An Efficient Outer Space Algorithm for Generalized Linear Multiplicative Programming Problem

HONGWEI JIAO1,2, WENJIE WANG2, RONGJIANG CHEN2, YOULIN SHANG1, AND JINGBEN YIN2

1Postdoctoral Research Station of Control Science and Engineering, Henan University of Science and Technology, Luoyang 471023, China
2Postdoctoral Research Base, Henan Institute of Science and Technology, Xinxiang 453003, China

Corresponding authors: Hongwei Jiao (jiaohongwei@hist.edu.cn), Rongjiang Chen (chenrj0324@126.com), and Youlin Shang (matingshang@sina.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 11871196, in part by the China Postdoctoral Science Foundation under Grant 2017M622340, in part by the Key Scientific and Technological Research Projects of Henan Province under Grant 202102210147 and Grant 192102210114, and in part by the Science and Technology Climbing Program of Henan Institute of Science and Technology under Grant 2018JY01.

ABSTRACT Base on the outer space search and the branch-and-bound framework, this paper presents an efficient outer space branch-and-bound algorithm for globally solving generalized linear multiplicative programming problem. First of all, we convert the problem into an equivalent problem. Then, by utilizing a direct relaxation method, we establish the linear relaxed problem to compute the lower bound of the global optimal value of the equivalent problem. By subsequently subdividing the initial outer space rectangle and solving a series of linear relaxed problems, the proposed algorithm is convergent to the global optimal solution of the primal problem. Finally, compared with some known algorithms, numerical experiments are given to demonstrate the feasibility and effectiveness of the proposed algorithm.

INDEX TERMS Generalized linear multiplicative programming problem, global optimization, linear relaxed problem, branch-and-bound algorithm.

I. INTRODUCTION

We investigate the following generalized linear multiplicative programming problem:

\[
\begin{align*}
\min f(x) &= \sum_{i=1}^{p} \sum_{j=1}^{n} c_{ij}x_{j} + e_{i}(\sum_{j=1}^{n} d_{ij}x_{j} + g_{i}) \\
\text{s.t. } x \in X &= \{x \mid Ax \leq b, \ x \geq 0\},
\end{align*}
\]

where \( p \geq 2 \), \( \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \) and \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i}, \ i = 1, \ldots, p \), are all bounded linear (affine) functions defined on \( X, A \in R^{m \times n}, b \in R^{m}, X \) is a bounded polyhedron set. Because the problem (MP) have a wide of applications in microeconomics [1], data mining/pattern recognition [2], plant layout design [3], system reliability analysis and optimization [4]–[10], optimal resource allocation for Power-Efficient MC-NOMA [11] and other fields [12]–[14], and it has potential theoretical and computational difficulty. So that, it has attracted the attentions of many researchers and practitioners.

In the past few decades, a large number of algorithms have been proposed for the problem (MP) and its special form. Most of these algorithms are based on the branch and bound framework, for examples, by utilizing convex envelope and concave envelope of logarithmic function to construct linear relaxation, Shen and Jiao [15] proposed a branch-and-bound algorithm for linear multiplicative programming problem with exponents; by utilizing two-level linear relaxation to construct linear relaxed problem, Jiao [16] proposed a branch-and-bound algorithm for generalized linear multiplicative programming problem of generalized polynomial forms; based on partitioning and searching outcome space region, Gao et al. [17] presented an outcome space branch-and-bound algorithm for the special form of the problem (MP); based on utilizing the convex envelope of bilinear function, and by successsively subdividing and
searching the outcome space rectangle of the affine functions
\[ \sum_{j=1}^{n} d_{ij}x_{j} + g_{i}, \quad i = 1, \ldots, p \]
Yin et al. [18] proposed a new outcome space branch-and-bound algorithm for solving the general form of problem (MP); by using new linearizing technique, Wang et al. [19] presented a novel algorithm for the problem (MP); based on simplex partition and search, wang et al. [20] proposed a simplex branch-and-bound algorithm for the problem (MP). In addition to the branch and bound algorithm mentioned above, some other types of algorithms are also proposed to solve the problem (MP), such as, Phuong and Tuy [21] and Chen and Jiao [22] proposed two monotonic optimization methods for generalized linear fractional programming problem and generalized linear multiplicative programming problems; by utilizing the concept of level set, Liu and Zhao [10] gave a level set method for solving the linear multiplicative programming problem with exponents. Recently, Shen et al. [23]–[25] proposed three polynomial time approximation algorithms for generalized linear fractional multiplicative programming problem; Zhao and Zhao [12] presented a convex relaxation method for generalized linear multiplicative programming problem with generalized linear multiplicative constraints; based on geometric programming solver, Zhao and Yang [13] also proposed an inner approximation algorithm for generalized linear multiplicative programming problem with generalized linear multiplicative constraints; Jiao et al. [5] formulated an outcome space method for generalized linear multiplicative programming problem with linear constraints; these methods can be also used to solve the problem (MP) considered in this paper.

But as far as we know, apart from [18], most of the algorithms mentioned above require that \( \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \geq 0 \) and \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} \geq 0 \), it is to say, the algorithm for the problem (MP) with that \( \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \) and \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} \) can acquire arbitrary values has been still little studied in the literatures. Thus, it is still very necessary to propose a new practical algorithm for the general form of the problem (MP).

This paper will present a new outer space branch-and-bound algorithm for globally solving generalized linear multiplicative programming problem (MP). First of all, we convert the problem (MP) into an equivalent nonconvex programming problem (EQ). Next, by utilizing the direct linear relaxed technique, we establish the linear relaxed problem of the equivalent problem (EQ). By subsequently partitioning the outer space rectangle of the images of \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i}, \quad i = 1, \ldots, p \), and solving a series of linear relaxed problems, the proposed algorithm is convergent to the global minimum of the primal problem (MP). Compared with the existent algorithms, the proposed algorithm have the following some potential practical and computational advantages:

(i) The mathematical modelling of the considered problem has a more general form, which does not impose any special signal restrictions on each linear function \( \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \) and \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i}, \quad i = 1, \ldots, p \), in the objective function of the problem (MP).

(ii) The branching search take place in the outer space rectangle of the images of \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i}, \quad i = 1, \ldots, p \), rather than the rectangle of variable region, which will greatly reduce the dimension of the searched space and decrease the required computational efforts since \( n \) usually far exceeds \( p \) in many practical problems.

(iii) During the process of the branch-and-bound search, all subproblems that need to be solved in each iteration are all linear programming problems, and the scale of these linear programming problems which can be solved by the known LP solver remains unchanged.

(iv) Numerical experiments show that the proposed algorithm has the stronger robustness, and which can be used to solve all test problems to get their global optimal solutions and global optimal values with the higher computational efficiency.

The remaining section of the paper is organized as follows. Firstly, we convert the problem (MP) into an equivalent problem (EQ) in Section 2, and by utilizing the direct relaxation technique, we establish its linear relaxed problem. Secondly, we propose a new outer space branch-and-bound algorithm for the problem (EQ) in Section 3, and derive its global convergence. Thirdly, some numerical experiments are reported in Section 4, and numerical results demonstrate the stronger robustness and the higher efficiency of the algorithm. Finally, we present some concluding remarks in Section 5 and give some future works.

II. EQUIVALENT PROBLEM AND ITS LINEAR RELAXED PROGRAMMING PROBLEM

In this section, we firstly will convert the problem (MP) into an equivalent problem. Next, we will drive the linear relaxed problem of the equivalent problem.

Without losing generality, for each \( j = 1, \ldots, n \), we assume that \( l_{j}^{0} = \min_{x \in \mathbb{X}} x_{j} \) and \( u_{j}^{0} = \max_{x \in \mathbb{X}} x_{j} \). And for each \( i = 1, \ldots, p \), we let \( L_{i}^{0} = \min_{x \in \mathbb{X}} \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} \) and \( U_{i}^{0} = \max_{x \in \mathbb{X}} \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} \), which are easily computed by solving \( 2p \) linear programming problems. Let \( F^{0} = \{ \alpha \in \mathbb{R}^{0} | L_{i}^{0} \leq \alpha \leq U_{i}^{0}, \quad i = 1, \ldots, p \} \), we consider the following problem:

\[
\begin{align*}
\text{min} & \quad G(x, \alpha) = \sum_{i=1}^{p} \alpha_{i} \left( \alpha_{i} \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \right) \\
\text{s.t.} & \quad \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} - \alpha_{i} = 0, \quad i = 1, \ldots, p, \\
& \quad x \in \mathbb{X}, \quad \alpha \in F^{0}.
\end{align*}
\]

(EQ)
Obviously, the problems (MP) and (EQ) have the same global optimal solutions and global optimal values. In the following, we will present a direct linear relaxation method for constructing the linear relaxed problem of the problem (EQ), which is given by the following Theorem 1.

**Theorem 1:** Let \( F = \{ \alpha \in \mathbb{R}^p \mid L_i \leq \alpha_i \leq U_i, i = 1, \ldots, p \} \subseteq F^0 \), for any \( x \in X \) and \( \alpha \in F \), define the functions \( G(x, \alpha), \ GRC(x, \alpha) \) and \( GRC^C(x, \alpha) \) as follows:

\[
G(x, \alpha) = \sum_{i=1}^{p} \sum_{j=1}^{n} c_{ij} x_j + e_i, \\
GRC(x, \alpha) = \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} L_j x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij} U_j x_j + e_i \alpha_i, \\
GRC^C(x, \alpha) = \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} U_j x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij} L_j x_j + e_i \alpha_i.
\]

Then, we have the following conclusions:

(i) For any \( x \in X \) and \( \alpha \in F \), we have

\[
GRC(x, \alpha) \leq G(x, \alpha) \leq GRC^C(x, \alpha);
\]

(ii) since

\[
\lim_{\|U-L\| \to 0} GRC(x, \alpha) = \lim_{\|U-L\| \to 0} G(x, \alpha) = \lim_{\|U-L\| \to 0} GRC^C(x, \alpha).
\]

**Proof:** (i) Since

\[
G(x, \alpha) - GRC(x, \alpha) = \sum_{i=1}^{p} \alpha_i \left( \sum_{j=1}^{n} c_{ij} x_j + e_i \right) - \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} L_j x_j - \sum_{j=1, c_{ij} < 0}^{n} c_{ij} U_j x_j + e_i \alpha_i \]

\[
= \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} L_j x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij} U_j x_j + e_i \alpha_i \]

\[
\geq 0
\]

and

\[
GRC(x, \alpha) - G(x, \alpha) = \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} U_j x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij} L_j x_j + e_i \alpha_i \]

\[
= \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} U_j x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij} L_j x_j + e_i \alpha_i \]

\[
\geq 0
\]

from the above inequalities, we can get that, for any \( x \in X \) and \( \alpha \in F \),

\[
GRC(x, \alpha) \leq G(x, \alpha) \leq GRC^C(x, \alpha).
\]

(ii) since

\[
G(x, \alpha) - GRC(x, \alpha) = \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij}(U_j - L_j)x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij}(U_j - L_j)x_j
\]

\[
\leq \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij}(U_j - L_j)x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij}(U_j - L_j)x_j
\]

\[
= \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij}(U_j - L_j)x_j + \sum_{j=1, c_{ij} < 0}^{n} c_{ij}(U_j - L_j)x_j
\]

\[
\leq \sum_{i=1}^{p} \sum_{j=1, c_{ij} > 0}^{n} c_{ij} u_j - \sum_{j=1, c_{ij} < 0}^{n} c_{ij} l_j
\]

and the proof is completed. \( \diamond \)
Let $F = \{\alpha \in \mathbb{R}^p \mid L_i \leq \alpha_i \leq U_i, i = 1, \ldots, p\}$ be $F^0$ or its a sub-rectangle, by Theorem 1, we can establish the linear relaxed problem $(LP_F^F)$ of the problem $(EQ)$ over $F$ as follows:

$$\min G_{RC}(x, \alpha) = \sum_{i=1}^{p} \left( \sum_{j=1, c_j > 0}^{n} c_{ij} L_j x_j \right) + \sum_{j=1, c_j < 0}^{n} c_{ij} U_j x_j + e(\alpha)$$

$$s.t. \sum_{j=1}^{n} d_{ij} x_j + g_i - \alpha_i = 0, \quad i = 1, \ldots, p,$$

$$x \in X, \quad \alpha \in F.$$ 

By the constructing method for the linear relaxed problem, over the same rectangle $F$, it is obvious that the feasible region of the problem $(LP_F^F)$ contain the feasible region of the problem $(EQ)$, and the optimal value of the problem $(LP_F^F)$ is less than or equal to that of the problem $(EQ)$. That is to say, the problem $(LP_F^F)$ can provide a valid lower bound for the problem $(EQ)$ over the same rectangle $F$.

### III. OUTER SPACE ALGORITHM AND ITS CONVERGENCE

In this section, we firstly introduce a rectangular bisection method, then based on the branch-and-bound framework, combine the rectangular bisection method and the former linear relaxed problem together, a new outer space branch-and-bound algorithm is proposed for globally solving the problem $(MP)$.

#### A. RECTANGULAR BISECTION METHOD

In the algorithm, the branching process take place in the outer space rectangle of the affine functions $\sum_{j=1}^{n} d_{ij} x_j + g_i, i = 1, \ldots, p$. Without loss of generality, let $F^{k-1} = \{\alpha \in \mathbb{R}^p \mid L_i \leq \alpha_i \leq U_i, i = 1, \ldots, p\}$ be $F^0$ or a sub-rectangle of $F^0$, which is selected for the next segmentation, the proposed rectangular bisection method is described as follows. Let

$q = \arg \max \{U_i - L_i \mid i = 1, 2, \ldots, p\}$

be subscript of maximum edge of rectangle $F^{k-1}$, partition $F^{k-1}$ into two new sub-rectangles

$$F^{k,1} = \{\alpha \in \mathbb{R}^p \mid L_i \leq \alpha_i \leq \frac{L_i + U_i}{2}, i = q; L_i \leq \alpha_i \leq U_i, \quad i = 1, 2, \ldots, p, i \neq q\}$$

and

$$F^{k,2} = \{\alpha \in \mathbb{R}^p \mid \frac{L_i + U_i}{2} \leq \alpha_i \leq U_i, i = q; L_i \leq \alpha_i \leq \frac{L_i + U_i}{2}, \quad i = 1, 2, \ldots, p, i \neq q\}.$$ 

Without loss of generality, assume $\{F^k\}$ be a nested rectangular subsequence which be formed by the rectangular partitioning process. Obviously, by the exhaustiveness of the rectangular bisection method, we can get that $\lim_{k \to \infty} F^k = \alpha^*$. 

#### B. OUTER SPACE BRANCH-AND-BOUND ALGORITHM

Combine the above rectangular bisection method and the former linear relaxed problem together, a new outer space branch-and-bound algorithm is proposed for solving the problem $(MP)$ as follows.

**Algorithm Steps**

1. **Step 1.** Let the initial iteration number $k = 0$ and the termination error condition $\varepsilon \geq 0$, by solving linear programming problems, for each $i = 1, 2, \ldots, p$, compute

$$L_i^0 = \min_{x \in X} \sum_{j=1}^{n} d_{ij} x_j + g_i,$$

and

$$U_i^0 = \max_{x \in X} \sum_{j=1}^{n} d_{ij} x_j + g_i,$$

let the initial rectangle $F^0 = \{\alpha \in \mathbb{R}^p \mid L_i^0 \leq \alpha_i \leq U_i^0, \quad i = 1, \ldots, p\}$, and let the initial active node set $\mathcal{Y}_0 = \{F^0\}$.

Solve the problem $(LP_{F^0})$ to obtain its optimal solution $(x^0, \alpha^0)$ and optimal value $LB(F^0)$, respectively. Let the lower bound $LB_0 = LB(F^0)$ and the upper bound $UB_0 = G(x^0, \alpha^0)$. If $UB_0 - LB_0 \leq \varepsilon$, then the algorithm terminates, and obtain the optimal solution $(x^0, \alpha^0)$ of the $(EQ)$ and the optimal solution $x^0$ of the $(MP)$ over $F^0$, respectively. Otherwise, let the deleted rectangle set $\Omega = \emptyset, k = 1$, and go on Step 2.

2. **Step 2.** Let $UB_k = UB_{k-1}$, partition $F^{k-1}$ into two new sub-rectangles $F^{k,1}$ and $F^{k,2}$, let $\tilde{F}^k$ be the new set of the partitioned sub-rectangles, and let $\Omega = \Omega \cup \{F^{k-1}\}$ be the set of the deleted rectangle.

For each rectangle $F^{k,i} \in \tilde{F}^k$, solve the problem $(LP_{F^{k,i}})$ to obtain its optimal solution $(x^{k,i}, \alpha^{k,i})$ and optimal value $LB(F^{k,i})$. If $LB(F^{k,i}) > UB_k$, then let $\hat{F}^k := \tilde{F}^k \backslash F^{k,i}$ and $\Omega = \Omega \cup \{F^{k,i}\}$. Otherwise, let $UB_k = \min \{UB_k, G(x^{k,i}, \alpha^{k,i})\}$ be the new upper bound.

3. **Step 3.** Let $\mathcal{Y}_k = (\mathcal{Y}_{k-1} \backslash \{F^{k-1}\}) \cup \{\hat{F}^k\}$ be the remaining partitioned rectangle set, and let $LB_k = \max_{F \in \mathcal{Y}_k} LB(F)$ be the new lower bound.

4. **Step 4.** If $UB_k - LB_k \leq \varepsilon$, then the algorithm stops, and we get the optimal solution $(x^k, \alpha^k)$ of the $(EQ)$ and the optimal solution $x^k$ of the $(MP)$, respectively. Otherwise, let $k = k + 1$ and return to Step 2.

#### C. CONVERGENCE OF OUTER SPACE ALGORITHM

In the subsection, the global convergence of the outer space branch-and-bound algorithm is given as follows.

If the algorithm terminates going through $k$ iterations, then, at the $k_{th}$ iteration, we can get the feasible solution $x^k$ of the problem $(MP)$ and the feasible solution $(x^k, \alpha^k)$ of the
problem (EQ) by solving the \((LP_{F^k})\), where
\[
\alpha_i^k = \sum_{j=1}^{n} d_{ij}x_j^{k} + g_i, \quad i = 1, 2, \ldots, p.
\]
Let \(v\) be the global optimal value of the primal problem (MP).
From the convergent terminating condition, the updating upper bound methods, the updating lower bound methods, the equivalence of the problem (MP) and (EQ) and Theorem 1, we can obtain that
\[
G(x^k, \alpha^k) \leq \text{LB}_k + \epsilon, \quad \text{LB}_k \leq v, \quad v \leq G(x^k, \alpha^k)
\]
and
\[
f(x^k) = G(x^k, \alpha^k).
\]
Combining the several inequalities together, we can follow that
\[
v \leq f(x^k) \leq v + \epsilon.
\]
Therefore, if the proposed algorithm terminates after going through \(k\) iterations, then \(x^k\) is an \(\epsilon\)-global optimal solution of the problem (MP).

**Theorem 2:** If the proposed algorithm generates an infinite sequence \(\{x^k\}\) of solution, then the limitation \(x^*\) of \(\{x^k\}\) will be a global optimal solution of the problem (MP).

**Proof:** If the proposed algorithm generates an infinite sequence of solution, then, by solving the linear relaxed programming problem \((LP_{F^k})\), we can get a feasible solution sequence \(\{x^k\}\) of the problem (MP) and a feasible solution sequence \(\{(x^k, \alpha^k)\}\) of the problem (EQ), where
\[
\alpha_i^k = \sum_{j=1}^{n} d_{ij}x_j^{k} + g_i, \quad i = 1, 2, \ldots, p.
\]
From the continuity of the linear function \(\alpha_i = \sum_{j=1}^{n} d_{ij}x_j + g_i\),
\[
\alpha_i^k = \left(\sum_{j=1}^{n} d_{ij}x_j^{k} + g_i\right) \in [L_i^k, U_i^k]
\]
and the exhaustiveness of the rectangular bisection method, for any \(i \in \{1, 2, \ldots, p\}\), we have that
\[
\sum_{j=1}^{n} d_{ij}x_j^{k} + g_i = \lim_{k \to \infty} \sum_{j=1}^{n} d_{ij}x_j^{k} + g_i = \lim_{k \to \infty} \left[L_i^k, U_i^k\right] = \alpha_i^*.
\]
So that, \((x^*, \alpha^*)\) is a feasible solution of the problem (EQ), by the Theorem 1, and also since \(\{\text{LB}_k\}\) is an increasing lower bound sequence with that \(\text{LB}_k \leq v\), we can get that
\[
G(x^*, \alpha^*) \geq v \geq \lim_{k \to \infty} \text{LB}_k = \lim_{k \to \infty} G_{RC}(x^k, \alpha^k) = G(x^*, \alpha^*).
\]
So, by the continuity of the function \(f(x)\) and the above inequalities, we can get that
\[
\lim_{k \to \infty} \text{LB}_k = v = G(x^*, \alpha^*) = f(x^*) = \lim_{k \to \infty} f(x^k).
\]
Therefore, the limitation \(x^*\) of \(\{x^k\}\) is a global optimal solution for the problem (MP), the proof is completed.

**D. THE COMPLEXITY OF THE PROPOSED ALGORITHM**

**Definition 1:** A nonempty compact hyper-rectangle with sides parallel to the axes is denoted by
\[
F = [L_1, U_1] \times \ldots \times [L_p, U_p] \subset R^p.
\]
The diameter of a hyper-rectangle \(F \subset R^p\) is
\[
\delta(F) = \max[\|\alpha - \alpha'\|_2 : \alpha, \alpha' \in F] = \sqrt{(U_1 - L_1)^2 + \ldots + (U_p - L_p)^2}.
\]

**Theorem 3:** For the proposed branch-and-bound algorithm, assumed that, for a given feasible hyper-rectangle \(F^0\), there exists a fixed positive number \(C\) and an accuracy \(\epsilon\). In addition, we also assume that the branching process will eventually subdivide the hyper-rectangle into \(s = 2^p\) smaller sub-hyper-rectangles. Then, in the worst case, the number of iterations of the proposed algorithm by dividing the hyper-rectangle \(F^0\) can be given by the following expression:
\[
\sum_{v=0}^{\log_2 \frac{C \delta(F)}{\epsilon}} 2^{p\cdot v},
\]
where \(z = \left[\log_2 \frac{C \delta(F)}{\epsilon}\right] - \frac{\delta(F)}{\epsilon}\).

**Proof:** The proof method is similar to the proof Theorem 5 in [14].

It can be seen that the number of iterations of the proposed algorithm is exponential growth, but when the size of \(p\) satisfies \(p \ll n\), our algorithm has an advantage in solving large-scale problem (MP), and the reader can refer to [14].

**IV. NUMERICAL EXPERIMENT**

To test and verify the computational performance and robustness of the algorithm, with the given convergent error \(\epsilon = 1.0e - 6\), some numerical problems are coded by Matlab 2014a in a microcomputer with Win10 system, Intel(R) Xeon(R) CPU E5-2620 v4 @2.10GHz processor and 16.0GB RAM memory, these test problems and their numerical results are listed as follows.

**Problem 1** (Yin et al. [18]; Wang et al. [20]; Shen and Huang [25]):

\[
\begin{align*}
\min \quad 3x_1 - 4x_2 + (x_1 + 2x_2 - 1.5)(2x_1 - x_2 + 4) \\
+ (x_1 - 2x_2 + 8.5)(2x_1 + x_2 - 1) \\
\text{s.t.} \quad 5x_1 - 8x_2 & \geq -24, \\
5x_1 + 8x_2 & \leq 44, \\
6x_1 - 3x_2 & \leq 15, \\
4x_1 + 5x_2 & \geq 10, \\
x_1 & \geq 0.
\end{align*}
\]
Problem 2 (Yin et al. [18]; Wang et al. [19]; Wang et al. [20]; Shen and Huang [25]):
\[
\begin{align*}
\min (0.813396x_1 + 0.67442x_2 + 0.305038x_3 + 0.129742x_4 + 0.217796) \\
\times (0.224508x_1 + 0.063458x_2 + 0.93223x_3 + 0.528763x_4 + 0.091947) \\
s.t. \quad 0.488509x_1 + 0.063458x_2 + 0.945686x_3 + 0.210704x_4 \leq 3.562809, \\
-0.324014x_1 - 0.501754x_2 - 0.719204x_3 + 0.099562x_4 \leq -0.052215, \\
0.445225x_1 - 0.346896x_2 + 0.637939x_3 - 0.257623x_4 \leq 0.42792, \\
-0.202821x_1 + 0.647361x_2 + 0.920135x_3 - 0.983091x_4 \leq 0.84095, \\
-0.886420x_1 - 0.802444x_2 - 0.305441x_3 - 0.180123x_4 \leq -1.353686, \\
-0.515999x_1 - 0.424820x_2 + 0.897498x_3 + 0.187268x_4 \leq 2.137251, \\
-0.591515x_1 + 0.060581x_2 - 0.427365x_3 + 0.579388x_4 \leq -0.290987, \\
0.423524x_1 + 0.940496x_2 - 0.437944x_3 - 0.742941x_4 \leq 0.37362, \\
x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]

Problem 3 (Gao et al. [17]; Yin et al. [18]; Wang et al. [19]; Wang et al. [20]; Shen and Huang [25]):
\[
\begin{align*}
\min (x_1 + x_2)(x_1 - x_2 + 7) \\
s.t. \quad 2x_1 + x_2 \leq 14, \\
x_1 + x_2 \leq 10, \\
-4x_1 + x_2 \leq 0, \\
2x_1 + x_2 \geq 0, \\
x_1 + 2x_2 \geq 6, \\
x_1 + 2x_2 \leq 0, \\
x_1 - x_2 \leq 3, \\
x_1 + x_2 \geq 0, \\
x_1 - x_2 + 7 \geq 0, \\
x_1, x_2 \geq 0.
\end{align*}
\]

Problem 4 (Yin et al. [18]; Wang et al. [19]; Wang et al. [20]; Shen and Huang [25]):
\[
\begin{align*}
\min x_1 + (2x_1 - 3x_2 + 13)(x_1 + x_2 - 1) \\
s.t. \quad -x_1 + 2x_2 \leq 8, \\
-x_2 \leq -3, \\
x_1 + 2x_2 \leq 12, \\
x_1 - 2x_2 \leq -5, \\
x_1, x_2 \geq 0.
\end{align*}
\]

Problem 5 (Yin et al. [18]; Wang et al. [20]; Shen and Huang [25]):
\[
\begin{align*}
\min -x_1^2 - x_2^2 + (-x_1 - 3x_2 + 2)(4x_1 + 3x_2 + 1) \\
s.t. \quad x_1 + x_2 \leq 5, \\
-x_1 + x_2 \leq 6, \\
x_1, x_2 \geq 0.
\end{align*}
\]

Problem 6 (Yin et al. [18]; Wang et al. [20]; Shen and Huang [25]):
\[
\begin{align*}
\min -2x_1^2 - x_2^2 - 2 \\
+(-2x_1 - 3x_2 + 2)(4x_1 + 6x_2 + 2) \\
+(-3x_1 + 5x_2 + 2)(6x_1 + 8x_2 + 1) \\
s.t. \quad 2x_1 + x_2 \leq 10, \\
-x_1 + 2x_2 \leq 10, \\
x_1, x_2 \geq 0.
\end{align*}
\]

Problem 7 (Yin et al. [18]; Wang et al. [19]; Shen and Huang [25]):
\[
\begin{align*}
\min x_1 + (x_1 - x_2 + 5)(x_1 + x_2 - 1) \\
s.t. \quad -2x_1 - 3x_2 \leq -9, \\
3x_1 - x_2 \leq 8, \\
x_1 + 2x_2 \leq 8, \\
x_1 + 2x_2 \leq 12, \\
x_1 \geq 0.
\end{align*}
\]

Problem 8 (Yin et al. [18]; Wang et al. [19]; Shen and Huang [25]):
\[
\begin{align*}
\min (x_1 + x_2)(x_1 - x_2) \\
+(x_1 + x_2 + 1)(x_1 - x_2 + 1) \\
s.t. \quad x_1 + 2x_2 \leq 20, \\
x_1 - 3x_2 \leq 20, \\
1 \leq x_1, x_2 \leq 3.
\end{align*}
\]

Problem 9 (Yin et al. [18]; Wang et al. [19]; Shen and Huang [25]):
\[
\begin{align*}
\min (x_1 + x_2)(x_1 - x_2) \\
+(x_1 + x_2 + 2)(x_1 - x_2 + 2) \\
s.t. \quad x_1 + 2x_2 \leq 20, \\
x_1 - 3x_2 \leq 20, \\
1 \leq x_1, x_2 \leq 4.
\end{align*}
\]

Problem 10 (Yin et al. [18]; Wang et al. [19]; Shen and Huang [25]):
\[
\begin{align*}
\min 2x_1 - 2x_2 + 3(-2x_1 + 3x_2 + x_3 - 4) \\
+(-2x_1 + x_2 + 3 + 2)(x_1 + x_2 - 3x_3 + 5) \\
+(-2x_1 - x_2 + 2x_3 + 7)(4x_1 - x_2 - 2x_3 - 5) \\
s.t. \quad x_1 + x_2 + x_3 \leq 10, \\
x_1 - 2x_2 + x_3 \leq 10, \\
-2x_1 + 2x_2 + 3x_3 \leq 10, \\
x_1 + x_2 + 3x_3 \geq 6, \\
x_1 \geq 1, x_2 \geq 1, x_3 \geq 1.
\end{align*}
\]

Problem 11 (Yin et al. [18]):
\[
\begin{align*}
\min \sum_{i=1}^{p} \left( \sum_{j=1}^{n} c_{ij}x_j + e_i \right) \left( \sum_{j=1}^{n} \delta_{ij}x_j + g_i \right) \\
s.t. \quad Ax \leq b, \\
x \geq 0.
\end{align*}
\]

where, for each \( i = 1, \ldots, p \), \( j = 1, \ldots, n \), \( c_{ij} \) and \( \delta_{ij} \) are randomly generated in the interval \([0, 1]\); for each \( i = 1, \ldots, p \), \( e_i \) and \( g_i \) are randomly generated in the interval \([0, 100]\); each element of \( A \) is randomly generated in the interval \([0, 1]\); each
TABLE 1. Comparison of numerical results for Problems 1-10.

| Problems | Refs.          | Optimal solution     | Optimal value | Iterations | Time(s) |
|----------|----------------|----------------------|---------------|------------|---------|
| 1        | This paper     | (0.0000, 3.0000)     | -2.50000      | 182        | 7.07510 |
|          | Yin et al. [18]| (0.0000, 3.0000)     | -2.50000      | 30         | 1.54421 |
|          | Wang et al. [20]| (0, 3)              | -2.5          | 1          | 0.063   |
|          | Shen and Huang [26]| (0, 3)       | -2.5          | 1          | 0.047   |
| 2        | This paper     | (1.3148, 0.1396, 0.0000, 0.4233) | 0.89019       | 22         | 0.51458 |
|          | Yin et al. [18]| (1.3148, 0.1396, 0.0000, 0.4233) | 0.89019       | 0          | 0.05622 |
|          | Wang et al. [19]| (1.3148, 0.1396, 0, 0.4233) | 0.8902        | 1          | 0.061   |
|          | Wang et al. [20]| (1.3148, 0.1396, 0, 0.4233) | 0.8902        | 1          | 0.188   |
|          | Shen and Huang [26]| (1.3148, 0.1396, 0, 0.4233) | 0.8902        | 1          | 0.093   |
| 3        | This paper     | (2.0000, 8.0000)     | 10.00000      | 27         | 0.70909 |
|          | Gao et al. [17]| (2, 8)               | 10            | 53         | 0.3     |
|          | Yin et al. [18]| (2.0000, 8.0000)     | 10.00000      | 0          | 0.04965 |
|          | Wang et al. [19]| (2, 8)              | 10            | 1          | 0.066   |
|          | Wang et al. [20]| (2, 8)              | 10            | 48         | 5.078   |
|          | Shen and Huang [26]| (2, 8)       | 10            | 1          | 0.046   |
| 4        | This paper     | (0.0000, 4.0000)     | 3.00000       | 172        | 3.53071 |
|          | Yin et al. [18]| (0.0000, 4.0000)     | 3.00000       | 0          | 0.05613 |
|          | Wang et al. [19]| (0, 4)              | 3             | 1          | 0.0727  |
|          | Wang et al. [20]| (0, 4)              | 3             | 2          | 0.203   |
|          | Shen and Huang [26]| (0, 4)       | 3             | 1          | 0.046   |
| 5        | This paper     | (0.0000, 5.0000)     | -233.00000    | 83         | 2.14846 |
|          | Yin et al. [18]| (0.0000, 5.0000)     | -233.00000    | 1          | 0.14337 |
|          | Wang et al. [20]| (0, 5)              | -233          | 4          | 0.3232  |
|          | Shen and Huang [26]| (0, 5) | -233          | 1          | 0.047   |
| 6        | This paper     | (0.0000, 0.0000)     | 4.00000       | 15         | 0.35513 |
|          | Yin et al. [18]| (0.0000, 0.0000)     | 4.00000       | 9          | 0.36751 |
|          | Wang et al. [20]| (0, 0)              | 4             | 7          | 0.6336  |
|          | Shen and Huang [26]| (0, 0)       | 4             | 1          | 0.062   |
| 7        | This paper     | (0.0000, 4.0000)     | 3.00000       | 89         | 2.49274 |
|          | Yin et al. [18]| (0.0000, 4.0000)     | 3.00000       | 0          | 0.05496 |
|          | Wang et al. [19]| (0, 4)              | 3             | 1          | 0.0693  |
|          | Shen and Huang [26]| (0, 4)       | 3             | 1          | 0.062   |
| 8        | This paper     | (1.0000, 3.0000)     | -13.00000     | 50         | 0.95348 |
|          | Yin et al. [18]| (1.0000, 3.0000)     | -13.00000     | 0          | 0.05962 |
|          | Wang et al. [19]| (1, 3)             | -13           | 1          | 0.0868  |
|          | Shen and Huang [20]| (1, 3)       | -13           | 1          | 0.047   |
| 9        | This paper     | (1.0000, 4.0000)     | -22.00000     | 52         | 1.02796 |
|          | Yin et al. [18]| (1.0000, 4.0000)     | -22.00000     | 0          | 0.05179 |
|          | Wang et al. [19]| (1, 4)            | -22           | 1          | 0.0849  |
|          | Shen and Huang [26]| (1, 4)       | -22           | 1          | 0.046   |
| 10       | This paper     | (5.5556, 1.7778, 2.6667) | -112.754      | 602        | 19.0745 |
|          | Yin et al. [18]| (5.5556, 1.7778, 2.6667) | -112.754      | 16         | 0.50701 |
|          | Wang et al. [19]| (5.5556, 1.7778, 2.6667) | -112.754      | 5          | 1.1491  |
|          | Shen and Huang [26]| (5.5556, 1.7778, 2.6667) | -112.754      | 1          | 0.062   |

Element of b is randomly generated in the interval [0, n]. Use the proposed algorithm in this paper to solve this problem, numerical results are given in Table 2.

In Table 2, some notations have been used in column headers as follows:

- **n**: Number of variables;
- **p**: Number of sums;
- **m**: Rows of matrix A;
- **Avg.NT**: Average iteration numbers;
- **Std.NT**: Standard deviation of iteration numbers;
- **Avg.Time**: Average CPU time in seconds;
- **Std.Time**: Standard deviation of CPU time.

From numerical results of Table 1, for Problems 1-10, compared with the known algorithms in the literature, the algorithm proposed in this paper can not only obtain the same optimal solution and optimal value, but also they have almost the same higher computational efficiency.

From numerical results of Problem 11 in Table 2, by solving randomly generated large-scale generalized linear multiplicative programming problem, and with the increase of the scale of the Problem 11, our algorithm has the higher computational efficiency than that of Ref. [18].
On the whole, numerical results of Tables 1 and 2 show that the algorithm can globally solve all test problems 1-11 with the robustness and effectiveness.

**Explanatory remarks:** In Table 2, Avg (Std).NT in the proposed algorithm is 0, this is because, under the same random parameter condition, when using the method proposed in this paper to solve Problem 11, the algorithm meets the termination condition in Step 1, that is to say, when the algorithm solves the relaxation problem in Step 1, we get an approximate global optimal solution of the original problem (MP), and the algorithm is terminated. Since we set the initial iteration number \( k_{0} = 0 \) in Step 1, so that the output iteration number is 0. Obviously, compared with the numerical results of Problem 11 in [18], this shows that the proposed algorithm in this paper has higher computational efficiency than that of [18].

## V. CONCLUSION

In this article, based on the branch-and-bound framework, we propose a novel outer space algorithm for globally solving generalized linear multiplicative programming problem (MP) with only assumption that each affine function \( \sum_{j=1}^{n} c_{ij}x_{j} + e_{i} \) and \( \sum_{j=1}^{n} d_{ij}x_{j} + g_{i} \) can acquire arbitrary value. By utilizing the equivalent transformation and the direct linear relaxation method, the initial generalized linear multiplicative programming problem (MP) can be converted into a sequence of linear relaxed problems. By subsequently subdividing the initial outer space rectangle of \( R^{p} \), and by subsequently solving a sequence of linear relaxed problems, the proposed algorithm is convergent to the global optimal solution of the initial problem (MP). During the branch-and-bound searching process, all subproblems that require to be solved in each iteration of the algorithm are all linear programming problems, and the scale of these linear programming problems still remains unchanged. Finally, comparing with some existent algorithms, numerical results demonstrate that the proposed algorithm has the higher computational efficiency.

In the future, based on the convex relaxation bounding technique and the branch-and-bound framework, the proposed outer space branch-and-bound method can be extended to solve globally minimizing generalized concave multiplicative programming problem.

## ACKNOWLEDGMENT

The authors would like thank to editor and the anonymous referees for their valuable comments and suggestions, which have greatly improved the earlier version of this articles.

## REFERENCES

[1] H. Konno and T. Kuno, "Generalized linear multiplicative and fractional programming," *Ann. Oper. Res.*, vol. 25, no. 1, pp. 147–161, Dec. 1990.

[2] H.-S. Ryoo and N. V. Sahinidis, "Global optimization of multiplicative programs," *J. Global Optim.*, vol. 26, pp. 387–418, Aug. 2003.

[3] C. Xue and H. Jiao, "An approximate algorithm for solving generalized linear multiplicative programming," *J. Henan Normal Univ., Natural Sci.*, vol. 36, no. 5, pp. 13–15, 2008.

[4] P. Shen, B. Huang, and L. Wang, "Range division and linearization algorithm for a class of linear ratios optimization problems," *J. Comput. Appl. Math.*, vol. 350, pp. 324–342, Apr. 2019.

[5] H. W. Jiao, S. Y. Liu, and Y. Q. Chen, "Global optimization algorithm for a generalized linear multiplicative programming," *J. Appl. Math. Comput.*, vol. 40, nos. 1–2, pp. 551–568, Oct. 2012.

[6] T. Matsui, "NP-hardness of linear multiplicative programming and related problems," *J. Global Optim.*, vol. 9, no. 2, pp. 113–119, Sep. 1996.
WENJIE WANG received the B.S. degree in mathematics and applied mathematics from Henan Normal University, China, in 1984, the M.S. degree in control science and engineering from the Henan University of Science and Technology, China, in 2002, and the Ph.D. degree in operational research and control from Shanghai University. He is currently a Full Professor with the School of Mathematics and Statistics, Henan University of Science and Technology. He has published more than 100 research monographs. He has published extensively and holds numerous patents in his research areas. His current research interests include optimization algorithms, system optimization, system engineering, and optimal control.

RONGJIANG CHEN received the B.S. degree in mathematics and applied mathematics from Henan Normal University, China, in 1984, the M.S. degree in control science and engineering from the Henan University of Science and Technology, China, in 2002, and the Ph.D. degree in operational research and control from Shanghai University. He is currently a Full Professor with the School of Mathematics and Statistics, Henan University of Science and Technology. He has published more than 100 research monographs. He has published extensively and holds numerous patents in his research areas. His current research interests include optimization algorithms, system optimization, system engineering, and optimal control.

HONGWEI JIAO received the B.S. degree in mathematics and applied mathematics and the M.S. degree in applied mathematics from Henan Normal University, China, in 2003 and 2006, respectively, and the Ph.D. degree in applied mathematics from Xidian University. He is currently an Associate Professor with the School of Mathematical Sciences, Henan Institute of Science and Technology, China. He has published extensively and holds numerous patents in his research areas. His current research interests include optimization algorithms, system engineering, and optimal control.

YOU LIN SHANG received the B.S. degree in mathematics and applied mathematics from Henan Normal University, China, in 1994, and the master’s degree from Zhengzhou University, in 2009. He is currently an Associate Professor with the Department of Mathematics, Henan Institute of Science and Technology, China. He has published over 30 research monographs. His research interests include software engineering, computer application, optimization design, product design optimization, manufacturing information systems, global optimization algorithms, nonlinear systems, and optimal control theory.

JINGBEN YIN received the B.S. degree in mathematics and applied mathematics from Henan University, China, in 1994, and the master’s degree from Zhengzhou University, in 2009. He is currently a Professor with the Department of Mathematics, Henan Institute of Science and Technology, China. He has published extensively and holds numerous patents in his research areas. His current research interests include optimization algorithms, system optimization, system engineering, and optimal control theory.