

RESEARCH

Open Access



An accelerating algorithm for globally solving nonconvex quadratic programming

Li Ge^{1,2*}  and Sanyang Liu¹

*Correspondence:
geli163@163.com

¹School of Mathematics and Statistics, Xidian University, Xi'an, China

²School of Mathematical Sciences, Henan Institute of Science and Technology, Xinxiang, China

Abstract

To globally solve a nonconvex quadratic programming problem, this paper presents an accelerating linearizing algorithm based on the framework of the branch-and-bound method. By utilizing a new linear relaxation approach, the initial quadratic programming problem is reduced to a sequence of linear relaxation programming problems, which is used to obtain a lower bound of the optimal value of this problem. Then, by using the deleting operation of the investigated regions, we can improve the convergent speed of the proposed algorithm. The proposed algorithm is proved to be convergent, and some experiments are reported to show higher feasibility and efficiency of the proposed algorithm.

Keywords: Nonconvex quadratic programming; Global optimization; Linear relaxation approach; Deleting technique; Branch and bound

1 Introduction

In this paper, we consider the following nonconvex quadratic programming problem (NQP):

$$(NQP): \begin{cases} \min f(y) = c^T y + y^T Q y \\ \text{s.t. } y \in Y^0 = \{y \in R^n \mid Ay \leq b\}, \end{cases}$$

where $Q = (q_{jk})_{n \times n}$ is a real $n \times n$ symmetric matrix; $A = (a_{jk})_{m \times n}$ is a real $m \times n$ matrix, and $c, y \in R^n$; $b \in R^m$.

The (NQP) is an important class of global optimization problems, and it is not only because they have broad applications in the fields of statistics and design engineering, optimal control, economic equilibria, financial management, production planning, combinatorial optimization, and so on [1, 2], but also because many other nonlinear problems can be transformed into this form [3–7], such as the following linear multiplicative programming (LMP) problem:

$$(LMP): \begin{cases} \min f(y) = \sum_{i=1}^p (c_i^T y + d_i)(e_i^T y + f_i) \\ \text{s.t. } y \in Y^0 = \{y \in R^n \mid Ay \leq b\}, \end{cases}$$

where $p \geq 0, c_i^T = (c_1, c_2, \dots, c_n), e_i^T = (e_1, e_2, \dots, e_n) \in R^n; A = (a_{jk})_{m \times n}$ is a real $m \times n$ matrix, and $d_i, f_i \in R, i = 1, 2, \dots, n; b \in R^m$. Obviously, the (LMP) can be easily transformed into the (NQP). So, the applications of the (NQP) also include all the applications of the (LMP).

In the last decades, many solution methods have been developed for globally solving the (NQP) and its special forms. For example, branch-and-cut algorithm [8] and branch-and-bound algorithm [9] for box-constrained nonconvex quadratic programs; two efficient algorithms for linear constrained quadratic programming (LCQP) by Cambini et al. [10] and Li et al. [11]; branch-and-reduce approach by Gao et al. [12]; finite branch-and-bound approach by Burer and Vandenbussche [13]; branch-and-bound method based on outer approximation and linear relaxation by Al-Khayyal et al. [14]; two simplicial branch-and-bound algorithms by Linderoth [15] and Raber [16]; branch-and-cut algorithm by Sherali et al. [17] and Audet et al. [18], and so on. In addition, a branch-and-bound algorithm for finding a global optimal solution for a nonconvex quadratic program with convex quadratic constraints has been proposed by Cheng Lu et al. [19]; based on the different properties of a quadratic function, five different branch-and-bound approaches for solving the (NQP) have been proposed in Refs. [20–24].

Based on the above-mentioned methods, we will present an accelerating algorithm for effectively solving the (NQP) in this paper. Firstly, we present a new linear relaxation technique, which can be used to construct the linear relaxation programming (LRP) of the (NQP). In addition, we divide the initial feasible region into a number of sub-regions and use deleting techniques to reduce the scope of the investigation. Then, the initial problem (NQP) is converted into a series of linear relaxation programming subproblems. We can prove that the solutions of these subproblems can converge to the global optimal value of the original problem (NQP). Finally, numerical results and comparison with the methods mentioned in recent literature show that our algorithm works as well as or better than those methods.

This paper is organized as follows. In Sect. 2, we describe a new linear relaxation approach for establishing the linear relaxation programming of the (NQP). In Sect. 3, a range deleting technique is given for improving the convergent speed of the proposed algorithm, then a range division and deleting algorithm is proposed and its convergence is proved. Some numerical experiments are reported to show the feasibility and effectiveness of our algorithm in Sect. 4.

2 New linear relaxation approach

In this section, we will show how to construct the (LRP) for the (NQP). Let the i th row of the matrix Q be Q_i , and let $z_i = Q_i y = \sum_{k=1}^n q_{ik} y_k, i = 1, 2, \dots, n$, we obtain

$$z = (z_1, z_2, \dots, z_n)^T = (Q_1 y, Q_2 y, \dots, Q_n y)^T = Qy.$$

By introducing new variables $z_i, i = 1, 2, \dots, n$, the function $f(y)$ can be expressed as follows:

$$f(y, z) = c^T y + y^T z = \sum_{i=1}^n c_i y_i + \sum_{i=1}^n y_i z_i,$$

and the set D is defined as

$$D = \{y \mid Ay \leq b, y \in R^n\}.$$

We compute the initial variable bounds by solving the following linear programming problems:

$$\begin{aligned} \underline{y}_i^0 &= \min_{y \in D} y_i, & \bar{y}_i^0 &= \max_{y \in D} y_i, & i &= 1, 2, \dots, n, \\ \underline{z}_i^0 &= \min_{y \in D} Q_i y, & \bar{z}_i^0 &= \max_{y \in D} Q_i y, & i &= 1, 2, \dots, n, \end{aligned}$$

and let

$$\begin{aligned} Y^0 &= \{y \in R^n \mid \underline{y}_i^0 \leq y_i \leq \bar{y}_i^0, i = 1, 2, \dots, n\}, \\ Z^0 &= \{z \in R^n \mid \underline{z}_i^0 \leq z_i \leq \bar{z}_i^0, i = 1, 2, \dots, n\}, \\ \Omega^0 &= \{(y, z) \in R^{2n} \mid \underline{y}_i^0 \leq y_i \leq \bar{y}_i^0, \underline{z}_i^0 \leq z_i \leq \bar{z}_i^0, i = 1, 2, \dots, n\}. \end{aligned}$$

Based on the above discussion, we can transform the initial problem into the following equivalent problem (EP):

$$(EP): \begin{cases} \min f(y, z) = \sum_{i=1}^n c_i y_i + \sum_{i=1}^n y_i z_i \\ \text{s.t. } Ay \leq b, \\ z_i = \sum_{k=1}^n q_{ik} y_k, i = 1, 2, \dots, n, \\ (y, z) \in \Omega^0. \end{cases}$$

For globally solving the (NQP), the computation of lower bounds for this problem and its subdivided subproblems is the principal operation in the establishment of a branch-and-bound procedure. A lower bound on the optimal value of the (NQP) and its subdivided subproblems can be gotten by solving a linear relaxation programming of the (EP), which can be derived by a new linear relaxation approach.

Let

$$\begin{aligned} Y &= \{y \in R^n \mid -\infty \leq \underline{y}_i \leq y_i \leq \bar{y}_i \leq +\infty, i = 1, 2, \dots, n\} \subseteq Y^0, \\ Z &= \{z \in R^n \mid -\infty \leq \underline{z}_i \leq z_i \leq \bar{z}_i \leq +\infty, i = 1, 2, \dots, n\} \subseteq Z^0 \end{aligned}$$

for any $y \in Y, z \in Z, i \in \{1, 2, \dots, n\}$, define the functions

$$\begin{aligned} \varphi(y_i) &= y_i^2, & \varphi^l(y_i) &= (y_i + \bar{y}_i)y_i - \frac{(y_i + \bar{y}_i)^2}{4}, \\ \varphi^u(y_i) &= (y_i + \bar{y}_i)y_i - y_i \bar{y}_i, \\ \varphi(z_i) &= z_i^2, & \varphi^l(z_i) &= (z_i + \bar{z}_i)z_i - \frac{(z_i + \bar{z}_i)^2}{4}, \\ \varphi^u(z_i) &= (z_i + \bar{z}_i)z_i - \underline{z}_i \bar{z}_i, & \varphi(y_i, z_i) &= y_i z_i, \end{aligned}$$

$$\begin{aligned} \varphi^l(y_i, z_i) &= \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i)z_i + (\underline{z}_i + \bar{z}_i)y_i - \frac{(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)^2}{4} + \underline{y}_i \bar{y}_i + \underline{z}_i \bar{z}_i \right], \\ \varphi^u(y_i, z_i) &= \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i)z_i + (\underline{z}_i + \bar{z}_i)y_i - (\underline{y}_i + \underline{z}_i)(\bar{y}_i + \bar{z}_i) + \frac{(\underline{y}_i + \bar{y}_i)^2}{4} + \frac{(\underline{z}_i + \bar{z}_i)^2}{4} \right]. \end{aligned}$$

Since $\varphi(y_i) = y_i^2$ is a convex function of y_i over $[\underline{y}_i, \bar{y}_i]$, its affine concave envelope is $\varphi^u(y_i) = (\underline{y}_i + \bar{y}_i)y_i - \underline{y}_i \bar{y}_i$. Moreover, the tangential supporting function for $\varphi(y_i)$ is parallel with the $\varphi^u(y_i)$, thus the point of tangential support will occur at $\frac{\underline{y}_i + \bar{y}_i}{2}$, and the corresponding tangential supporting function is $\varphi^l(y_i) = (\underline{y}_i + \bar{y}_i)y_i - \frac{(\underline{y}_i + \bar{y}_i)^2}{4}$. Hence, by the geometric property of the function $\varphi(y_i)$, we have

$$\varphi^l(y_i) = (\underline{y}_i + \bar{y}_i)y_i - \frac{(\underline{y}_i + \bar{y}_i)^2}{4} \leq y_i^2 \leq (\underline{y}_i + \bar{y}_i)y_i - \underline{y}_i \bar{y}_i = \varphi^u(y_i). \tag{1}$$

Similarly, it follows that

$$\varphi^l(z_i) \leq \varphi(z_i) \leq \varphi^u(z_i).$$

Furthermore, assume that $(y_i + z_i)$ is a single variable, then $(y_i + z_i)^2$ is a convex function about the univariate variable $(y_i + z_i)$ over the interval $[\underline{y}_i + \underline{z}_i, \bar{y}_i + \bar{z}_i]$. By the conclusion above, we have

$$(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)(y_i + z_i) - \frac{(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)^2}{4} \leq (y_i + z_i)^2 \tag{2}$$

and

$$(y_i + z_i)^2 \leq (\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)(y_i + z_i) - (\underline{y}_i + \underline{z}_i)(\bar{y}_i + \bar{z}_i). \tag{3}$$

By (1)–(3), we can obtain

$$\begin{aligned} \varphi(y_i, z_i) &= \frac{1}{2} [(y_i + z_i)^2 - y_i^2 - z_i^2] \\ &\geq \frac{1}{2} \left[(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)(y_i + z_i) - \frac{(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)^2}{4} \right] \\ &\quad - \frac{1}{2} [(\underline{y}_i + \bar{y}_i)y_i - \underline{y}_i \bar{y}_i + (\underline{z}_i + \bar{z}_i)z_i - \underline{z}_i \bar{z}_i] \\ &= \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i)z_i + (\underline{z}_i + \bar{z}_i)y_i - \frac{(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)^2}{4} + \underline{y}_i \bar{y}_i + \underline{z}_i \bar{z}_i \right] \\ &= \varphi^l(y_i, z_i), \end{aligned}$$

and

$$\begin{aligned} \varphi(y_i, z_i) &= \frac{1}{2} [(y_i + z_i)^2 - y_i^2 - y_k^2] \\ &\leq \frac{1}{2} [(\underline{y}_i + \underline{z}_i + \bar{y}_i + \bar{z}_i)(y_i + z_i) - (\underline{y}_i + \underline{z}_i)(\bar{y}_i + \bar{z}_i)] \end{aligned}$$

$$\begin{aligned}
 & -\frac{1}{2} \left[((\underline{y}_i + \bar{y}_i)y_i - \frac{(\underline{y}_i + \bar{y}_i)^2}{4}) + (\underline{z}_i + \bar{z}_i)z_i - \frac{(\underline{z}_i + \bar{z}_i)^2}{4} \right] \\
 & = \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i)z_i + (\underline{z}_i + \bar{z}_i)y_i - (\underline{y}_i + \underline{z}_i)(\bar{y}_i + \bar{z}_i) + \frac{(\underline{y}_i + \bar{y}_i)^2}{4} + \frac{(\underline{z}_i + \bar{z}_i)^2}{4} \right] \\
 & = \varphi^u(y_i, z_i).
 \end{aligned}$$

Hence, we have

$$\varphi^l(y_i, z_i) \leq \varphi(y_i, z_i) \leq \varphi^u(y_i, z_i). \tag{4}$$

Furthermore, we define the difference functions as follows:

$$\begin{aligned}
 \Delta(y_i) &= \varphi(y_i) - \varphi^l(y_i), & \nabla(y_i) &= \varphi^u(y_i) - \varphi(y_i), \\
 \Delta(z_i) &= \varphi(z_i) - \varphi^l(z_i), & \nabla(z_i) &= \varphi^u(z_i) - \varphi(z_i), \\
 \Delta(y_i, z_i) &= \varphi(y_i, z_i) - \varphi^l(y_i, z_i), & \nabla(y_i, z_i) &= \varphi^u(y_i, z_i) - \varphi(y_i, z_i).
 \end{aligned}$$

Since $\Delta(y_i)$ is convex about y_i , for any $y_i \in [\underline{y}_i, \bar{y}_i]$, it follows that $\Delta(y_i)$ can attain the maximum at the point \underline{y}_i or \bar{y}_i . Then

$$\max_{y_i \in [\underline{y}_i, \bar{y}_i]} \Delta(y_i) = \varphi(\bar{y}_i) - \varphi^l(\bar{y}_i) = \varphi(\underline{y}_i) - \varphi^l(\underline{y}_i) = \frac{(\bar{y}_i - \underline{y}_i)^2}{4}.$$

On the other hand, since $\nabla(y_i)$ is concave about y_i , for any $y_i \in [\underline{y}_i, \bar{y}_i]$, $\nabla(y_i)$ can attain the maximum at the point $\frac{\underline{y}_i + \bar{y}_i}{2}$, i.e.,

$$\max_{y_i \in [\underline{y}_i, \bar{y}_i]} \nabla(y_i) = \varphi^u\left(\frac{\underline{y}_i + \bar{y}_i}{2}\right) - \varphi\left(\frac{\underline{y}_i + \bar{y}_i}{2}\right) = \frac{(\bar{y}_i - \underline{y}_i)^2}{4}.$$

So, we have

$$\max_{y_i \in [\underline{y}_i, \bar{y}_i]} \Delta(y_i) = \max_{y_i \in [\underline{y}_i, \bar{y}_i]} \nabla(y_i) \rightarrow 0, \quad \text{as } |\bar{y}_i - \underline{y}_i| \rightarrow 0, \tag{5}$$

that is, $\Delta(y_i), \nabla(y_i) \rightarrow 0$, as $\|\bar{y} - \underline{y}\| \rightarrow 0$.

Similarly, we can prove that

$$\max_{z_i \in [\underline{z}_i, \bar{z}_i]} \Delta(z_i) = \max_{z_i \in [\underline{z}_i, \bar{z}_i]} \nabla(z_i) \rightarrow 0, \quad \text{as } |\bar{z}_i - \underline{z}_i| \rightarrow 0, \tag{6}$$

and $\Delta(z_i), \nabla(z_i) \rightarrow 0$, as $\|\bar{z} - \underline{z}\| \rightarrow 0$.

Define

$$\begin{aligned}
 \Delta(y_i + z_i) &= (y_i + z_i)^2 - \left[(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)(y_i + z_i) - \frac{(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)^2}{4} \right], \\
 \nabla(y_i + z_i) &= (\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)(y_i + z_i) - (\underline{y}_i + \underline{z}_i)(\bar{y}_i + \bar{z}_i) - (y_i + z_i)^2.
 \end{aligned}$$

Using the similar method, we can get the following conclusions:

$$\begin{aligned} & \max_{(y_i+z_i) \in [(\underline{y}_i+\underline{z}_i), (\bar{y}_i+\bar{z}_i)]} \Delta(y_i+z_i) \\ &= \max_{(y_i+z_i) \in [(\underline{y}_i+\underline{z}_i), (\bar{y}_i+\bar{z}_i)]} \nabla(y_i+z_i) \rightarrow 0, \quad \text{as } \|\bar{y}-\underline{y}\| \rightarrow 0, \|\bar{z}-\underline{z}\| \rightarrow 0. \end{aligned} \tag{7}$$

Since

$$\begin{aligned} \Delta(y_i, z_i) &= \varphi(y_i, z_i) - \varphi^l(y_i, z_i) \\ &= y_i z_i - \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i) z_i + (\underline{z}_i + \bar{z}_i) y_i - \frac{(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)^2}{4} + \underline{y}_i \bar{y}_i + \underline{z}_i \bar{z}_i \right] \\ &= \frac{1}{2} [(y_i + z_i)^2 - y_i^2 - z_i^2] \\ &\quad - \frac{1}{2} \left[(\underline{y}_i + \bar{y}_i) z_i + (\underline{z}_i + \bar{z}_i) y_i - \frac{(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)^2}{4} + \underline{y}_i \bar{y}_i + \underline{z}_i \bar{z}_i \right] \\ &= \frac{1}{2} \left[(y_i + z_i)^2 - \left((\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)(y_i + z_i) - \frac{(\underline{y}_i + \bar{y}_i + \underline{z}_i + \bar{z}_i)^2}{4} \right) \right] \\ &\quad + \frac{1}{2} \left[((\underline{y}_i + \bar{y}_i) y_i - \underline{y}_i \bar{y}_i - y_i^2) + ((\underline{z}_i + \bar{z}_i) z_i - \underline{z}_i \bar{z}_i - z_i^2) \right] \\ &= \frac{1}{2} \Delta(y_i + z_i) + \frac{1}{2} \nabla(y_i) + \frac{1}{2} \nabla(z_i) \\ &\leq \frac{1}{2} \max_{(y_i+z_i) \in [(\underline{y}_i+\underline{z}_i), (\bar{y}_i+\bar{z}_i)]} \Delta(y_i+z_i) + \frac{1}{2} \max_{y_i \in [\underline{y}_i, \bar{y}_i]} \nabla(y_i) + \frac{1}{2} \max_{z_i \in [\underline{z}_i, \bar{z}_i]} \nabla(z_i), \end{aligned}$$

by (5)–(7), we can get that

$$\Delta(y_i, z_i) \rightarrow 0, \quad \text{as } \|\bar{y}-\underline{y}\| \rightarrow 0, \|\bar{z}-\underline{z}\| \rightarrow 0. \tag{8}$$

Similarly, we can prove that $\nabla(y_i, z_i) \rightarrow 0$, as $\|\bar{y}-\underline{y}\| \rightarrow 0, \|\bar{z}-\underline{z}\| \rightarrow 0$.

For convenience, without loss of generality, for any $y \in Y \subseteq Y^0, z \in Z \subseteq Z^0$, define

$$f^L(y, z) = \sum_{i=1}^n c_i y_i + \sum_{i=1}^n \varphi^l(y_i, z_i).$$

By utilizing the convexity of a univariate quadratic function, we establish an effective method for generating the linear underestimation and linear overestimation of the functions y_i^2, z_i^2 , and $y_i z_i$, respectively. By the conclusions above, the linear relaxation underestimation function $f^L(y, z)$ of the function $f(y, z)$ for the (EP) can be established. Thus, by the former discussions, we can construct the corresponding approximation linear relaxation programming (LRP) of the (EP) as follows:

$$\text{(LRP):} \quad \begin{cases} \min f^L(y, z) = \sum_{i=1}^n c_i y_i + \sum_{i=1}^n \varphi^l(y_i, z_i) \\ \text{s.t. } Ay \leq b, \\ z_i = \sum_{k=1}^n q_{ik} y_k, i = 1, 2, \dots, n, \\ (y, z) \in \Omega^0. \end{cases}$$

By (4), it is obvious that

$$y_i z_i - \varphi^l(y_i, z_i) \geq 0,$$

then

$$\begin{aligned} f(y, z) - f^L(y, z) &= \sum_{k=1}^n c_k y_k + \sum_{k=1}^n y_k z_k - \left[\sum_{k=1}^n c_k y_k + \sum_{k=1}^n \varphi^l(y_k, z_k) \right] \\ &= \sum_{k=1}^n [y_k z_k - \varphi^l(y_k, z_k)] \\ &\geq 0. \end{aligned}$$

Thus, we have $f(y, z) - f^L(y, z) \geq 0$, i.e., $f(y, z) \geq f^L(y, z)$.

Furthermore, we have

$$\begin{aligned} f(y, z) - f^L(y, z) &= \sum_{k=1}^n [y_k z_k - \varphi^l(y_k, z_k)] \\ &= \sum_{k=1}^n \Delta(y_k, z_k). \end{aligned}$$

By (8), we have that $\Delta(y_i, z_i) \rightarrow 0$, as $\|\bar{y} - \underline{y}\| \rightarrow 0$, $\|\bar{z} - \underline{z}\| \rightarrow 0$.

Therefore, we have that

$$f(y, z) - f^L(y, z) \rightarrow 0 \quad \text{as } \|\bar{y} - \underline{y}\| \rightarrow 0, \|\bar{z} - \underline{z}\| \rightarrow 0.$$

Remark 1 From above, we only need to solve the (LRP) instead of solving the (EP) to obtain the lower and upper bounds of the optimal value in problem (NQP).

Remark 2 Based on the construction of the (LRP), each feasible point of the (NQP) in the sub-range Ω is also feasible to LRP, and the global minimum value of LRP is not more than that of the (NQP) in the sub-range Ω . Thus, the (LRP) can provide a valid lower bound for the global optimum value of problem NQP in the sub-range Ω .

Remark 3 The conclusions above ensure that the linear relaxation programming LRP can infinitely approximate the (NQP), as $\|Y\| \rightarrow 0$ (obviously, $\|Z\| \rightarrow 0$), this will guarantee the global convergence of the proposed algorithm.

3 Accelerating branch-and-bound algorithm and its convergence

Now we establish an accelerating branch-and-bound algorithm based upon the former linear relaxation approach for globally solving the (NQP). We will introduce the algorithm process at first and then give the convergence analysis of the algorithm.

3.1 Branching process

The critical operation of the branching process is iteratively subdividing the initial range Ω^0 into some sub-ranges, each sub-range denoted by Ω is concerned with a node of the branch and bound, such that any infinite iterative sequence of partition sets shrinks to a singleton. Here, we will adopt a standard range bisection approach, which is adequate to ensuring global convergence of the proposed algorithm. Detailed process is described as follows.

For any selected sub-range $\Omega' \in \Omega^0$, $Y' = [\underline{y}', \bar{y}'] \subseteq Y^0$, let $q \in \arg \max\{\bar{y}'_i - \underline{y}'_i : i = 1, 2, \dots, n\}$, and $y_m = (\underline{y}'_q + \bar{y}'_q)/2$, then divide Ω' into two sub-ranges Ω'_1 and Ω'_2 by subdividing the interval $[\underline{y}'_q, \bar{y}'_q]$ into two subintervals $[\underline{y}'_q, y_m]$ and $[y_m, \bar{y}'_q]$.

From the above branching operation, we can notice that the intervals $[\underline{z}', \bar{z}']$ of z will never be partitioned by the branching processes. Hence, these branching operations only occur in a space of dimension n , i.e., the proposed algorithm economizes the required computations.

3.2 Range deleting technique

At the k th iteration of the algorithm, for any rectangle $\Omega^k \subseteq \Omega^0$, we want to check whether Ω^k contains a global optimal solution of the (EP)(Ω^0), where

$$\Omega^k = \Omega_1^k \times \Omega_2^k \times \dots \times \Omega_{q-1}^k \times \Omega_q^k \times \Omega_{q+1}^k \times \dots \times \Omega_n^k,$$

with

$$\Omega_q^k = \{(y_q, z_q) \in R^2 \mid \underline{y}_q^k \leq y_q^k \leq \bar{y}_q^k, \underline{z}_q^0 \leq z_q^0 \leq \bar{z}_q^0\}.$$

Theorem 3.1 *Assume that \bar{f} is a known upper bound of the optimal value v of the (EP), for any sub-range $\Omega^k \subseteq \Omega^0$, the following conclusions hold: (i) If $ELB^k > \bar{f}$, then there is no global optimal solution for the (EP) over Ω^k ; (ii) If $ELB^k \leq \bar{f}$, then we have: for any $\tau \in \{1, 2, \dots, n\}$, if $c_\tau > 0$, then the sub-range $\bar{\Omega}^k$ does not contain any global optimal solution of the (EP); else if $c_\tau < 0$, then the sub-range $\underline{\Omega}^k$ does not contain any global optimal solution of the (EP), where*

$$ELB^k = \sum_{i=1}^n \min\{c_i \underline{y}_i^k, c_i \bar{y}_i^k\} + \sum_{i=1}^n \min\{\underline{y}_i^k \underline{z}_i^0, \underline{y}_i^k \bar{z}_i^0, \bar{y}_i^k \underline{z}_i^0, \bar{y}_i^k \bar{z}_i^0\},$$

$$\rho_\tau^k = \frac{\bar{f} - RLB^k + \min\{c_\tau \underline{y}_\tau^k, c_\tau \bar{y}_\tau^k\}}{c_\tau}, \quad \tau = 1, \dots, n,$$

$$\bar{\Omega}^k = \Omega_1^k \times \Omega_2^k \times \dots \times \Omega_{\tau-1}^k \times \bar{\Omega}_\tau^k \times \Omega_{\tau+1}^k \times \dots \times \Omega_n^k,$$

$$\underline{\Omega}^k = \Omega_1^k \times \Omega_2^k \times \dots \times \Omega_{\tau-1}^k \times \underline{\Omega}_\tau^k \times \Omega_{\tau+1}^k \times \dots \times \Omega_n^k,$$

with

$$\bar{\Omega}_\tau^k = \{(y_\tau, z_\tau) \in R^2 \mid \rho_\tau^k \leq y_\tau^k \leq \bar{y}_\tau^k, \underline{z}_\tau^0 \leq z_\tau^0 \leq \bar{z}_\tau^0\} \cap \Omega_\tau^k,$$

$$\underline{\Omega}_\tau^k = \{(y_\tau, z_\tau) \in R^2 \mid \underline{y}_\tau^k \leq y_\tau^k \leq \rho_\tau^k, \underline{z}_\tau^0 \leq z_\tau^0 \leq \bar{z}_\tau^0\} \cap \Omega_\tau^k.$$

Proof (i) If $ELB^k > \bar{f}$, then for all $(y, z) \in \Omega^k$,

$$\begin{aligned} f(y, z) &= \sum_{i=1}^n c_i y_i + \sum_{i=1}^n y_i z_i \\ &\geq \sum_{i=1}^n \min\{c_i y_i^k, c_i \bar{y}_i^k\} + \sum_{i=1}^n \min\{y_i^k z_i^0, y_i^k \bar{z}_i^0, \bar{y}_i^k z_i^0, \bar{y}_i^k \bar{z}_i^0\} \\ &= ELB^k > \bar{f}. \end{aligned}$$

Therefore, there is no global optimal solution for the (EP) over Ω^k .

(ii) If $ELB^k \leq \bar{f}$, for any $\tau \in \{1, 2, \dots, n\}$, then consider the following two cases.

Case 1: If $c_\tau > 0$, for all $(y, z) \in \bar{\Omega}^k$, we have $y_\tau > \rho_\tau$, i.e., $c_\tau y_\tau > \bar{f} - ELB^k + \min\{c_\tau y_\tau^k, c_\tau \bar{y}_\tau^k\}$.

Furthermore, we can get that

$$\begin{aligned} f(y, z) &= \sum_{i=1, i \neq \tau}^n c_i y_i + c_\tau y_\tau + \sum_{i=1}^n y_i z_i \\ &\geq \sum_{i=1, i \neq \tau}^n \min\{c_i y_i^k, c_i \bar{y}_i^k\} + c_\tau y_\tau + \sum_{i=1}^n \min\{y_i^k z_i^0, y_i^k \bar{z}_i^0, \bar{y}_i^k z_i^0, \bar{y}_i^k \bar{z}_i^0\} \\ &> \sum_{i=1, i \neq \tau}^n \min\{c_i y_i^k, c_i \bar{y}_i^k\} + \sum_{i=1}^n \min\{y_i^k z_i^0, y_i^k \bar{z}_i^0, \bar{y}_i^k z_i^0, \bar{y}_i^k \bar{z}_i^0\} + \bar{f} \\ &\quad - ELB^k + \min\{c_\tau y_\tau^k, c_\tau \bar{y}_\tau^k\} \\ &= ELB^k + \bar{f} - ELB^k \\ &= \bar{f}. \end{aligned}$$

Thus, we have

$$f(y, z) > \bar{f}.$$

Therefore, the range $\bar{\Omega}^k$ does not contain any global optimal solution of the (EP).

Case 2: If $c_\tau < 0$, then for any $(y, z) \in \bar{\bar{\Omega}}^k$, we have $y_\tau < \rho_\tau$, i.e., $c_\tau y_\tau > \bar{f} - ELB^k + \min\{c_\tau y_\tau^k, c_\tau \bar{y}_\tau^k\}$.

Similar to the proof of Case 1, we can get that

$$f(y, z) > \bar{f},$$

thus the range $\bar{\bar{\Omega}}^k$ does not contain any global optimal solution of the (EP). Thus, the proof is completed. \square

According to Theorem 3.1, we can construct the following range reduction technique to reject the whole investigated sub-range Ω^k or delete a part of the investigated sub-range Ω^k which does not contain any global optimal solution of the (EP).

Range deleting algorithm

Calculate ELB^k , if $ELB^k > \bar{f}$, then let $\Omega^k = \emptyset$. Otherwise, for each $\tau \in \{1, \dots, n\}$, if $c_\tau > 0$ and $\rho_\tau^k < \bar{y}_\tau^k$, then let $\bar{y}_\tau^k = \rho_\tau^k$ and $\bar{\Omega}^k$ with $\bar{\Omega}_\tau^k = \{(y_\tau, z_\tau) \in R^2 \mid y_\tau^k \leq y_\tau \leq \bar{y}_\tau^k, z_\tau^0 \leq z_\tau \leq \bar{z}_\tau^0\}$;

else if $c_\tau < 0$ and $\rho_\tau^k > \underline{y}_\tau^k$, then let $\underline{y}_\tau^k = \rho_\tau^k$ and $\overline{\Omega}_\tau^k$ with $\overline{\Omega}_\tau^k = \{(y_\tau, z_\tau) \in R^2 \mid \underline{y}_\tau^k \leq y_\tau^k \leq \overline{y}_\tau^k, \underline{z}_\tau^0 \leq z_\tau^0 \leq \overline{z}_\tau^0\}$.

3.3 Branch-and-bound algorithm

Assume that we have gotten the set of active nodes Λ_k at the k th iteration of the algorithm, and each node is associated with a sub-range $\Omega \subseteq \Omega^0$ for all $\Omega \in \Lambda_k$. For each sub-range Ω , use the proposed range deleting technique to compress the sub-range Ω , still denote the remaining sub-range by Ω , and obtain a lower bound $LB(\Omega)$ of the optimum value of the (NQP) by solving the (LRP) in Ω .

Let $f_r(\Omega)$ and $(y_r(\Omega), z_r(\Omega))$ be the optimal value and the optimal solution of the (LRP) over the sub-range Ω , respectively. Combining the former linear relaxation method, branching process, and range deleting technique, the basic steps of the proposed accelerating algorithm for globally solving the (NQP) may be stated as follows.

Algorithm statement

Initialization step. Let the initial iteration counter $k := 0$, the initial set of active nodes $\Lambda_0 = \{\Omega^0\}$, the initial upper bound $\bar{f} = +\infty$, the convergent error $\varepsilon > 0$, and the initial collection of feasible points $F := \emptyset$.

Solve the (LRP) over Ω^0 , obtain its optimal solution $y^0 := y_r(\Omega^0)$, $z^0 := z_r(\Omega^0)$ and optimal objective function value $LB_0 := f_r(\Omega^0)$. If (y^0, z^0) is feasible to the (NQP), then let $\bar{f} = f(y^0, z^0)$ and $F = F \cup \{(y^0, z^0)\}$. If $\bar{f} - LB_0 \leq \varepsilon$, then algorithm stops with (y^0, z^0) as an ε -global optimum solution for the (NQP). Otherwise, proceed with the following *Range deleting step*.

Range deleting step. For each sub-range Ω , use the proposed range deleting technique in Sect. 3.2 to contract each sub-range Ω and still denote the remaining sub-range by Ω .

Range division step. Apply the presented range division approach to subdivide Ω^k into two new sub-ranges, and denote the collection of new subdivided sub-ranges by $\widehat{\Omega}^k$.

Renewing the lower and upper bound step. If $\widehat{\Omega}^k \neq \emptyset$ for each $\Omega \in \widehat{\Omega}^k$, then solve the LRP(Ω) to get $f_r(\Omega)$ and $(y_r(\Omega), z_r(\Omega))$. Let $LB(\Omega) := f_r(\Omega)$, if $LB(\Omega) > \bar{f}$, set $\widehat{\Omega}^k := \widehat{\Omega}^k \setminus \Omega$. The residual subdivided set is now $\Lambda_k := (\Lambda_k \setminus \Omega^k) \cup \widehat{\Omega}^k$, and renew the lower bound $LB_k := \inf_{\Omega \in \Lambda_k} LB(\Omega)$.

If the midpoint y_{mid} of Y is feasible to problem (NQP), set $z_{\text{mid}} = Qy_{\text{mid}}$, then let $F := F \cup \{(y_{\text{mid}}, z_{\text{mid}})\}$. Furthermore, if $y_r(\Omega)$ is feasible to problem (NQP), then let $F := F \cup \{(y_r(\Omega), z_r(\Omega))\}$.

If $F \neq \emptyset$, renew the upper bound $\bar{f} := \min_{(y,z) \in F} f(y, z)$, and denote the known best feasible solution by $(y_{\text{best}}, z_{\text{best}}) := \operatorname{argmin}_{(y,z) \in F} f(y, z)$.

Termination checking step. If $\bar{f} - LB_k \leq \varepsilon$, the algorithm ends with \bar{f} and y_{best} as the ε -global optimum value and a global optimal solution of problem (NQP). Otherwise, set $k := k + 1$ and pick out an active node Ω^{k+1} satisfying $\Omega^{k+1} = \operatorname{argmin}_{\Omega \in \Lambda_k} LB(\Omega)$, and return to *Range deleting step*.

3.4 Convergence analysis

In this subsection, the convergence of this algorithm is discussed as follows.

Theorem 3.2 *Suppose that the feasible region D of the (NQP) is nonempty, then the above algorithm either terminates finitely with a global optimum solution of the (NQP), or generates an infinite sequence $\{y^k\}$ of which any accumulation point will be a global optimal solution of the (NQP).*

Proof If the algorithm terminates finitely, it terminates at the k th iteration, where $k \geq 0$. Upon termination, by steps in the algorithm, we get that $\bar{f} - LB_k \leq \varepsilon$. By the first five steps in the algorithm, there exists a feasible solution y^* for the (NQP), which satisfies that $\bar{f} = f(y^*)$, and $f(y^*) - LB_k \leq \varepsilon$. According to the algorithm, we have that $LB_k \leq \nu$. Since y^* is a feasible solution of the (NQP), we have $f(y^*) \geq \nu$. Combining the above inequalities, we get that $\nu \leq f(y^*) \leq LB_k + \varepsilon \leq \nu + \varepsilon$, i.e., $\nu \leq f(y^*) \leq \nu + \varepsilon$. Therefore, y^* is a global ε -optimum solution for the (NQP).

If the algorithm is infinite, according to the algorithm, we have that $\{LB_k\}$ is a non-decreasing sequence and it has an upper bound $\min_{y \in D} f(y)$, which ensures the existence of the limit $LB := \lim_{k \rightarrow \infty} LB_k \leq \min_{y \in D} f(y)$. Since the sequence $\{y^k\}$ is included in a compact set Y^0 , there must exist a convergent subsequence $\{y^s\} \subseteq \{y^k\}$ and assume that $\lim_{s \rightarrow \infty} y^s = y^*$. Then by the branching process and deleting method, there must exist a decreasing subsequence $\{Y^r\} \subset Y^s$, where $(Y^s, Z^s) \in \Lambda_s$ with $y^r \in Y^r$, $LB_r = LB(Y^r) = f^L(y^r)$ and $\lim_{r \rightarrow \infty} Y^r = \{y^*\}$. By the continuity of function $\Psi_0(y)$, we have

$$\begin{aligned} \lim_{r \rightarrow \infty} LB_r &= \lim_{r \rightarrow \infty} f^L(y^r) \\ &= \lim_{r \rightarrow \infty} f(y^r) \\ &= f(y^*). \end{aligned}$$

Then, since Y^0 is a closed set and $\{y^k\} \subset Y^0$, obviously, we have that $y^* \in Y^0$, i.e., y^* is a feasible solution of the (NQP). Thus, the proof is completed. □

4 Numerical experiments

To verify the performance of the proposed algorithm, some numerical experiments are reported and compared with the known methods [19–25]. The algorithm is implemented by Matlab 2016a, and the tests are run in a microcomputer with Intel(R) Xeon(R) processor of 2.4 GHz, 4 GB of RAM memory, under the Win10 operational system. We used linprog solver to solve all linear programming problems, and the convergent error is set to ε in our procedure. For these test examples, the numerical results compared with the current algorithms are demonstrated in Tables 1 and 2, where the following notations have

Table 1 Numerical comparisons for test Examples 1–6

E.g.	Methods	Opt. val.	Opt. sol.	Iter.	Time (s)
1	ours	10.00000	(2.0000, 8.0000)	3	0.35725
	[25]	10.0000090	(1.9999998, 7.9999988)	41	0.02
	[26]	10	(2, 8)	27	10.83
	[27]	10	(2, 8)	53	0.3
2	ours	3.00000	(0.0000, 4.0000)	8	0.01791
	[28]	3.00000	(0.0000, 4.0000)	25	0.750
3	ours	0.890185	(1.3148, 0.1396, 0.0, 0.4233)	1	0.38982
	[27]	0.8902	(1.3148, 0.1396, 0.0, 0.4233)	1	0.1880
	[29]	0.890193	(1.314792, 0.13955, 0.0, 0.423286)	6	
4	ours	-16.22662	(0.0, 3.6403, 0.0, 2.9029, 1.9388, 0.0)	5	0.59005
5	ours	-3.00000	(3.0000, 3.0000)	30	4.15970
6	ours	-1.06250	(0.7500, 2.0000)	3	0.26845

Table 2 Numerical comparisons with Ref. [12, 30] for Example 7

Dimension	Methods	Opt. val.	Iter.	Time (s)
$n = 5$	[12]	-25.0	141	10.11
	[30]	-25.0	12	0.0187
	ours	-25.0	1	0.01254
$n = 10$	[12]	-100.0	283	21.86
	[30]	-100.0	31	0.3342
	ours	-100.0	7	0.25649
$n = 20$	[12]	-400.0	651	47.00
	[30]	-400.0	86	5.9396
	ours	-400.0	15	2.73556
$n = 30$	[12]	-900.0	965	106.33
	[30]	-900.0	204	44.8577
	ours	-900.0	18	11.25635
$n = 50$	[12]	-2500.0	1891	304.30
	ours	-2500.0	21	17.35219
$n = 80$	ours	-6400.0	37	40.45623
$n = 100$	ours	-10,000.0	51	44.35865
$n = 150$	ours	-22,500.0	66	112.99298

been used in column headers: Opt.Val.: optimal value; Opt.Sol.: optimal solution; Iter.: the number of iterations.

Example 1 (Refs. [25–27])

$$\left\{ \begin{array}{l} \min(y_1 + y_2)(y_1 - y_2 + 7) \\ \text{s.t. } 2y_1 + y_2 \leq 14, \\ \quad y_1 + y_2 \leq 10, \\ \quad -4y_1 + y_2 \leq 0, \\ \quad 2y_1 + y_2 \geq 6, \\ \quad y_1 + 2y_2 \geq 6, \\ \quad y_1 - y_2 \leq 3, \\ \quad y_1 \leq 5, \\ \quad y_1 + y_2 \geq 0, \\ \quad y_1 - y_2 + 7 \geq 0. \end{array} \right.$$

Example 2 (Ref. [28])

$$\left\{ \begin{array}{l} \min y_1 + (2y_1 - 3y_2 + 13)(y_1 + y_2 - 1) \\ \text{s.t. } -y_1 + 2y_2 \leq 8, \\ \quad -y_2 \leq -3, \\ \quad y_1 + 2y_2 \leq 12, \\ \quad y_1 - 2y_2 \leq -5, \\ \quad y_1, y_2 \geq 0. \end{array} \right.$$

Example 3 (Refs. [27, 29])

$$\left\{ \begin{array}{l} \min(0.813396y_1 + 0.67440y_2 + 0.305038y_3 + 0.129742y_4 + 0.217796) \\ \quad \times (0.224508y_1 + 0.063458y_2 + 0.932230y_3 + 0.528736y_4 + 0.091947) \\ \text{s.t. } 0.488509y_1 + 0.063565y_2 + 0.945686y_3 + 0.210704y_4 \leq 3.562809, \\ \quad -0.324014y_1 - 0.501754y_2 - 0.719204y_3 + 0.099562y_4 \leq -0.052215, \\ \quad 0.445225y_1 - 0.346896y_2 + 0.637939y_3 - 0.257623y_4 \leq 0.427920, \\ \quad -0.202821y_1 + 0.647361y_2 + 0.920135y_3 - 0.983091y_4 \leq 0.840950, \\ \quad -0.886420y_1 - 0.802444y_2 - 0.305441y_3 - 0.180123y_4 \leq -1.353686, \\ \quad -0.515399y_1 - 0.424820y_2 + 0.897498y_3 + 0.187268y_4 \leq 2.137251, \\ \quad -0.591515y_1 + 0.060581y_2 - 0.427365y_3 + 0.579388y_4 \leq -0.290987, \\ \quad 0.423524y_1 + 0.940496y_2 - 0.437944y_3 - 0.742941y_4 \leq 0.373620, \\ \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0, y_4 \geq 0. \end{array} \right.$$

Example 4

$$\left\{ \begin{array}{l} \min 6.5y_1 - 0.5y_1^2 - y_2 - 2y_3 - 3y_4 - 2y_5 - y_6 \\ \text{s.t. } y_1 + 2y_2 + 8y_3 + y_4 + 3y_5 + 5y_6 \leq 16, \\ \quad -8y_1 - 4y_2 - 2y_3 + 2y_4 + 4y_5 - y_6 \leq -1, \\ \quad 2y_1 + 0.5y_2 + 0.2y_3 - 3y_4 - y_5 - 4y_6 \leq 24, \\ \quad 0.2y_1 + 2y_2 + 0.1y_3 - 4y_4 + 2y_5 + 2y_6 \leq 12, \\ \quad -0.1y_1 - 0.5y_2 + 2y_3 + 5y_4 - 5y_5 + 3y_6 \leq 3, \\ \quad 0 \leq y_i \leq 10, i = 1, 2, \dots, 6. \end{array} \right.$$

Example 5

$$\left\{ \begin{array}{l} \min 2y_1 + 3y_2 - 2y_1^2 - 2y_2^2 + 2y_1y_2 \\ \text{s.t. } -y_1 + y_2 \leq 1, \\ \quad y_1 - y_2 \leq 1, \\ \quad -y_1 + 2y_2 \leq 3, \\ \quad 2y_1 - y_2 \leq 3, \\ \quad 0 \leq y_1 \leq 15, 0 \leq y_2 \leq 15. \end{array} \right.$$

Example 6

$$\left\{ \begin{array}{l} \min y^T Q y + c^T y \\ \text{s.t. } A y \leq b, \\ \quad y \in Y^0 = \{0 \leq y_1 \leq 2, 0 \leq y_2 \leq 2\}, \end{array} \right.$$

where $c = (2, 4)^T, b = (1, 2, 4, 3, 1)^T, Q = \begin{pmatrix} -1 & 2 \\ 2 & -4 \end{pmatrix}, A = \begin{pmatrix} -4 & 2 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & -4 \end{pmatrix}$.

Example 7 (Ref. [12])

$$\begin{cases} \min - \sum_{i=1}^n y_i^2 \\ \text{s.t. } \sum_{i=1}^j y_i \leq j, j = 1, 2, \dots, n, \\ y_i \geq 0, i = 1, 2, \dots, n. \end{cases}$$

Table 2 lists the numerical results of our algorithm, Gao’s algorithm [12], and Jiao’s algorithm [30] on Example 7. By comparing the numerical results in Table 2, we can conclude that our algorithm applied to Example 7 is superior to Gao’s algorithm [12] and Jiao’s algorithm [30] in terms of number of iterations and time.

Some randomly generated test examples with a large scale number of variables and constraints are used to validate the robustness of the proposed algorithm. These randomly generated examples and their computational results are given as follows.

Example 8

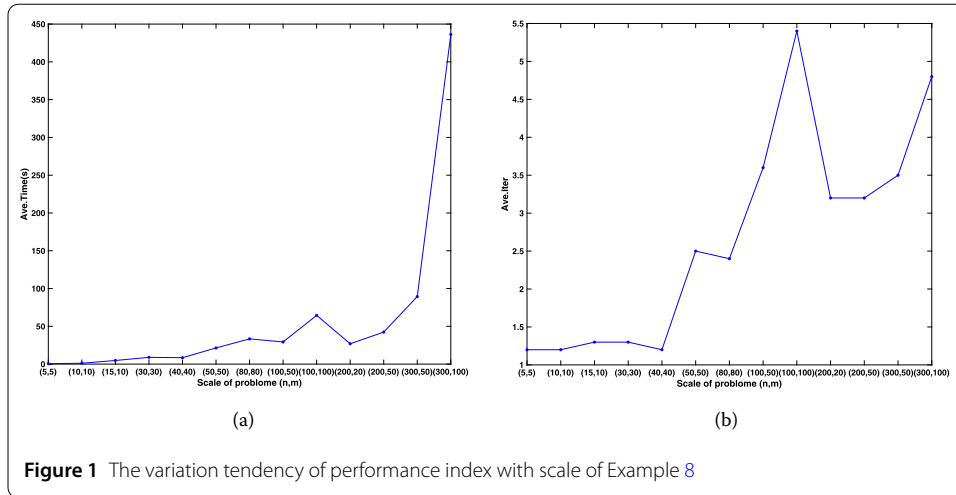
$$\begin{cases} \min y^T Q y + c^T y \\ \text{s.t. } A y \leq b, \\ y \in Y^0 = \{-10 \leq y_i \leq 10, i = 1, 2, \dots, n\}, \end{cases}$$

where $Q_{n \times n}$ is a real symmetric matrix, all the real elements of $Q_{n \times n}$, $A_{m \times n}$, and $c_{n \times 1}$ are randomly generated in Interval $[-2, 2]$, all the real elements of $b_{m \times 1}$ are randomly generated in Interval $[1, 10]$. For Example 8, we solved 10 different random instances for each size and presented statistics of the results. The computational results are summarized in Table 3, where the following notations have been used in column headers: Ave.Iter.: the average number of iterations; Ave.Time (s): the average CPU execution time for the algorithm in seconds; m : the number of constraints; n : the number of variables.

From Table 3 and Fig. 1, it can be seen that, when m and n are below 50, the algorithm can find the global optimal solution in a short time and with lower iteration number. As the problem size becomes larger, the average number of iterations and the average CPU time of our algorithm are also increased, but they are not very sensitive to the problem size.

Table 3 Numerical results for Example 8

n	m	Ave.Iter.	Ave.Time (s)
5	5	1.2	0.5089
10	10	1.2	1.1423
15	10	1.3	4.7854
30	30	1.3	8.9569
40	40	1.2	8.4528
50	50	2.5	21.3985
80	80	2.4	33.3828
100	50	3.6	29.3468
100	100	5.4	64.5159
200	20	3.2	26.9366
200	50	3.2	42.3695
300	50	3.5	89.3652
300	100	4.8	436.2315



From the experimental results in Table 3, we can see that the proposed algorithm with the given convergent error can be used to globally solve the (NPQ) with a large scale number of constraints and variables. The results in Tables 1–3 show that our algorithm is both feasible and efficient.

5 Concluding remarks

In this paper, we propose a new branch-and-bound algorithm for globally solving the non-convex quadratic programming problem. In this algorithm, we present a new linear relaxation method, which can be used to derive the linear programs relaxation problem of the investigated problem (NPQ). To accelerate the computational speed of the proposed branch-and-bound algorithm, an interval deleting rule is used to reduce the investigated regions. By subsequently partitioning the initial region and solving a sequence of linear programs relaxation problems, the proposed algorithm is convergent to the global optima of the initial problem (NPQ). Finally, compared with some existent algorithms, numerical results show higher computational efficiency of the proposed algorithm.

Acknowledgements

The authors are grateful to the responsible editor and the anonymous referees for their valuable comments and suggestions, which have greatly improved the earlier version of this paper.

Funding

This work is supported by the National Natural Science Foundation of China (No. 61373174, No. 11501168, No. 11602184); the Science and Technology Key Project of Education Department of Henan Province (No. 16A110030).

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors contributed equally to the manuscript, and they read and approved the final manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 15 April 2018 Accepted: 4 July 2018 Published online: 16 July 2018

References

1. Khammash, M.H.: Synthesis of globally optimal controllers for robust performance to unstructured uncertainty. *IEEE Trans. Autom. Control* **41**(2), 189–198 (1996)

2. Kedem, G., Watanabe, H.: Graph-optimization techniques for IC layout and compaction. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **3**(1), 12–20 (1984)
3. Lodwick, W.A.: Preprocessing nonlinear functional constraints with applications to the pooling problem. *INFORMS J. Comput.* **4**, 119–131 (1992)
4. Floudas, C.A., Visweswaran, V.: Primal-relaxed dual global optimization approach. *J. Optim. Theory Appl.* **78**(2), 187–225 (1993)
5. Li, Y., Kang, L., Garis, H.D., Kang, Z., Liu, P.: A robust algorithm for solving nonlinear programming problems. *Int. J. Comput. Math.* **79**(5), 523–536 (2002)
6. Shen, C., Xue, W., Pu, D.: A globally convergent trust region multidimensional filter SQP algorithm for nonlinear programming. *Int. J. Comput. Math.* **86**(12), 2201–2217 (2009)
7. Wang, H., Liu, F., Gu, C., Pu, D.: An infeasible active-set QP-free algorithm for general nonlinear programming. *Int. J. Comput. Math.* **94**(5), 884–901 (2016)
8. Vandenbussche, D., Nemhauser, G.L.: A branch-and-cut algorithm for nonconvex quadratic programs with box constraints. *Math. Program.* **102**(3), 559–575 (2005)
9. Burer, S., Vandenbussche, D.: Globally solving box-constrained nonconvex quadratic programs with semidefinite-based finite branch-and-bound. *Comput. Optim. Appl.* **43**(2), 181–195 (2009)
10. Cambini, R., Sodini, C.: Decomposition methods for solving nonconvex quadratic programs via branch and bound. *J. Glob. Optim.* **33**(3), 313–336 (2005)
11. Li, H.M., Zhang, K.C.: A decomposition algorithm for solving large-scale quadratic programming problems. *Appl. Math. Comput.* **173**(1), 394–403 (2006)
12. Gao, Y., Xue, H., Shen, P.: A new rectangle branch-and-reduce approach for solving nonconvex quadratic programming problems. *Appl. Math. Comput.* **168**(2), 1409–1418 (2005)
13. Burer, S., Vandenbussche, D.: A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations. *Math. Program.* **113**(2), 259–282 (2008)
14. Al-Khayyal, F.A., Larsen, C., Voorhis, T.V.: A relaxation method for nonconvex quadratically constrained quadratic programs. *J. Glob. Optim.* **6**(3), 215–230 (1995)
15. Linderoth, J.: A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Math. Program.* **103**(2), 251–282 (2005)
16. Raber, U.: A simplicial branch-and-bound method for solving nonconvex all-quadratic programs. *J. Glob. Optim.* **13**(4), 417–432 (1998)
17. Sherali, H.D., Adams, W.P.: *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*, vol. 31. Springer, Berlin (1999)
18. Audet, C., Hansen, P., Jaumard, B., Savard, G.: A branch and cut algorithm for nonconvex quadratically constrained quadratic programming. *Math. Program.* **87**(1), 131–152 (2000)
19. Lu, C., Deng, Z., Jin, Q.: An eigenvalue decomposition based branch-and-bound algorithm for nonconvex quadratic programming problems with convex quadratic constraints. *J. Glob. Optim.* **67**(3), 475–493 (2017)
20. Gao, Y., Shang, Y., Zhang, L.: A branch and reduce approach for solving nonconvex quadratic programming problems with quadratic constraints. *Oper. Res. Trans.* **9**(2), 9–20 (2005)
21. Qu, S.J., Zhang, K.C., Ji, Y.: A global optimization algorithm using parametric linearization relaxation. *Appl. Math. Comput.* **186**(1), 763–771 (2007)
22. Shen, P., Duan, Y., Ma, Y.: A robust solution approach for nonconvex quadratic programs with additional multiplicative constraints. *Appl. Math. Comput.* **201**(1–2), 514–526 (2008)
23. Shen, P., Liu, L.: A global optimization approach for quadratic programs with nonconvex quadratic constraints. *Chin. J. Eng. Math.* **25**(5), 923–926 (2008)
24. Shen, P., Wang, C.: Linear decomposition approach for a class of nonconvex programming problems. *J. Inequal. Appl.* **2017**(1), 74 (2017)
25. Chen, Y., Jiao, H.: A nonisolated optimal solution of general linear multiplicative programming problems. *Comput. Oper. Res.* **36**(9), 2573–2579 (2009)
26. Gao, Y., Wu, G., Ma, W.: A new global optimization approach for convex multiplicative programming. *Appl. Math. Comput.* **216**(4), 1206–1218 (2010)
27. Wang, C.F., Liu, S.Y., Shen, P.P.: Global minimization of a generalized linear multiplicative programming. *Appl. Math. Model.* **36**(6), 2446–2451 (2012)
28. Jiao, H.W., Liu, S.Y., Zhao, Y.F.: Effective algorithm for solving the generalized linear multiplicative problem with generalized polynomial constraints. *Appl. Math. Model.* **39**(23–24), 7568–7582 (2015)
29. Thoai, N.V.: A global optimization approach for solving the convex multiplicative programming problem. *J. Glob. Optim.* **1**(4), 341–357 (1991)
30. Jiao, H., Liu, S., Lu, N.: A parametric linear relaxation algorithm for globally solving nonconvex quadratic programming. *Appl. Math. Comput.* **250**, 973–985 (2015)