}_{mt}^{\Omega_q}) &= \exp(\overline{Y}_{mt}^{\Omega_q}) - B_{mt}^{\Omega_q}(1 - \ln B_{mt}^{\Omega_q} + \overline{Y}_{mt}^{\Omega_q}) \\
&= \exp(\underline{Y}_{mt}^q)(1 - u_{mt}^q + u_{mt}^q \ln u_{mt}^q).
\end{aligned}$$

Therefore, we have

$$\Delta_{\max}^q = \Lambda_{\max}^q = \exp(\underline{Y}_{mt}^q)(1 - u_{mt}^q + u_{mt}^q \ln u_{mt}^q).$$

Together with (10), from the above results we can follow that $u_{mt}^q \to 1$ and $\Delta_{\max}^q = \Lambda_{\max}^q \to 0$, as $q \to \infty$. This means that

$$\max_{y \in \Omega_q} |F_{mt}^{\Omega_q}(y) - f_{mt}(y)| = \max_{y \in \Omega_q} |H_{mt}^{\Omega_q}(y) - f_{mt}(y)| \to 0 \quad \text{as } q \to \infty,$$

and the proof is complete. □

Lemma 3 *Assumption as in Lemma 2. Then*

$$\max_{y\in\Omega_q} |L_m^{\Omega_q}(y) - \Phi_m(y)| \to 0 \quad as\ q \to \infty,$$

where $m = 0, 1 \ldots, 2p + M$.

Proof It follows immediately from Lemma 2 and the definitions of $L_m^{\Omega_q}(y)$ and $\Phi_m(y)$. □

3 Branch and bound algorithm and its convergence

To globally solve the problem (P2) which is equivalent to the primal problem (P), a branch and bound algorithm is developed based on the former linear relaxation programming. This algorithm needs to solve a sequence of linear programming over the initial rectangle Ω or partitioned subrectangle $\Omega^{q(k)}$ in order to search a global optimal solution of (P2). The basic idea of this method is to generate nonincreasing upper bounds and nondecreasing lower bounds until they approximate to enough closely each other. The proposed algorithm is based on partitioning Ω into subrectangles, and each is associated with a node of the branch and bound tree, and each node is associated with a LRP on the corresponding subrectangle. At any stage k ($k \geq 1$) of the algorithm, suppose that we have a collection of active nodes indexed by Q_k, say, each is associated with a rectangle $\Omega^{q(k)} \subseteq \Omega$, $\forall\, q(k) \in Q_k$. For each such node $q(k)$, we will have computed a lower bound $LB_{q(k)}$ via solution of LRP($\Omega^{q(k)}$), so that the lower bound of optimal value of problem (P2) at the stage k is given by $LB(k) = \min\{LB_{q(k)}, q(k) \in Q_k\}$. We now select an active node $q(k)$ such that $LB(k) = LB_{q(k)}$ for further considering. Then we partition the selected rectangles into two subrectangles according to the following branching rules. For these two subrectangles the feasibility checking are applied, in order to identify whether the subrectangles should be eliminated. Computing the lower bounds for each new undeleted nodes by solving the corresponding LRP as before. If necessary and possible, we will update the upper bound of incumbent solution V^*. Then, the active nodes collection Q_k will satisfy $LB_{q(k)} < V^*, \forall\, q(k) \in Q_k$, for each stage k. Upon fathoming any nonimproving nodes, we obtain a collection of active nodes for the next stage, and this process is repeated until convergence is obtained.

The critical element in guaranteeing convergence to a global minimum is the choice of a suitable branching rule. In this paper, we choose a simple and standard bisection rule. This method is sufficient to ensure convergence since it drives all the intervals to zero for the variables that are associated with the term that yields the greatest discrepancy in the employed approximation along any infinite branch of the branch and bound tree.

Branching rule:

Assume that a rectangle $\Omega^{q(k)} = \{y \mid \underline{y}_n^q \leq y_n^q \leq \overline{y}_n^q,\ n = 1, \ldots, 2p+N\} \subseteq \Omega$ is going to be partitioned. Then the selection of the branching variable y_λ which possesses the maximum length in $\Omega^{q(k)}$ and the partitioning of $\Omega^{q(k)}$ is done using the following rule. Let $\lambda = \arg\max\{\overline{y}_n^q - \underline{y}_n^q,\ n = 1, \ldots, 2p + N\}$, and partition

$\Omega^{q(k)}$ by bisecting the interval $[\underline{y}_\lambda^q, \overline{y}_\lambda^q]$ into the subintervals $[\underline{y}_\lambda^q, (\underline{y}_\lambda^q + \overline{y}_\lambda^q)/2]$ and $[(\underline{y}_\lambda^q + \overline{y}_\lambda^q)/2, \overline{y}_\lambda^q]$.

After these preparations we can now formulate our branch and bound algorithm. The basic steps of the algorithm are summarized in the following.

Algorithm statement:

Step 0 A convergence tolerance ϵ is selected, and the iteration counter k is set to zero, $Q_k = \{1\}$, $q(k) = 1$, $\Omega^{q(k)} = \Omega^1 = \Omega$. Set the initial upper $V^* = \infty$. Solve LRP($\Omega^{q(k)}$), and denote the corresponding optimal solution and optimal value by $(\hat{y}(\Omega^{q(k)}), LB_{q(k)})$, and let the initial lower bound $LB(k) = LB_{q(k)}$. If $\hat{y}(\Omega^{q(k)})$ is feasible for (P2), then update upper V^*. If $V^* - LB(k) \leq \epsilon$, then stop with $\hat{y}(\Omega^{q(k)})$ as the prescribed solution to problem (P2). Otherwise, proceed to Step 1.

Step 1 Choose a branching variable y_λ to partition $\Omega^{q(k)}$ to get two subrectangles $\Omega^{q(k)\cdot 1}$ and $\Omega^{q(k)\cdot 2}$ according to the selected branching rule. Replace $q(k)$ by these two new node indices $q(k)\cdot 1$ and $q(k)\cdot 2$ in Q_k.

Step 2 For each new node index $q(k)\cdot \nu$ where $\nu = 1, 2$, compute

$$\begin{aligned}\underline{\Phi}_m(\nu) = & \sum_{t=1,\ c_{mt}>0}^{\Gamma_m} c_{mt} \exp(\underline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}) \\ & + \sum_{t=1,\ c_{mt}<0}^{\Gamma_m} c_{mt} \exp(\overline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}) \quad \text{for } m = 1, \ldots, 2p + M,\end{aligned}$$

where c_{mt}, $\underline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}$ and $\overline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}$ have been defined in Sect. 2.2. If $\underline{\Phi}_m(\nu) > 0$ for some $m \in \{1, 2, \ldots, 2p + M\}$, that is, the problem (P2) is infeasible for $\Omega^{q(k)\cdot\nu}$, then the corresponding index $q(k)\cdot\nu$ will be eliminated from Q_k. If $\Omega^{q(k)\cdot\nu}$ $(\nu = 1, 2)$ are all been eliminated, then go to Step 4.

Step 3 For undeleted subrectangle update the corresponding parameters $A_{mt}^{\Omega^{q(k)\cdot\nu}}$, $B_{mt}^{\Omega^{q(k)\cdot\nu}}$, $\underline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}$ and $\overline{Y}_{mt}^{\Omega^{q(k)\cdot\nu}}$ as defined in Sect. 2.2. Solve problem LRP($\Omega^{q(k)\cdot\nu}$) where $\nu = 1$ or $\nu = 2$ or $\nu = 1, 2$, and denote the obtained optimal solutions and optimal values by $(\hat{y}(\Omega^{q(k)\cdot\nu}),\ LB_{q(k)\cdot\nu})$. Then if $\hat{y}(\Omega^{q(k)\cdot\nu})$ is feasible for (P2), update the upper bound $V^* = \min\{V^*,\ \Phi_0(\hat{y}(\Omega^{q(k)\cdot\nu}))\}$. If $LB_{q(k)\cdot\nu} > V^*$, then delete the corresponding node.

Step 4 Fathom any nonimproving nodes by setting $Q_{k+1} = Q_k - \{q(k) \in Q_k \mid LB_{q(k)} \geq V^* - \epsilon\}$. If $Q_{k+1} = \emptyset$ then stop, and V^* is the optimal value, $y^*(\kappa)$ with $\kappa \in \kappa_0$ is a global optimal solution, where $\kappa_0 = \{\kappa \mid V^* = \Phi_0(y^*(\kappa))\}$. Otherwise, $k = k + 1$.

Step 5 Set the lower bound $LB(k) = \min\{LB_{q(k)} \mid q(k) \in Q_k\}$, then select an active node $q(k) \in \arg\min\{LB_{q(k)}\}$ for further considering, and go to Step 1.

Next, we will give the convergence of the algorithm.

Theorem 2 *Suppose that problem* (P2) *has a global optimal solution, and let* Φ_0^* *be the global optimal value of* (P2). *Then*:

(i) (*For the case* $\epsilon > 0$): *The algorithm always terminates after finitely many iterations yielding a global* ϵ*-optimal solution* y^* *and a global* ϵ*-optimal value* V^* *for problem* (P2) *in the sense that*

$$y^* \in \Omega, \quad V^* - \epsilon \le \Phi_0^* \le V^* \text{ with } V^* = \Phi_0(y^*);$$

and

(ii) (*For the case* $\epsilon = 0$): *If the algorithm does not terminate after finitely many iterations, then every accumulation point of a infinite sequence* $\{y^k\}$ *generated by the algorithm is a global optimal solution for problem* (P2).

Proof (i) If the algorithm is finite, then it terminates after finitely many iterations $k,\ k \ge 0$. Upon termination, by solving problem LPR($\Omega^{q(k)}$) for some $\Omega^{q(k)} \subseteq \Omega$, we can find a feasible solution y^* and a upper bound V^* of Φ_0^* for problem (P2) satisfying $y^* \in \Omega,\ V^* = \Phi_0(y^*)$ and

$$LB_{q(k)} \ge V^* - \epsilon, \quad \forall\, q(k) \in Q_k.$$

By Lemma 1, it is easy to show by standard arguments for branch and bound algorithm that

$$LB_{q(k)} \le \Phi_0^*.$$

Since y^* is a feasible solution for problem (P2),

$$V^* = \Phi_0(y^*) \ge \Phi_0^*.$$

Taken together, the three previous statements imply that

$$V^* - \epsilon \le LB_{q(k)} \le \Phi_0^* \le V^*.$$

Therefore, $V^* - \epsilon \le \Phi_0^* \le V^*$.

(ii) Suppose that the algorithm is infinite. Then, since the bisection of rectangles is exhaustive [see Horst and Tuy 2003], by the branching rule the proposed branch and bound algorithm can always generate an infinite rectangle sequence $\{\Omega_k\}$ with $\Omega_k = [\underline{y}^k, \overline{y}^k]$ such that $\Omega_{k+1} \subseteq \Omega_k \subseteq R^{2p+N}$ and $\bigcap_{k=1}^{\infty} \Omega_k = \{y^*\}$ with $\Omega_1 = \Omega$. Let $\{y^k\}$ be a infinite sequence satisfying $y^k \in \Omega_k$ for any k, then we clearly have $\lim_{k\to\infty} y^k = y^*$. Note that Ω is a compact set, and so, without loss of generality, there exists a subsequence $\{y^l\}$ of $\{y^k\}$ such that y^l is the optimal solution of RLP(Ω_q) with $LB(l) = L_0^{\Omega_l}(y^l)$ and $\lim_{l\to\infty} y^l = y^*$. Since the nondecreasing sequence $\{LB(l)\}$ is bounded from above by $\min\limits_{y\in F} \Phi_0(y) = \Phi_0^*$, where F denotes the feasible region of problem (P2), then there exists a limit such that

$$\lim_{l\to\infty} LB(l) \triangleq LB \le \Phi_0^*. \tag{11}$$

In addition, for any given l, by $y^l \in \Omega_l$ we have

$$\Phi_m(y^l) - L_m^{\Omega_l}(y^l) \le \max_{y\in\Omega_l} |\Phi_m(y) - L_m^{\Omega_l}(y)|, \quad m = 1, \ldots, 2p + M. \tag{12}$$

Letting $\{y^*\} = \Omega^*$, and using Lemma 3, by taking the limit as $l \to \infty$ the above inequality (12) implies that

$$\Phi_m(y^*) \le L_m^{\Omega^*}(y^*) \le 0, \quad \text{for } m = 1, \ldots, 2p + N,$$

and hence, y^* is feasible for problem (P2). Moreover, by Lemma 3 and (11) it is clear that

$$\Phi_0(y^*) = \lim_{l\to\infty} \Phi_0(y^l) = \lim_{l\to\infty} L_0^{\Omega_l}(y^l) = \lim_{l\to\infty} LB(l) \le \Phi_0^*.$$

Therefore, y^* is a global optimization solution for problem (2), and the proof is complete. □

4 Numerical experiment

To verify the performance of the proposed global optimization algorithm, there exist at least two computational issues to be considered in the following.

The first one involves in computing the scalars l_j, u_j, L_j and U_j under Assumption 1. The Bernstein Algorithm (Nataray and Kotecha 2004; Berz and Hoffstatter 1998) has established an important tool for finding bounds on the range of multivariate polynomials, and so, these scalars l_j, u_j, L_j and U_j can be obtained on the given initial box X via Bernstein Algorithm such that $0 \le n_j(x) \le u_j$, $0 < L_j \le d_j(x) \le U_j$.

The second one concerns the lower bound computing process which is obtained by solving the linear relaxation programming (LRP) from Sect. 2. This linear programming can be solved by some existing efficient and known methods. Here we adopt the simplex algorithm to solve the linear relaxation programming, and so the complement of the proposed global optimization algorithm will depend upon the simplex algorithm.

The algorithm is coded with C++ and some test problems are implemented on a Celeron IV (1693 MHz) microcomputer. Numerical results show that the proposed algorithm can globally solve the problem (P).

Below we only describe some of these sample problems and the corresponding computational results.

Example 1

$$(\mathrm{P}) \quad \begin{cases} \min \quad h(x) = -\dfrac{-x_1^2 + 3x_1 - x_2^2 + 3x_2 + 3.5}{x_1 + 1} \\ \qquad\qquad\quad - \dfrac{x_2}{x_1^2 - 2x_1 + x_2^2 - 8x_2 + 20} \\ \text{s.t.} \quad 2x_1 + x_2 \le 6, \\ \qquad 3x_1 + x_2 \le 8, \\ \qquad -x_1 + x_2 \ge -1, \\ \qquad X = \{x : 1 \le x_1 \le 3,\ 1 \le x_2 \le 3\}, \end{cases}$$

For solving this problem, we firstly give the upper and lower bounds of $n_1(x)$, $n_2(x)$, $d_1(x)$ and $d_2(x)$. Clearly, the upper and lower bounds of $n_2(x)$ and $d_1(x)$

are $u_2 = 3,\ l_2 = 1$ and $U_1 = 4,\ L_1 = 2$, respectively. For all $x \in X$, by using the $(3, 3)$th Bernstein polynomial, $U_2 = 16$ and $L_2 = 4$ can be obtained, and by using the $(5, 5)$th Bernstein polynomial, we have $u_1 = 8.3,\ l_1 = 3.5$.

Secondly, with $\epsilon = 1.0\text{E-}8$, the algorithm found a global ϵ-optimal minimum $V^* = -4.060819161$ after 2638 iterations at the global ϵ-optimal solution $(x_1, x_2)^{\mathrm{T}} = (1.0,\ 1.743823132)$, and the CPU time of the algorithm is 16.23 s.

Example 2

$$
(\mathrm{P}) \begin{cases} \min \quad h(x) = a_1 \dfrac{-x_1^2 + 3x_1 + 2x_2^2 + 3x_2 + 3.5}{x_1 + 1} \\ \qquad\qquad\quad - a_2 \dfrac{x_2}{x_1^2 - 2x_1 + x_2^2 - 8x_2 + 20} \\ \text{s.t.} \quad 3x_1 + x_2 \le 8, \\ \qquad\ \ x_1 - x_1^{-1}x_2 \le 1, \\ \qquad\ \ 2x_1x_2^{-1} + x_2 \le 6, \\ \qquad\ \ X = \{x : 1 \le x_1 \le 3,\ 1 \le x_2 \le 3\}, \end{cases}
$$

where $a_1 = 0.25,\ a_2 = -1.75$.

Similar to Example 1, for this problem, the upper and lower bounds of $n_2(x)$, $d_1(x)$ and $d_2(x)$ are the same as Example 1. By using the $(3, 3)$th Bernstein polynomial, for $n_1(x)$ we can obtain $u_1 = 33.1666667$ and $l_1 = 8.5$ for all $x \in X$.

With $\epsilon = 1.0\text{E-}8$, the algorithm found the global ϵ-optimal minimum $V^* = 0.883868686$ after 420 iterations at the global ϵ-optimal solution

$$(x_1, x_2)^{\mathrm{T}} = (1.618033989,\ 1.0),$$

and the CPU time of the algorithm is 3.52 s.

Example 3

$$
(\mathrm{P}) \begin{cases} \min \quad h(x) = a_1 \frac{n_1(x)}{d_1(x)} + a_2 \frac{n_2(x)}{d_2(x)} \\ \text{s.t.} \quad 2x_1^{-1} + x_1x_2 \le 4, \\ \qquad\ \ x_1 + 3x_1^{-1}x_2 \le 5, \\ \qquad\ \ x_1^2 - 3x_2^3 \le 2, \\ \qquad\ \ X = \{x : 1 \le x_1 \le 3,\ 1 \le x_2 \le 3\}, \end{cases}
$$

where $a_1 = -1.35,\ a_2 = 12.99$, and

$$
\begin{aligned}
n_1(x) &= x_1^2x_2^{0.5} - 2x_1x_2^{-1} + x_2^2 - 2.8x_1^{-1}x_2 + 7.5, \\
d_1(x) &= x_1x_2^{1.5} + 1, \\
n_2(x) &= x_2 + 0.1, \\
d_2(x) &= x_1^2x_2^{-1} - 3x_1^{-1} + 2x_1x_2^2 - 9x_2^{-1} + 12.
\end{aligned}
$$

For this problem, we firstly apply to the equivalence transformation

$$
\begin{aligned}
\frac{n_1(x)}{d_1(x)} &= \frac{x_1x_2(x_1^2x_2^{0.5} - 2x_1x_2^{-1} + x_2^2 - 2.8x_1^{-1}x_2 + 7.5)}{x_1x_2(x_1x_2^{1.5} + 1)} \\
&= \frac{x_1^3x_2^{1.5} - 2x_1^2 + x_1x_2^3 - 2.8x_2^2 + 7.5x_1x_2}{x_1^2x_2^{2.5} + x_1x_2} \triangleq \frac{n_1'(x)}{d_1'(x)}.
\end{aligned}
$$

Similarly,

$$\frac{n_2(x)}{d_2(x)} = \frac{x_1 x_2^2 + 0.1 x_1 x_2}{x_1^3 - 3x_2 + 2x_1^2 x_2^3 - 9x_1 + 12 x_1 x_2} \triangleq \frac{n_2'(x)}{d_2'(x)}.$$

Then, for the upper and lower bounds of $d_1'(x)$ and $n_2'(x)$, we can easily obtain $U_1 = 149.2961148,\ L_1 = 2$, and $u_2 = 27.9,\ l_2 = 1.1$. For $n_2'(x)$, by using the $(4, 4)$th Bernstein polynomial, we have $U_2 = 601$ and $L_2 = 3$. In order to get u_1 and l_1, we let $x_2^{0.5} = \hat{x}_2$, then the given box $X = [1, 3] \times [1, 3]$ is transformed into $[1, 3] \times [1, \sqrt{3}]$, consequently, $u_1 = 246.113037$ and $l_1 = 4.7$ are given by using the $(4, 7)$th Bernstein polynomial.

With $\epsilon = 1.0\text{E} - 7$, the algorithm found the global ϵ-optimal minimum $V^* = -1.96149893$ after 15243 iterations at the global ϵ-optimal solution

$$(x_1, x_2)^{\mathrm{T}} = (2.698690670, 1.20758556),$$

and the CPU time of the algorithm is 130.62 s.

5 Concluding remarks

In this paper, a global optimization algorithm is presented for solving the sum of generalized polynomial ratios problem (P) on a nonconvex feasible region, which arises in various engineering design problems. In the algorithm an equivalent problem (P2) is introduced firstly, then a linear relaxation programming is presented based on linear underestimating functions of the objective and constraint functions of problem (P2). Hence the lower bounding subproblems are linear programming problems that can be solved by some existing efficient and known methods. These characteristics offer computational advantages that can enhance the efficiencies of the proposed algorithm. The algorithm was applied to several test problems, the convergence of the global minimum was achieved in all case.

It is hoped in practice, the proposed algorithm and the ideas used in this paper will offer some valuable tools for solving the sum of nonlinear ratios problems on a nonconvex feasible region.

Acknowledgments The authors are grateful to the responsible editor and the anonymous referees for their valuable comments and suggestions, which have greatly improved the earlier version of this paper. This research is supported by the National Natural Science Foundation of China under Grant 10671057, and the Henan Province Soft Science Research Program (No. 0513030920).

References

Konno H, Fukaisi K (2000) A branch and bound algorithm for solving low rank linear multiplicative and fractional programming problems. J Global Optim 18:283–299

Gotoh JK, Konno H (2001) Maximization of the ratio of two convex quadratic functions over a polytope. Comput Optim Appl 20:43–60

Tuy H, Thach PT, Konno H (2004) Optimization of polynomial fractional functions. J Global Optim 29:19–44

Konno H, Thach PT, Tuy H (1997) Optimization on low rank nonconvex structures. Kluwer, Dordrecht
Tuy H (1998) Convex analysis and global optimization. Kluwer, Dordrecht
Phuong NTH, Tuy H (2003) A unified monotonic approach to generalized linear fractional programming. J Global Optim 26:229–259
Bensen HP (2002a) Using concave envelopes to globally solve the nonlinear sum of ratios problem. J Global Optim 22:343–364
Bensen HP (2002b) Globally optimization algorithm for the nonlinear sum of ratios problem. J Optim Theory Appl 112(1):1–29
Konno H, Abe N (1995) Minimization of the sum of three linear fractional functions. J Global Optim 15:419–432
Konno H, Yamshita H (1997) Minimization of the sum and the product of several linear fractional functions. Tokyo Institute of Technology, Technical report
Nataray PSV, Kotecha K (2004) Global optimization with higher order inclusion function forms, part 1: a combined Taylor–Bernstein form. Reliable Comput 1:27–44
Berz M, Hoffstatter G (1998) Computation and application of Taylor polynomials with interval remaider bounds. Reliable Comput 4:83–97
Horst R, Tuy H (2003) Global optimization: deterministic approaches, 3rd edn. Springer, Berlin Heidelberg New York