Journal of Global Optimization **15:** 433–447, 1999.
© 1999 *Kluwer Academic Publishers. Printed in the Netherlands.*

Heuristic Methods for Linear Multiplicative Programming

X.J. LIU, T. UMEGAKI and Y. YAMAMOTO
Institute of Policy and Planning Sciences, University of Tsukuba, Tsukuba, Ibaraki 305-8573, Japan
e-mail: yamamoto@shako.sk.tsukuba.ac.jp

(Accepted in revised form 28 July 1999)

Abstract. The linear multiplicative programming is the minimization of the product of affine functions over a polyhedral set. The problem with two affine functions reduces to a parametric linear program and can be solved efficiently. For the objective function with more than two affine functions multiplied, no efficient algorithms that solve the problem to optimality have been proposed, however Benson and Boger have proposed a heuristic algorithm that exploits links of the problem with concave minimization and multicriteria optimization. We will propose a heuristic method for the problem as well as its modification to enhance the accuracy of approximation. Computational experiments demonstrate that the method and its modification solve randomly generated problems within a few percent of relative error.

1. Introduction

The linear multiplicative programming is the minimization of the product of p affine functions $c_i^\top x + d_i$ over a polyhedral set D, i.e.,

$$P \quad \left| \begin{array}{ll} \text{minimize} & f(x) = \prod_{i=1}^{p} (c_i^\top x + d_i) \\ \text{subject to} & x \in D, \end{array} \right.$$

where $D := \{x | x \in R^n;\ Ax \leqslant b\}$ is the polyhedral set defined by the system of m linear inequalities $Ax \leqslant b$. As will be seen in the next section, we can assume without loss of generality that the p affine functions $c_i^\top x + d_i$ are positive on D.

When $p = 2$, problem P has been extensively investigated by, for example, Swarup [17–19], Forgo [3,4], Konno-Kuno-Yajima [5], Konno-Kuno [6,7] and proved to be NP-hard by Matsui [10] (see also Pardalos-Vavasis [11]). The algorithms proposed in these papers are mainly based on the fact that an optimal solution of P is among the vertices to be encountered while solving the parametric linear program

$$\left| \begin{array}{ll} \text{minimize} & \theta(c_1^\top x + d_1) + (1-\theta)(c_2^\top x + d_2) \\ \text{subject to} & x \in D \end{array} \right.$$

for $\theta \in [0, 1]$ or

$$\left|\begin{array}{ll} \text{minimize} & c_1^{\mathsf{T}}x + d_1 \\ \text{subject to} & x \in D;\ c_2^{\mathsf{T}}x + d_2 = \eta \end{array}\right.$$

for $\eta > 0$.

For problem P with $p > 2$ there have been proposed several algorithms by, for example, Thoai [20], Kuno-Yajima-Konno [9], Ryoo-Sahinidis [13] and Kuno [8]. See also Tuy [21] for the dimension reduction technique by dualization. Thoai [20] exploited the equivalence between problem P and problem

$$\left|\begin{array}{ll} \text{minimize} & \prod_{i=1}^{p} y_i \\ \text{subject to} & x \in D \\ & c_i^{\mathsf{T}}x + d_i \leqslant y_i \text{ for } i = 1, 2, \ldots, p, \end{array}\right.$$

called the master problem. If one knows the set, say Y, of points $y \in R^p$ such that (x, y) is a feasible solution of the master problem for some $x \in R^n$, problem P clearly reduces to the minimization of $\prod_{i=1}^{p} y_i$ on Y. He proposed an outer approximation method which enumerates the linear inequalities defining Y. The master problem considered by Kuno, Yajima and Konno [9] is

$$\left|\begin{array}{ll} \text{minimize} & \sum_{i=1}^{p} \xi_i (c_i^{\mathsf{T}}x + d_i) \\ \text{subject to} & x \in D. \end{array}\right.$$

Let $h(\xi)$ be the optimal value of this problem for $\xi \in R^p$ and let $\Xi := \{\xi | \xi \in R^p;\ \xi \geqslant 0;\ \prod_{i=1}^{p} \xi_i \geqslant 1\}$. They showed the reduction of problem P to the minimization of $h(\xi)$ on Ξ and developed an outer approximation method. The methods of Ryoo and Sahinidis [13] and of Kuno [8] are based on the combination of branch-and-bound and cutting plane methods.

Let C be the $p \times n$ matrix of rows c_i^{T} for $i = 1, 2, \ldots, p$ and consider the following multicriteria problem (see for example Steuer [14]).

$$MC \quad \left|\begin{array}{ll} \text{vector minimize} & Cx \\ \text{subject to} & x \in D. \end{array}\right.$$

Then clearly an optimal solution of P is an efficient point of problem MC, where a point $\overline{x} \in D$ is said to be *efficient* if there is no point $x \in D$ such that $Cx \leqslant C\overline{x}$ and $Cx \neq C\overline{x}$. Together with the pseudoconcavity of the objective function f (see the next section), we see that there is an optimal solution of P among the efficient vertices of D. Exploiting this property, Benson-Boger [2] proposed a heuristic method for problem P, which they named the *efficient point search heuristic*. Their computational experiment for randomly generated problems supports that their method efficiently provides fairly good approximate solutions of rather large problems for $p = 2, 3, 4$, and 5.

In this paper we will propose a heuristic algorithm and its modification to enhance the accuracy of approximate solutions to be obtained. In Section 2 we show that the assumption that $c_i^\mathsf{T} x + d_i > 0$ for all $x \in D$ does not deteriorate the generality of the problem. After introducing Benson-Boger's method in Section 3, we propose a heuristic method *efficient face search heuristic* in Section 4 as well as its modification. In Section 5 some computational results will be reported. Some remarks will be given in the last section.

2. Assumption

For a subset K of $\{1, 2, \dots, p\}$ let $|K|$ denote the number of indices of K and let

$$D(K) := \left\{ x \;\middle|\; \begin{array}{ll} x \in D & \\ c_i^\mathsf{T} x + d_i \leqslant 0 & \text{for } i \in K \\ c_i^\mathsf{T} x + d_i \geqslant 0 & \text{for } i \notin K \end{array} \right\}.$$

Clearly problem P reduces to the family of problems $P(K)$:

$$P(K) \left| \begin{array}{ll} \text{minimize} & \prod_{i=1}^{p} (c_i^\mathsf{T} x + d_i) \\ \text{subject to} & x \in D(K). \end{array} \right.$$

If $|K|$ is odd, the objective function takes a nonpositive value on $D(K)$, while a nonnegative value on $D(K)$ if $|K|$ is even. Therefore to find a globally optimal solution of P, we have only to solve problems $P(K)$ for K of odd cardinality which is rewritten as

$$\left| \begin{array}{ll} \text{maximize} & (\prod_{i \in K} -(c_i^\mathsf{T} x + d_i)) \times (\prod_{i \notin K} (c_i^\mathsf{T} x + d_i)) \\ \text{subject to} & x \in D(K). \end{array} \right.$$

LEMMA 2.1. *Suppose $|K|$ is odd. Then the optimal value of $P(K)$ is zero if and only if there is an $i \in \{1, 2, \dots, p\}$ such that $c_i^\mathsf{T} x + d_i = 0$ for all $x \in D(K)$.*

Proof. The 'if' part is trivial. To show the 'only if' part, suppose the contrary, that is, for each $i \in \{1, 2, \dots, p\}$ there is a point $x^i \in D(K)$ such that $c_i^\mathsf{T} x^i + d_i \neq 0$. Let x be their barycenter $(1/p) \sum_{i=1}^{p} x^i$. Then one readily sees that $x \in D(K)$, $c_i^\mathsf{T} x + d_i < 0$ for $i \in K$ and $c_i^\mathsf{T} x + d_i > 0$ for $i \notin K$. This implies the negative optimal value of $P(K)$, and consequently a contradiction. □

Note that whether $D(K)$ is contained in $\{x \mid c_i^\mathsf{T} x + d_i = 0\}$ for some $i \in \{1, 2, \dots, p\}$ can be readily checked by solving at most p linear programs. When this does not occur, an equivalent form of problem $P(K)$ would be

$$\left| \begin{array}{ll} \text{maximize} & \sum_{i \in K} \log(-(c_i^\mathsf{T} x + d_i)) + \sum_{i \notin K} \log(c_i^\mathsf{T} x + d_i) \\ \text{subject to} & x \in D(K). \end{array} \right.$$

By virtue of the monotonicity and concavity of the logarithmic function this is a concave maximization problem, which is rather an easy problem to solve globally. To sum up, we obtain

LEMMA 2.2. *If $D(K) \neq \emptyset$ for some K of odd cardinality, problem P reduces to a number of concave maximization problems,*

and we will assume throughout this paper that

ASSUMPTION 2.3. *$D(K) = \emptyset$ for every K of odd cardinality.*

LEMMA 2.4. *Under Assumption 2.3, $D = D(K)$ for a unique subset $K \subseteq \{1, 2, \ldots, p\}$ of even cardinality. Moreover*

$$D \subseteq \left\{ x \;\middle|\; \begin{array}{ll} c_i^{\mathsf{T}} x + d_i < 0 & \text{for } i \in K \\ c_i^{\mathsf{T}} x + d_i > 0 & \text{for } i \notin K \end{array} \right\}. \tag{2.1}$$

Proof. From Assumption 2.3 there is a $K \subseteq \{1, 2, \ldots, p\}$ of even cardinality with nonempty $D(K)$. If there is a point $x \in D(K)$ satisfying $c_i^{\mathsf{T}} x + d_i = 0$ for some $i \in K$, then it belongs also to $D(K \backslash \{i\})$. This implies that $D(K \backslash \{i\}) \neq \emptyset$ and contradicts Assumption 2.3. If $c_i^{\mathsf{T}} x + d_i = 0$ for some $i \notin K$, then $x \in D(K \cup \{i\})$. This is again a contradiction. Therefore for any $x \in D(K)$ we obtain

$$c_i^{\mathsf{T}} x + d_i < 0 \quad \text{for } i \in K, \; c_i^{\mathsf{T}} x + d_i > 0 \quad \text{for } i \notin K. \tag{2.2}$$

Next suppose that $K \neq K'$ and $D(K)$ and $D(K')$ are both nonempty. Choose arbitrarily points $x \in D(K)$ and $x' \in D(K')$. Then the line segment $[x, x']$ connecting them contains a point z of $D(K)$ which satisfies $c_i^{\mathsf{T}} z + d_i = 0$ for some $i \in (K \backslash K') \cup (K' \backslash K)$. This contradicts (2.2). Therefore only one $D(K)$ is nonempty. Since the union of all $D(K)$'s coincides with D, we obtain that this $D(K)$ coincides with D and (2.1). □

Let z_i and Z_i be the infimum and supremum of $c_i^{\mathsf{T}} x + d_i$ over D for $i = 1, 2, \ldots, p$ and let $[z_i, Z_i]$ denote the closed interval between z_i and Z_i with the convention that infinite endpoints are excluded. Then the above argument is summarized in the following theorem.

THEOREM 2.5. *If $0 \notin [z_i, Z_i]$ for all $i = 1, 2, \ldots, p$, then $D = D(K)$ for a unique $K \subseteq \{1, 2, \ldots, p\}$. If $|K|$ is odd, problem P reduces to a single concave maximization problem, and if $|K|$ is even, it reduces to problem P with all terms $c_i^{\mathsf{T}} x + d_i$ positive on D by reversing the signs of c_i and d_i for $i \in K$. If $0 \in [z_i, Z_i]$ for some i, solving a number of concave maximization problems will yield an optimal solution of problem P.*

Therefore we will hereafter assume without loss of generality that

$c_i^{\mathsf{T}} x + d_i > 0$ for all $i = 1, 2, \ldots, p$ and for all $x \in D$.

Under this assumption the objective function f is pseudoconcave on D (see Avriel et al. [1]), which implies the existence of an optimal solution of P among the vertices of D.

3. Benson–Boger's heuristic

Exploiting the property that an optimal solution of P is an efficient vertex of the multicriteria problem MC, Benson and Boger [2] proposed a heuristic method. Let $d = (d_1, d_2, \dots, d_p)^\top$ and for $w, y \in R^p$ let $Q(w, y)$ be the linear program

$$Q(w, y) \quad \left| \begin{array}{ll} \text{minimize} & w^\top Cx \\ \text{subject to} & x \in D;\ Cx + d \leqslant y. \end{array} \right.$$

Then their method is outlined as follows.

Benson–Boger's Heuristic Method
Step 1
find $\overline{y} = (\overline{y}_1, \dots, \overline{y}_p)$ and $\underline{y} = (\underline{y}_1, \dots, \underline{y}_p)$ such that
$\underline{y} \leqslant Cx + d \leqslant \overline{y}$ holds for any $x \in D$;
solve $\max\{\alpha \mid \alpha \geqslant 0;\ x \in D;\ Cx + d \leqslant \overline{y} + \alpha(\underline{y} - \overline{y})\}$ to yield
an optimal solution (x^*, α^*);
$y^* \leftarrow \overline{y} + \alpha^*(\underline{y} - \overline{y})$;
$y^j \leftarrow y^* + (j/s)(\overline{y} - y^*)$ for $j = 0, 1, \dots, s$;
choose an appropriate positive value ω;
$w^0 \leftarrow (1, 1, \dots, 1)^\top$;
$w^1 \leftarrow (\omega, 1, \dots, 1)^\top, \dots, w^p \leftarrow (1, 1, \dots, \omega)^\top$;
Step 2
for all combinations of w^k and y^j do
solve $Q(w^k, y^j)$ to yield an optimal solution $\overline{x}^{kj}$;
find an efficient face σ^{kj} containing $\overline{x}^{kj}$ of D;
solve $\min\{(\nabla f(\overline{x}^{kj}))^\top x \mid x \in \sigma^{kj}\}$ to yield an optimal solution $\tilde{x}^{kj}$;
end for
Step 3
$(k^*, j^*) \leftarrow \operatorname{argmin}_{k=0,\dots,p;\ j=0,\dots,s} f(\tilde{x}^{kj})$
$\tilde{x} \leftarrow \tilde{x}^{k^* j^*}$

The point $\tilde{x}$ in Step 3 is the approximate solution provided by the method. Since w^k is a positive weight vector, the solution $\overline{x}^{kj}$ of $Q(w^k, y^j)$ is an efficient point (see Steuer [14]) of D. Then an efficient face, a face consisting of efficient points, σ^{kj} containing $\overline{x}^{kj}$ is searched for and a solution, which is a vertex of D, of $\min\{(\nabla f(\overline{x}^{kj}))^{\mathsf{T}}x|x \in \sigma^{kj}\}$ is stored as a candidate for the solution. The key lemma of Benson–Boger [2] for finding an efficient face is as follows.

LEMMA 3.1 (Benson–Boger [2], Theorem 3.3). *Let $\overline{x}^{kj}$ be an optimal solution of $Q(w^k, y^j)$ and let $\overline{y} := C\overline{x}^{kj} + d$. Let $\overline{u}$ be a dual optimal solution of $Q(w^k, \overline{y})$ corresponding to the constraint $Cx + d \leqslant \overline{y}$. Then $\sigma^{kj} := \{x \mid x \in D;\ (w^k + \overline{u})^{\mathsf{T}}Cx = (w^k + \overline{u})^{\mathsf{T}}C\overline{x}^{kj}\}$ is an efficient face containing $\overline{x}^{kj}$.*

Finding an efficient face according to this lemma will not cost much because $Q(w^k, \overline{y})$ differs from $Q(w^k, y^j)$, which has already been solved, in only the right hand side constant vector. This will contribute to the efficiency of the method. But roughly speaking, the total of $3(s + 1)(p + 1)$ linear programs are solved to obtain an approximate solution $\tilde{x}$.

To see how small the face σ^{kj} can be, let $(\overline{v}, \overline{u})$ be a dual optimal solution of $Q(w^k, \overline{y})$, i.e., $(\overline{v}, \overline{u})$ solves

$$\left|\begin{array}{ll} \text{maximize} & -b^{\mathsf{T}}v - (\overline{y} - d)^{\mathsf{T}}u \\ \text{subject to} & A^{\mathsf{T}}v + C^{\mathsf{T}}u = -C^{\mathsf{T}}w^k \\ & v \geqslant 0;\ u \geqslant 0. \end{array}\right.$$

Let x' be a point of the face σ^{kj}, then

$$\overline{v}^{\mathsf{T}}Ax' = -(w^k + \overline{u})^{\mathsf{T}}Cx' = -(w^k + \overline{u})^{\mathsf{T}}C\overline{x}^{kj} = \overline{v}^{\mathsf{T}}A\overline{x}^{kj}.$$

Since $\overline{y} = C\overline{x}^{kj} + d$, and hence $\overline{x}^{kj}$ remains optimal to problem $Q(w^k, \overline{y})$, the complementarity of slackness holds between $\overline{x}^{kj}$ and $(\overline{v}, \overline{u})$. Hence we have $\overline{v}^{\mathsf{T}}(A\overline{x}^{kj} - b) = 0$. This together with the above equation implies

$$\overline{v}^{\mathsf{T}}Ax' = \overline{v}^{\mathsf{T}}b.$$

Therefore, if the strict complementarity of slackness holds between $\overline{x}^{kj}$ and $\overline{v}$, x' satisfies by equality all the constraints binding at $\overline{x}^{kj}$. Namely

$$\sigma^{kj} \subseteq \{x \mid x \in D;\ a_i^{\mathsf{T}}x = b_i \text{ for all constraints } i \text{ binding at } \overline{x}^{kj}\},$$

where a_i^{T} denotes the ith row of A, and b_i denotes the ith element of b. Particularly, when $\overline{x}^{kj}$ is a vertex of D, it is likely that σ^{kj} ends up to be the zero-dimensional face consisting of $\overline{x}^{kj}$ alone.

The higher the dimension of the efficient face to be found is, the wider the region we can search for an efficient vertex. Therefore the way of finding an efficient face would be room for improvement of Benson–Boger's heuristic method.

4. Efficient face search heuristic

Let e be a p-dimensional positive vector and consider the linear system

$$A^{\mathsf{T}} v + C^{\mathsf{T}} u = 0;\ \ v \geqslant 0;\ \ u \geqslant e. \tag{4.1}$$

LEMMA 4.1. *For a solution* $(\overline{v}, \overline{u})$ *of (4.1) let* σ *be a face of* D *defined by*

$$\sigma := \{x \mid x \in D;\ a_i^{\mathsf{T}} x = b_i \text{ for } i \text{ such that } \overline{v}_i > 0\},$$

where a_i^{T} *is the* i*th row of* A *and* b_i *is the* i*th component of* b*. Then* σ *is an efficient face, i.e., every point of* σ *is an efficient point of* MC*.*

Proof. We shall show that every point $\overline{x}$ of σ is an optimal solution of

$$\left|\begin{array}{ll} \text{minimize} & \overline{u}^{\mathsf{T}} C x \\ \text{subject to} & Ax \leqslant b. \end{array}\right.$$

Clearly $\overline{v}$ is a feasible solution of the dual problem

$$\left|\begin{array}{ll} \text{maximize} & -b^{\mathsf{T}} v \\ \text{subject to} & A^{\mathsf{T}} v + C^{\mathsf{T}} \overline{u} = 0;\ \ v \geqslant 0, \end{array}\right.$$

and from the definition of σ we see that $\overline{x}$ together with $\overline{v}$ satisfies the complementarity of slackness condition $\overline{v}^{\mathsf{T}}(A\overline{x} - b) = 0$. Therefore $\overline{x}$ is an optimal solution of the above primal linear program. Since $\overline{u}$ is a positive vector, $\overline{x}$ is an efficient point of MC (see Steuer [14]). □

From the viewpoint of widening the search region for an approximate solution of P, the higher dimensional face σ would be more desirable. Namely, the number of positive components of $\overline{v}$ should be as small as possible. Therefore we propose to solve the following linear program to obtain an efficient face.

$$\left|\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} v_i \\ \text{subject to} & A^{\mathsf{T}} v + C^{\mathsf{T}} u = 0;\ \ v \geqslant 0;\ \ u \geqslant e. \end{array}\right.$$

Now we are ready to describe our heuristic method.

Heuristic method

Step 1

choose an appropriate positive value ω;

$w^0 \leftarrow (1, 1, \ldots, 1)^{\mathsf{T}}$;

$w^1 \leftarrow (\omega, 1, \ldots, 1)^{\mathsf{T}}, \ldots, w^p \leftarrow (1, 1, \ldots, \omega)^{\mathsf{T}}$;

Step 2

for all w^k do

solve $\min\{(w^k)^{\mathsf{T}} C x \mid Ax \leqslant b\}$ to yield an optimal solution $\overline{x}^k$;

$J^k \leftarrow \{i \mid i \in \{1, 2, \ldots, m\};\ a_i^{\mathsf{T}} \overline{x}^k = b_i\}$;

solve $\min\{\sum_{i \in J^k} v_i \mid \sum_{i \in J^k} v_i a_i + C^{\mathsf{T}} u = 0;\ v_i \geqslant 0;\ u \geqslant e\}$ to

yield an optimal solution $(\overline{v}^k, \overline{u}^k)$;
$\hat{J}^k \leftarrow \{i \mid i \in J^k;\ \overline{v}_i^k > 0\}$;
$\sigma^k \leftarrow \{x \mid x \in D;\ a_i^\mathsf{T} x = b_i \text{ for } i \in \hat{J}^k\}$;
solve $\min\{(\nabla f(\overline{x}^k))^\mathsf{T} x \mid x \in \sigma^k\}$ to yield an optimal solution $\tilde{x}^k$;
end for

Step 3

$k^* \leftarrow \mathrm{argmin}_{k=0,\dots,p} f(\tilde{x}^k)$
$\tilde{x} \leftarrow \tilde{x}^{k^*}$

In Step 1 weight vectors $w^0, w^1, \dots, w^p$ are chosen in the same manner as Benson and Boger proposed. We solve $p+1$ linear programs in Step 2 to yield $\overline{x}^0, \dots, \overline{x}^p$ and set J^k be the index set of binding constraints at $\overline{x}^k$. A higher dimensional efficient face is searched for by solving another linear program, and then $(\nabla f(\overline{x}^k))^\mathsf{T} x$ is minimized on the face to obtain a candidate solution $\tilde{x}^k$. Therefore we solve $3(p+1)$ linear programs to obtain an approximate solution $\tilde{x}$.

Although we aim at minimizing the number of positive components of $\overline{v}^k$ by minimizing $\sum_{i \in J^k} v_i$ in Step 2, it does not always work, which stimulates us to an improvement. In the modified heuristic method we keep the best candidate solution as $\tilde{x}^{k^*}$, and then for all $j \in \hat{J}^{k^*}$ we check if there is an efficient face whose binding constraints have indices in $\hat{J}^{k^*} \backslash \{j\}$. Then we minimize $(\nabla f(\overline{x}^{k^*}))^\mathsf{T} x$ on the face to be found.

We carried out a preliminary experiment for the modification in which the minimization of $\sum_{i \in \hat{J}^k \backslash \{j\}} v_i$ is done for all $j \in \hat{J}^k$ and for all $k = 0, 1, \dots, p$. Although this modification produced by far more candidate solutions than the above heuristic method, it requires solving additional $2 \times \sum_{k=0}^{p} |\hat{J}^k|$ of linear programs. In most of the instances tested in the experiment the best approximate solutions were found among the candidates $\tilde{x}^{k^*j}$ generated from $\hat{J}^{k^*} \backslash \{j\}$, where k^* is the index such that $f(\tilde{x}^{k^*}) = \min_{k=0,1,\dots,p} f(\tilde{x}^k)$. Therefore we propose the following modification, that is, doing Step 4 for $\hat{J}^{k^*}$ alone. The computational experiments will show that the modification is worth doing when a more accurate approximate solution is desired.

Modified heuristic method

Step 1

choose an appropriate positive value ω;
$w^0 \leftarrow (1, 1, \dots, 1)^\mathsf{T}$;
$w^1 \leftarrow (\omega, 1, \dots, 1)^\mathsf{T}, \dots, w^p \leftarrow (1, 1, \dots, \omega)^\mathsf{T}$;

Step 2

for all w^k do
solve $\min\{(w^k)^\mathsf{T} Cx \mid Ax \leqslant b\}$ to yield an optimal solution $\overline{x}^k$;
$J^k \leftarrow \{i \mid i \in \{1, 2, \dots, m\};\ a_i^\mathsf{T} \overline{x}^k = b_i\}$;
solve $\min\{\sum_{i \in J^k} v_i \mid \sum_{i \in J^k} v_i a_i + C^\mathsf{T} u = 0;\ v_i \geqslant 0;\ u \geqslant e\}$ to
yield an optimal solution $(\overline{v}^k, \overline{u}^k)$;

$\hat{J}^k \leftarrow \{i \mid i \in J^k;\ \overline{v}_i^k > 0\}$;
$\sigma^k \leftarrow \{x \mid x \in D;\ a_i^\top x = b_i$ for $i \in \hat{J}^k\}$;
solve $\min\{(\nabla f(\overline{x}^k))^\top x \mid x \in \sigma^k\}$ to yield an optimal solution $\tilde{x}^k$;
end for

Step 3

$k^* \leftarrow \mathrm{argmin}_{k=0,\dots,p} f(\tilde{x}^k)$

Step 4

for all $j \in \hat{J}^{k^*}$ do
solve $\min\{\sum_{i\in\hat{J}^{k^*}\setminus\{j\}} v_i \mid \sum_{i\in\hat{J}^{k^*}\setminus\{j\}} v_i a_i + C^\top u = 0;\ v_i \geqslant 0;\ u \geqslant e\}$ to yield an optimal solution $(\overline{v}^{k^*j}, \overline{u}^{k^*j})$;
$\hat{J}^{k^*j} \leftarrow \{i \mid i \in \hat{J}^{k^*}; \overline{v}_i^{k^*j} > 0\}$;
$\sigma^{k^*j} \leftarrow \{x|x \in D; a_i^\top x = b_i$ for $i \in \hat{J}^{k^*j}\}$;
solve $\min\{(\nabla f(\overline{x}^{k^*}))^\top x|x \in \sigma^{k^*j}\}$ to yield an optimal solution $\tilde{x}^{k^*j}$;
end for

Step 5

let $\tilde{x}$ be the point that attains $\min\{f(\tilde{x}^{k^*}), \min_{j\in\hat{J}^{k^*}} f(\tilde{x}^{k^*j})\}$

5. Computational experiments

We wrote a code in C for our heuristic method and modified heuristic method, and carried out the experiment on HP Apollo Model 715/75. The linear program solver we used is LPAKO ver.4.0f provided by Park [12]. Following Benson-Boger's way of experiment in [2] we generated 240 problem instances. Namely we set

$$D = \{x \mid Tx \geqslant t;\ 1 \leqslant x_j \leqslant \hat{t} \text{ for } j = 1, 2, \dots, n\},$$

where $T = (t_{ij})$ is an $m \times n$ matrix whose elements are randomly chosen from $\{1, 2, \dots, 10\}$, t is an m-dimensional vector whose ith component is defined by $t_i = \sum_{j=1}^n t_{ij}^2$, and $\hat{t} = \max_{i=1,2,\dots,m} t_i$. We generated the coefficient vectors c_i of the objective function f by randomly drawing elements from the set $\{1, 2, \dots, 10\}$ and we set $d_i = 0$ for all i. We used $\omega = 9$ to generate $p + 1$ weight vectors w^k in Step 1 for the both methods. To find an optimal solution of problem P we enumerated all of the efficient vertices of problem MC by ADBASE developed by Steuer [15,16].

The numerical results for the heuristic method and the modified heuristic method are shown in the Tables I and II. Each row of the tables gives average statistics of ten problem instances except for the sixth and seventh columns. The fourth column *efficient* gives the average number of efficient vertices found by ADBASE. As proposed in Benson-Boger [2], to evaluate the approximate solution obtained we use the efficiency rating r defined by

$$r = \frac{z_{\max} - \tilde{z}}{z_{\max} - z_{\min}},$$

Table I. Computational results for heuristic method

p	m	n	efficient	r	r_{best}	r_{worst}	time
2	25	20	21.7	0.999997	1	0.999992	0.21
	25	30	17.0	0.999934	1	0.999890	0.36
	30	40	24.1	0.998704	1	0.969217	0.69
	40	30	33.0	0.999578	0.999951	0.997889	0.57
	40	50	28.5	0.999273	0.999997	0.992651	1.62
	50	40	38.3	0.999019	0.999917	0.997966	1.43
	50	60	45.6	0.985371	0.999062	0.916732	3.67
	60	70	55.9	0.940832	1	0.886512	6.82
3	25	20	331.0	0.999948	1.000000	0.999777	0.28
	25	30	756.4	0.999135	0.999992	0.995413	0.35
	30	40	681.3	0.995911	0.999597	0.990016	1.06
	40	30	727.3	0.999028	1.000000	0.995499	0.79
	40	50	1809.0	0.989379	0.999441	0.918471	1.67
	50	40	1645.7	0.990051	0.999010	0.989852	1.38
	50	60	2332.3	0.974496	0.991431	0.899949	3.81
4	25	20	1789.7	0.999186	1.000000	0.993378	0.28
	25	30	6732.2	0.995488	0.999687	0.990098	0.94
	30	40	13618.7	0.999766	1	0.994115	0.99
	40	30	17645.0	0.990003	0.999344	0.941095	0.96
	40	50	20234.5	0.992779	0.998805	0.990076	2.13
	50	40	33192.1	0.963425	0.997352	0.903949	1.71
5	10	20	843.7	0.998356	0.999901	0.975573	0.27
	20	10	452.0	0.974928	0.997666	0.948753	0.15
	25	30	24815.6	0.899201	0.967415	0.844599	1.93

where $\tilde{z}$ is $f(\tilde{x})$, the objective function value of the approximate solution obtained by each of our methods, and $z_{\max}$ and $z_{\min}$ are the maximum and minimum objective function values, respectively among efficient vertices of problem MC. Note that $z_{\min}$ is the optimal value of P and that the closer is $\tilde{z}$ to $z_{\min}$, the closer is the efficiency rating r to 1. The average efficiency ratings are shown in the fifth column under r. The best and worst efficiency ratings of ten instances are shown in the sixth and seventh columns under r_{best} and r_{worst}. We write $r_{\text{best}} = 1$ when $\tilde{z} = z_{\min}$, i.e., the method provides an exact optimal solution, which should be distinguished from $r_{\text{best}} = 1.000000$ meaning $|1 - r_{\text{best}}| < 10^{-6}$. We will use the same notation in Table II. The last column *time* gives the average CPU time in seconds that the heuristic method needed.

Table II. Computational results for modified heuristic method

p	m	n	efficient	r	r_{best}	r_{worst}	time	r_{BB}
2	25	20	21.7	1.000000	1	1.000000	0.25	1.000
	25	30	17.0	0.999995	1	0.999988	0.42	1.000
	30	40	24.1	1.000000	1	0.999992	0.78	1.000
	40	30	33.0	0.999997	1.000000	0.999982	0.61	0.999
	40	50	28.5	0.999991	1.000000	0.999935	1.66	1.000
	50	40	38.3	0.999936	0.999992	0.999893	1.45	0.999
	50	60	45.6	0.998955	0.999654	0.976732	4.17	1.000
	60	70	55.9	0.992580	1	0.916112	7.45	1.000
3	25	20	331.0	0.999991	1.000000	0.999987	0.34	0.985
	25	30	756.4	0.999935	1.000000	0.999881	0.39	0.960
	30	40	681.3	0.999802	0.999966	0.997695	1.22	0.987
	40	30	727.3	0.999971	1.000000	0.999517	0.91	0.993
	40	50	1809.0	0.997156	0.999755	0.941718	2.02	0.920
	50	40	1645.7	0.998923	0.999181	0.990021	1.68	0.993
	50	60	2332.3	0.992008	0.999439	0.945631	4.26	0.995
4	25	20	1789.7	0.999912	1.000000	0.999548	0.38	0.998
	25	30	6732.2	0.999022	0.999973	0.997893	1.11	0.992
	30	40	13618.7	0.999941	1	0.999869	1.25	0.986
	40	30	17645.0	0.997706	0.999512	0.942831	1.19	0.978
	40	50	20234.5	0.998237	0.999945	0.991421	2.17	0.968
	50	40	33192.1	0.990254	0.999070	0.959515	1.84	0.969
5	10	20	843.7	0.999344	0.999994	0.996888	0.29	0.993
	20	10	452.0	0.991058	0.999159	0.977657	0.16	0.998
	25	30	24815.6	0.958771	0.987431	0.896523	2.23	0.995

Table III. Computational results for larger problems

p	m	n	$(z_{\text{h}} - z)/z$	$(z_{\text{mh}} - z)/z$	$(z_{\text{BB}} - z)/z$
2	70	80	0.2587	0.0000	28.2708
	80	90	0.1407	0.0069	0.6644
3	60	70	0.6680	0.0000	2.1154
	70	80	3.1909	0.0957	8.8030
4	50	60	1.3577	0.5232	0.7096
	60	70	2.3446	0.0848	4.4663
5	30	40	1.3143	0.0000	4.5893
	40	50	0.2564	0.0000	1.9552

We observe in Table I that the quality of the approximate solutions obtained deteriorates slightly as the size of problem increases. But the average efficiency ratings and even the worst efficiency ratings are very close to 1.0 irrespective of p as long as the problem size is relatively small. Note also that the growth of CPU time is very mild with the increase of problem size.

The result of the modified heuristic method for the same problem instances is shown in Table II, where we see better efficiency ratings than those in Table I. The efficiency ratings reported in Benson-Boger [2] are also given in the right most column under r_{BB}. Benson-Boger's heuristic method performs quite well when $p = 2$. It accounts for this phenomenon that their method could be viewed as applying the parametric method proposed in Appendix for discrete values of parameter α. See Theorem A.1. Note that the problem instances solved here, though generated in the same manner, are not identical to those in Benson-Boger [2]. Hence simple comparison should not lead to any conclusion.

To compare the efficiency of the methods for larger problems, we generated additional 80 problem instances of larger size in the same manner. Since the problem instances are too large to apply ADBASE to enumerate all the efficient vertices, instead of the efficiency rating we compared the objective function values z_{h}, z_{mh} and z_{BB} yielded by the three methods: heuristic method, modified heuristic method and Benson-Boger's method. We set $z := \min\{z_{\mathrm{h}}, z_{\mathrm{mh}}, z_{\mathrm{BB}}\}$ and show in Table III the average statistics of relative errors $(z_{\mathrm{h}} - z)/z$, $(z_{\mathrm{mh}} - z)/z$ and $(z_{\mathrm{BB}} - z)/z$ for ten problem instances except for the case of $(p, m, n) = (2, 70, 80)$ where, eliminating an outlier, the averages for nine instances are reported. Note that the column of $(z_{mh} - z)/z$ contains a good many zeroes. This means that the modified heuristic method provided the best solution for almost all problem instances and hence its superiority over the other two methods.

Acknowledgments

The authors thank Akiko Yoshise and Maiko Shigeno for their comments on the earlier version of this paper. They are grateful to Prof. R.E. Steuer for providing ADBASE, without which the proposed heuristic method could not have been evaluated. They thank also anonymous referees and the editors for their valuable comments. The third author is supported by Grant-in-Aid for Scientific Research of the Ministry of Education, Science and Culture, Grant No. 10680419.

Appendix

We will propose a new parametric method that yields an optimal solution for problem P when $p = 2$.

Suppose that each term $c_i x + d_i$ is bounded on D for $i = 1, 2, \dots, p$ and let $\underline{y}$ and $\overline{y}$ be p-dimensional vectors such that $\underline{y} \leqslant Cx + d \leqslant \overline{y}$ holds for all $x \in D$. For a nonempty subset $J \subseteq \{1, 2, \dots, p\}$ and for $\alpha \in [0, 1]$ let us consider the vector

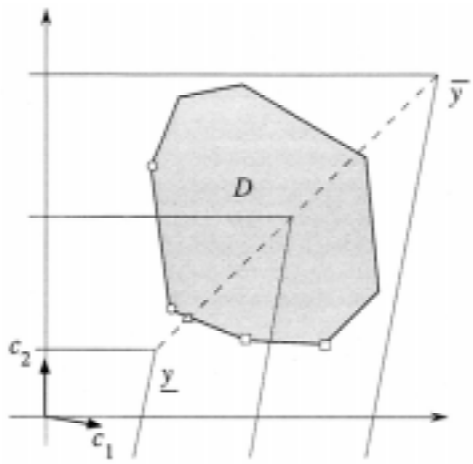

Figure 1. Parametric method for P with $p = 2$.

minimization problem

$$P_J(\alpha)\left|\begin{array}{ll} \text{vector minimize} & C_J x \\ \text{subject to} & x \in D;\ Cx + d \leqslant (1-\alpha)\overline{y} + \alpha \underline{y}, \end{array}\right.$$

where C_J is the matrix of rows c_j^{T} for $j \in J$.

LEMMA A.1. *Every optimal solution of P is an efficient point of $P_J(\alpha)$ for some nonempty proper subset J of* $\{1, 2, \ldots, p\}$ *and for some* $\alpha \in [0, 1]$.

Proof. Let x^* be an optimal solution of P and let

$$\alpha^* := \max\{\alpha \mid Cx^* + d \leqslant (1-\alpha)\overline{y} + \alpha\underline{y}\}.$$

Then clearly $\alpha^* \in [0, 1]$ and $C_{J'}x^* + d_{J'} = (1-\alpha^*)\overline{y}_{J'} + \alpha^*\underline{y}_{J'}$, holds for some nonempty subset J' of $\{1, 2, \ldots, p\}$, where $d_{J'}$, $\overline{y}_{J'}$ and $\underline{y}_{J'}$ denote the vectors consisting of the jth components for $j \in J'$ of d, $\overline{y}$ and $\underline{y}$, respectively. Let J be $\{1, 2, \ldots, p\}\backslash J'$ when $J' \neq \{1, 2, \ldots, p\}$, an arbitrary proper subset of $\{1, 2, \ldots, p\}$ when $J' = \{1, 2, \ldots, p\}$ and suppose x^* is not an efficient point of $P_J(\alpha^*)$. Namely, we suppose there is a point $\tilde{x} \in D$ such that $C\tilde{x} + d \leqslant (1-\alpha^*)\overline{y} + \alpha^*\underline{y}$, $C_J\tilde{x} \leqslant C_J x^*$ and $C_J\tilde{x} \neq C_J x^*$. Since $C_{J'}\tilde{x} + d_{J'} \leqslant C_{J'}x^* + d_{J'}$ by the definition of α^*, this clearly leads to a contradiction. □

When $p = 2$, the multicriteria problem $P_J(\alpha)$ reduces to an ordinary linear program and we obtain the following theorem.

THEOREM A.1. *When $p = 2$, an optimal solution of P is among the vertices to be encountered while solving the two parametric linear programming $P_{\{1\}}(\alpha)$ and $P_{\{2\}}(\alpha)$.*

This theorem provides us with a new parametric approach to a solution of P for $p = 2$.

New parametric method
Step 1
for $i = 1$ and 2 do
solve $P_{\{i\}}(\alpha)$ parametrically for $\alpha \in [0, 1]$ and store the vertices to be encountered
end for
Step 2
evaluate the function f at the vertices and choose one with the minimum value

The figure illustrates the case of two-dimension, where the bold arrows show c_1^{T} and c_2^{T}. Solving $P_{\{1\}}(\alpha)$ parametrically, one yields the vertices denoted by a circle, while the vertices denoted by a square are enumerated in solving $P_{\{2\}}(\alpha)$. The triangle shows the point where the both parametric linear programs end up.

When $p > 2$, solving even p parametric linear programs will not always provide an optimal solution of P. As Theorem A.1 shows, if one enumerates all efficient points for every nonempty proper subset J of $\{1, 2, \ldots, p\}$ and for every value of parameter α, one would obtain an optimal solution among the vertices to be enumerated.

References

1. M. Avriel, W.E. Diewert, S. Schaible and I. Zang (1988), Generalized Convexity, New York: Plenum Press.
2. H.P. Benson and G.M. Boger (1997), Multiplicative programming problems: analysis and efficient point search heuristic, *Journal of Optimization Theory and Applications* 94: 487–510.
3. F. Forgo (1975), The solution of special quadratic programming problem, *SZIGMA* 8: 53–59 (in Hungarian).
4. F. Forgo (1988), *Nonconvex Programming*, Budapest, Hungary: Akademiai Kiado.
5. H. Konno, T. Kuno and Y. Yajima (1992), Parametric simplex algorithm for a class of NP-complete problems whose average number of steps is polynomial, *Computational Optimization and Applications* 1: 227–239.
6. H. Konno and T. Kuno (1992), Linear mutliplicative programming, *Mathematical Programming* 56: 51–64.

7. H. Konno and T. Kuno (1995), Multiplicative programming, in: R. Horst and P.M. Pardalos (eds.), *Handbook of Global Optimization* (pp. 369–405) Dordrecht, Boston/London: Kluwer Academic Publishers.
8. T. Kuno (1999), A finite branch-and-bound algorithm for linear multiplicative programming, ISE-TR-99-159. Institute of Information Sciences and Electronics, University of Tsukuba.
9. T. Kuno, Y. Yajima and H. Konno (1993), An outer approximation method for minimizing the product of several convex functions on a convex set, *Journal of Global Optimization* 3: 325–335.
10. T. Matsui (1996), NP-hardness of linear multiplicative programming and related problems, *Journal of Global Optimization* 9: 113–119.
11. P.M. Pardalos and S.A. Vavasis (1991), Quadratic progamming with one negative eigenvalue is NP-hard, *Journal of Global Optimization* 1: 15–22.
12. S. Park (1997) LPAKO ver4.0f code for Linear Programming Program, Seoul National University.
13. H.S. Ryoo and N.V. Sahinidis (1996), A branch-and-reduce approach to global optimization, *Journal of Global Optimization* 8: 107–138.
14. R.E. Steuer (1986), *Multiple Criteria Optimization: Theory, Computation, and Application*. New York: John-Wiley and Sons.
15. R.E. Steuer (1994), Random problem generation and the computation of efficient extreme points in multiple objective linear programming, *Computational Optimization and Applications* 3: 333–347.
16. R.E. Steuer (1995), Manual for the ADBASE: Multiple Objective Linear Programming, mimeo.
17. K. Swarup (1966a), Programming with indefinite quadratic function with linear constraints, *Cahiers du Centre d'Études de Recherche Opérationnelle* 8: 132–136.
18. K. Swarup (1966b), Indefinite quadratic programming, *Cahiers du Centre d'Études de Recherche Opérationnelle* 8: 217-222.
19. K. Swarup (1966c), Quadratic programming, *Cahiers du Centre d'Études de Recherche Opérationnelle* 8: 223–234.
20. N.V. Thoai (1991), A global optimization approach for solving the convex multiplicative programming problem, *Journal of Global Optimization* 1: 341–357.
21. H. Tuy (1991), Polyhedral annexation, dualization and dimension reduction technique in global optimization, *Journal of Global Optimization* 1: 229–244.