

# Solving the sum-of-ratios problem by a stochastic search algorithm

Wei-Ying Wu · Ruey-Lin Sheu · Ş. İlker Birbil

Received: 17 April 2006 / Accepted: 2 February 2008 / Published online: 15 May 2008  
© Springer Science+Business Media, LLC. 2008

**Abstract** In spite of the recent progress in fractional programming, the sum-of-ratios problem remains untoward. Freund and Jarre proved that this is an NP-complete problem. Most methods overcome the difficulty using the deterministic type of algorithms, particularly, the branch-and-bound method. In this paper, we propose a new approach by applying the stochastic search algorithm introduced by Birbil, Fang and Sheu to a transformed image space. The algorithm then computes and moves sample particles in the  $q - 1$  dimensional image space according to randomly controlled interacting electromagnetic forces. Numerical experiments on problems up to sum of eight linear ratios with a thousand variables are reported. The results also show that solving the sum-of-ratios problem in the image space as proposed is, in general, preferable to solving it directly in the primal domain.

**Keywords** Sum-of-ratios problems · Min-max problems · Dinkelbach-type method · Branch-and-bound method · Stochastic search method

## 1 Introduction

Consider the sum-of-ratios problem of the following form:

$$(P) \quad \min_{x \in D} \sum_{s=1}^q \frac{f_s(x)}{g_s(x)},$$

where  $f_s: \mathbb{R}^n \rightarrow \mathbb{R}$  and  $g_s: \mathbb{R}^n \rightarrow \mathbb{R}$  are continuous on  $D$  and  $g_s(x) > 0$ ,  $\forall x \in D$ ,  $s = 1, 2, \dots, q$ . We assume that  $D \subset \mathbb{R}^n$  is compact and connected. Problem (P) arises in many

---

W.-Y. Wu  
Department of Statistics and Probability, Michigan State University, East Lansing, MI 48824, USA

R.-L. Sheu (✉)  
Department of Mathematics, National Cheng Kung University, Tainan 701, Taiwan  
e-mail: rsheu@mail.ncku.edu.tw

Ş. İ. Birbil  
Faculty of Engineering and Natural Sciences, Sabancı University, Orhanlı-Tuzla, 34956 Istanbul, Turkey

**Table 1** Recent numerical results on the sum-of-ratios problems

Reference	$q$	$n$	Ratio type	Approach
[7]	2	2	Linear	Concave minimization (image sp.)
[16]	8	10	Linear	$B\&B$ (image sp.)
[18]	12	40	Linear	$B\&B$ (image sp.)
[21]	7	10	Linear	Monotonic optimization (image sp.)
[5]	2	2	Nonlinear	$B\&B$ (image sp.)
[6]	3	4	Nonlinear	$B\&B$ (domain sp.)
[12]	1	50	Nonlinear	Interior point method (image sp.)

different contexts. For example, Almqvist and Levin [1] analyzed a multistage stochastic shipping problem, whose model has a deterministic equivalence to (P). Konno and Inori [17] formulated a bond portfolio optimization problem in the form of (P). Other applications can be found in Frenk and Schaible [11].

When  $q = 1$ , (P) is a single-ratio problem. In this case, if  $f_1$  is nonnegative convex and  $g_1$  positive concave, then  $f_1/g_1$  is strictly quasi-convex with a unique optimum (see Schaible [2, 3]). When  $q \geq 2$ , even for the simplest case in which  $f_s, g_s$  are affine functions, problem (P) is a global optimization problem that may have multiple local optima [23]. In particular, Freund and Jarre [12] proved that (P) is NP-complete, indicating that an efficient algorithm may not exist. So far, most studies have focused on the linear ratios (see [7, 15–18]), although there are a few studies on the nonlinear ratios (see [4–6, 12]). Table 1 above lists the recent literature and the largest problem solved therein.

As we can see from the table, the branch-and-bound approach is the most popular. It divides either the image space of  $f_s$  and/or  $g_s$ , or divides the domain of (P) into several parts, while solving linear(nonlinear) programs to obtain the bounds. Because of the combinatorial nature of the branch-and-bound approach, the computational complexity grows exponentially as the dimension of the problem becomes larger. Therefore, it is quite difficult to handle a high dimensional feasible region or go beyond the sum of 10 linear ratios.

In contrast to deterministic algorithms, we propose to use a method equipped with a random mechanism. The term “randomness” does not mean to find the global optimum totally by chance. Indeed, most probabilistic algorithms adopt deterministic policies to direct the local search, whereas repeatedly random sampling is used only to escape non-global local optima. Commonly used stochastic algorithms include simulated annealing [14], evolutionary methods [20], multilevel methods [22], and partitioning methods [24]. Most methods are not just heuristics but have a proof for convergence in probability.

In this paper, we are particularly interested in the performance of Birbil, Fang and Sheu’s stochastic algorithm [9] for solving (P). The feature is to mimic the basic electromagnetism theory of physics to produce some sort of interacting forces among a population of charged particles. For this reason, it is called the Electromagnetism-like mechanism (EM in short). Basically, the EM method draws a finite population of sample particles and move them within a given working space. The particles iterate only according to their objective function values, but not relating to any linearity, convexity, or differentiability of  $f_s, g_s$ . Naturally, the working space for EM can be just  $D$  itself, or some specially designed image space of  $f_s, g_s$ . In view that a typical large scale problem always has a large  $n$  whereas  $q$  is in general less than 50, we shall try to work the EM method in the image space and call it the EM image space approach.

To this end, we propose to replace  $f_s/g_s$  with an independent variable  $\gamma_s$  for all but the first ratio  $f_1/g_1$ . In other words, we set  $\gamma_s = f_s/g_s$  for  $s = 2, 3, \dots, q$  and determine the range of each  $\gamma_s$  by solving single ratio problems. Then, the EM method is applied to the  $(q - 1)$ -dimensional space of  $\gamma_2, \gamma_3, \dots, \gamma_q$ . When a point is sampled, its associate function value can be computed by minimizing the remaining ratio  $f_1(x)/g_1(x)$ . In this way, we decompose (P) into a series of single ratio problems each of which is solved by the Dinkelbach algorithm. The motivation for doing this lies in the fact that the Dinkelbach procedure for solving a single ratio problem is exactly the Newton method [13]. The theoretical derivation of the EM image space approach constitutes the first part of the paper.

Numerically, although the convergence of EM is asymptotic, against the well-known nonlinear optimization solver KNITRO equipped with the multistar feature, our examples demonstrate that, for large scale sum-of-linear-ratios problems with  $n \gg q$ , the EM image space approach outperforms KNITRO with multistar. Finally, our numerical examples suggest that the EM image space approach is, in general, preferable to applying EM directly on  $D$  (the EM domain space approach). The numerical results constitute the second part of the paper.

## 2 EM review

In this section, we summarize important results of EM (for details, see [9]). The method looks for the global optimal solutions of a generic nonlinear function over a box constraint. Specifically, it takes the following form:

$$\begin{aligned} \min f(x) \\ \text{s.t. } x \in S, \end{aligned}$$

where  $f: R^n \rightarrow R$  is a nonlinear function and

$$S = \{x \in R^n \mid -\infty < l_k \leq x_k \leq u_k < \infty, k = 1, \dots, n\}$$

is a hyper-cube(rectangular) in  $R^n$ .

Suppose we select  $m$  sample particles  $\{1, 2, \dots, m\}$  whose positions in  $S$  are  $x^1, \dots, x^m$ , respectively. Let  $best$  be an integer in  $\{1, 2, \dots, m\}$  that marks the particle with the smallest object value and its position with  $x^{best}$ .

To mimic the electromagnetism theory, imagine that the particle  $i$  is charged by an amount  $Q^i$  as follows:

$$Q^i = \exp\left(-n \frac{f(x^i) - f(x^{best})}{\sum_{i=1}^n f(x^i) - f(x^{best})}\right). \tag{1}$$

In this way, each  $Q^i$  is normalized to have a value in  $(0, 1]$  and higher charges always imply smaller  $f$ -values. Given a pair of charged particles  $i$  and  $j$  and assuming that  $f(x^i) < f(x^j)$ , we denote the interacting force  $F^{(i,j)}$  as

$$F^{(i,j)} = (x^i - x^j) \frac{Q^i Q^j}{\|x^i - x^j\|^2} \tag{2}$$

and define

$$F^{(j,i)} = -F^{(i,j)}.$$

In other words,  $F^{(i,j)}$  always points from a particle having a relatively higher objective value to the other. The magnitude of  $F^{(i,j)}$  is proportional to the product of their charges, and also inversely proportional to the square of the distance. The total force  $F^i$  exerted on particle  $i$  can then be computed by adding all component forces vectorially:

$$F^i = \sum_{j \neq i}^m \tau_j F^{(i,j)}, \tag{3}$$

where  $\tau_j = 1$  if  $f(x_i) < f(x_j)$ , and  $\tau_j = -1$  if  $f(x_i) \geq f(x_j)$ . But there are two exceptions in the algorithm. First, at each iteration, the particle  $best$  never moves. That is,  $F^{best} = 0$ . Secondly, the farthest particle away from  $best$ , say particle  $p$ , has a positive probability to reverse component force. Specifically, choose any

$$x^p \in \arg \max \{ \|x^{best} - x^i\|, \quad i = 1, 2, \dots, m \}.$$

Then,

$$F^p = \sum_{j \neq p}^m (-1)^{n_j} \tau_j F^{(i,j)} \tag{4}$$

where  $n_j, j = 1, \dots, \hat{p}, \dots, m$ , is a Bernoulli random variable with a success probability  $\mu \in (0, 1)$  such that

$$n_j = \begin{cases} 0, & \text{with probability } 0 < \mu < 1; \\ 1, & \text{with probability } 1 - \mu. \end{cases}$$

Using  $F^i$  as the moving direction, the particle  $i$  moves to its new location

$$x^i = x^i + \lambda_i \frac{F^i}{\|F^i\|}, \tag{5}$$

where  $\lambda_i$  is a random step size to keep  $x^i$  in the box. After arriving at the new positions, the objective values and the associated charges of all particles are re-evaluated. Then, the process starts afresh. If we define  $Y_k$  to be the  $m$  positions at iteration  $k$ , then  $Y_k$  is a random variable whose state space is  $S^m = S \times S \dots \times S$ . Consequently, the family of random variables  $\{Y_k : k = 0, 1, 2 \dots\}$  constitutes a stochastic process. Since the location of the particles at the next iteration relies only on their current positions, it is a time homogeneous Markov chain.

Birbil et al. [9] showed that, under mild assumptions, EM converges to global optimum with probability one. Let  $x^*$  be a global minimum solution of  $\min\{f(x) : x \in S\}$ . Given  $\varepsilon > 0$ , define the  $\varepsilon$ -neighborhood of  $x^*$  by

$$B_\varepsilon^* = \{x \in S \mid |f(x) - f(x^*)| \leq \varepsilon\}.$$

Also, define  $\chi_{B_\varepsilon^*}(x)$  as the characteristic function

$$\chi_{B_\varepsilon^*}(x) = \sum_{i=1}^m 1_{B_\varepsilon^*}(x^i).$$

They have the following assumptions:

1.  $\mu(B_\varepsilon^*) > 0$ , where  $\mu$  is the Lebesgue measure on  $R^n$ .
2.  $f : S \rightarrow R$  is a lower bounded measurable function with respect to the Lebesgue measure  $\mu$ .

- 3. The collection of  $m$  ( $m > n$ ) vectors generated by the algorithm at every iteration has a full affine rank, i.e.,

$$\text{rank}(\{x^1, x^2, \dots, x^m\}) = n.$$

**Theorem 2.1** *Under the above assumptions, EM converges with probability one, i.e.,*

$$\lim_{k \rightarrow \infty} P\{\chi_{B_\varepsilon^*}(Y_k) \neq 0\} = 1.$$

### 3 Theoretical results

In this section, we address the EM image space approach for solving (P). Our idea is to fix  $q - 1$  ratios at a parametric vector  $\gamma \in R^{q-1}$  and to solve the remaining single-ratio problem with a set of  $q - 1$  constraints.

Let the image space of the last  $q - 1$  ratios be

$$I_{q-1} = \left\{ \left( \frac{f_2(x)}{g_2(x)}, \dots, \frac{f_q(x)}{g_q(x)} \right) \mid x \in D \right\}$$

and

$$\gamma = (\gamma_2, \dots, \gamma_q) \in I_{q-1}.$$

Define

$$\begin{aligned} (P_\gamma) \quad & \min_{x \in D} \frac{f_1(x)}{g_1(x)} + \gamma_2 + \dots + \gamma_q \\ & \text{s.t.} \quad \frac{f_s(x)}{g_s(x)} = \gamma_s, \quad \forall s = 2, \dots, q. \end{aligned} \tag{6}$$

Equivalently,  $(P_\gamma)$  can be written as

$$\begin{aligned} T(\gamma) = \min_{x \in D_\gamma} \quad & \frac{f_1(x)}{g_1(x)} + \gamma_2 + \dots + \gamma_q \\ \text{s.t.} \quad & D_\gamma = \{x \in D \mid f_s(x) - \gamma_s g_s(x) = 0, \quad \forall s = 2, \dots, q\}. \end{aligned} \tag{7}$$

This is a single ratio problem with  $q - 1$  constrains. The following theorem asserts that problem (P) can be solved by minimizing  $T(\gamma)$  over  $I_{q-1}$ , i.e.,

$$\begin{aligned} (P') \quad & \inf T(\gamma) \\ & \text{s.t.} \quad \gamma \in I_{q-1}. \end{aligned} \tag{8}$$

**Theorem 3.1** *The optimal values of (P) and (P') are the same.*

*Proof* Let  $x^*$  and  $M$  be the optimal solution and the optimal value of (P), respectively. Likewise, let  $M_1$  be the optimal value of (P'). Since

$$\left( \frac{f_2(x^*)}{g_2(x^*)}, \dots, \frac{f_q(x^*)}{g_q(x^*)} \right) = \gamma^* \in I_{q-1},$$

we have,

$$\begin{aligned}
 M_1 &\leq T(\gamma^*) \\
 &= \min_{x \in D_{\gamma^*}} \frac{f_1(x)}{g_1(x)} + \gamma_2^* + \dots + \gamma_q^* \\
 &\leq \frac{f_1(x^*)}{g_1(x^*)} + \gamma_2^* + \dots + \gamma_q^* \\
 &= M.
 \end{aligned}
 \tag{9}$$

On the other hand, given  $\varepsilon > 0$ , there is some  $\gamma^\varepsilon = (\gamma_2^\varepsilon, \dots, \gamma_q^\varepsilon) \in I_{q-1}$  such that

$$T(\gamma^\varepsilon) \leq M_1 + \varepsilon. \tag{10}$$

Since  $\frac{f_s(x)}{g_s(x)}, \forall s = 2, \dots, q$ , are continuous and  $D_{\gamma^\varepsilon}$  is compact, we can find  $x_{\gamma^\varepsilon} \in D_{\gamma^\varepsilon} \subset D$  that minimizes  $(P_{\gamma^\varepsilon})$ . Then,

$$\begin{aligned}
 T(\gamma^\varepsilon) &= \frac{f_1(x_{\gamma^\varepsilon})}{g_1(x_{\gamma^\varepsilon})} + \gamma_2^\varepsilon + \dots + \gamma_q^\varepsilon \\
 &= \frac{f_1(x_{\gamma^\varepsilon})}{g_1(x_{\gamma^\varepsilon})} + \frac{f_2(x_{\gamma^\varepsilon})}{g_2(x_{\gamma^\varepsilon})} + \dots + \frac{f_q(x_{\gamma^\varepsilon})}{g_q(x_{\gamma^\varepsilon})} \\
 &\geq M.
 \end{aligned}
 \tag{11}$$

Combining (10) and (11) to obtain

$$M \leq M_1 + \varepsilon, \quad \forall \varepsilon > 0.$$

Let  $\varepsilon \rightarrow 0$  and by (9), we prove that  $\gamma^*$  is a limit point of  $\{\gamma^\varepsilon\}$ ;  $x^*$  is a limit point of  $\{x_{\gamma^\varepsilon}\}$ ; and

$$M = M_1.$$

□

In practice, optimization problems with inequality constraints are usually easier than those with equality constraints. Therefore, we further replace  $D_\gamma$  with

$$\Delta_\gamma = \{x \in D \mid f_s(x) - \gamma_s g_s(x) \leq 0, s = 2, \dots, q\},$$

and consider for  $\beta = (\beta_2, \beta_3, \dots, \beta_q) \in I_{q-1}$  the following problem:

$$\begin{aligned}
 (\Phi_\beta) \quad &\min_{x \in \Delta_\beta} \frac{f_1(x)}{g_1(x)} \\
 &s.t. \quad \Delta_\beta = \{x \in D \mid f_s(x) - \beta_s g_s(x) \leq 0, \forall s = 2, \dots, q\}.
 \end{aligned}
 \tag{12}$$

Let  $x_\beta$  be an optimal solution of  $(\Phi_\beta)$  and define

$$\Theta(\beta) = \sum_{s=1}^q \frac{f_s(x_\beta)}{g_s(x_\beta)}. \tag{13}$$

Then, we consider

$$(P'') \quad \inf_{\beta \in I_{q-1}} \Theta(\beta).$$

**Theorem 3.2** *The optimal values of (P') and (P'') are the same.*

*Proof* Let  $M_1$  and  $M_2$  be the optimal value of (P') and (P''), respectively. Since

$$D_\beta \subseteq \Delta_\beta, \quad \forall \beta \in I_{q-1},$$

this implies

$$\Theta(\beta) \leq T(\beta).$$

Thus

$$M_2 = \inf_{\beta \in I_{q-1}} \Theta(\beta) \leq \inf_{\beta \in I_{q-1}} T(\beta) = M_1. \tag{14}$$

Conversely, given  $\varepsilon > 0$ , there exists some  $\beta_\varepsilon \in I_{q-1}$  such that

$$\Theta(\beta_\varepsilon) \leq M_2 + \varepsilon. \tag{15}$$

By the same arguments as in the proof of Theorem 3.1, we can find  $x_{\beta_\varepsilon} \in \Delta_{\beta_\varepsilon} \subset D$  such that

$$\Theta(\beta_\varepsilon) = \sum_{s=1}^q \frac{f_s(x_{\beta_\varepsilon})}{g_s(x_{\beta_\varepsilon})}.$$

This implies

$$M \leq \Theta(\beta_\varepsilon). \tag{16}$$

By the Theorem 3.1, (16) and (15),

$$M_1 = M \leq \Theta(\beta_\varepsilon) \leq M_2 + \varepsilon. \tag{17}$$

Letting  $\varepsilon$  tend to 0, we show that

$$M = M_1 = M_2.$$

□

Although we have proved (P) and (P'') have the same optimal value,  $I_{q-1}$  is only compact, connected but may be not a box. However, it is not difficult to find a prescribed hyper-cube that covers  $I_{q-1}$ . Let

$$L_s = \min_{x \in D} \frac{f_s(x)}{g_s(x)}, \quad \forall s = 2, \dots, q;$$

$$U_s = \max_{x \in D} \frac{f_s(x)}{g_s(x)}, \quad \forall s = 2, \dots, q.$$

Then,

$$B = B_2 \times \dots \times B_q \supseteq I_{q-1}$$

with  $B_s = [L_s, U_s], s = 2, \dots, q$ . To modify (P'') as an optimization problem over  $B$  rather than just over  $I_{q-1}$ , we choose a large constant  $\bar{M} \gg M$  and let  $\Theta(\beta) = \bar{M}$  for all  $\beta \in B \setminus I_{q-1}$  such that

$$\Delta_\beta = \{x \in D \mid f_s(x) - \beta_s g_s(x) \leq 0, \forall s = 2, \dots, q\} = \emptyset.$$

Then define

$$\psi(\beta) = \begin{cases} \Theta(\beta), & \text{if } \beta \in I_{q-1}; \\ \overline{M}, & \text{if } \Delta_\beta = \emptyset; \\ \frac{\sum_{s=1}^q f_s(x_\beta)}{g_s(x_\beta)}, & x_\beta \text{ solves } (\Phi_\beta), \text{ if } \beta \in B \setminus I_{q-1}, \Delta_\beta \neq \emptyset. \end{cases} \tag{18}$$

Consider

$$(P''') \quad \inf_{\beta \in B} \psi(\beta).$$

**Theorem 3.3**  $(P''')$  and  $(P'')$  have the same optimal value.

*Proof* Let  $M_3$  be the optimal value of  $(P''')$ . By the definition of  $\psi(\beta)$ , if  $\Delta_\beta = \emptyset$ , then

$$\psi(\beta) = \overline{M} > M.$$

Moreover, if  $\beta \in B \setminus I_{q-1}$ ,  $\Delta_\beta \neq \emptyset$ , then  $x_\beta \in \Delta_\beta \subset D$  and

$$\psi(\beta) = \sum_{s=1}^q \frac{f_s(x_\beta)}{g_s(x_\beta)} \geq M.$$

By Theorems (3.1) and (3.2), the minimum value of  $\Theta(\beta)$  over  $I_{q-1}$  is  $M$ . Since  $\psi(\beta) = \Theta(\beta)$  for  $\beta \in I_{q-1}$ , we thus know that the minimum of  $\psi(\beta)$  over  $B$  must occur on  $I_{q-1}$ . This proves

$$M = M_3. \tag{□}$$

### 4 The algorithm

We have seen that solving (P) amounts to solving  $(P''')$  in the image space  $I_{q-1} = \left\{ \left( \frac{f_2(x)}{g_2(x)}, \dots, \frac{f_q(x)}{g_q(x)} \right) \mid x \in D \right\}$ . The objective function  $\psi(\beta)$  of  $(P''')$ , defined by the minimization problem  $(\Phi_\beta)$  in (12), is nonconvex on  $B = B_1 \times \dots \times B_s \times \dots \times B_{q-1}$  where  $B_s = [L_s, U_s]$ ,  $s = 1, \dots, q - 1$  and

$$L_s = \min_{x \in D} \frac{f_s(x)}{g_s(x)}; \tag{19}$$

$$U_s = \max_{x \in D} \frac{f_s(x)}{g_s(x)}. \tag{20}$$

To speed up the computation, we deal with only a sub-interval  $[l_s^k, u_s^k] \subset [L_s, U_s]$  at iteration  $k$ . Let

$$B^k = [l_2^k, u_2^k] \times \dots \times [l_q^k, u_q^k]$$

and

$$(P_k''') \quad \inf_{\beta \in B^k} \psi(\beta), \tag{21}$$

where  $l_s^k, u_s^k$  are the  $k$ -th parameters in the Dinkelbach algorithm for solving  $L_s$  and  $U_s$ , respectively. The first step of the Dinkelbach algorithm requires a point  $\underline{x}^0 \in D$  with which



one computes  $u_s^0 = l_s^0 = \frac{f_s(x^0)}{g_s(x^0)}$ , for all  $s = 2, \dots, q$ . Therefore,  $B^0$  is just a singleton  $(l_2^0, \dots, l_q^0)$ . For other  $B^k, k = 1, 2, \dots$ , compute  $l_s^k$  and  $u_s^k$  separately by solving parametric subproblems, for  $s = 2, \dots, q$ ,

$$\min_{x \in D} \{f_s(x) - l_s^{k-1} g_s(x)\} = f_s(\underline{x}_s^k) - l_s^{k-1} g_s(\underline{x}_s^k) \tag{22}$$

and

$$\max_{x \in D} \{f_s(x) - u_s^{k-1} g_s(x)\} = f_s(\bar{x}_s^k) - u_s^{k-1} g_s(\bar{x}_s^k). \tag{23}$$

Then, let

$$l_s^k = \frac{f_s(\underline{x}_s^k)}{g_s(\underline{x}_s^k)},$$

and

$$u_s^k = \frac{f_s(\bar{x}_s^k)}{g_s(\bar{x}_s^k)}.$$

By this way,  $\{l_s^k\}$  decrease monotonically to  $L_s$  and  $\{u_s^k\}$  increase monotonically to  $U_s$  such that  $B^0 \subset B^1 \dots \subset B^k \dots$  and eventually  $(P_k''')$  coincides with  $(P''')$  in the limit.

For each  $(P_k''')$ , we select  $m$  sample particles  $\{\beta_k^1, \beta_k^2, \dots, \beta_k^m\} \subset B^k$  each of which is a  $q - 1$  dimensional vector. Evaluating  $\psi(\beta_k^i)$  for each particle requires to check whether

$$\Delta_{\beta_k^i} = \{x \in D \mid f_s(x) - (\beta_k^i)_s g_s(x) \leq 0, \forall s = 2, \dots, q\} \tag{24}$$

is an empty set. If  $f_s$  and  $g_s$  are affine, this problem can be solved by the phase-one of linear programming. Otherwise, we determine by the following sequence of optimization problems. First, let

$$C_q = \min_{x \in D} f_q(x) - (\beta_k^i)_q g_q(x). \tag{25}$$

We have the following lemma.

**Lemma 4.1** *The optimal value  $C_q \leq 0$ .*

*Proof* Clearly,  $(\beta_k^i)_q \in [L_q, U_q]$ . Given that  $D$  is compact and  $\frac{f_q(x)}{g_q(x)}$  is continuous on  $D$ , there exists  $x' \in D$  such that

$$L_q = \min_{x \in D} \frac{f_q(x)}{g_q(x)} = \frac{f_q(x')}{g_q(x')} \leq (\beta_k^i)_q.$$

Since  $g_q(x) > 0$  for all  $x \in D$ , we have

$$f_q(x') - (\beta_k^i)_q g_q(x') \leq 0,$$

which shows that  $C_q \leq 0$ . □

Let  $x_q$  be the optimal solution of (25). By  $C_q \leq 0$ , we can use  $x_q$  as an initial point to solve

$$\begin{aligned} C_{q-1} &= \min_{x \in D} f_{q-1}(x) - (\beta_k^i)_{q-1} g_{q-1}(x) \\ & \text{s.t. } f_q(x) - \beta_k g_q(x) \leq 0. \end{aligned} \tag{26}$$

and let  $x_{q-1}$  be the optimal solution of (26). If  $C_{q-1} > 0$ , then  $\Delta_{\beta_k^i} = \emptyset$ . Otherwise, we use  $x_{q-1}$  as an initial point to solve

$$C_{q-2} = \min_{x \in D} f_{q-2}(x) - (\beta_k^i)_{q-2} g_{q-2}(x) \quad \text{s.t.} \quad f_{q-1}(x) - \beta_{q-1} g_{q-1}(x) \leq 0; f_q(x) - \beta_q g_q(x) \leq 0. \tag{27}$$

In general, we consider, for  $v = 1, 2, \dots, q - 2$ ,

$$C_{q-v} = \min_{x \in D} f_{q-v}(x) - (\beta_k^i)_{q-v} g_{q-v}(x) \quad \text{s.t.} \quad \{x | f_\mu(x) - \beta_\mu g_\mu(x) \leq 0, \mu = q - v + 1, \dots, q\}. \tag{28}$$

and let  $x_{q-v}$  be the optimal solution of (28). If at any stage  $C_{q-v} > 0$ ,  $\Delta_{\beta_k^i} = \emptyset$ . Otherwise,  $x_{q-v}$  will be the initial solution for solving  $C_{q-v-1}$ .

**Lemma 4.2** *The set  $\Delta_{\beta_k^i}$  is non-empty if and only if every  $C_\mu, \mu = 2, 3, \dots, q - 1$  is non-positive.*

By starting from any  $x \in D$ , we can use MATLAB to solve for  $C_q, C_{q-1}, \dots, C_2$  sequentially. The procedure is especially designed for MATLAB because it requires to input an initial feasible solution when solving a non-linear programming.

Theorem 2.1 implies that the algorithm must converge asymptotically, but does not show how soon the convergence can be observed. To prevent from terminating the algorithm prematurely, our stopping rule consists of three stages. Among the  $m$  particles in the image space  $I_{q-1}$ , the one with the smallest  $\psi$  value is called the *best* particle. If it is kept unchanged for  $K_1$  consecutive iterations, the algorithm enters the second stage which will randomly sample  $a_1$  additional particles. After a few iterations, if the best one remains unchanged for  $K_2 - K_1$  consecutive times, the algorithm shifts into the final stage with an additional  $a_2$  particles. The algorithm comes to a complete stop when the best particle has not moved for  $K_3 - K_2$  consecutive times. Applying this scheme, we can start with a smaller set of sample particles, and gradually increase the sample size. Moreover, the total number of iterations cannot exceed a preset parameter *Maxiter*. The execution time is thus controlled by the parameters  $K_1, K_2, K_3$  and *Maxiter*.

#### 4.1 EM image space algorithm for solving (P)

Step 1. Select  $x_0 \in D$  and calculate

$$u_s^0 = l_s^0 = \frac{f_s(x_0)}{g_s(x_0)}, \quad \forall s = 2, \dots, q.$$

Set  $k = 1, k_1 = 0, l = 0$ . Choose four positive integers for *Maxiter* and  $K_1, K_2, K_3$ .

Step 2. Determine the search range  $B^k = [l_2^k, u_2^k] \times \dots \times [l_q^k, u_q^k]$  by solving

$$\min_{x \in D} \{f_s(x) - l_s^{k-1} g_s(x)\} = f_s(\underline{x}_s^k) - l_s^{k-1} g_s(\underline{x}_s^k)$$

and

$$\max_{x \in D} \{f_s(x) - u_s^{k-1} g_s(x)\} = f_s(\bar{x}_s^k) - u_s^{k-1} g_s(\bar{x}_s^k).$$

Then compute

$$l_s^k = \frac{f_s(\underline{x}_s^k)}{g_s(\underline{x}_s^k)}, \quad \forall s = 2, \dots, q,$$

and

$$u_s^k = \frac{f_s(\bar{x}_s^k)}{g_s(\bar{x}_s^k)}, \quad \forall s = 2, \dots, q.$$

Step 3. Select randomly  $m$  sample points  $\{\beta_k^1, \beta_k^2, \dots, \beta_k^m\} \subset B^k$ . For  $i = 1, \dots, m$ ,

Step 3.1. compute  $\psi(\beta_k^i)$  by (18);

Step 3.2. find

$$\beta_k^{best} \in \operatorname{argmin}\{\psi(\beta_k^i) : i = 1, \dots, m\},$$

$$\beta_k^p \in \operatorname{argmax}\{\|\beta_k^{best} - \beta_k^i\| : i = 1, \dots, m\};$$

Step 3.3. compute the charges  $Q_k^i$  by

$$Q_k^i = \exp\left(-n \frac{\psi(\beta_k^i) - \psi(\beta_k^{best})}{\sum_{i=1}^m \psi(\beta_k^i) - \psi(\beta_k^{best})}\right);$$

Step 3.4. calculate the electromagnetic force  $F_k^i, i \neq p$  by

$$F_k^{(i,j)} = (\beta_k^i - \beta_k^j) \frac{Q_k^i Q_k^j}{\|\beta_k^i - \beta_k^j\|^2},$$

$$F_k^i = \sum_{j \neq i}^m \tau_j F_k^{(i,j)},$$

where  $\tau_j = 1$  if  $\psi(\beta_k^i) < \psi(\beta_k^j)$ , and  $\tau_j = -1$  if  $\psi(\beta_k^i) \geq \psi(\beta_k^j)$ . For  $i = p$ ,

$$F_k^p = \sum_{j \neq p}^m (-1)^{n_j} \tau_j F_k^{(i,j)},$$

where  $n_j$ 's are Bernoulli random variables such that

$$n_j = \begin{cases} 0, & \text{with prob. } 0 < \mu < 1; \\ 1, & \text{with prob. } 1 - \mu. \end{cases} \quad j = 1, \dots, \hat{p}, \dots, m.$$

Step 3.5. Move  $\beta_k^i, i \neq best$  to

$$\beta_k^i = \beta_k^i + \lambda_k^i \frac{F_k^i}{\|F_k^i\|}$$

where  $\lambda_k^i$  is a random step size to keep  $\beta_k^i$  in  $B^k$ .

Step 4. If  $k < Maxiter$  and  $l + k_1 < K_3$ ,

Step 4.1. set  $k = k + 1$ .

Step 4.2. if  $\beta_k^{best} = \beta_{k-1}^{best}$ ,

set  $k_1 = l + k_1 + 1$ . Otherwise, set  $k_1 = 0$ .

- Step 4.3. if  $k_1 = K_1$ , set  $l = K_1$ ,  $k_1 = 0$  and set  $m = m + a_1$ ;  
 Step 4.4. if  $k_1 = K_2$ , set  $l = K_2$ ,  $k_1 = 0$  and set  $m = m + a_2$ ;  
 Step 4.5. go to Step 2.

## 5 Numerical examples

To test the numerical performance of the EM image space approach, we have compiled a set of test problems involving linear as well as nonlinear ratios. The implementation was done in MATLAB 7.2 on Intel Core Duo 1.66GHz Notebook with 1.5G memory. The main computation effort for minimizing the sum of  $q$  ratios with a set of  $m$  sample points is to solving  $m$  single-ratio problems each with  $q - 1$  constraints by the Dinkelbach method, a Newton-type algorithm whose convergence is expected to be very fast. The Dinkelbach procedure is implemented by MATLAB subroutine fmincon. To compare our results, we solve the same set of test problems using the nonlinear optimization solver KNITRO with multistart option, and as the modeling language, we use AMPL.

*Example 5.1* In this example, we give a set of 16 fractional functions  $F_1, \dots, F_{16}$  and consider the sum of the first four, six, eight,  $\dots$ , sixteen of them. They are

$$\begin{aligned}
 F_1 &= \frac{x_1 + 0.5x_2 + x_3 + 5}{x_1 + 2x_2 + 3x_3 + 5}; & F_2 &= \frac{2x_1 - x_3 - 5}{4x_1 + 2x_2 + x_3 + 5}; \\
 F_3 &= \frac{x_1 - 0.2x_2 + 3x_3 + 5}{x_1 + 5x_2 + x_3 + 1}; & F_4 &= \frac{x_1 - 2x_2 - 4x_3 + 1}{x_2 + 3x_3 + 5}; \\
 F_5 &= \frac{-x_1 + x_2 - x_3}{2x_2 + 3x_3 + 1}; & F_6 &= \frac{x_1 - 4x_2 - 2x_3 + 2}{-x_1 + 2x_2 + x_3 + 11}; \\
 F_7 &= \frac{2x_1 + 2x_2 - 5x_3 + 1}{3x_1 + x_2 + x_3 + 1}; & F_8 &= \frac{x_1 + x_2 + 2x_3 + 1}{x_1 + 2x_2 + x_3 + 1}; \\
 F_9 &= \frac{x_1 - 3x_2 + x_3 + 2}{x_1 + x_3 + 3}; & F_{10} &= \frac{x_1 + 3x_2 + x_3 + 2}{2x_1 + x_2 + x_3 + 1}; \\
 F_{11} &= \frac{-x_1 - 3x_2 + x_3}{x_1 + 3x_2 + x_3 + 1}; & F_{12} &= \frac{-x_1 + 5x_2 - x_3}{3x_1 + x_2 + x_3 + 6}; \\
 F_{13} &= \frac{3x_1 - x_2 + x_3 - 1}{x_1 + 2x_2 + 2}; & F_{14} &= \frac{-x_2 + 2x_3 + 1}{2x_1 + 4x_2 + x_3 + 3}; \\
 F_{15} &= \frac{-x_1 - 2x_2 + x_3 + 3}{3x_1 + 3x_2 + x_3 + 2}; & F_{16} &= \frac{-3x_1 - x_2 + x_3 - 2}{x_1 + 2x_2 + 6x_3 + 1}.
 \end{aligned}$$

This example also has a constraint set  $D$  as follows:

$$D = \{0 \leq x \in R^3 \mid 2x_1 + x_2 + x_3 \leq 10, x_1 + 6x_2 + 3x_3 \leq 10, \\
 x_1 + 3x_2 + 2x_3 \leq 10, x_1 + 2x_2 + x_3 \leq 10\}.$$

The purpose of Example 5.1 is to verify the validity of the EM image space approach. With a small scale problem in three variables, we were allowed to confirm the results more easily. The nature of the example, however, forces our method to work in a higher dimensional space than  $D$ . As expected, the results in Table 2 show that the EM image space approach, though correctly worked out most of the problems except for the last sum-of-16-ratios, is in general slower than KNITRO.

The sum-of-16-ratios problem is quite tricky. It has two local minima, one at (1.3157, 1.4471, 0) with the value 2.3106 and the other at (5, 0, 0) with the optimal value 1.4867.

**Table 2** EM image space approach for Example 5.1 with parameters  $m = 10, \mu = 0.45, (K_1, K_2, K_3, K_4) = (5, 8, 15, \infty), (a_1, a_2) = (5, 5), Maxiter = 30$

$q$	EM image space approach		KNITRO (multistart)	
	Obj. value	Run time (s)	Obj. value	Run time (s)
4	0.23617	30.29	0.23609	0
6	0.0275	27.64	0.02750	0
8	-0.6513	25.43	-0.65126	0
10	1.4083	16.54	1.40828	0
12	1.6128	25.37	1.61279	0
14	2.2427	17.375	2.24275	0
16	2.4813	29.4688	2.310583	0
*16	1.6348	7091	(1.48671)	(0.156)

The former is a non-global local minimum which has a very large basin of attraction, constituting a region at least 95% of  $D$ . It can be visualized by noting both local minima have the third component 0 so that we can project the graph of the function onto the domain  $\{(x_1, x_2, 0) \mid 0 \leq x_1 \leq 5, 0 \leq x_2 \leq 5/3\}$ , which then gives Fig. 1. Applying directly the local solver like MATLAB or KNITRO failed to escape the large basin (the bottom right of the graph). Incorporating KNITRO with multistart option, however, reached  $(5, 0, 0)$  after a couple of trials. This solution is shown in the last row of Table 2.

For this problem, the EM image space approach had to work on a 15-dimensional space. We let EM run for 100 iterations, increase the probability of reversing the search directions to 0.95, and used up to 600 sample particles at a time. In spite of the efforts, the solution (when converted back to  $D$ ) merely escaped the wrong basin but the solution quality, suffering from a much higher dimensional approach, was not as good. See Table 2 row (\*16). We also believe that the cause to the poor performance of the EM image space should be on the dimension, not on the large basin because the EM domain space approach (applying EM directly on  $D$ , see next section) worked out the global optimal solution in 4.29s (last row in Table 6).

*Example 5.2* In this example, we consider the sum of two nonlinear ratios,  $F_1$  and  $F_2$ :

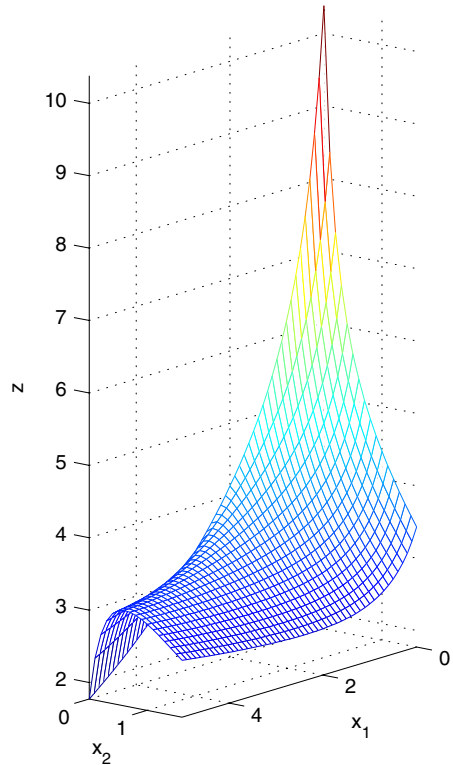
$$F_1 = \frac{\cos\left(\sqrt{x_1^2 + 3x_2^2}\right) + x_2}{\cos(x_1) + 3x_2^2 + 2}; \quad F_2 = \frac{\cos(x_1) + x_2^2 + x_2}{x_1^2 + x_2 + 1},$$

with the constraint  $D$  defined by

$$D = \{(x_1, x_2) \in R^2 \mid x_1 + 2x_2 \leq 6, 2x_1 - x_2 \leq 1, x_1, x_2 \geq 0\}.$$

The function  $\cos\left(\sqrt{x_1^2 + 3x_2^2}\right)$  has a difficult spiral contour and the objective function  $F_1 + F_2$  has at least two local minima on  $D$ . Starting from  $(0.2, 0.1)$ , MATLAB found a non-global local minimum at  $(0.5, 0)$  with a value 1.00703. Both the EM image space approach and KNITRO with multistart option had managed to escape  $(0.5, 0)$  and arrived at the global optimal solution  $(0.9361, 0.8722)$  (see Table 3).

**Fig. 1** The graph of the sum-of-16-ratios on the domain  $\{(x_1, x_2, 0) \mid 0 \leq x_1 \leq 5, 0 \leq x_2 \leq 5/3\}$  where a local minimum is located at  $(1.3157, 1.4471, 0)$  and the global solution is at  $(5, 0, 0)$



**Table 3** EM image space approach for Example 5.2 with parameters  $m=3, (K_1, K_2, K_3) = (2, 2, 5), (a_1, a_2) = (3, 5), Maxiter = 50, \mu = 0.75$

EM image sp. approach		KNITRO (multistart)		Matlab	
Obj. value	Run time	Obj. value	Run time	Obj. value	Run time
0.9467	25.84	0.9467	0.016	1.00703	0.063

*Example 5.3* In this example, the EM image space approach was tested on some large scale sum-of-linear-ratios problems. The test problems minimize  $\sum_{s=1}^q \frac{c_s^t x + p_s}{d_s^t x + q_s}$  subject to  $D = \{x \mid Ax \leq b, x \geq 0\}$  with  $c_s, d_s \in \mathbb{R}^n; p_s, q_s \in \mathbb{R}$  for  $s = 1, 2, \dots, q$  and  $A \in \mathbb{R}^p \times \mathbb{R}^n; b \in \mathbb{R}^p$ . In particular, vectors  $d_s$  are non-negative and  $q_s$  are positive so that the denominators are strictly positive over the constraint. In this example,  $p$  is fixed at 5,  $n$  is up to 1000 and the components in all vectors were randomly generated in MATLAB format. They can be retrieved from the website: <http://www.math.ncku.edu.tw/english/rlsheu.html>.

Due to the scale of the problems, we incorporated the EM image space approach with a local search strategy as in [8]. It allows the current best point  $\beta^{best}$  to search for a better point around a  $\delta$ -neighborhood of  $\beta^{best}$ . The search is done along each coordinate direction for a number of trials specified by *LSITER* (stands for “local search iterations”). The parameters  $\delta$  and *LSITER* are predefined parameters. A replacement is made for  $\beta^{best}$  once there is a better solution found. This local search, introduced by Birbil and Fang [8], does not utilize

**Table 4** Comparison of EM image space approach and MATLAB for large scale problems

$(q, n)$	EM image space approach		Matlab	
	Obj. value	Run time (s)	Obj. value	Run time (s)
(2, 200)	-10.9583	28.06	-10.9647	9.37
(4, 200)	-9.1297	20.89	-3.0332	8.4
(6, 200)	-4.4149	22.8	-2.5102	10.28
(8, 200)	-4.1672	57.56	-3.6906	5.5
(2, 1000)	-1.377	38.56	Not available over 2 hours	
(4, 1000)	0.9788	102.68	Not available over 2 hours	
(6, 1000)	1.2779	1147	1.27	1666
(8, 1000)	2.7051	629.2	2.7011	1639

Parameters used by EM image space approach in Example 5.3

$(q, n)$	$m$	$(K_1, K_2, K_3)$	$(a_1, a_2)$	Maxiter	$\nu$	$\delta$	LSITER
(2, 200)	4	(3, 3, 3)	(3, 5)	50	0.56	0.25	5
(4, 200)	5	(2, 2, 0)	(4, 0)	50	0.56	0.45	5
(6, 200)	5	(2, 2, 0)	(4, 5)	50	0.56	0.45	5
(8, 200)	3	(2, 2, 3)	(3, 3)	50	0.56	0.5	5
(2, 1000)	4	(2, 2, 0)	(2, 1)	50	0.56	0.45	2
(4, 1000)	6	(2, 4, 0)	(2, 2)	50	0.56	0.5	5
(6, 1000)	7	(2, 8, 12)	(2, 2)	50	0.56	0.45	10
(8, 1000)	6	(2, 4, 4)	(5, 5)	50	0.56	0.7	20

any first order information, hence is flexible and easy to implement. It improves the solution quality but is time consuming. It is worthwhile especially in the final stage of large scale problems.

We first solved each problem by the EM image space approach with the local search, and then by MATLAB subroutine *fmincon*. Since  $n \gg q$ , our EM image space approach works fine. The results and their comparison are given in Table 4. These testing problems, ranging from  $(q, n) = (2, 200)$  to  $(q, n) = (8, 1000)$ , are by far the largest in the literature. Moreover, many of them have multiple local minima. For  $(q, n) = (2, 200)$ , the EM image space approach and MATLAB found the same local minimum but MATLAB was quicker. For  $(q, n) = (4, 200)$ ,  $(6, 200)$ ,  $(8, 200)$ , MATLAB was trapped by a local minimum while our EM approach was able to escape. For  $(q, n) = (2, 1000)$  and  $(q, n) = (4, 1000)$ , MATLAB failed to find a local minimum over 2 h. For  $(q, n) = (6, 1000)$  and  $(q, n) = (8, 1000)$ , both methods approached the same solution. MATLAB had a slightly better solution quality but spent longer time.

The free version of AMPL allows the user to run mathematical programs for up to 300 variables. We thus apply KNITRO with multistart option whenever  $n = 200$ . It solved the same problem for 200 times (default value) using 200 randomly generated start points. See Table 5 for the results.

Unlike Examples 5.1–5.2 where the problem dimensions were low, even with the multistart option KNITRO failed to escape the non-global local minima in this example. One way to understand the phenomenon is to consider an optimization problem whose feasible domain

**Table 5** Comparison of EM image space approach and KNITRO with multistart for  $n = 200$ 

$(q, n)$	EM image space approach		KNITRO (multistart)	
	Obj. value	Run time (s)	Obj. value	Run time (s)
(2, 200)	-10.9583	28.06	-10.96434 (-10.96458)	0.281 (47.734)
(4, 200)	-9.1297	20.89	-3.03288 (-3.03318)	0.234 (43.281)
(6, 200)	-4.4149	22.80	-2.51021 (-2.51172)	0.203 (38.969)
(8, 200)	-4.1672	57.56	-3.69056 (-3.69062)	0.234 (42.531)

is an  $n$ -dimensional cube each side of which is 1, and suppose the basin of the attraction for the global optimum is an  $n$ -dimensional sub-cube with each side 0.9. Then, a randomly generated point has a probability of  $(0.9)^n$  to fall inside the basin. When  $n = 200$ , the probability is  $7.06 \times 10^{-10}$  with which KNITRO with multistart option is almost impossible to generate a point inside the basin in 200 trials.

## 6 EM domain space approach

In this section, we use some numerical examples to compare the EM image space approach with the EM domain space approach. To apply EM on  $D$ , we first extend the feasible domain  $D$  into a hyper-cube  $B = [l_1, u_1] \times \cdots \times [l_n, u_n]$  where  $l_i = \min_{x \in D} x_i$  and  $u_i = \max_{x \in D} x_i$ . Define

$$\bar{f}(x) = \begin{cases} \sum_{s=1}^q \frac{f_s(x)}{g_s(x)}, & \text{if } x \in D; \\ \bar{M} \text{ (a large constant)}, & \text{if } x \in B - D. \end{cases}$$

Then we can apply EM to solve  $\min_{x \in B} \bar{f}(x)$ . This approach is more straight forward and incurs no extra overhead computation like the image space approach. When  $n$  is small, it should enjoy certain edge. However, as the following examples suggest, the high nonlinearity of  $\bar{f}(x)$  may offset the advantage in dimensions.

*Example 6.1* We use EM to solve Example 5.1 directly on  $D$ . The results are summarized in Table 6. Since we observed that the EM domain space approach usually produced solutions of poor quality, the following comparison was thus made by forcing EM on the domain to run extensively ( $Maxiter = 100000$ ) until the runtime has exceeded that used by the image space in Example 5.1. As we can see, for  $q = 4$  to  $q = 14$ , EM in the domain space obtained a less precise solution within the same time frame. Very surprisingly, the EM domain space approach solved correctly the 16-ratio problem in 4.29 s.

*Example 6.2* We used EM to solve Example 5.3 directly on  $D$ . From Table 7, it is evident when  $n \gg q$  the EM image space approach has a huge advantage in dimensionality and thus has performed much better.

Finally, we propose another reason why it is worthwhile to work in the image space. The replacement of  $f_s/g_s$  with  $\gamma_s$  suppresses the nonlinearity of the former into the latter single linear variable. As a result, the objective function  $\psi(\beta)$  in the image space has a much more regular shape than the original sum-of-ratios function.



**Table 6** EM domain space approach for Example 5.1 with parameters  $m = 10, (K_1, K_2, K_3, K_4) = (5, 8, 15, \infty), (a_1, a_2) = (3, 5), \mu = 0.45$

$q$	EM image space approach		EM domain space approach	
	Obj. value	Run time (s)	Obj. value	Run time (s)
4	0.2361	30.29	0.2381	30.29
6	0.0275	27.64	0.0308	27.64
8	-0.6513	25.43	-0.6496	25.43
10	1.4083	16.54	1.4141	16.54
12	1.6128	25.37	1.6166	25.37
14	2.2427	17.375	2.2486	29.48
16	1.6348	7091	1.4867	4.29

**Table 7** EM domain space approach for Example 5.3 with parameters  $m = 50, \mu = 0.56, (K_1, K_2, K_3, K_4) = (2, 2, 0, \infty), (a_1, a_2) = (4, 5)$  and  $Maxiter = 5000$

$(q, n)$	EM image space approach		EM domain space approach	
	Obj. value	Run time (s)	Obj. value	Run time (s)
(2, 200)	-10.9583	28.06	1.0221	558.57
(4, 200)	-9.1297	20.89	0.5434	554.84
(6, 200)	-4.4149	22.80	0.5915	563.78
(8, 200)	-4.1672	57.56	-0.2362	567.25

*Example 6.3* (Benson [6]) This problem has two nonlinear ratios with three variables. The  $\epsilon$ -optimal value reported in [6] is  $-6.12$ . Table 8 below shows that both the EM image space approach and the EM domain space approach solved the problem, but again the later was less precise and slower. Figure 2 shows that the objective function  $\psi(\beta)$  is a one-dimensional quasi-convex function. It is unimodular and is easier to solve than the original sum-of-non-linear-ratios problem.

$$F_1 = \frac{x_1^2 - 4x_1 + 2x_2^2 - 8x_2 + 3x_3^2 - 12x_3 - 56}{x_1^2 - 2x_1 + x_2^2 - 2x_2 + x_3 + 20};$$

$$F_2 = \frac{2x_1^2 - 16x_1 + x_2^2 - 8x_2 - 2}{2x_1 + 4x_2 + 6x_3}$$

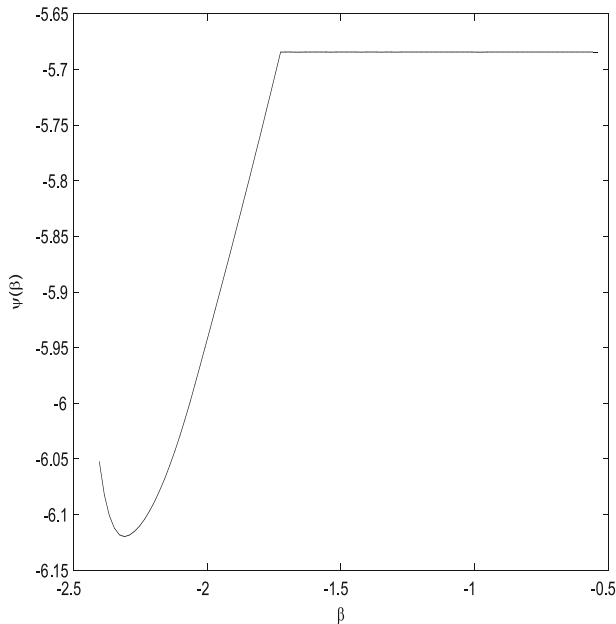
$$D = \{(x_1, x_2, x_3) \mid x_1 + x_2 + x_3 \leq 10, -x_1 - x_2 + x_3 \leq 4, x_1, x_2, x_3 \geq 1\}$$

### 7 Conclusion

In this paper, we reduce the sum-of-ratios problems into a sequence of single ratio problems and propose a random search algorithm to incorporate the Dinkelbach algorithm for solving the transformed problem ( $P''$ ). Since most of our examples have multiple local minima, local

**Table 8** Applying EM to both image and domain space using parameters  $m = 3$ ,  $\mu = 0.48$ ,  $(K_1, K_2, K_3) = (2, 3, 5)$ ,  $(a_1, a_2) = (3, 5)$ ,  $Maxiter = 10000$

EM image sp. approach		EM domain sp. approach	
Obj. value	Runtime	Obj. value	Runtime
-6.1198	28.3906	-6.1196	28.4063



**Fig. 2** The graph of  $\psi(\beta)$  is quasi-convex in Example 6.3

search algorithms, such as *fmincon* in **MATLAB**, may easily miss the global minimizer. On the other hand, although **KNITRO** along with multistart has done very well in low dimensional examples (Examples 5.1–5.2), it was insufficient for large scale problems in Example 5.3. In contrast, our EM image space approach replaces the nonlinear function  $f_s/g_s$  with a linear independent variable  $\gamma_s$ . When  $n \gg q$ , it minimizes a better behaved function  $\psi(\beta)$  in the much smaller image space.

Evaluating  $\psi(\beta)$ , however, create computational difficulty. Since  $\psi(\beta)$  is defined implicitly by a constrained minimization problem, deterministic type of algorithms using the first order information of the function, such as *fmincon* in **MATLAB** or **KNITRO** are, in general, insufficient for solving the considered optimization problem (P'''). This is why we picked a random search algorithm like EM to work out  $\psi(\beta)$ . The paper confirms the feasibility of combining the transformation (P'''), the Dinkelbach algorithm, and the EM random procedure, in spite that it is still time consuming and yet not precise enough. We hope that these drawbacks can be resolved by an improved version (especially with an efficient local search strategy) of EM.

**Acknowledgment** This research work was partially supported by the National Science Council of Taiwan under Project NSC 93-2215-M-006-017.

## References

1. Almqy, Y., Levin, O.: A class of fractional programming problems. *Oper. Res.* **19**, 57–67 (1971)
2. Avriel, M., Diewert, W.E., Schaible, S., Zang, I.: *Generalized Concavity*. Plenum Press (1988)
3. Barros, A.I., Frenk, J.B.G., Schaible, S., Zhang, S.: A new algorithm for generalized fractional programs. *Math. Program.* **72**, 147–175 (1996)
4. Benson, H.P.: Global optimization of nonlinear sums of ratios. *J. Math. Anal. Appl.* **263**, 301–315 (2001)
5. Benson, H.P.: Global optimization algorithm for the nonlinear sum of ratios problem. *J. Optim. Theor. Appl.* **112**, 1–29 (2002)
6. Benson, H.P.: Using concave envelopes to globally solve the nonlinear sum of ratios problems. *J. Glob. Optim.* **22**, 343–364 (2002)
7. Benson, H.P.: On the global optimization of sums of linear fractional functions over a convex set. *J. Optim. Theor. Appl.* **121**, 19–39 (2004)
8. Birbil, Ş.İ., Fang, S.C.: An electromagnetism-like mechanism for global optimization. *J. Glob. Optim.* **25**, 263–282 (2003)
9. Birbil, Ş.İ., Fang, S.C., Sheu, R.L.: On the convergence of a population-based global optimization algorithm. *J. Glob. Optim.* **30**, 301–318 (2004)
10. Cambini, A., Martein, L., Schaible, S.: On maximizing a sum of ratios. *J. Inform. Optim. Sci.* **10**, 65–79 (1989)
11. Frenk, J.B.G., Schaible, S.: Fractional programming. In: Floudas, C.A., Pardalos, P.M. (eds.) *Encyclopedia of Optimization*, vol. II, pp. 162–172. Kluwer Publishers, Dordrecht (1989)
12. Freund, R.W., Jarre, F.: Solving the sum-of-ratios problem by an interior-point method. *J. Glob. Optim.* **19**, 83–102 (2001)
13. Ibaraki, T.: Parametric approaches to fractional programs. *Math. Program.* **26**, 345–362 (1983)
14. Ingber, L.: Simulated annealing: practice versus theory. *Math. Comput. Modell.* **18**, 29–57 (1993)
15. Konno, H., Abe, N.: Minimization of the sum of three linear fractional functions. *J. Glob. Optim.* **15**, 419–432 (1999)
16. Konno, H., Fukaishi, K.: A branch and bound algorithm for solving low rank linear multiplicative and fractional programming problems. *J. Glob. Optim.* **18**, 283–299 (2000)
17. Konno, H., Inori, M.: Bond portfolio optimization by bilinear fractional programming. *J. Oper. Res. Soc. Jpn.* **32**, 143–158 (1989)
18. Kuno, T.: A branch-and-bound algorithm for maximizing the sum of several linear ratios. *J. Glob. Optim.* **22**, 155–174 (2002)
19. Kuno, T., Yajima, Y., Konno, H.: An outer approximation method for minimizing the product of several convex functions on a convex set. *J. Glob. Optim.* **3**, 325–335 (1993)
20. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag xvi+340 (1994)
21. Phuong, N.T.H., Tuy, H.: A unified monotonic approach to generalized linear fractional programming. *J. Glob. Optim.* **26**, 229–259 (2003)
22. Rinnooy-Kan, A.H.G., Timmer, G.T.: Stochastic global optimization methods. II. Multilevel methods. *Math. Program.* **39**, 57–78 (1987)
23. Schaible, S.: Recent results in fractional programming. *Oper. Res. Verfahren* **23**, 271–272 (1977)
24. Wood, G.R.: Multidimensional bisection applied to global optimisation. *Comput. Math. Appl.* **21**, 161–172 (1991)