Computers & Operations Research 40 (2013) 2301–2307

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Maximizing for the sum of ratios of two convex functions over a convex set ☆

Peiping Shen *, Weimin Li, Xiaodi Bai

College of Mathematics and Information Science, Henan Normal University, Xinxiang 453007, PR China

ARTICLE INFO

Available online 30 March 2013

Keywords:
Fractional programming
Global optimization
Sum of ratios
Branch and bound

ABSTRACT

This paper presents an algorithm for globally maximizing a sum of convex–convex ratios problem with a convex feasible region, which does not require involving all the functions to be differentiable and requires that their sub-gradients can be calculated efficiently. To our knowledge, little progress has been made for globally solving this problem so far. The algorithm uses a branch and bound scheme in which the main computational effort involves solving a sequence of linear programming subproblems. Because of these properties, the algorithm offers a potentially attractive means for globally solving the sum of convex–convex ratios problem over a convex feasible region. It has been proved that the algorithm possesses global convergence. Finally, the numerical experiments are given to show the feasibility of the proposed algorithm.

1. Introduction

The fractional programming is one of the most successful fields today in nonlinear optimization problems. The sum of ratios (SOR) problem is a special class of optimization between fractional programs, and has attracted the interest of practitioners and researchers for at least 30 years. This is because, from a practical point of view, the (SOR) has a number of important applications. Included among these are applications in areas such as transportation planning, government contracting, economics and finance [1–4]. From a research point of view, the (SOR) poses significant theoretical and computational difficulties. Even in the simplest case where the ratios are all linear, i.e., the numerators and denominators are affine functions, their sum is neither quasiconvex nor quasiconcave though each of them has both properties. This is mainly due to the fact that the (SOR) is a global optimization problem, i.e., it is known to generally possess multiple local optima that are not globally optimal. The sum of ratios problem therefore fall into the domain of global optimization [5,6]. One can find the details of this development in Refs. [7–9] and the corresponding bibliographies appearing therein.

Many global optimization algorithms have been proposed for solving the linear sum of ratios fractional programs, i.e., the numerators and denominators are all affine functions and the feasible region is a polyhedron (see [10–13], for example). Recently, some solution algorithms have been developed for solving globally the nonlinear sum of ratios problem. For instance, Freund and Jarre [14] proposed an interior-point approach for the convex–concave ratios with convex constraints; Yang et al. [15] presented a conical partition algorithm for the sum of DC ratios; Benson [16,17] gave two branch-and bound algorithms for the concave–convex ratios; Shen et al. [18–21] developed global optimization algorithms for the nonlinear sum of ratios problem.

In this paper we consider the following sum of ratios problem:

$$(\mathrm{P})\quad \begin{cases} v=\max & \sum_{j=1}^{m} h_j(x)=\sum_{j=1}^{m}\dfrac{f_j(x)}{g_j(x)} \\ \text{s.t.} & x\in X, \end{cases}$$

where $f_j(x)$ and $g_j(x)$, $j=1,\dots,m$, are real-valued convex functions defined on X, X is a nonempty, compact convex set in R^n, and, for each $j=1,\dots,m$, $f_j(x)\geq 0$ and $g_j(x)>0$ for all $x\in X$.

The problem (P) is called a nonconcave fractional program, and may arise in practical applications, for instance, the maximally predictable portfolio problem ($m=1$) [22], and the projective geometry problems ($m\geq 2$) including multiview triangulation, camera resectioning and homography estimation [23]. It should be noted that although the literature on nonconvex optimization has rapidly increased in recent years, most research papers either only deal with the theoretical aspects of the problem or are concerned only with finding Kuhn–Tucker points or local solutions rather than global optima. Also, the problem (P) is different from the problems considered in [13–21]. Specially, the feasible set and each numerator in the objective function in (P) are not the same as the corresponding ones in [21], although both problem (P) and the one considered in [21] are the sum of convex–convex ratios problem. So it is difficult to apply these solution methods in the

* Corresponding author.
E-mail address: shenpeiping@163.com (P. Shen).
☆ Research supported by National Natural Science Foundation of China (11171094) and Fundamental and Frontier Technology Research Projects of Henan Province (132300410285).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.03.012

above Refs. [13–21] directly to problem (P) since their algorithms are based on the oneself structure of the corresponding problems. As a result, to my knowledge, most of the theoretical and algorithmic work for problem (P) applies only to single-ratio cases, i.e., $m=1$ (see [24–27]) or to special cases of (P) (see [10–13], for example). Additionally, Frenk and Schaible [28] and Schaible [9] have encouraged that more research be done into the solutions of nonconcave fractional programs.

The purpose of this paper is to present a branch and bound algorithm for globally solving the nonlinear sum of ratios problem (P), which does not require involving all the functions to be differentiable and requires that their sub-gradients can be calculated efficiently. In the algorithm, the linear bounding functions related to the objective and constraint functions of (P) are first formed, based on the characteristics of problem (P). Thus a linear relaxation programming is then derived for providing an upper bound of the optimal value to problem (P). The main computational effort of the algorithm only involves in solving a sequence of linear programming subproblems that do not grow in size from iteration to iteration. Additionally, the algorithm uses simplices, rather than more complicated polytopes, as partition elements in the branch and bound search. This keeps the number of constraints in the linear program subproblems to minimum. Finally, the convergence of the algorithm is proved and numerical experiments are given to illustrate the feasibility of the algorithm.

The remainder of this paper is organized as follows. The initial simplex and simplicial partition process, the upper and lower bounding process and the fathoming process used in this approach are defined and studied in Section 2. Section 3 presents the algorithm for solving (P), and shows the convergence property of the algorithm. In Section 4, we give the results of solving some numerical examples with the algorithm.

2. Key algorithm processes

In this section, in order to present the branch and bound algorithm for solving (P), we first explain the four key processes: successively refined partitioning of the feasible set; estimation of upper and lower bounds for the optimal value of the objective function; and deleting procedure over each subset generated by the partitions.

The partition process consists in a successive simplicial partition of the initial simplex S^0 following in an exhaustive subdivision rule, i.e., such that any infinite nested sequence of partition sets generated through the algorithm shrinks to a singleton. A commonly used exhaustive subdivision rule is the standard bisection.

The upper bounding process is two-fold. First, for each simplex S created by the branching process, the upper bounding process seeks an upper bound for the maximum of the objective function taken over $X \cap S$. Second, for each step $k \ge 0$ of the algorithm, the upper bounding process seeks an upper bound UB_k for the global optimal value of problem (P).

The lower bounding process consists in estimating a lower bound LB_k for the objective function value by enclosing all feasible points found while computing the upper bounds of the optimum of problem (P).

The deleting process consists in deleting each simplex in which there is no feasible solution for further consideration.

Next, we will give the detail processes, respectively.

2.1. Initial simplex and simplicial partition

The partition process iteratively subdivides an n-dimension simplex S^0 containing X into n-dimension subsimplices. This process helps the algorithm identifies a location of a global optimal solution in X for problem (P). Throughout the algorithm, each simplex created by this branching process is n-dimensional and will be called an *n-simplex*.

An initial simplex S^0 which tightly encloses X can be constructed as follows [29]:

$$S^0 = \left\{ x \in R^n | x_i \ge \gamma_i, i = 1, \ldots, n, \sum_{i=1}^{n} x_i \le \gamma \right\},$$

where $\gamma = \max\{\sum_{i=1}^{n} x_i | x \in X\}$ and $\gamma_i = \min\{x_i | x \in X\}$, $i = 1, \ldots, n$. Then the vertex set of S^0 is

$$\{V_0^0, V_1^0, \ldots, V_n^0\},$$

where $V_0^0 = (\gamma_1, \ldots, \gamma_n)$ and $V_i^0 = (\gamma_1, \ldots, \gamma_{i-1}, \alpha_i, \gamma_{i+1}, \ldots, \gamma_n)$ with

$$\alpha_i = \gamma - \sum_{t \ne i} \gamma_t, \quad i = 1, \ldots, n.$$

Next, the subdivision of simplices is defined in the following way. At each stage of the branch and bound algorithm, a subsimplex of S^0 is subdivided into two simplices by the branching process. To explain this process, assume without loss of generality that a subsimplex of S^0 to be subdivided is S with the vertex set $\{V_0, V_1, \ldots, V_n\}$. Let c be the midpoint of the longest edge $[V_d, V_e]$ of S. Then $\{S_1, S_2\}$ is called a simplicial bisection of S, where the vertex set of S_1 is $\{V_0, V_1, \ldots, V_{d-1}, c, V_{d+1}, \ldots, V_n\}$, and the vertex set of S_2 is $\{V_0, V_1, \ldots, V_{e-1}, c, V_{e+1}, \ldots, V_n\}$.

It follows easily that this simplicial partition process is exhaustive, i.e., if $\{S^k\}$ denotes a nested subsequence of simplices (i.e., $S^{k+1} \subset S^k$, for all k) formed by the branching process, then for some unique point $\bar{x} \in R^n$,

$$\bigcap_k S^k = \lim_{k \to \infty} S^k = \{\bar{x}\}.$$

2.2. Upper bounding

Let $S = \{V_0, V_1, \ldots, V_n\}$ represent a typical *n-simplex* created by the partition process of the algorithm, and let

$$\hat{s} = \frac{1}{n+1} \sum_{i=0}^{n} V_i$$

denote the barycenter of S. Consider the subproblem

$$(\mathrm{P}(S)) \quad \begin{cases} v(S) = \max & \sum_{j=1}^{m} h_j(x) \\ \quad\quad \text{s.t.} & x \in X \cap S. \end{cases}$$

To explain how the upper bounding process finds an upper bound $UB(S)$ for $v(S)$, we first need to give two affine functions $lf_j^S(x)$ and $lg_j^S(x)$ over S such that

$$lf_j^S(x) \ge f_j(x) \quad \text{and} \quad lg_j^S(x) \le g_j(x), \ \forall x \in S, \ j = 1, \ldots, m.$$

Notice that S is a simplex and $f_j(x)$ is convex. So from [6] it follows that for each $x \in S$, $lf_j^S(x)$ is a concave envelope for $f_j(x)$ over S given by

$$lf_j^S(x) = \sum_{i=0}^{n} \alpha_i f(V_i), \quad j = 1, \ldots, m, \tag{1}$$

where $x = \sum_{i=0}^{n} \alpha_i V_i$ with $\alpha_i \ge 0$ and $\sum_{i=0}^{n} \alpha_i = 1$. On the other hand, since for each $j = 1, \ldots, m$, $g_j(x)$ is a convex function on R^n, it follows that $g_j(x)$ has the sub-gradient everywhere in their domains. Thus, the affine function $lg_j^S(x) : R^n \to R$ is defined for each $x \in R^n$ by

$$lg_j^S(x) = g_j(\hat{s}) + \langle p_j^{\hat{s}}, x - \hat{s} \rangle \tag{2}$$

satisfying

$$lg_j^S(x) \le g_j(x)$$

for all $x \in R^n$, where $p_j^{\hat{s}}$ denotes any sub-gradient of g_j at $\hat{s}$.