# A Global Optimization Approach for Solving the Convex Multiplicative Programming Problem[1]

NGUYEN VAN THOAI[2]
*Fachbereich IV-Mathematik, Universität Trier, 5500 Trier, Germany.*

(Accepted: 16 October 1991)

**Abstract.** We consider a *convex multiplicative programming problem* of the form

$$\min\{f_1(x) \cdot f_2(x) : x \in X\},$$

where $X$ is a compact convex set of $\mathbb{R}^n$ and $f_1, f_2$ are convex functions which have nonnegative values over $X$.

Using two additional variables we transform this problem into a problem with a special structure in which the objective function depends only on two of the $(n+2)$ variables. Following a decomposition concept in global optimization we then reduce this problem to a *master problem* of minimizing a quasi-concave function over a convex set in $\mathbb{R}^2_+$. This master problem can be solved by an outer approximation method which requires performing a sequence of simplex tableau pivoting operations. The proposed algorithm is finite when the functions $f_i$, $(i = 1, 2)$ are affine-linear and $X$ is a polytope and it is convergent for the general convex case.

**Key words.** Multiplicative programming, global optimization, decomposition, outer approximation.

## 1. Introduction

We are dealing with a problem of minimizing the product of two convex functions over a compact convex set in $\mathbb{R}^n$ (*convex multiplicative programming problem*) which we denote by $(MP)$.

In Swarup (1966), Bector and Dahl (1974), Schaible (1976), Aneja *et al.* (1984), and Pardalos (1988) a special class of (MP) when the objective function is a product of two linear functions and the feasible set is a polytope (the *linear case*) was considered in connection with solving certain classes of quadratic programming problems. Efficient methods for the linear case which were established based on different parametric approaches are due to Gabasov and Kirillova (1980), Konno and Kuno (1989), and Tuy and Tam (1990). In Kuno and Konno (1990) a parametric algorithm was also applied to solving the general convex case.

In recent years a new direction of mathematical programming called ‚global optimization has attracted the attention of many mathematicians as well as engineers and economists. Results in this field provide algorithms for solving a great variety of problems for which the standard methods fail. An excellent

[1] Partly supported by the Deutsche Forschungsgemeinschaft Project CONMIN.
[2] On leave from Institute of Mathematics Hanoi, Vietnam.

*Journal of Global Optimization* **1**: 341–357, 1991.

representation of the most important deterministic methods in global optimization is given in the book of Horst and Tuy (1990).

It is the purpose of the present article to propose a new method for solving the convex multiplicative programming problem by applying certain global optimization techniques. The main idea of this method is to transform problem $(MP)$ into a problem in $\mathbb{R}^{n+2}$ whose objective function depends only upon two of the $n+2$ variables. The special structure of the transformed problem suggests reducing it to a so-called *master problem* in $\mathbb{R}^2_+$ that can be solved by a simple outer approximation procedure even when its feasible set is not explicitly available.

Computational experiments show that this method is quite efficient. The main advantages affecting numerical efficiency are:

(a) within the outer approximation procedure for solving the master problem in $\mathbb{R}^2_+$ the methods discussed in Horst *et al.* (1988) for computing all new vertices of a polytope generated from a given polytope by an affine cut can be very efficiently applied and

(b) the construction of cutting planes requires mainly only the well-known simplex tableau pivoting operations.

The paper is organized as follows. In Section 2, a conceptual algorithm is established. This basic algorithm is then implemented for the linear case and the convex case in Sections 3 and 4, respectively. Illustrative examples and computational experiments are reported in Section 5. In Section 6 we discuss an extension of our method for solving a generalized multiplicative programming problem of minimizing the product of more than two convex functions over a compact convex set.

## 2. The Basic Algorithm

Consider the following Convex Multiplicative Programming Problem

$$(MP)\qquad \begin{array}{ll} \min & f_1(x)\cdot f_2(x) \\ \text{s.t.} & g_i(x)\leq 0,\ i=1,\ldots,m \\ & x\geq 0 \end{array}$$

where $f_1$, $f_2$ and $g_i$, $(i=1,\ldots,m)$ are convex functions on $\mathbb{R}^n$.

Defining

$$g(x)=\max\{g_i(x): i=1,\ldots,m\} \tag{2.1}$$

we obtain a convex function $g(x)$, and the feasible set of problem $(MP)$, denoted by $X$, can be formulated as

$$X:=\{x\in\mathbb{R}^n:\ g(x)\leq 0,\ x\geq 0\} \tag{2.2}$$

One usually assumes that the set $X$ is bounded and $f_1, f_2$ are nonnegative in $X$. In fact, we can assume without loss of generality that

$$f_1(x) \geqslant 0 \quad \text{and} \quad f_2(x) \geqslant 0 \quad \text{for all } x \in \mathbb{R}^n, \tag{2.3}$$

because, under the assumption that $f_1, f_2$ are nonnegative in $X$, replacing $f_1$ and $f_2$ by the convex functions $\max\{0; f_1(x)\}$ and $\max\{0; f_2(x)\}$, respectively, does not have any influence on the solutions of problem (MP).

From assumption (2.3), by using two additional variables $y_1$, $y_2$, we can transform problem $(MP)$ into the following problem:

$$\min y_1 \cdot y_2 \tag{2.4}$$

$$\text{s.t. } g(x) \leqslant 0 \tag{2.5}$$

$$-y_1 + f_1(x) \leqslant 0 \tag{2.6}$$

$$(AP) \qquad -y_2 + f_2(x) \leqslant 0 \tag{2.7}$$

$$x \geqslant 0 \tag{2.8}$$

$$y = (y_1, y_2)^T \geqslant 0 \tag{2.9}$$

Let us denote the convex feasible set of $(AP)$ by $\Omega$.

Although problem $(AP)$ has now $n+2$ variables, the above transformation is worthwhile since the objective function of $(AP)$ does only depend on the two variables $y_1$, $y_2$. This special structure suggests applying a decomposition concept in global optimization that reduces $(AP)$ to the following *master problem* in $\mathbb{R}^2$ (cf. Horst and Tuy (1990)).

$$(CP) \qquad \min\{y_1 \cdot y_2 : y \in D \subset \mathbb{R}^2\}$$

where

$$D := \{y \geqslant 0 : (\exists x \geqslant 0) \text{ such that (2.5)–(2.7) are fulfilled}\} \tag{2.10}$$

Since the set $D$, being the projection of the convex set $\Omega$ on $\mathbb{R}^2$, is convex and the function $F(y) = y_1 \cdot y_2$ is quasi-concave (as shown in Konno and Kuno (1989)), we see that $(CP)$ is an ordinary concave minimization problem. Moreover, we have

PROPOSITION 1. (a) *The set of optimal solutions of problem* $(CP)$ *is bounded.* (b) *If* $y^*$ *is an optimal solution of* $(CP)$, *then every point* $x^* \in \mathbb{R}^n$ *satisfying* $(y^*, x^*) \in \Omega$ *is an optimal solution of* $(MP)$.

*Proof.* (a) It is easy to see that each optimal solution $(y, x)$ of problem (AP) satisfies $y_i = f_i(x)$, $i = 1, 2$. But $f_i (i = 1, 2)$ are convex functions on $\mathbb{R}^n$, and hence bounded over the bounded set $X$. Therefore, the set of all optimal solutions of (AP) is bounded, and hence its projection on $y$-space is bounded as well.

(b) Problem $(AP)$ can be rewritten as $\min_{x \in D} \min\{y_1 \cdot y_2 : (y, x) \in \Omega\}$. Therefore, if $y^*$ is an optimal solution of $(CP)$ and $x^* \in \mathbb{R}^n$ such that $(y^*, x^*) \in \Omega$, then $(y^*, x^*)$ is obviously an optimal solution of problem $(AP)$ which implies that $x^*$ is an optimal solution of $(MP)$. □

The above consideration suggests to solve the concave minimization problem ($CP$), instead of the original problem ($MP$). As discussed in Horst and Thoai (1989), for concave programming problems in small dimensions it is suitable to apply the following Outer Approximation Algorithm.

ALGORITHM 1.
*Initialization*. Construct a cube

$$S^0 = \{y \in \mathbb{R}^2 : 0 \leq y_i \leq \gamma_i < +\infty, i = 1, 2\} \tag{2.11}$$

which contains the bounded set of optimal solutions to ($CP$). Let $V(S^0)$ denote the vertex set of $S^0$.
Set $k \leftarrow 0$.

*Iteration k*. Solve $\min\{y_1 \cdot y_2 : y \in S^k\} = \min\{y_1 \cdot y_2 : y \in V(S^k)\}$ obtaining a solution $y^k$.

*k.1.* If $y^k \in D$, then the algorithm terminates yielding $y^k$ as an optimal solution of ($CP$).

*k.2.* Otherwise, construct an affine function $l_k(y) = c^k y + c_0^k$ such that $l_k(y^k) > 0$ and $l_k(y) \leq 0$ for all $y \in D$. Build $S^{k+1} = S^k \cap \{y \in \mathbb{R}^2 : l_k(y) \leq 0\}$. Compute the vertex set $V(S^{k+1})$ (from $V(S^k)$) and go to iteration $k + 1$.

When implementing the above procedure we have to perform the following basic operations at each iteration $k$:

(a) the computation of the set $V(S^{k+1})$ from $V(S^k)$.
(b) the check whether $y^k$ belongs to $D$, and, if not, the construction of a cut $l_k(y)$.

Moreover, when Algorithm 1 has detected an optimal solution $y^k$ of ($CP$) we have to deduce an optimal solution of the original problem ($MP$) from $y^k$.

In general (a) is a troublesome problem for the number of vertices of $S^k$ can grow exponentially. However, as shown in Horst *et al.* (1988), if this problem is considered in a space of small dimension such as $\mathbb{R}^q$ with $q \leq 20$, then the algorithms discussed there can be regarded as relatively efficient. In connection with our problem where $q = 2$ we shall see later in Section 5 that the polytopes $S^k$ usually have a few vertices and hence the computation of the sets $V(S^k)$ only requires a negligible amount of time.

The basic operations (b) will be dealt with in the next two sections for the case that the functions $f_i$, ($i = 1, 2$) and $g_i$, ($i = 1, \ldots, m$) are all affine-linear and for the general convex case, respectively.

## 3. The Linear Case

In this section we consider the linear case of problem ($MP$), i.e.,

$$(\widetilde{MP}) \qquad \begin{aligned} \min \quad & (\alpha_1^T x + \beta_1) \cdot (\alpha_2^T x + \beta_2) \\ \text{s.t.} \quad & \tilde{A} x \leq \tilde{b} \\ & x \geq 0 \end{aligned}$$

where $\alpha_i \in \mathbb{R}^n$, $\beta_i \in \mathbb{R} (i = 1, 2)$, $\tilde{A}$ is an $(\tilde{m} \times n)$-matrix and $\tilde{b} \in \mathbb{R}^{\tilde{m}}$. (As usual we denote the transpose of a vector (matrix) by $\cdot^{\mathrm{T}}$).

The corresponding problem $(AP)$ that we introduced in Section 2 becomes

$$(\widetilde{AP}) \qquad \begin{aligned} & \min y_1 \cdot y_2 \\ & \text{s.t. } By + Ax \leq b \\ & \qquad\qquad y \geq 0 \\ & \qquad\qquad x \geq 0 \end{aligned}$$

where $B$ is an $(m \times 2)$-matrix, $A$ an $(m \times n)$-matrix and $b \in \mathbb{R}^{\mathrm{m}}$ with $m = \tilde{m} + 2$. More precisely we have

$$B = \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}, \qquad A = \begin{pmatrix} \tilde{A} \\ \alpha_1^T \\ \alpha_2^T \end{pmatrix}, \qquad b = \begin{pmatrix} \tilde{b} \\ -\beta_1 \\ -\beta_2 \end{pmatrix}. \tag{3.1}$$

Thus, the feasible set of the concave minimization problem $(CP)$ has here the form

$$D := \{ y \geq 0 : (\exists x \geq 0)\ By \leq b - Ax \} . \tag{3.2}$$

The following proposition provides a representation of $D$ which we use to implement Algorithm 1 for the linear case. For a related representation, see Lemma 1 in Tuy (1985).

Let $e = (1, \ldots, 1)^T \in \mathbb{R}^m$.

PROPOSITION 2. *The set $D$ defined by* (3.2) *is a polyhedral convex set consisting of all points $y \geq 0$ satisfying*

$$(LP) \qquad \max\{ (By - b)^T z : -A^T z \leq 0,\ e^T z \leq 1,\ z \geq 0 \} = 0 .$$

*Proof.* Since the feasible set of problem $(AP)$ is a polyhedral convex set and $D$ is its image under a linear transformation from $\mathbb{R}^{n+2}$ to $\mathbb{R}^2$, it follows that $D$ is a polyhedral convex set as well (cf. Rockafellar (1970), Theorem 19.3). Now, let us assert that the set $D$ consists of all points $y \geq 0$ such that

$$\min\{ t : -Ax + te \geq By - b,\ x \geq 0,\ t \geq 0 \} = 0 \tag{3.3}$$

Indeed, let $y \geqslant 0$ satisfy (3.3). Then there exists a point $x \geqslant 0$ such that $-Ax + te \geqslant By - b$ and $t = 0$. This implies by definition that $y \in D$. Conversely, for each $y \in D$ we have, from the definition of $D$: $(\exists x \geqslant 0)$ $By \leqslant b - Ax + te$ for all $t \geqslant 0$, which implies (3.3).

Finally, it is easy to verify that the linear program $(LP)$ is the dual of (3.3) and hence the proposition follows. □

The above assertion provides a hyperplane which separates the set $D$ from an arbitrary point outside $D$.

PROPOSITION 3. *For each point $\bar{y} \in \mathbb{R}^2_+ \setminus D$ there exists a vertex $\bar{z}$ of the polytope*

$$Q = \{z : -A^T z \leqslant 0, e^T z \leqslant 1, z \geqslant 0\}.$$

*such that the affine function*

$$l(y) = \bar{z}^T By - \bar{z}^T b$$

*satisfies $l(y) \leqslant 0$ for all $y \in D$, and $l(\bar{y}) > 0$.*

*Proof.* Since for each $y \geqslant 0$ the linear function $(By - b)^T z$ attains its maximum at a vertex of the polytope $Q$, and this maximum, being equal to the optimal value of the dual problem in (3.3), is nonnegative, it follows from Proposition 2 that there exists a vertex (basic feasible point) $\bar{z}$ of $Q$ satisfying

$$\begin{aligned} l(y) &= \bar{z}^T By - \bar{z}^T b = (By - b)^T \bar{z} \\ &\leqslant \max\{(By - b)^T z : -A^T z \leqslant 0, e^T z \leqslant 1, z \geqslant 0\} = 0 \text{ for all } y \in D \end{aligned}$$

and

$$\begin{aligned} &0 < l(\bar{y}) = (B\bar{y} - b)^T \bar{z} \leqslant \max\{(B\bar{y} - b)^T z : -A^T z \leqslant 0, e^T z \leqslant 1, z \geqslant 0\} \\ &\text{for } \bar{y} \not\in D. \end{aligned}$$

□

By means of the above two propositions we can establish the following algorithm for the linear case $(\widetilde{MP})$.

ALGORITHM 2.

*Initialization*. Construct a cube $S^0$ and its vertex set $V(S^0)$ as in Algorithm 1. Set $k \leftarrow 0$.

*Iteration k*. Solve $\min\{y_1 \cdot y_2 : y \in V(S^k)\}$ obtaining a solution $y^k$. Apply the simplex method to the following linear program:

$$(LP_k) \qquad \max\{(By^k - b)^T z : -A^T z \leqslant 0, e^T z \leqslant 1, z \geqslant 0\}.$$

*k.1*. If a basic feasible point $z^k$ with $(By^k - b)^T z^k > 0$ is obtained, then construct the affine function

$$l_k(y) = (z^k)^T By - (z^k)^T b.$$

Set $S^{k+1} = S^k \cap \{y \in \mathbb{R}^2 : l_k(y) \leq 0\}$. Compute $V(S^{k+1})$ and go to iteration $k+1$.

*k.2.* Otherwise, the optimal value of $(LP_k)$ is equal to zero. Then $y^k \in D$ (Proposition 2), and hence $y^k$ solves $(CP)$. The algorithm terminates.

PROPOSITION 4. *Algorithm 2 terminates after a finite number of iterations.*

*Proof.* Since for all $k$ the common feasible set of linear programs $(LP_k)$ is a polytope having a finite number of vertices, we only need to show that, whenever the algorithm does not terminate at iteration $k$, the point $z^k$ must be different from $z^q$ for all $q = 0, \ldots, k-1$. Since $S^k = S^0 \cap \{y : l_q(y) \leq 0,\ q = 0, \ldots, k-1\}$, and $y^k \in S^k$ we have $l_q(y^k) = (By^k - b)^T z^q \leq 0$ for all $q = 0, \ldots, k-1$. Therefore, $z^k \neq z^q$ for $q = 0, \ldots, k-1$ because $(By^k - b)^T z^k > 0$. □

Let Algorithm 2 terminate at some iteration $k$ with $y^k \in D$. We show how we can obtain an optimal solution of $(\widetilde{MP})$ from $y^k$. We have

$$\max\{(By^k - b)^T z : -A^T z \leq 0,\ e^T z \leq 1,\ z \geq 0\} = 0 .$$

Assume further that the last row of last simplex tableau has the form

| | |
|---|---|
| $z^*$ | $\bar{c}_1, \ldots, \bar{c}_{m+n+1}$ |

with the optimal value $z^* = 0$ and $\bar{c}_j \leq 0$, $j = 1, \ldots, m+n+1$. (Note that the matrix $A^T$ has $n$ rows and $m$ columns).

From the duality theory of linear programming it follows that the point $(x^k, t^k) \in \mathbb{R}^{n+1}$ with $x_i^k = -\bar{c}_{m+i}$, $i = 1, \ldots, n$ and $t^k = -\bar{c}_{m+n+1}$ is an optimal solution of the dual problem and we have

$$\min\{t : -Ax + te \geq By^k - b,\ x \geq 0,\ t \geq 0\} = 0 .$$

This implies that $(y^k, x^k)$ is a feasible point to problem $\widetilde{AP}$ and hence it follows from Proposition 1 that $x^k$ is an optimal solution of problem $(\widetilde{MP})$.

It is worth noting that the main operation in Algorithm 2 is solving the linear programs $(LP_k)$. Since each of these programs differs from others only in the objective function, these programs can be solved efficiently by successively pivoting the simplex tableaux which correspond to the polytope $\{z : -A^T z \leq 0,\ e^T z \leq 1,\ z \geq 0\} \subset \mathbb{R}^m$.

Algorithm 2 is illustrated in Section 5, Example 1.

## 4. The Convex Case

In this section we are dealing with the general convex multiplicative programming problem $(MP)$ as formulated in Section 2.

The method which we propose here can be regarded as a natural extension of the above method for the linear case. The main idea of going from Algorithm 2 to an algorithm for the convex case can be explained as follows.

The convex feasible set $\Omega$ of problem $(AP)$ will be iteratively approximated by a sequence of convex polyhedral sets, say $\Omega^0, \Omega^1, \ldots$, such that $\Omega^0 \supset \Omega^1 \supset \cdots \supset \Omega$. Each polyhedral set $\Omega^k$ is defined by a system of the form

$$B^k y + A^k x \leq b^k\,, \quad y, x \geq 0 \tag{4.1}$$

where $y \in \mathbb{R}^2$, $x \in \mathbb{R}^n$ and $B^k$, $A^k$, $b^k$ are of appropriate sizes.

Let $S^k$ be a polytope that contains the projection of the optimal solution set of problem $(AP)$ on the $y$-space, and let $y^k$ be an optimal solution of problem $\min\{y_1 \cdot y_2 : y \in V(S^k)\}$. (Recall that by $V(S)$ we always denote the vertex set of a polytope $S$.) Furthermore, let $t^k$ be the optimal value of the linear subproblem

$$(LP_k) \qquad \max\{(B^k y^k - b^k)^T z : (-A^k)^T z \leq 0,\, e^T z \leq 1,\, z \geq 0\}$$

and $(x^k, t^k)$ an optimal solution of its dual problem

$$\min\{t : -A^k x + te \geq B^k y^k - b^k\,, \quad x, t \geq 0\}$$

The following three cases can occur:

CASE 1. $t^k = 0$ and $(y^k, x^k) \in \Omega$, i.e. $(y^k, x^k)$ satisfies the constraints (2.5)–(2.9) in problem $(AP)$. In this case, it follows from Proposition 2 that $y^k \in D^k$, where $D^k$ denotes the projection of $\Omega^k$ on $y$-space and $(y^k, x^k) \in \Omega^k$. Since $\Omega^k \supset \Omega$, it follows that $x^k$ is an optimal solution of the original problem $(MP)$.

CASE 2. $t^k = 0$ and $(y^k, x^k) \not\in \Omega$. To continue the procedure in this case we shall construct a set $\Omega^{k+1}$ by cutting off a part of $\Omega^k$ which contains the point $(y^k, x^k)$. More precisely, we construct an affine linear function $L_k(y, x)$ such that

$$L_k(y^k, x^k) > 0 \text{ and } L_k(y, x) \leq 0 \quad \forall (y, x) \in \Omega \tag{4.2}$$

The set $\Omega^{k+1}$ is then generated by adding the constraint $L_k(y, x) \leq 0$ to the system defining $\Omega^k$.

CASE 3. $t^k > 0$. From Proposition 2 we thus have $y^k \not\in D^k$ and hence $(y^k, x^k) \not\in \Omega$. In this case we shall construct two cutting planes. The one according to Proposition 3 cuts off $y^k$ to build $S^{k+1}$ and the other is used to construct $\Omega^{k+1}$ as in Case 2.

Now, we are in a position to present an algorithm for solving the general convex multiplicative programming problem.

ALGORITHM 3.

*Initialization*. Construct $S^0$ and $V(S^0)$ as in Algorithm 2. Construct a convex polyhedral set $\Omega^0$ that contains $\Omega$ and is defined by a system of the form (4.1) with $k = 0$. Solve $\min\{y_1 \cdot y_2 : y \in V(S^0)\}$ obtaining $y^0$.
Set $k \leftarrow 0$.

*Iteration k.* Solve linear subproblem $(LP_k)$ obtaining a basic optimal solution $z^k$ and an optimal solution $(x^k, t^k)$ of its dual problem (as in Section 3).

*k.1.* If $t^k = 0$ and $(y^k, x^k) \in \Omega$, then $x^k$ is an optimal solution of problem $(MP)$. The algorithm terminates.

*k.2.* If $t^k = 0$ and $(y^k, x^k) \not\in \Omega$, then set $S^{k+1} \leftarrow S^k$, $y^{k+1} \leftarrow y^k$ and go to *k.4*.

*k.3.* If $t^k > 0$, then construct $l_k(y) = (z^k)^T B^k y - (z^k)^T b^k$; set $S^{k+1} = S^k \cap \{y \in \mathbb{R}^2 : l_k(y) \leqslant 0\}$, compute $V(S^{k+1})$, and solve $\min\{y_1 \cdot y_2 : y \in V(S^{k+1})\}$ to obtain $y^{k+1}$.

*k.4.* Choose $L_k(y, x)$ satisfying (4.2). To construct $\Omega^{k+1}$, determine $B^{k+1}$, $A^{k+1}$ and $b^{k+1}$ by adding the constraint $L_k(y, x) \leqslant 0$ to the system defining $\Omega^k$ and go to iteration $k + 1$.

Before proving convergency of the above algorithm we give some details on implementation.

At the beginning of the procedure a first polyhedral convex set $\Omega^0 = \{(y, x) \in \mathbb{R}^{n+2} : B^0 y + A^0 x \leqslant b^0, (y, x) \geqslant 0\}$ can be defined by the following system:

$$\begin{aligned} e^T x &\leqslant \beta \\ -y_1 + (\alpha^{1,0})^T x &\leqslant \beta_{1,0} \\ -y_2 + (\alpha^{2,0})^T x &\leqslant \beta_{2,0} \\ y, x &\geqslant 0 \end{aligned} \tag{4.3}$$

where $e = (1, \ldots, 1)^T \in \mathbb{R}^n$, $\beta$ is a number (large enough) such that

$$X \subset X^0 := \{x \in \mathbb{R}^n : x \geqslant 0, e^T x \leqslant \beta\} \tag{4.4}$$

with $X$ being the bounded convex feasible set of problem $(MP)$ as defined in (2.2), $\alpha^{i,0} (i = 1, 2)$ are subgradients of $f_i (i = 1, 2)$, respectively, at some point $x^0 \in \mathbb{R}^n_+$, (e.g., $x^0 = 0$), and $\beta_{i,0} = (x^0)^T \alpha^{i,0} - f_i(x^0)$ $(i = 1, 2)$.

At iteration $k$, assume that $(y^k, x^k) \not\in \Omega$, i.e., $\max\{g(x^k); -y_1^k + f_1(x^k); -y_2^k + f_2(x^k)\} > 0$. Define $h(y, x) = \max\{g(x^k); -y_i^k + f_1(x^k); -y_2^k + f_2(x^k)\}$. Then $h(y, x)$ is obviously a convex function and we can construct an affine linear function $L_k(y, x)$ by

$$L_k(y, x) = \begin{cases} -y_i + (\alpha^k)^T x + f_i(x^k) - (\alpha^k)^T x^k, & \text{if } h(y^k, x^k) = -y_i^k + f_i(x^k) \\ & \text{for an } i \in \{1, 2\} \\ (\alpha^k)^T x + g(x^k) - (\alpha^k)^T x^k, & \text{if } h(y^k, x^k) = g(x^k) \end{cases} \tag{4.5}$$

where $\alpha^k$ denotes a subgradient at point $x^k$ of $f_i(x)$ if $h(y^k, x^k) = -y_i^k + f_i(x^k)$ for an $i \in \{1, 2\}$, or of $g(x)$ otherwise.

From the general theory of outer approximation methods it is easy to verify that $L_k(y, x)$ satisfies condition (4.2) (cf., e.g., Horst *et al.* (1987), Horst and Tuy (1990)). Thus, the set $\Omega^{k+1}$ is determined by adding the constraint $L_k(y, x) \leqslant 0$ to the system defining $\Omega^k$.

PROPOSITION 5. Assume that throughout Algorithm 3 the sets $\Omega^k$, $k \geqslant 0$ are constructed as described above. Then, when the algorithm does not terminate after a finite number of iterations, it generates an infinite sequence $\{(y^k, x^k)\}$, every accumulation point $(y^*, x^*)$ of which satisfies that $y^*$ is an optimal solution of problem $(CP)$ while $x^*$ is an optimal solution of the original problem $(MP)$.

*Proof.* At each iteration $k$ the algorithm generates a point $(y^k, x^k)$. Therefore, if the algorithm never terminates it must generate an infinite sequence $\{(y^k, x^k)\}$. Let $(y^*, x^*)$ be an accumulation point of this sequence. (Note that $(y^*, x^*)$ exists, since $y^k \in S^0$ and $x^k \in X^0$ for all $k$, where $S^0$ and $X^0$ are compact sets defined by (2.11) and (4.4), respectively).

If for each $k$ the cutting function is constructed as in (4.5), then it follows from an outer approximation concept (cf., e.g., Horst *et al.* (1987), Horst and Tuy (1990)) that $(y^*, x^*) \in \Omega$. Therefore $y^* \in D$, which implies that $y^*$ solves the concave programming problem $(CP)$, and hence, from Proposition 1, that $x^*$ solves the original problem $(MP)$. □

An illustrative example of Algorithm 3 is given in Section 5 (Example 2).

## 5. Illustrative Examples and Computational Experiments

Next, we give two examples to illustrate Algorithms 2 and 3, respectively.

EXAMPLE 1. We consider a linear multiplicative programming problem with following input data.

$\tilde{m} = 8$,
$\tilde{n} = 4$,

$$\tilde{A} = \begin{pmatrix} .488509 & .063565 & .945686 & .210704 \\ -.324014 & -.501754 & -.719204 & .099562 \\ .445225 & -.346896 & .637939 & -.257623 \\ -.202821 & .647361 & .920135 & -.983091 \\ -.886420 & -.802444 & -.305441 & -.180123 \\ -.515399 & -.424820 & .897498 & .187268 \\ -.591515 & .060581 & -.427365 & .579388 \\ .423524 & .940496 & -.437944 & -.742941 \end{pmatrix},$$

$$\tilde{b} = \begin{pmatrix} 3.562809 \\ -.052215 \\ .427920 \\ .840950 \\ -1.353686 \\ 2.137251 \\ -.290987 \\ .373620 \end{pmatrix}$$

$\alpha_1^T = (.813396,\ .674440,\ .305038,\ .129742)$, $\beta_1 = .217796$,
$\alpha_2^T = (.224508,\ .063458,\ .932230,\ .528736)$, $\beta_2 = .091947$

A first cube in $\mathbb{R}^2$:
$S^0 = \{y \in \mathbb{R}^2 : 0 \leq y_i \leq 100,\ i = 1, 2\}$

**Iteration 0:
Current best outer approximation point in $y$-space:
$y^0 = (.000000,\ .000000)^T$,
$F(y^0) = y_1^0 \cdot y_2^0 = .000000$,
A basic optimal solution of linear subproblem $(LP_0)$:
$z^0 = (0.0,\ 0.0,\ 0.0,\ 0.0,\ .448213,\ 0.0,\ .020334,\ 0.0,\ .531453,\ 0.0)^T$,
Optimal value of linear subproblem: .728404,
Cutting plane:
$l_0(y) = -.531453y_1 + .728404 \leq 0$

**Iteration 1:
Current best outer approximation point in $y$-space:
$y^1 = (100.000000,\ .000000)^T$,
$F(y^1) = .000000$,
A basic optimal value of linear subproblem $(LP_1)$:
$z^1 = (0.0,\ 0.0,\ 0.0,\ 0.0,\ .229767,\ 0.0,\ 0.0,\ .154495,\ 0.0,\ .615738)^T$,
Optimal value of linear subproblem: .309925,
Cutting plane:
$l_1(y) = -.615738y_2 + .309925 \leq 0$

**Iteration 2:
Current best outer approximation point in $y$-space:
$y^2 = (1.370590,\ .503340)^T$,
$F(y^2) = .689872$,
A basic optimal value of linear subproblem $(LP_2)$:
$z^2 = (0.0,\ 0.0,\ .082410,\ 0.0,\ .258050,\ 0.0,\ 0.0,\ .220977,\ 0.0,\ .438563)T$,
Optimal value of linear subproblem: .051071,
Cutting plane:
$l_2(y) = -.438563y_2 + .271817 \leq 0$
...

This problem was solved after 6 iterations.

Optimal solution of the concave minimization problem in $y$-space:

$$y^* = (1.436281,\ .619790)^T.$$

Optimal solution of the original linear multiplicative programming problem:

$$x^* = (1.314792,\ .139555,\ .000000,\ .423286)^T.$$

Optimal function value: .890193.

The maximal number of vertices of a polytope in $\mathbb{R}^2$ generated throughout the algorithm was 6.

EXAMPLE 2. We consider a convex multiplicative programming problem of form $(MP)$ in $\mathbb{R}^2$ with

$$f_1(x) = (x_1 + 2x_2 + 3x_2 + 0.2)^2 ,$$
$$f_2(x) = 0.1 \exp\left(x_1 + \frac{x_2}{2} + \frac{x_3}{3} + 0.5\right)$$
$$g_1(x) = \left|x_1 + \frac{1}{2} x_2 + \frac{2}{3} x3\right|^{1.5} - 4.2 ,$$
$$g_2(x) = \left(x_1 + \frac{x_2}{2} + \frac{x_3}{3} - 2\right)^2 - 3.5 ,$$
$$g_3(x) = -x_1 + 3,$$
$$g_4(x) = -x_2 + 4,$$
$$g_5(x) = -x_3 + 3 .$$

We set

$$g(x) = \max\{g_i(x) : i = 1, \ldots, 5\} .$$

A first cube in $\mathbb{R}^2$:
$s^0 = \{y \in \mathbb{R}^2 : 0 \leq y_i \leq 100,\ i = 1, 2\}$.
Computing a subgradient of $f_1$ and $f_2$, respectively, at point 0 we determine a first polyhedral set $\Omega^0$ by:

$$\begin{aligned} x_1 + x_2 + x_3 &\leq 10. \\ -y_1 + 0.4x_1 + 0.8x_2 + 1.2x_3 &\leq -0.04 \\ -y_2 + 0.164872x_1 + 0.082436x_2 + 0.054957x_3 &\leq -0.164872 \\ y_1, y_2, x_1, x_2, x_3 &\geq 0 \end{aligned}$$

**Iteration 0:
Current best outer approximation point in $y$-space:
$y^0 = (0.000000,\ 0.000000)^T$,
$F(y^0) = y_1^0 \cdot y_2^0 = 0.000000$,
$x^0 = (0.0,\ 0.0,\ 0.0)^T$, $t^0 = 0.164872$
$-y_1^0 + f_1(x^0) = 0.040000$, $-y_2^0 + f_2(x^0) = 0.164872$, $g(x^0) = 0.500000$
Cutting planes:
$l_0(y) = -y_2 + 0.164872 \leq 0$,
$L_0(y, x) = -4.0x_1 - 2.0x_2 - 1.333333x_3 + 0.5 \leq 0$

**Iteration 1:
Current best outer approximation point in $y$-space:
$y^1 = (0.000000,\ 100.000000)^T$,
$F(y^1) = 0.000000$,
$x^1 = (0.104545,\ 0.0,\ 0.0)^T$, $t^1 = 0.081818$

$-y_1^1 + f_1(x^1) = 0.092748$, $-y_2^1 + f_2(x^1) = -99.816958$, $g(x^1) = 0.092748$
Cutting planes:
$l_1(y) = -0.909091y_1 + 0.081818 \leq 0$,
$L_1(y, x) = -y_1 + 0.609091x_1 + 1.218182x_2 + 1.827273x_3 + 0.029070 \leq 0$

**Iteration 2:
Current best outer approximation point in $y$-space:
$y^2 = (0.090000, 0.164872)^T$,
$F(y^2) = 0.014838$,
$x^2 = (0.115892, 0.008319, 0.000000)^T$, $t^2 = 0.019793$
$-y_1^2 + f_1(x^2) = 0.020576$, $-y_2^2 + f_2(x^2) = 0.021030$, $g(x^2) = 0.034206$
Cutting planes:
$l_2(y) = -0.960414y_2 + 0.178139 \leq 0$,
$L_2(y, x) = -3.759897x_1 - 1.879948x_2 - 1.253299x_3 + 0.485588 \leq 0$
...

This problem was solved after 10 iterations.

Optimal solution of the concave minimization problem in $y$-space:

$$y^* = (0.108354, 0.187606)^T,$$

Optimal solution of the original convex multiplicative programming problem:

$$x^* = (0.129171, 0.000000, 0.000000)^T,$$

Optimal function value: 0.020328.
The maximal number of vertices of a polytope in $\mathbb{R}^2$ generated throughout the algorithm was 4.

A set of randomly generated problems was used to test the above algorithms. The test runs were performed on an IBM-PS2 computer, Modell 80, using codes written in FORTRAN 77. The test problems are of following three types.

*Type 1:*

$$\begin{array}{ll} \min & (\alpha_1^T x + \beta_1) \cdot (\alpha_2^T x + \beta_2) \\ \text{s.t.} & Ax \leq b \\ & x \geq 0. \end{array}$$

*Type 2:*

$$\begin{array}{ll} \min & f_1(x) \cdot f_2(x) \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{array}$$

with

$$f_1(x) = 0.1\Big(0.2 + \sum_{i=1}^{n} ix_i\Big)^2, \quad f_2(x) = 0.1\left|x_1 + \sum_{i=2}^{p} \frac{i-1}{i} x_i\right|^{1.5} + 0.6\,.$$

*Type 3:*

$$\begin{aligned} \min \quad & f_1(x) \cdot f_2(x) \\ \text{s.t.} \quad & Ax \leqslant b \\ & g_i(x) \leqslant 0, \; i = 1, 2, 3 \\ & x \geqslant 0 \end{aligned}$$

with

$$f_1(x) = \left(0.2 + \sum_{i=1}^{n} i x_i\right)^2, \quad f_2(x) = 0.1 \exp\left(0.5 + \sum_{i=1}^{n} \frac{x_i}{i}\right),$$

$$g_1(x) = \left| x_1 + \sum_{i=2}^{p} \frac{i-1}{i} x_i \right|^{1.5} - 0.5, \quad g_2(x) = \left(-2 + \sum_{i=1}^{n} \frac{x_i}{i}\right)^2 - 0.2,$$

$$g_3(x) = -x_1 - x_n + 1.$$

The $(m \times n)$-matrix $A$, $m$-vector $b$ (for Types 1–3) and the $n$-vectors $\alpha_1$, $\alpha_2$ and numbers $\beta_1$, $\beta_2$ (for Type 1) were generated by a random number generator similar to the one described in Horst and Thoai (1989). Computational results on some typical problems are given in Tables I–III. It is worth noting that for all test problems the number of vertices of the polytopes $S^k$ is very small and hence the vertex calculation performed by the method of Horst–Thoai–Vries (1988) only requires a negligible amount of time (approximately 1% of CPU-time).

For all problems we have taken $\gamma_1 = \gamma_2 = 10^6$ to construct a first cube $S^0 \subset \mathbb{R}^2_+$. For $k \leqslant 10$, the cut $l_k(y) \leqslant 0$ was constructed whenever a vertex $z^k$ of the feasible set of problem $(LP_k)$ was found with $(By^k - b)^T z^k \geqslant \min\{0.1, t^k\} > 0$, and thereafter whenever $(By^k - b)^T z^k \geqslant \min\{0.4, t^k\} > 0$. (Recall that $t^k$ denotes the optimal value of problem $(LP_k)$). The algorithms terminated whenever $t^k \leqslant 10^{-6}$.

Table I. Computational results on problems of Type 1

| Problem no. | $m$ | $n$ | Number of iterations | Maximal number of vertices of a polytope $S^k$ | CPU-time (sec.) |
|---|---|---|---|---|---|
| 1 | 10 | 20 | 9 | 8 | 1.85 |
| 2 | 20 | 20 | 16 | 10 | 3.65 |
| 3 | 22 | 20 | 14 | 9 | 4.05 |
| 4 | 20 | 30 | 16 | 10 | 5.04 |
| 5 | 35 | 50 | 20 | 11 | 19.88 |
| 6 | 45 | 60 | 26 | 12 | 81.22 |
| 7 | 45 | 100 | 30 | 14 | 240.50 |
| 8 | 60 | 100 | 30 | 13 | 291.12 |
| 9 | 70 | 100 | 30 | 13 | 511.38 |
| 10 | 70 | 120 | 31 | 16 | 560.75 |
| 11 | 100 | 100 | 32 | 15 | 635.06 |
| 12 | 102 | 150 | 40 | 16 | 2823.13 |
| 13 | 102 | 190 | 40 | 16 | 3187.93 |
| 14 | 72 | 199 | 45 | 20 | 2298.88 |
| 15 | 110 | 199 | 49 | 16 | 8068.84 |

Table II. Computational results on problems of Type 2

| Problem no. | $m$ | $n$ | Number of iterations | Maximal number of vertices of a polytope $S^k$ | Number of cuts $L_k(y,x) \leq 0$ | CPU-time (sec.) |
|---|---|---|---|---|---|---|
| 1 | 10 | 20 | 6 | 4 | 1 | 2.03 |
| 2 | 20 | 20 | 18 | 14 | 14 | 20.02 |
| 3 | 22 | 20 | 36 | 17 | 32 | 27.32 |
| 4 | 20 | 30 | 46 | 15 | 15 | 44.69 |
| 5 | 35 | 50 | 25 | 14 | 18 | 119.50 |
| 6 | 45 | 60 | 22 | 11 | 15 | 278.01 |
| 7 | 45 | 100 | 47 | 20 | 15 | 480.43 |
| 8 | 60 | 100 | 32 | 14 | 17 | 1221.19 |
| 9 | 70 | 100 | 43 | 47 | 19 | 1972.87 |
| 10 | 70 | 120 | 38 | 23 | 19 | 1948.81 |

Table III. Computational results on problems of Type 3

| Problem no. | $m$ | $n$ | Number of iterations | Maximal number of vertices of a polytope $S^k$ | CF | CG | CPU-time (sec.) |
|---|---|---|---|---|---|---|---|
| 1 | 5 | 5 | 29 | 7 | 10 | 6 | 4.04 |
| 2 | 5 | 10 | 33 | 7 | 13 | 9 | 8.44 |
| 3 | 10 | 20 | 42 | 13 | 15 | 5 | 13.47 |
| 4 | 20 | 30 | 78 | 16 | 15 | 6 | 79.15 |
| 5 | 30 | 50 | 75 | 16 | 18 | 8 | 270.06 |
| 6 | 40 | 80 | 38 | 14 | 8 | 5 | 637.82 |
| 7 | 50 | 80 | 48 | 16 | 11 | 6 | 1609.75 |
| 8 | 50 | 100 | 97 | 15 | 28 | 6 | 1902.59 |
| 9 | 20 | 120 | 71 | 24 | 18 | 4 | 335.85 |
| 10 | 40 | 120 | 43 | 13 | 10 | 6 | 1510.81 |

*Notation:*
CF: Number of cuts $L_k(y,x) \leq 0$ constructed according to the convex constraints $-y_i + f_i(x) \leq 0$, $i = 1, 2$.
CG: Number of cuts $L_k(y,x) \leq 0$ constructed according to $g(x) \leq 0$.

Since the feasible set of problems of Type 2 is polyhedral, the cuts $L_k(y,x) \leq 0$ were constructed according to the convex constraints $-y_i + f_i(x) \leq 0$, $i = 1, 2$.

For solving problems of Type 3, Algorithm 3 was slightly modified that at each iteration $k$ the Operation $k$.4 (construction of the cut $L_k(y,x) \leq 0$) is only performed for the case $t^k \leq 10^{-6}$ and $(y^k, x^k) \not\in \Omega$.

## 6. A Generalized Convex Multiplicative Programming Problem

The algorithms in the previous sections can immediately be applied for solving the following generalized convex multiplicative programming problem, denoted by $(GMP)$.

$$(GMP) \qquad \begin{array}{ll} \min & \prod_{i=1}^{q} f_i(x) \\ \text{s.t.} & g_i(x) < 0\,, \quad i = 1, \ldots, m \\ & x \geq 0 \end{array}$$

where $f_i$, $(i = 1, \ldots, q)$ and $g_i$, $(i = 1, \ldots, m)$ are convex functions over $\mathbb{R}^n$ and $f_i(x) \geq \varepsilon > 0$ for $i = 1, \ldots, q$.

Defining $g(x) = \max\{g_i(x) : i = 1, \ldots, m\}$ and using $q$ additional variables $y_1, \ldots, y_q$ we transform $(GMP)$ into

$$(GAP)\qquad \begin{aligned} &\min \quad \prod_{i=1}^{q} y_i && (6.1)\\ &\text{s.t.} \quad g(x) \leq 0 && (6.2)\\ &-y_i + f_i(x) \leq 0, \quad (i = 1, \ldots, q) && (6.3)\\ &x \geq 0 && (6.4)\\ &y_i \geq \varepsilon > 0, \quad (i = 1, \ldots, q). && (6.5) \end{aligned}$$

The master problem according to $(GAP)$ has then the form

$$(GCP) \qquad \min\left\{\prod_{i=1}^{q} y_i : y \in D \subset \mathbb{R}^q\right\}$$

with

$$D := \{y \geq 0 : (\exists x \geq 0) \text{ such that } (6.2)\text{–}(6.5) \text{ are fulfilled}\}. \qquad (6.6)$$

While solving Problem $(GCP)$ a method for the calculation of the sets $V(S^k)$ in $\mathbb{R}^q$ is required. As mentioned in Section 2, the methods discussed in Horst *et al.* (1988) can be efficiently applied for $q \leq 20$.

## Acknowledgement

The author would like to thank Professors Hoang Tuy and Reiner Horst for helpful discussions.

## References

Aneja, Y. P., Aggarwal, V. and Nair, K. (1984), On a Class of Quadratic Programming, *European Journal of Oper. Res.* **18**, 62–70.

Bector, C. R. and Dahl, M. (1974), Simplex Type Finite Iteration Technique and Reality for a Special Type of Pseudo-Concave Quadratic Functions, *Cahiers du Centre d'Etudes de Recherche Operationnelle* **16**, 207–222.

Gabasov, R. and Kirillova, F. M. (1980), *Linear Programming Methods, Part 3* (Special Problems), Minsk (in Russian).

Hoffman, K. L. (1981), A Method for Globally Minimizing Concave Functions over Convex Sets, *Mathematical Programming* **20**, 22–32.

Horst, R. and Thoai, N. V. (1989), Modification, Implementation and Comparison of Three Algorithms for Globally Solving Linearly Constrained Concave Minimization Problems, *Computing* **42**, 271–289.

Horst, R. and Tuy, H. (1990), *Global Optimization: Deterministic Approaches*, Springer-Verlag.

Horst, R., Thoai, N. V. and Tuy, H. (1987), Outer Approximation by Polyhedral Convex Sets, *Oper. Res. Spektrum* **9**, 153–159.

Horst, R., Thoai, N. V., and Tuy, H. (1989), On an Outer Approximation Concept in Global Optimization, *Optimization* **20**, 255–264.

Horst, R., Thoai, N. V., and de Vries, J. (1988), On Finding New Vertices and Redundant Constraints in Cutting Plane Algorithms for Global Optimization, *Oper. Res. Letters* **7**, 85–90.

Konno, H. and Kuno, T. (1989), Linear Multiplicative Programming, IHSS 89–13, Institute of Human and Social Sciences, Tokyo Institute of Technology.

Kuno, T. and Konno, H. (1990), A Parametric Successive Underestimation Method for Convex Multiplicative Programming Problems, IHSS 90–19, Institute of Human and Social Sciences, Tokyo Institute of Technology.

Pardalos, P. M. (1988), Polynomial Time Algorithms for Some Classes of Constrained Nonconvex Quadratic Problems, Preprint, Computer Science Department, The Pennsylvania State University.

Rockafellar, R. T. (1970), *Convex Analysis*, Princeton University Press, Princeton, N.J.

Schaible, S. (1976), Minimization of Ratios, *J. Optimization Theory Appl.* **19**, 347–352.

Swarup, K. (1966), Programming with Indefinite Quadratic Function with Linear Constraints, *Cahiers de Centre d'Etudes de Recherche Operationnelle* **8**, 133–136.

Tuy, H. (1985), Concave Minimization under Linear Constraints with Special Structure, *Optimization* **16**, 335–352.

Tuy, H. (1990), The Complementary Convex Structure in Global Optimization, Preprint, Institute of Mathematics, Hanoi.

Tuy, H. and Tam, B. T. (1990), An Efficient Solution Method for Rank Two Quasiconcave Minimization Problems, Preprint, Institute of Mathematics, Hanoi.