SIAM J. OPTIM.
Vol. 29, No. 3, pp. 1844–1869

A LINEAR-TIME ALGORITHM FOR GLOBALLY MAXIMIZING THE SUM OF A GENERALIZED RAYLEIGH QUOTIENT AND A QUADRATIC FORM ON THE UNIT SPHERE*

LONG-FEI WANG† AND YONG XIA‡

Abstract. We study the problem, which we refer to as problem (P), of maximizing the sum of a generalized Rayleigh quotient and a quadratic form on the unit sphere. The computational complexity is first analyzed. Then we reformulate (P) as a new univariate optimization problem (P_α). Though the objective function has no closed-form expression, with the help of an extended S-lemma, the function value evaluation of (P_α) is reduced to minimizing a nonsmooth univariate convex function and hence can be efficiently solved by bisection search. We propose a novel approach for overestimating the objective function of (P_α), which leads to a new efficient branch-and-bound algorithm for solving (P_α). We show that, with a high probability, the new algorithm can find a global ϵ-approximation solution of (P_α) in linear time in terms of the number of nonzero elements of the input matrices. All tested numerical results demonstrate that the new algorithm highly outperforms not only the software BARON but also the recent global optimization algorithm based on Lipschitz bounds.

Key words. fractional programming, quadratic constrained quadratic programming, Rayleigh quotient, semidefinite programming, S-lemma, branch and bound

AMS subject classifications. 90C26, 90C32, 90C20, 90C22

DOI. 10.1137/18M1164639

1. Introduction.

We study the problem of maximizing the sum of a generalized Rayleigh quotient and a quadratic form on the unit sphere, which was first proposed by Zhang [29, 30]:

$$\text{(P)} \qquad \begin{aligned} \max \quad & f(x) = \frac{x^T B x}{x^T W x} + x^T D x \\ \text{s.t.} \quad & x \in S := \{x \in \mathbb{R}^n : \ \|x\| = 1\}, \end{aligned}$$

where B, W, and D are $n \times n$ symmetric matrices, W is assumed to be positive definite so that the denominators are positive, and $\|\cdot\|$ denotes the standard ℓ_2-norm.

Problem (P) can be regarded as a class of regularized spectral optimization problems [15]. Recently, more real applications of (P) in the downlink of a multiuser MIMO system [20] and the sparse Fisher discriminant analysis in pattern recognition [6, 9, 25] were reported by Zhang [29, 30]. In addition, simply setting $D = 0$ in (P) yields the well-known single generalized Rayleigh quotient optimization problem, which is related to the classical eigenvalue problem [31]. Thus, (P) is a normalization of maximizing the sum of two generalized Rayleigh quotients:

$$\max_{x \neq 0} \ \frac{x^T B x}{x^T W x} + \frac{x^T D x}{x^T V x}, \tag{1.1}$$

*Received by the editors January 10, 2018; accepted for publication (in revised form) May 8, 2019; published electronically July 11, 2019.
https://doi.org/10.1137/18M1164639
Funding: This research was supported by the National Natural Science Foundation of China under grants 11822103, 11571029, and 11771056, and by the Beijing Natural Science Foundation grant Z180005.

†LMIB of the Ministry of Education, School of Mathematics and System Sciences, Beihang University, Beijing, 100191, People's Republic of China (lfking@buaa.edu.cn).
‡Corresponding author. LMIB of the Ministry of Education, School of Mathematics and System Sciences, Beihang University, Beijing, 100191, People's Republic of China (yxia@buaa.edu.cn).

where W and V are positive definite. Here we could provide a new special "application" of (1.1). Generally, the so-called "most-fair" point is defined to be the solution at which the maximal value between $\frac{f_1(x)}{f_2(x)}$ and $\frac{f_2(x)}{f_1(x)}$ is close to 1 as much as possible. Suppose both f_1 and f_2 take positive values. It is not difficult to show that finding the most-fair point is equivalent to minimizing the sum of $\frac{f_1(x)}{f_2(x)}$ and $\frac{f_2(x)}{f_1(x)}$; see [7]. Thus, we get a special case of (1.1) if f_1 and f_2 are both positive definite quadratic forms.

In the area of fractional programming, problem (P) belongs to the quadratically constrained sum-of-quadratic-ratios; see the references in [13]. Setting $D = B$ or $D = W$ reduces (P) to the simultaneous optimization of absolute and relative terms, which dates back to Schaible [22] in 1984. Recently, (P) was equivalently embedded into the problem of maximizing the sum-of-linear-ratios over the cone of positive semidefinite matrices [26].

As a nonconvex programming problem on the sphere, (P) is a fractional extension of the classical trust-region subproblem [5], which was shown recently to have a linear-time approximation scheme [12, 23]. Problem (P) also plays a role in the partial Lagrangian function of quadratic fractional programming with two quadratic constraints [18, 28] with direct applications in computer vision [8].

Problem (P) contains a sum-of-linear-ratios optimization as a special case. When the input matrices B, W, and D are all diagonal (see, for example, the last two test examples in [17]), by replacing the variables x_i^2 by new nonnegative variables y_i, (P) reduces to the problem of maximizing the sum of a linear function and a linear ratio over the standard simplex. It is well known that minimizing the sum of two linear ratios over a general polyhedron is NP-hard [16]. One contribution of this paper is to show that the all-diagonal case of (P) can be efficiently and polynomially solved.

Generally, problem (P) is difficult to solve. As shown by Example 1.1 in [29], there could be a few local nonglobal maximizers of (P). Moreover, even finding the critical points of (P) is nontrivial; see [29, 30]. Recently, (P) was shown in [26] to have a fully polynomial time approximation scheme (FPTAS). In this paper, we can show that (P) can be reformulated as a nonconvex quadratic optimization problem with seven quadratic constraints including one ellipsoidal constraint, and hence is polynomially solvable. However, that does not mean that there is an efficient algorithm for globally solving (P).

In order to globally solve (P), Nguyen, Sheu, and Xia [17] first recast (P) as a univariate maximization problem, where the functional evaluations correspond to solving semidefinite programming (SDP) problems when $n \geq 3$ is assumed (noting that the other cases $n = 1, 2$ are trivial). Then, a "two-stage" heuristic algorithm was proposed to solve the univariate optimization. It first subdivides the feasible region into coarse intervals and then applies the quadratic fit line search in each interval. It is conjectured based on a large number of numerical experiments that the univariate objective function is unimodal. Recently, X. H. Wang, L. F. Wang, and Xia [24] established a Lipschitz upper bounding approach for the univariate function and then proposed an efficient branch-and-bound (L-BB) algorithm. Their numerical results demonstrated that Algorithm L-BB outperforms the two-stage algorithm.

In this paper, we first show (P) can be converted to an equivalent quadratic constrained quadratic programming, which implies the polynomial solvability of (P), though the corresponding algorithm is not practical. Interestingly, when all the input matrices are diagonal, we show there is an efficient $O(n^2)$ time algorithm for globally solving (P). For the general (P), we reformulate it as maximizing a new univariate

function. Based on an extended S-lemma with equality, the function evaluation is also shown to be equivalent to an SDP subproblem when $n \geq 3$. The classical bisection search approach is employed to solve the SDP subproblem by iteratively solving a series of maximum eigenvalue problems. Then, we develop an adaptive branch-and-bound algorithm to globally solve (P) based on a novel easy-to-compute and high-quality overestimation of the black-box univariate function. To find a global ϵ-approximation solution of (P), surprisingly, our new algorithm has a linear-time computational complexity in terms of the number of nonzero elements of the input matrices based on probabilistic analysis. Numerical results demonstrate that our new algorithm highly outperforms not only the approach to solve the QCQP reformulation using the nonconvex optimization software BARON but also the existing global optimization Algorithm L-BB.

The remainder of this paper is organized as follows. In section 2, we show (P) is polynomial solvable, though there is no practical algorithm. Then, supposing all input matrices are diagonal, we show that (P) can be efficiently solved in $O(n^2)$ time. Section 3 presents the existing global optimization algorithm for solving (P) based on Lipschitz bounds. In section 4, we present a new univariate reformulation of (P) and show that the function evaluation can be efficiently solved by bisection search. In section 5, we propose a new branch-and-bound algorithm to solve the univariate maximization problem and show that the computational complexity could be linear time in terms of the number of nonzero elements of the input matrices. In section 6, we do numerical experiments and report numerical comparison between BARON, the algorithm based on Lipschitz bounds, and our new algorithm.

Throughout this paper, we assume $n \geq 3$. Denote by $v(\cdot)$ the optimal value of the problem $(\cdot)$. The notation ":=" means "define." Denote by $A \succeq 0$ (resp., $A \preceq 0$) a positive (resp., negative) semidefinite matrix A. In particular, $A \succ 0$ denotes that A is positive definite. The maximal and minimal eigenvalues of A are denoted by $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$, respectively. For two symmetric matrices A and B, the inner product is denoted by $A \bullet B = \mathrm{tr}(AB^T) = \sum_{i,j} A_{ij}B_{ij}$. $\mathrm{Diag}(a_1, \ldots, a_n)$ returns a diagonal matrix with diagonal elements $a_1, \ldots, a_n$. For a real number a, $\lceil a \rceil$ returns the largest integer less than or equal to a. Denote by $[a, b]$ and (a, b) the intervals $\{x \in \mathbb{R} : a \leq x \leq b\}$ and $\{x \in \mathbb{R} : a < x < b\}$, respectively.

2. Polynomial solvability. Consider the general quadratic constrained quadratic programming problem

$$\text{(GQCQP)} \qquad \max_x \{q_0(x) : \ q_i(x) \leq 0, \ i = 1, \ldots, p\},$$

where $q_0(x), q_1(x), \ldots, q_p(x)$ are general quadratic functions. Given $\epsilon \in (0, 1)$, a vector $\widehat{x}$ is called an ϵ-approximation solution of (GQCQP) if $q_i(\widehat{x}) \leq \epsilon$ for $i = 1, \ldots, p$ and $q_0(\widehat{x}) \geq v(\text{GQCQP}) - \epsilon$. We assume that $p \geq 1$ is a fixed integer and at least one of the $q_i(x)$ $(i \in \{1, \ldots, p\})$ is strictly convex. Then based on Barvinok's construction [2], Bienstock proposed an algorithm that either proves that (GQCQP) is infeasible or computes an ϵ-approximation solution of (GQCQP) in polynomial time with respect to the number of bits in the data and $\log \epsilon^{-1}$ [3].

We reformulate (P) as a special case of (GQCQP). Without loss of generality, we assume $B \succ 0$ and $D \succ 0$ in (P), since $W \succ 0$ and for any α and β it holds that

$$\forall x \in S : \ f(x) = \frac{x^T B x}{x^T W x} + x^T D x = \frac{x^T (B + \alpha W) x}{x^T W x} + x^T (D + \beta I) x - \alpha - \beta.$$

By introducing additional variables s and t to replace x^TBx/x^TWx and x^TWx, respectively, we can reformulate (P) as the following quadratic constrained quadratic programming problem:

$$\text{(QCQP)} \qquad \max_{x\in\mathbb{R}^n, s,t\in\mathbb{R}} \left\{ s + x^TDx : \ st = x^TBx,\ x^TWx = t,\ x^Tx = 1 \right\}.$$

We notice that the feasible region of (QCQP) is bounded. Define $M_t := \lambda_{\max}(W)$ and $M_s = \lambda_{\max}(B)/\lambda_{\min}(W)$. Let (x, s, t) be any feasible solution to (QCQP). We have $s^2 \le M_s^2$ and $t^2 \le M_t^2$. Therefore, we can further reformulate (QCQP) as the following QCQP with inequality constraints:

$$\begin{aligned} \max_{x\in\mathbb{R}^n,\ s,t\in\mathbb{R}} \ & s + x^TDx \\ \text{s.t.} \quad & st \le x^TBx \le st,\ t \le x^TWx \le t,\ 1 \le x^Tx \le 1, \\ & x^Tx + s^2 + t^2 \le 1 + M_s^2 + M_t^2, \end{aligned} \tag{2.1}$$

where the redundant constraint (2.1) is added so that the assumption on the existence of an ellipsoidal constraint holds.

Let γ be any constant number satisfying

$$\gamma > \frac{\lambda_{\min}(W) + \lambda_{\max}(B)}{\lambda_{\min}^2(W)}.$$

For any

$$\epsilon \in \left(0, \frac{\gamma\lambda_{\min}^2(W) - \lambda_{\min}(W) - \lambda_{\max}(B)}{\gamma\lambda_{\min}(W)(1 + \lambda_{\min}(W))}\right),$$

applying Bienstock's algorithm to approximately solve (QCQP) (2.1) yields an ϵ-approximation solution in polynomial time, denoted by $(\widehat{x}, \widehat{s}, \widehat{t})$. Then it holds that

$$\widehat{s}\widehat{t} \le \widehat{x}^TB\widehat{x} + \epsilon,\ \widehat{x}^TW\widehat{x} \le \widehat{t} + \epsilon,\ 1 - \epsilon \le \widehat{x}^T\widehat{x} \le 1 + \epsilon, \tag{2.2}$$

$$\widehat{s} + \widehat{x}^TD\widehat{x} \ge v(\text{QCQP}) - \epsilon. \tag{2.3}$$

The definition of ϵ implies that $\epsilon < \lambda_{\min}(W)/(1 + \lambda_{\min}(W))$ and hence

$$\epsilon < (1 - \epsilon)\lambda_{\min}(W) \le (1 - \epsilon)\frac{\widehat{x}^TW\widehat{x}}{\|\widehat{x}\|^2} \le \widehat{x}^TW\widehat{x},$$

where the last inequality follows from the third inequality in (2.2). Then, according to the first two inequalities in (2.2) and the definitions of γ and ϵ, one can verify that

$$\widehat{s} \le \frac{\widehat{x}^TB\widehat{x} + \epsilon}{\widehat{x}^TW\widehat{x} - \epsilon} \le \frac{\widehat{x}^TB\widehat{x}}{\widehat{x}^TW\widehat{x}} + \gamma\epsilon. \tag{2.4}$$

It follows from (2.4) and (2.3) that

$$f(\widehat{x}) = \frac{\widehat{x}^TB\widehat{x}}{\widehat{x}^TW\widehat{x}} + \widehat{x}^TD\widehat{x} \ge \widehat{s} + \widehat{x}^TD\widehat{x} - \gamma\epsilon \ge v(\text{QCQP}) - (1 + \gamma)\epsilon = v(\text{P}) - (1 + \gamma)\epsilon.$$

It turns out that $\widehat{x}$ remains a $(1+\gamma)\epsilon$-approximation solution of (P). As a conclusion, we have the following result.

PROPOSITION 2.1. *Both* (QCQP) *and* (P) *can be approximately solved in polynomial time under the bit model of computing.*

To the best of our knowledge, the above polynomial-time algorithm does not work in practice.

In the following, we show that the diagonal case of (P) could be efficiently solved in polynomial time. Under the assumption that B, W, and D are all diagonal (or simultaneously diagonalized), (P) is rewritten as

$$(\mathrm{P_D}) \qquad \max_{\sum_{i=1}^n x_i^2=1} \ \frac{\sum_{i=1}^n b_i x_i^2}{\sum_{i=1}^n w_i x_i^2} + \sum_{i=1}^n d_i x_i^2.$$

Introducing $z_i := x_i^2$ for $i = 1, \ldots, n$, we get an equivalent reformulation of $(\mathrm{P_D})$:

$$(\mathrm{P_z}) \qquad \max \frac{\sum_{i=1}^n b_i z_i}{\sum_{i=1}^n w_i z_i} + \sum_{i=1}^n d_i z_i$$
$$\text{s.t. } z \in \Delta := \left\{ z : \ \sum_{i=1}^n z_i = 1, \ z_i \geq 0, \ i = 1, \ldots, n \right\}.$$

Let z^* be an optimal solution of $(\mathrm{P_z})$. Define $\alpha^* = \sum_{i=1}^n w_i z_i^*$. Then, $(\mathrm{P_z})$ shares the same optimal solution with the following linear program:

$$(\mathrm{LP}) \qquad \max \frac{\sum_{i=1}^n b_i z_i}{\alpha^*} + \sum_{i=1}^n d_i z_i$$

$$(2.5) \qquad \text{s.t. } \sum_{i=1}^n w_i z_i = \alpha^*, \ z \in \Delta.$$

Since (LP) attains its optimal value at an extreme point of (2.5) and there are at most two nonzero elements for each extreme point, we can globally solve $(\mathrm{P_z})$ by enumerating all vectors in the simplex Δ with at most two nonzero elements. Without loss of generality, let $(z_1, z_2, 0, \ldots, 0)^T \in \Delta$ be such a candidate. We have $z_2 = 1 - z_1$ and the optimal z_1 is obtained by solving the following univariate optimization:

$$(2.6) \qquad \max_{z_1 \in [0,1]} \ \left\{ g(z_1) = \frac{b_1 z_1 + b_2(1 - z_1)}{w_1 z_1 + w_2(1 - z_1)} + d_1 z_1 + d_2(1 - z_1) \right\}.$$

It is easy to verify that the first derivative $g'(z_1)$ has at most two zero points and the second derivative $g''(z_1)$ does not change its sign in $[0, 1]$. It follows that problem (2.6) actually has a closed-form optimal solution. In order to globally solve $(\mathrm{P_z})$, it is sufficient to examine all $\binom{n}{2} = \frac{1}{2}n(n-1)$ such 2-combinations. As a conclusion, we have the following result.

THEOREM 2.1. *Suppose B, W, and D are all diagonal. Then* (P) *can be solved in $O(n^2)$ time.*

3. The existing Lipschitz global optimization approach. By introducing an additional variable μ to replace the fractional term $\frac{x^T B x}{x^T W x}$, one obtains the following univariate maximizing reformulation of (P) [17, 24]:

$$(\mathrm{P}_\mu) \qquad \max_{\mu \in [\underline{\mu}, \overline{\mu}]} \ q(\mu) := \mu + g(\mu),$$

where the lower and upper bounds of μ, denoted by $\underline{\mu}$ and $\overline{\mu}$, are the smallest and the largest generalized eigenvalues of the matrix pair (B, W), respectively, and $g(\mu)$ is a

homogeneous QCQP given by

$$g(\mu) = \max_{x^T(B-\mu W)x \geq 0,\ x \in S} x^T D x. \tag{3.1}$$

Strong Lagrangian duality holds for (3.1) and hence it can be globally solved by a semidefinite programming approach [17, 24]. The optimal Lagrangian multiplier of the constraint $x^T(B-\mu W)x \geq 0$ is denoted by η.

A global optimization algorithm for solving (P_μ) was proposed by X. H. Wang, L. F. Wang, and Xia [24]. It iteratively subdivides the feasible region $[\underline{\mu}, \overline{\mu}]$ into $\bigcup_{i=1}^{k}[\mu_i, \mu_{i+1}]$ with $\mu_1 = \underline{\mu}$ and $\mu_{k+1} = \overline{\mu}$. In each interval $[\mu_i, \mu_{i+1}]$, a Lipschitz continuous overestimation of $q(\mu)$ is given as follows:

$$\forall \mu \in [\mu_i, \mu_{i+1}]:\ q(\mu) \leq \bar{q}(\mu) := \min\{q_1(\mu), q_2(\mu)\}, \tag{3.2}$$

where

$$\begin{aligned} q_1(\mu) &= q(\mu_i) + (1 - \eta_i \lambda_{\min}(W))(\mu - \mu_i), \\ q_2(\mu) &= q(\mu_{i+1}) + (1 - \eta_{i+1} \lambda_{\max}(W))(\mu - \mu_{i+1}), \end{aligned}$$

and η_i, η_{i+1} are the optimal Lagrangian multipliers for $x^T(B-\mu W)x \geq 0$ in (3.1) with $\mu = \mu_i, \mu_{i+1}$, respectively. An immediate corollary of (3.2) is that

$$\forall \mu \geq \mu_i:\ q(\mu) \leq q(\mu_i) + \mu - \mu_i.$$

Moreover, (3.2) provides the following Lipschitz upper bound for $q(\mu)$ over $[\mu_i, \mu_{i+1}]$:

$$U_i = \max_{\mu \in [\mu_i, \mu_{i+1}]} \bar{q}(\mu) = \begin{cases} q(\mu_i) & \text{if } 1 - \eta_i \lambda_{\min}(W) \leq 0, \\ q(\mu_{i+1}) & \text{if } 1 - \eta_{i+1} \lambda_{\max}(W) \geq 0, \\ q_1(\mu_0) & \text{otherwise,} \end{cases}$$

where

$$\mu_0 = \frac{q(\mu_{i+1}) - \mu_{i+1} + \eta_{i+1}\mu_{i+1}\lambda_{\max}(W) - q(\mu_i) + \mu_i - \eta_i \mu_i \lambda_{\min}(W)}{\eta_{i+1}\lambda_{\max}(W) - \eta_i \lambda_{\min}(W)}.$$

Based on the above easy-to-obtain Lipschitz bounds, X. H. Wang, L. F. Wang, and Xia [24] proposed an efficient branch-and-bound algorithm (without the pruning operator) for solving (P_μ). We refer to it as Algorithm L-BB throughout this paper. It is guaranteed to terminate in at most $\lceil \frac{\overline{\mu} - \underline{\mu}}{\epsilon} \rceil$ steps and return an ϵ-approximately optimal solution of (P_μ), i.e., a feasible solution μ^* satisfying

$$v(\mathrm{P}_\mu) \geq q(\mu^*) \geq v(\mathrm{P}_\mu) - \epsilon.$$

Numerical results demonstrated in [24] that Algorithm L-BB greatly outperforms the "two-stage" heuristic algorithm proposed in [17].

4. A new univariate reformulation. By introducing a new additional variable α to replace the denominator $x^T W x$ rather than the whole ratio $\frac{x^T B x}{x^T W x}$, we obtain an alternative univariate maximizing reformulation of (P):

$$\max_{\alpha \in [\lambda_1, \lambda_n]} \left\{ \mathcal{G}(\alpha) := \max_{x^T W x = \alpha,\ x \in S} \left\{ \frac{x^T B x}{\alpha} + x^T D x \right\} \right\}, \tag{P_α}$$

where $\lambda_1 = \lambda_{\min}(W)$ and $\lambda_n = \lambda_{\max}(W)$.

Evaluating $\mathcal{G}(\alpha)$ is a homogeneous quadratic programming problem with two homogeneous quadratic constraints (HTQP). The hidden convexity of (HTQP) is shown in [27], which also shows that the semidefinite programming relaxation is tight.

Notice that strong duality for $g(\mu)$ in (3.1) follows from the extended S-lemma with inequality; see Theorem 5.17 in [19]. Extending to the equality case $\mathcal{G}(\alpha)$, we can similarly establish the corresponding extended S-lemma with equality.

THEOREM 4.1. *Let $n \geq 3$ and A, B be $n \times n$ real symmetric matrices. For any $\lambda_{\min}(B) < \beta < \lambda_{\max}(B)$, the following two statements are equivalent:*

(I) $\{x \in \mathbb{R}^n : \ x^T A x < \alpha, \ x^T B x = \beta, \ \|x\| = 1\} = \emptyset,$

(II) $\exists y \in \mathbb{R} : \ A - \alpha I + y(B - \beta I) \succeq 0.$

The proof of Theorem 4.1 is very similar to that of Theorem 5.17 in [19] and is hence omitted.

For any $\alpha \in (\lambda_1, \lambda_n) = (\lambda_{\min}(W), \lambda_{\max}(W))$, we have

$$\begin{aligned}
\mathcal{G}(\alpha) &= \inf_{\gamma} \ \left\{ \gamma : \left\{ -x^T \left(\frac{B}{\alpha} + D \right) x < -\gamma, \ x^T W x = \alpha, \ x \in S \right\} = \emptyset \right\} \\
&= \inf_{\gamma, \nu_1} \ \left\{ \gamma : -\frac{B}{\alpha} - D + \gamma I + \nu_1 (W - \alpha I) \succeq 0 \right\} \qquad (4.1) \\
&= \inf_{\nu_1, \nu_2} \ \left\{ \alpha \nu_1 + \nu_2 : D + \frac{B}{\alpha} - \nu_1 W - \nu_2 I \preceq 0 \right\} \qquad (4.2) \\
&= \inf_{\nu_1} \ \left\{ \alpha \nu_1 + \lambda_{\max} \left(D + \frac{B}{\alpha} - \nu_1 W \right) := h(\nu_1) \right\}, \qquad (4.3)
\end{aligned}$$

where (4.1) follows from Theorem 4.1, and (4.2) inherits from (4.1) with a new variable ν_2 introduced to replace $\gamma - \alpha\nu_1$.

Let $\nu_1^* \in [-\infty, +\infty]$ be an optimal solution to (4.3). For any $\|u\| = 1$, it follows from the fact that $h(0) \geq h(\nu_1^*)$ that

$$\begin{aligned}
\lambda_{\max}\left(D + \frac{B}{\alpha} \right) &\geq \alpha \nu_1^* + \lambda_{\max}\left(D + \frac{B}{\alpha} - \nu_1^* W \right) \\
&\geq \alpha \nu_1^* + u^T \left(D + \frac{B}{\alpha} - \nu_1^* W \right) u \\
&= (\alpha - u^T W u) \nu_1^* + u^T \left(D + \frac{B}{\alpha} \right) u.
\end{aligned}$$

By choosing u as u_1, u_2, the unit eigenvectors corresponding to the maximal and minimal eigenvalues of W, respectively, we obtain lower and upper bounds of ν_1^*:

$$\nu_1^* \in [\underline{\nu}, \overline{\nu}] := \left[\frac{u_1^T (D + \frac{B}{\alpha}) u_1 - \lambda_{\max}(D + \frac{B}{\alpha})}{\lambda_{\max}(W) - \alpha}, \frac{\lambda_{\max}(D + \frac{B}{\alpha}) - u_2^T (D + \frac{B}{\alpha}) u_2}{\alpha - \lambda_{\min}(W)} \right]. \qquad (4.4)$$

Consequently, the minimum of (4.3) is attained. Now, we have seen that the calculation of $\mathcal{G}(\alpha)$ is reduced to minimizing $h(\nu_1)$ over $[\underline{\nu}, \overline{\nu}]$.

It is trivial to see that $h(\nu_1)$ is convex but nonsmooth. Let u_ϵ^* be an approximation of the unit eigenvector corresponding to the maximum eigenvalue of the matrix $D + \frac{B}{\alpha} - \nu_1 W$ satisfying $\|u_\epsilon^*\| = 1$ and

$$\lambda_{\max}\left(D + \frac{B}{\alpha} - \nu_1 W \right) \geq u_\epsilon^{*T} \left(D + \frac{B}{\alpha} - \nu_1 W \right) u_\epsilon^* \geq \lambda_{\max}\left(D + \frac{B}{\alpha} - \nu_1 W \right) - \frac{\epsilon}{2}.$$

Then we can verify that

$$g(\nu_1) := \alpha - u_\epsilon^{*T} W u_\epsilon^* \tag{4.5}$$

is a $\frac{\epsilon}{2}$-subgradient of $h(\nu_1)$ at ν_1 as

$$\begin{aligned}
&h(\widetilde{\nu}_1) - h(\nu_1) \\
&\geq \alpha(\widetilde{\nu}_1 - \nu_1) + \lambda_{\max}\left(D + \frac{B}{\alpha} - \widetilde{\nu}_1 W\right) - u_\epsilon^{*T}\left(D + \frac{B}{\alpha} - \nu_1 W\right) u_\epsilon^* - \frac{\epsilon}{2} \\
&\geq \alpha(\widetilde{\nu}_1 - \nu_1) + u_\epsilon^{*T}\left(D + \frac{B}{\alpha} - \widetilde{\nu}_1 W\right) u_\epsilon^* - u_\epsilon^{*T}\left(D + \frac{B}{\alpha} - \nu_1 W\right) u_\epsilon^* - \frac{\epsilon}{2} \\
&= g(\nu_1) \cdot (\widetilde{\nu}_1 - \nu_1) - \frac{\epsilon}{2}.
\end{aligned}$$

The above inequality also implies that

$$g(\nu_1) > 0 \implies h(\widetilde{\nu}_1) > h(\nu_1) - \frac{\epsilon}{2} \ \forall \widetilde{\nu}_1 > \nu_1, \tag{4.6}$$

$$g(\nu_1) < 0 \implies h(\widetilde{\nu}_1) > h(\nu_1) - \frac{\epsilon}{2} \ \forall \widetilde{\nu}_1 < \nu_1. \tag{4.7}$$

Therefore, as summarized in the following, the classical bisection search approach is employed to solve (4.3), where the initial interval is dynamically updated rather than the overestimated $[\underline{\nu}, \overline{\nu}]$ given in (4.4).

Bisection algorithm.
1. Input: $B, W, D \in \mathbb{R}^{n\times n}$, α, and $\epsilon > 0$ (tolerance parameter).
2. If $g(0) = 0$, $\nu_1 := 0$ and go to step 5.
   If $g(0) > 0$, set $\overline{\nu} = 0$ and $\underline{\nu} = \max\{-2^i : g(-2^i) < 0, i = 0, 1, 2, \ldots\}$.
   If $g(0) < 0$, set $\underline{\nu} = 0$ and $\overline{\nu} = \min\{2^i : g(2^i) > 0, i = 0, 1, 2, \ldots\}$.
   Set the iteration number $k = \log_2(\overline{\nu} - \underline{\nu})$.
3. Let $\nu_1 = \frac{1}{2}(\underline{\nu} + \overline{\nu})$ and $k := k + 1$.
   If $g(\nu_1) = 0$ or $\overline{\nu} - \underline{\nu} < \frac{\epsilon}{\lambda_n - \lambda_1}$, go to step 5.
4. If $g(\nu_1) > 0$, update $\overline{\nu} = \nu_1$. If $g(\nu_1) < 0$, update $\underline{\nu} = \nu_1$. Go to step 3.
5. Output ν_1: an ϵ-approximately optimal solution of (4.3).

THEOREM 4.2. *Let $\epsilon > 0$ and $\epsilon_0 > 0$ be two small enough tolerances. For any $\alpha \in (\lambda_1 + \epsilon_0, \lambda_n - \epsilon_0)$, the bisection algorithm finds an ϵ-approximately optimal solution of (4.3), i.e., a solution ν_1 satisfying $h(\nu_1) \leq \mathcal{G}(\alpha) + \epsilon$, after at most*

$$\lceil 4\log_2(\epsilon_0^{-1}) + \log_2(\epsilon^{-1}) + c \rceil$$

iterations, where $c = 2\log_2(\epsilon_0\|D\| + \|B\|) + \log_2(\lambda_n - \lambda_1) + 5$.

Proof. The bisection algorithm returns an approximation solution

$$\nu_1^* = \frac{1}{2}(\underline{\nu}^* + \overline{\nu}^*),$$

where $[\underline{\nu}^*, \overline{\nu}^*]$ is the last interval before termination. It holds that $g(\underline{\nu}^*) < 0$, $g(\overline{\nu}^*) > 0$, and $\overline{\nu}^* - \underline{\nu}^* < \frac{\epsilon}{\lambda_n - \lambda_1}$. Therefore, it follows from (4.6) and (4.7) that

$$\begin{aligned}
h(\widetilde{\nu}_1) &> h(\underline{\nu}^*) - \frac{\epsilon}{2} \ \forall \widetilde{\nu}_1 < \underline{\nu}^*, \\
h(\widehat{\nu}_1) &> h(\overline{\nu}^*) - \frac{\epsilon}{2} \ \forall \widehat{\nu}_1 > \overline{\nu}^*.
\end{aligned}$$

Or, equivalently, we obtain

$$h(\underline{\nu}^*) < \min_{\nu_1<\underline{\nu}^*} h(\nu_1) + \frac{\epsilon}{2}, \quad h(\overline{\nu}^*) < \min_{\nu_1>\overline{\nu}^*} h(\nu_1) + \frac{\epsilon}{2}. \tag{4.8}$$

On the other hand, for any point $\nu_1 \in [\underline{\nu}^*, \overline{\nu}^*]$, we have

$$\begin{aligned}
h(\nu_1^*) - h(\nu_1) &= \alpha(\nu_1^* - \nu) + \lambda_{\max}\left(D + \frac{B}{\alpha} - \nu_1^* W\right) - \lambda_{\max}\left(D + \frac{B}{\alpha} - \nu_1 W\right) \\
&\le \alpha(\nu_1^* - \nu_1) + u^T\left(D + \frac{B}{\alpha} - \nu_1^* W\right)u - u^T\left(D + \frac{B}{\alpha} - \nu_1 W\right)u \\
&= (\alpha - u^T W u)(\nu_1^* - \nu_1) \\
&\le \max\{\alpha - u^T W u, u^T W u - \alpha\} \cdot |\nu_1^* - \nu_1| \\
&\le (\lambda_n - \lambda_1) \cdot |\nu_1^* - \nu_1| \qquad (4.9) \\
&\le (\lambda_n - \lambda_1) \cdot \frac{1}{2}(\overline{\nu}^* - \underline{\nu}^*) \\
&\le \frac{\epsilon}{2}, \qquad (4.10)
\end{aligned}$$

where u is a unit eigenvector corresponding to the maximal eigenvalue of $D + \frac{B}{\alpha} - \nu_1^* W$ so that the first inequality holds, and (4.9) follows from the fact that $\lambda_1 < \alpha < \lambda_n$ and $\lambda_1 \le u^T W u \le \lambda_n$.

The above inequalities (4.8) and (4.10) imply that

$$\begin{aligned}
h(\nu_1^*) &\le \min\left\{\min_{\nu_1\in[\underline{\nu}^*,\overline{\nu}^*]} h(\nu_1), h(\underline{\nu}^*), h(\overline{\nu}^*)\right\} + \frac{\epsilon}{2} \\
&\le \min\left\{\min_{\nu_1\in[\underline{\nu}^*,\overline{\nu}^*]} h(\nu_1), \min_{\nu_1<\underline{\nu}^*} h(\nu_1) + \frac{\epsilon}{2}, \min_{\nu_1>\overline{\nu}^*} h(\nu_1) + \frac{\epsilon}{2}\right\} + \frac{\epsilon}{2} \\
&\le \min_{\nu_1} h(\nu_1) + \epsilon \\
&= \mathcal{G}(\alpha) + \epsilon.
\end{aligned}$$

Thus, ν_1^* is an ϵ-approximately optimal solution of (4.3).

In the following, it is sufficient to overestimate the number of iterations in the worst case.

Let $[\underline{\nu}, \overline{\nu}]$ be defined in (4.4), which covers the dynamic initial interval generated by the above bisection algorithm. Therefore, the number of the iterations in step 2, denoted by K_1, is at most $\lceil \log_2(\overline{\nu} - \underline{\nu}) \rceil$.

The worst case in step 3 is that, after K_2 iterations, the bisection algorithm terminates as the length of the interval (i.e., $\frac{\overline{\nu}-\underline{\nu}}{2^K}$) is less than or equal to $\frac{\epsilon}{\lambda_n - \lambda_1}$. That is, it holds that $K_2 = \lceil \log_2\left(\frac{\overline{\nu}-\underline{\nu}}{\epsilon}\right) + \log_2(\lambda_n - \lambda_1) \rceil$.

According to (4.4), we have the following overestimation:

$$\begin{aligned}
\overline{\nu} - \underline{\nu} &= \frac{\lambda_{\max}(D + \frac{B}{\alpha}) - u_1^T(D + \frac{B}{\alpha})u_1}{\lambda_{\max}(W) - \alpha} + \frac{\lambda_{\max}(D + \frac{B}{\alpha}) - u_2^T(D + \frac{B}{\alpha})u_2}{\alpha - \lambda_{\min}(W)} \\
&\le \frac{2}{\epsilon_0}\left(\lambda_{\max}\left(D + \frac{B}{\alpha}\right) - \lambda_{\min}\left(D + \frac{B}{\alpha}\right)\right) \le \frac{4}{\epsilon_0}\left\|D + \frac{B}{\alpha}\right\| \\
&\le \frac{4}{\epsilon_0}\left(\|D\| + \left\|\frac{B}{\alpha}\right\|\right) < \frac{4}{\epsilon_0}\left(\|D\| + \frac{1}{\epsilon_0}\|B\|\right) = \frac{4}{\epsilon_0^2}\left(\epsilon_0\|D\| + \|B\|\right). \qquad (4.11)
\end{aligned}$$

Substituting (4.11) into $K_1 + K_2$ yields the total number of iterations in the worst case. The proof is complete. ∎

Note that (4.2) is the Lagrangian dual problem of the QCQP $\mathcal{G}(\alpha)$ with no duality gap. Let ν_1^* be the optimal solution of (4.3). Define $\nu_2^* = \lambda_{\max}(D + \frac{B}{\alpha} - \nu_1^* W)$. Then ν_1^* and ν_2^* are the optimal Lagrangian multipliers corresponding to the constraints $x^T W x = \alpha$ and $x^T x = 1$, respectively. The optimal solution x^* of the QCQP $\mathcal{G}(\alpha)$ remains optimal for the Lagrangian function:

$$\mathcal{G}(\alpha) = \max_{x \in S} \left\{ \frac{x^T B x}{\alpha} + x^T D x - \nu_1^* (x^T W x - \alpha) \right\}.$$

Let $\{u_1, \ldots, u_k\}$ be an orthogonal basis of the eigenspace corresponding to the maximum eigenvalue of $D + \frac{B}{\alpha} - \nu_1^* W$. Define $U = [u_1, \ldots, u_k]$. For any $v \in \mathbb{R}^k$ such that $\|v\| = 1$, we have $\|Uv\| = \|v\| = 1$. Therefore, $x^* := Uv$ is an optimal solution of the QCQP $\mathcal{G}(\alpha)$ for any $v \in \mathbb{R}^k$ satisfying $v^T U^T W U v = \alpha$ and $\|v\| = 1$. The latter quadratic system of v is easy to solve based on the spectral decomposition of $U^T W U$.

5. A new algorithm based on a novel bound.

5.1. A novel overestimation approach. As shown in the above section, we can efficiently evaluate $\mathcal{G}(\alpha)$ for any given α. However, there seems to be no explicit expression of $\mathcal{G}(\alpha)$. Our goal in this subsection is to provide an easy-to-compute and high-quality overestimation of $\mathcal{G}(\alpha)$ over any given interval $[\alpha_i, \alpha_{i+1}]$ with known solution information at the two endpoints.

For each α, let $\nu_1(\alpha)$ be an $\frac{\epsilon}{2}$-approximately optimal solution of (4.3) returned by the above bisection algorithm, i.e., $h(\nu_1(\alpha)) \le \mathcal{G}(\alpha) + \frac{\epsilon}{2}$.

Let u_ϵ^* ($\|u_\epsilon^*\| = 1$) be an approximation of the unit eigenvector corresponding to the maximum eigenvalue of $D + \frac{B}{\alpha} - \nu_1(\alpha) W$ satisfying

$$u_\epsilon^{*T} \left(D + \frac{B}{\alpha} - \nu_1(\alpha) W \right) u_\epsilon^* \ge \lambda_{\max} \left(D + \frac{B}{\alpha} - \nu_1(\alpha) W \right) - \frac{\epsilon}{2}.$$

Define

$$\mathcal{G}^\epsilon(\alpha) := \alpha \nu_1(\alpha) + u_\epsilon^{*T} \left(D + \frac{B}{\alpha} - \nu_1(\alpha) W \right) u_\epsilon^*. \tag{5.1}$$

It holds that

$$\mathcal{G}^\epsilon(\alpha) + \frac{\epsilon}{2} \ge h(\nu_1(\alpha)). \tag{5.2}$$

Moreover, one can trivially verify that

$$\mathcal{G}(\alpha) - \frac{\epsilon}{2} \le \mathcal{G}^\epsilon(\alpha) \le \mathcal{G}(\alpha) + \frac{\epsilon}{2}. \tag{5.3}$$

Following (5.2), we have

$$\begin{aligned} \mathcal{G}^\epsilon(\alpha_i) + \frac{\epsilon}{2} \ge h(\nu_1(\alpha_i)) &= \alpha_i \nu_1(\alpha_i) + \lambda_{\max} \left(D + \frac{B}{\alpha_i} - \nu_1(\alpha_i) W \right) \\ &\ge \alpha_i \nu_1(\alpha_i) + x^T \left(D + \frac{B}{\alpha_i} - \nu_1(\alpha_i) W \right) x \ \forall x \in S. \end{aligned} \tag{5.4}$$

Similarly, we obtain

$$\mathcal{G}^\epsilon(\alpha_{i+1}) + \frac{\epsilon}{2} \ge \alpha_{i+1} \nu_1(\alpha_{i+1}) + x^T \left(D + \frac{B}{\alpha_{i+1}} - \nu_1(\alpha_{i+1}) W \right) x \ \forall x \in S. \tag{5.5}$$

For any $\alpha \in [\alpha_i, \alpha_{i+1}]$, according to (4.3), it holds that

$$\begin{aligned}\mathcal{G}(\alpha) &= \min_{\nu_1}\left\{\alpha\nu_1 + \max_{x\in S}\ x^T\left(D + \frac{B}{\alpha} - \nu_1 W\right)x\right\}\\ &\le \min_{\nu_1}\left\{\alpha\nu_1 + \max_{x \text{ satisfies (5.4), (5.5)}} x^T\left(D + \frac{B}{\alpha} - \nu_1 W\right)x\right\}\\ &\le \min_{\nu_1}\left\{\alpha\nu_1 + \max_{(y^B,y^D,y^W)\in Y\subseteq\mathbb{R}^3}\ y^D + \frac{y^B}{\alpha} - \nu_1 y^W\right\}, \end{aligned} \tag{5.6}$$

where y^B, y^D, and y^W are introduced to replace x^TBx, x^TDx, and x^TWx, respectively, so that following (5.4), (5.5), (y^B, y^D, y^W) lies in the following set:

$$\begin{aligned}Y = \Bigg\{(y^B, y^D, y^W) \in \mathbb{R}^3 :\ &\mathcal{G}^\epsilon(\alpha_i) + \frac{\epsilon}{2} \ge \frac{y^B}{\alpha_i} + y^D - \nu_1(\alpha_i)y^W + \alpha_i\nu_1(\alpha_i),\\ &\mathcal{G}^\epsilon(\alpha_{i+1}) + \frac{\epsilon}{2} \ge \frac{y^B}{\alpha_{i+1}} + y^D - \nu_1(\alpha_{i+1})y^W + \alpha_{i+1}\nu_1(\alpha_{i+1})\Bigg\}.\end{aligned}$$

The inner maximizing problem of (5.6) in terms of (y^B, y^D, y^W) is a linear programming problem which enjoys strong Lagrangian duality, and hence is equivalent to its dual minimization problem. Consequently, the problem (5.6) can be recast as

$$\min_{\nu_1\in\mathbb{R}}\min_{\mu_1,\mu_2}\ \mu_1(\mathcal{G}^\epsilon(\alpha_i) - \alpha_i\nu_1(\alpha_i)) + \mu_2(\mathcal{G}^\epsilon(\alpha_{i+1}) - \alpha_{i+1}\nu_1(\alpha_{i+1})) + \alpha\nu_1 + \frac{\epsilon}{2} \tag{5.7}$$

$$\text{s.t. } \frac{1}{\alpha_i}\mu_1 + \frac{1}{\alpha_{i+1}}\mu_2 = \frac{1}{\alpha}, \tag{5.8}$$

$$\mu_1 + \mu_2 = 1, \tag{5.9}$$

$$\nu_1(\alpha_i)\mu_1 + \nu_1(\alpha_{i+1})\mu_2 = \nu_1, \tag{5.10}$$

$$\mu_1, \mu_2 \ge 0. \tag{5.11}$$

Equations (5.8) and (5.9) provide a unique solution of (μ_1, μ_2):

$$\mu_1 = \frac{\alpha_i(\alpha_{i+1} - \alpha)}{\alpha(\alpha_{i+1} - \alpha_i)}, \quad \mu_2 = \frac{\alpha_{i+1}(\alpha - \alpha_i)}{\alpha(\alpha_{i+1} - \alpha_i)}. \tag{5.12}$$

Since $\alpha_i \le \alpha \le \alpha_{i+1}$, the nonnegative constraints (5.11) are satisfied for the solution (5.12). Moreover, substituting the solution (5.12) into (5.10) yields

$$\nu_1 = \frac{\alpha_i\alpha_{i+1}}{\alpha_{i+1} - \alpha_i}(\nu_1(\alpha_i) - \nu_1(\alpha_{i+1}))\frac{1}{\alpha} + \frac{1}{\alpha_{i+1} - \alpha_i}(\nu_1(\alpha_{i+1})\alpha_{i+1} - \nu_1(\alpha_i)\alpha_i). \tag{5.13}$$

That is, the optimization problem (5.7)–(5.11) has a unique feasible solution, which is clearly the explicit optimal solution. Plugging (5.12) and (5.13) in (5.7) provides an overestimation for $\mathcal{G}(\alpha)$:

$$\mathcal{G}(\alpha) \le \overline{\mathcal{G}}(\alpha) := c_1\alpha + \frac{c_2}{\alpha} + c_3\ \forall\alpha \in [\alpha_i, \alpha_{i+1}], \tag{5.14}$$

where the constant coefficients are calculated via

$$c_1 = \frac{\alpha_{i+1}\nu_1(\alpha_{i+1}) - \alpha_i\nu_1(\alpha_i)}{\alpha_{i+1} - \alpha_i}, \tag{5.15}$$

$$c_2 = \alpha_i\alpha_{i+1}\left(c_1 - \frac{\mathcal{G}^\epsilon(\alpha_{i+1}) - \mathcal{G}^\epsilon(\alpha_i)}{\alpha_{i+1} - \alpha_i}\right), \tag{5.16}$$

$$c_3 = \frac{\alpha_{i+1}\mathcal{G}^\epsilon(\alpha_{i+1}) - \alpha_i\mathcal{G}^\epsilon(\alpha_i)}{\alpha_{i+1} - \alpha_i} - c_1(\alpha_{i+1} + \alpha_i) + \frac{\epsilon}{2}. \tag{5.17}$$

It is interesting to see that the overestimating function $\overline{\mathcal{G}}(\alpha)$ is tight with a tolerance $\frac{\epsilon}{2}$ at the two endpoints as one can verify from (5.3) that

$$\overline{\mathcal{G}}(\alpha_{i+1}) = \mathcal{G}^{\epsilon}(\alpha_{i+1}) + \frac{\epsilon}{2} \leq \mathcal{G}(\alpha_{i+1}) + \epsilon,\ \overline{\mathcal{G}}(\alpha_i) = \mathcal{G}^{\epsilon}(\alpha_i) + \frac{\epsilon}{2} \leq \mathcal{G}(\alpha_i) + \epsilon. \tag{5.18}$$

In conclusion, we obtain the following upper bound for $\mathcal{G}(\alpha)$ over $[\alpha_i, \alpha_{i+1}]$.

THEOREM 5.1. *Let c_1, c_2, and c_3 be defined as in* (5.15)–(5.17). *If*

$$c_1 < 0,\ c_2 < 0,\ \widetilde{\alpha} := \sqrt{\frac{c_2}{c_1}} \in (\alpha_i, \alpha_{i+1}), \tag{5.19}$$

then it holds that

$$\max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \mathcal{G}(\alpha) \leq \max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \overline{\mathcal{G}}(\alpha) = 2\sqrt{c_1 c_2} + c_3.$$

Otherwise, we have

$$\max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \mathcal{G}(\alpha) \leq \max\{\mathcal{G}(\alpha_i), \mathcal{G}(\alpha_{i+1})\} + \epsilon. \tag{5.20}$$

Proof. According to definition (5.14), under assumption (5.19), $\overline{\mathcal{G}}(\alpha)$ is strictly concave and $\widetilde{\alpha}$ is the unique maximizer of $\overline{\mathcal{G}}(\alpha)$. On the other hand, if (5.19) fails to hold, then $\overline{\mathcal{G}}(\alpha)$ is monotonic over $[\alpha_i, \alpha_{i+1}]$. Therefore, we obtain

$$\begin{aligned} \max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \mathcal{G}(\alpha) &\leq \max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \overline{\mathcal{G}}(\alpha) \\ &= \max\left\{\overline{\mathcal{G}}(\alpha_i), \overline{\mathcal{G}}(\alpha_{i+1})\right\} \qquad (5.21) \\ &\leq \max\{\mathcal{G}(\alpha_i), \mathcal{G}(\alpha_{i+1})\} + \epsilon, \qquad (5.22) \end{aligned}$$

where (5.21) holds due to the monotonicity of $\overline{\mathcal{G}}(\alpha)$ and (5.22) follows from (5.18). The proof of (5.20) is complete. ∎

5.2. A new branch-and-bound algorithm. In this subsection, we propose a new branch-and-bound algorithm for solving (P_α) based on the above novel upper bounding approach. Our goal is to find an ϵ-approximately optimal solution of (P_α), i.e., a feasible solution $\alpha^* \in [\lambda_1, \lambda_n]$ satisfying

$$v(\mathrm{P}_\alpha) \geq \mathcal{G}(\alpha^*) \geq v(\mathrm{P}_\alpha) - \epsilon.$$

Notice that our bounding approach based on the reformulation (4.3) works only when α is in the interior of $[\lambda_1, \lambda_n]$, i.e., $\alpha \in (\lambda_1, \lambda_n)$. It is necessary to carefully study $\mathcal{G}(\alpha)$ in the neighbourhood of λ_1 or λ_n.

For convenience, in the analysis we make the following assumption on W in the original problem (P). Actually, we show in the appendix that we can construct a perturbation of (P) where the assumption is *always* satisfied. Moreover, the perturbation not only has the same computational complexity as (P) but also provides an ϵ-approximately optimal solution of the original (P).

Assumption 5.1. Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{n-1} \leq \lambda_n$ be the n eigenvalues of the matrix W. There is an $\epsilon_1 > 0$ such that $\lambda_1 + \epsilon_1 \leq \lambda_2$ and $\lambda_{n-1} + \epsilon_1 \leq \lambda_n$.

LEMMA 5.1. *The objective function of* (P), *$f(x)$, is Lipschitz continuous over S:*

$$|f(x) - f(y)| \leq L\|x - y\|\ \forall x, y \in S, \tag{5.23}$$

where the Lipschitz constant L is given by

$$L \geq \frac{2\max\{|\lambda_{\max}(B)|, |\lambda_{\min}(B)|\}(\lambda_1 + \lambda_n)}{\lambda_1^2} + 2\max\{|\lambda_{\max}(D)|, |\lambda_{\min}(D)|\}. \tag{5.24}$$

Proof. The Lipschitz continuity of $f(x)$ over S follows from the compactness of S. The Lipschitz constant L is an overestimation of the norm of the derivative of $f(x)$, which reads

$$f'(x) = \frac{2Bx}{x^T W x} - \frac{2(x^T B x)Wx}{(x^T W x)^2} + 2Dx.$$

Then we have

$$\max_{x \in S} \|f'(x)\| \leq \frac{2\|B\|}{\lambda_1} + \frac{2\|B\|\lambda_n}{\lambda_1^2} + 2\|D\|,$$

where the matrix norm $\|\cdot\|$ is the spectral norm. For a symmetric matrix M, one can verify that $\|M\| = \max\{|\lambda_{\max}(M)|, |\lambda_{\min}(M)|\}$. The proof is complete. ∎

LEMMA 5.2. *Suppose Assumption* 5.1 *holds with an $\epsilon_1 > 0$ and L is defined in* (5.24). *For any $\epsilon \in (0, L)$, define*

$$\epsilon_2 = \frac{\epsilon_1 \epsilon^2}{2L^2}. \tag{5.25}$$

Then, for any $x \in S$ satisfying

$$x^T W x \leq \lambda_1 + \epsilon_2, \tag{5.26}$$

it holds that

$$|f(x) - \mathcal{G}(\lambda_1)| < L\sqrt{\frac{2\epsilon_2}{\epsilon_1}} = \epsilon \ \forall x \in S \text{ and } (5.26). \tag{5.27}$$

Proof. The definition of ϵ_2 implies that $\epsilon_2 \in (0, \epsilon_1)$. For any $x \in S$ satisfying (5.26), there exists $z \in \mathbb{R}^n$ such that $x = \sum_{i=1}^n z_i u_i$, $\sum_{i=1}^n z_i^2 = 1$, and so

$$\lambda_1 + \epsilon_2 \geq x^T W x = \sum_{i=1}^n z_i^2 \lambda_i \geq z_1^2 \lambda_1 + \lambda_2 \sum_{i=2}^n z_i^2 = z_1^2 \lambda_1 + (1 - z_1^2)\lambda_2.$$

This implies that

$$z_1^2 \geq \frac{\lambda_2 - \lambda_1 - \epsilon_2}{\lambda_2 - \lambda_1} = 1 - \frac{\epsilon_2}{\lambda_2 - \lambda_1} \geq 1 - \frac{\epsilon_2}{\epsilon_1}, \tag{5.28}$$

where Assumption 5.1 is used in the last inequality of (5.28). Therefore, it follows from (5.28) that

$$z_1 > 0 \implies \|x - u_1\|^2 = (z_1 - 1)^2 + \sum_{i=2}^n z_i^2 \leq \left(\frac{\epsilon_2}{\epsilon_1}\right)^2 + \frac{\epsilon_2}{\epsilon_1} < \frac{2\epsilon_2}{\epsilon_1},$$

$$z_1 < 0 \implies \|x - (-u_1)\|^2 = (z_1 + 1)^2 + \sum_{i=2}^n z_i^2 \leq \left(\frac{\epsilon_2}{\epsilon_1}\right)^2 + \frac{\epsilon_2}{\epsilon_1} < \frac{2\epsilon_2}{\epsilon_1}.$$

Since

$$\mathcal{G}(\lambda_1) = f(u_1) = f(-u_1),$$

it follows from (5.23) in Lemma 5.1 that (5.27) holds. ∎

THEOREM 5.2. *Suppose Assumption* 5.1 *holds with an* $\epsilon_1 > 0$. *For any* $\epsilon \in (0, L)$ *with* L *and* ϵ_2 *being defined as in* (5.24) *and* (5.25), *respectively, it holds that either*

$$\max_{\alpha\in[\lambda_1+\epsilon_2,\lambda_n-\epsilon_2]} \mathcal{G}(\alpha) = v(\mathrm{P}_\alpha) \tag{5.29}$$

or

$$\max\{\mathcal{G}(\lambda_1), \mathcal{G}(\lambda_n)\} \geq v(\mathrm{P}_\alpha) - \epsilon.$$

Proof. According to Lemma 5.2, we have

$$\max_{\alpha\in[\lambda_1,\lambda_1+\epsilon_2]} \mathcal{G}(\alpha) \leq \mathcal{G}(\lambda_1) + \epsilon. \tag{5.30}$$

Similarly, we can show that

$$\max_{\alpha\in[\lambda_n-\epsilon_2,\lambda_n]} \mathcal{G}(\alpha) \leq \mathcal{G}(\lambda_n) + \epsilon. \tag{5.31}$$

Suppose (5.29) does not hold true. Then

$$v(\mathrm{P}_\alpha) = \max\left\{\max_{\alpha\in[\lambda_1,\lambda_1+\epsilon_2]} \mathcal{G}(\alpha), \max_{\alpha\in[\lambda_n-\epsilon_2,\lambda_n]} \mathcal{G}(\alpha)\right\} \leq \max\{\mathcal{G}(\lambda_1), \mathcal{G}(\lambda_n)\} + \epsilon,$$

where the inequality follows from (5.30) and (5.31). The proof is complete. ∎

Our branch-and-bound algorithm works on a subdivision list

$$\lambda_1^\epsilon + \frac{3\epsilon_2}{4} = \alpha_1 < \cdots < \alpha_{k+1} = \lambda_n^\epsilon - \frac{3\epsilon_2}{4}, \tag{5.32}$$

where λ_1^ϵ and λ_n^ϵ are the approximations of λ_1 and λ_n, respectively, and they are required to satisfy

$$\lambda_1 \leq \lambda_1^\epsilon = u_1^{\epsilon T} W u_1^\epsilon \leq \lambda_1 + \frac{\epsilon_2}{4}, \tag{5.33}$$

$$\lambda_n - \frac{\epsilon_2}{4} \leq \lambda_n^\epsilon = u_n^{\epsilon T} W u_n^\epsilon \leq \lambda_n. \tag{5.34}$$

Then, according to (5.27) in Lemma 5.2, it holds that

$$\mathcal{G}^\epsilon(\lambda_1^\epsilon) := u_1^{\epsilon T} B u_1^\epsilon / \lambda_1^\epsilon + u_1^{\epsilon T} D u_1^\epsilon = f(u_1^\epsilon) \in \left(\mathcal{G}(\lambda_1) - \frac{\epsilon}{2}, \mathcal{G}(\lambda_1) + \frac{\epsilon}{2}\right). \tag{5.35}$$

Similarly, we have

$$\mathcal{G}^\epsilon(\lambda_n^\epsilon) := u_n^{\epsilon T} B u_n^\epsilon / \lambda_n^\epsilon + u_n^{\epsilon T} D u_n^\epsilon = f(u_n^\epsilon) \in \left(\mathcal{G}(\lambda_n) - \frac{\epsilon}{2}, \mathcal{G}(\lambda_n) + \frac{\epsilon}{2}\right). \tag{5.36}$$

In each iteration, we first select the interval $[\alpha_i, \alpha_{i+1}]$ from the $\{\alpha\}$-list that provides the maximal upper bound. Then we insert $\widetilde{\alpha}$ from (5.19), the maximizer of the overestimating function, into the $\{\alpha\}$-list (5.32) and increase k by one. Repeat

the process until a stopping criterion is reached. The detailed algorithm is presented as follows.

New branch-and-bound algorithm.

1. Input the tolerance $\epsilon > 0$. Calculate ϵ_2 (5.25).
   Compute the approximate eigenvalues $\lambda_1^\epsilon, \lambda_n^\epsilon$ and the approximate unit eigenvectors u_1^ϵ and u_n^ϵ satisfying (5.33) and (5.34), respectively.
   Calculate $\mathcal{G}^\epsilon(\lambda_1^\epsilon)$ and $\mathcal{G}^\epsilon(\lambda_n^\epsilon)$ as in (5.35) and (5.36).
   Initialize $LB = \mathcal{G}^\epsilon(\lambda_1^\epsilon) + \frac{\epsilon}{2}$ and $\alpha^* = \lambda_1^\epsilon$.
   If $LB < \mathcal{G}^\epsilon(\lambda_n^\epsilon) + \frac{\epsilon}{2}$, update $LB = \mathcal{G}^\epsilon(\lambda_n^\epsilon) + \frac{\epsilon}{2}$ and $\alpha^* = \lambda_n^\epsilon$.
   Set $\alpha_1 = \lambda_1^\epsilon + \frac{3\epsilon_2}{4}$, $\alpha_2 = \lambda_n^\epsilon - \frac{3\epsilon_2}{4}$, and $T = \emptyset$.
   Initialize the iteration (or functional evaluation) number $k = 2$.
   For $i = 1, 2$ compute $\mathcal{G}^\epsilon(\alpha_i)$ (5.1) and the approximation solution $\nu_1(\alpha_i)$.
   If $\mathcal{G}^\epsilon(\alpha_1) + \frac{\epsilon}{2} > LB$, update $LB = \mathcal{G}^\epsilon(\alpha_1) + \frac{\epsilon}{2}$ and $\alpha^* = \alpha_1$.
   If $\mathcal{G}^\epsilon(\alpha_2) + \frac{\epsilon}{2} > LB$, update $LB = \mathcal{G}^\epsilon(\alpha_2) + \frac{\epsilon}{2}$ and $\alpha^* = \alpha_2$.
2. Calculate $\tilde{\alpha}$ (5.19). Compute $\mathcal{G}^\epsilon(\tilde{\alpha})$ and the approximation solution $\nu_1(\tilde{\alpha})$.
   If $\mathcal{G}^\epsilon(\tilde{\alpha}) + \frac{\epsilon}{2} > LB$, update $LB = \mathcal{G}^\epsilon(\tilde{\alpha}) + \frac{\epsilon}{2}$ and $\alpha^* = \tilde{\alpha}$.
3. According to Theorem 5.1, compute the upper bounds:
   $UB_1 = \max_{\alpha \in [\alpha_1, \tilde{\alpha}]} \overline{\mathcal{G}}(\alpha)$, $UB_2 = \max_{\alpha \in [\tilde{\alpha}, \alpha_2]} \overline{\mathcal{G}}(\alpha)$.
   Update $T = T \cup \{(UB_1, \alpha_1, \tilde{\alpha})\} \cup \{(UB_2, \tilde{\alpha}, \alpha_2)\}$ and $k = k + 1$.
4. Find $(UB^*, \alpha_1, \alpha_2) = \arg\max_{(t,*,*) \in T} t$. If $UB^* \le LB + \epsilon$, stop and return α^* as an approximately optimal solution.
   Otherwise, update $T = T \setminus \{(UB^*, \alpha_1, \alpha_2)\}$ and go to step 2.

5.3. Complexity analysis. In this subsection, we analyze the complexity of the new branch-and-bound algorithm.

THEOREM 5.3. *The new branch-and-bound algorithm finds a global ϵ-approximation solution of* (P_α) *in at most*

$$\left\lceil \frac{32U\lambda_n^2(\lambda_n - \lambda_1)}{3\lambda_1^2\ \epsilon_2\epsilon} \right\rceil \tag{5.37}$$

iterations, where

$$U = \lambda_{\max}\left(D - \lambda_{\min}(D)\cdot I + \frac{B - \lambda_{\min}(B)\cdot I}{\lambda_1}\right) + |\lambda_{\min}(D)| + \frac{|\lambda_{\min}(B)|}{\lambda_1}. \tag{5.38}$$

Proof. According to Theorem 5.2 and relations (5.33) and (5.34), it is sufficient to estimate the number of iterations for approximately solving

$$\max_{\alpha \in [\lambda_1^\epsilon + \frac{3\epsilon_2}{4}, \lambda_n^\epsilon - \frac{3\epsilon_2}{4}]} \mathcal{G}(\alpha).$$

Suppose $(UB, \alpha_i, \alpha_{i+1}) \in T$ is selected to subdivide in the current iteration of our new algorithm. Then we have $UB^* = \max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \overline{\mathcal{G}}(\alpha)$. Without loss of generality, we assume that (5.19) holds in the interval $[\alpha_i, \alpha_{i+1}]$, since otherwise it follows from Theorem 5.1 that $LB \ge UB^* - \epsilon$, and hence the algorithm terminates.

The condition (5.19) implies that the overestimating function $\overline{\mathcal{G}}(\alpha)$ (5.14) is concave. Therefore, for any $\alpha \in [\alpha_i, \alpha_{i+1}]$, we have

$$\begin{aligned}\overline{\mathcal{G}}(\alpha) &\le \overline{\mathcal{G}}(\alpha_i) + \overline{\mathcal{G}}'(\alpha_i)(\alpha - \alpha_i)\\ &\le \mathcal{G}^{\epsilon}(\alpha_i) + \frac{\epsilon}{2} + \left(c_1 - \frac{c_2}{\alpha_i^2}\right)(\alpha - \alpha_i)\\ &\le \mathcal{G}^{\epsilon}(\alpha_i) + \frac{\epsilon}{2} - \frac{\alpha_{i+1}^2 - \alpha_i^2}{\alpha_i^2} c_1(\alpha_{i+1} - \alpha_i), \qquad (5.39)\end{aligned}$$

where the last two inequalities follow from (5.18) and (5.19), respectively.

Following (5.33) and (5.34), we obtain

$$\left[\lambda_1^{\epsilon} + \frac{3\epsilon_2}{4}, \lambda_n^{\epsilon} - \frac{3\epsilon_2}{4}\right] \subseteq \left[\lambda_1 + \frac{3\epsilon_2}{4}, \lambda_n - \frac{3\epsilon_2}{4}\right].$$

Then, for all $\alpha \in [\lambda_1^{\epsilon} + \frac{3\epsilon_2}{4}, \lambda_n^{\epsilon} - \frac{3\epsilon_2}{4}]$, according to (4.4), we have

$$\begin{aligned}|\nu_1(\alpha)| &\le \max\left\{\left|\frac{u_1^T(D + \frac{B}{\alpha})u_1 - \lambda_{\max}(D + \frac{B}{\alpha})}{\lambda_{\max}(W) - \alpha}\right|, \left|\frac{\lambda_{\max}(D + \frac{B}{\alpha}) - u_2^T(D + \frac{B}{\alpha})u_2}{\alpha - \lambda_{\min}(W)}\right|\right\}\\ &\le \frac{4}{3} \cdot \frac{\max_{u \in S}\left|u^T(D + \frac{B}{\alpha})u\right| + \left|\lambda_{\max}(D + \frac{B}{\alpha})\right|}{\epsilon_2}\\ &\le \frac{8}{3} \cdot \frac{\left|\lambda_{\max}(D - \lambda_{\min}(D) \cdot I + \frac{B - \lambda_{\min}(B) \cdot I}{\alpha})\right| + |\lambda_{\min}(D)| + \left|\frac{\lambda_{\min}(B)}{\alpha}\right|}{\epsilon_2}\\ &\le \frac{8U}{3\epsilon_2}.\end{aligned}$$

Therefore, following the definition (5.15), we have

$$-c_1(\alpha_{i+1} - \alpha_i) = -\alpha_{i+1}\nu_1(\alpha_{i+1}) + \alpha_i\nu_1(\alpha_i) \le (\alpha_{i+1} + \alpha_i)\frac{8U}{3\epsilon_2}. \qquad (5.40)$$

Substituting (5.40) into (5.39) yields

$$\begin{aligned}UB^* = \max_{\alpha \in [\alpha_i, \alpha_{i+1}]} \overline{\mathcal{G}}(\alpha) &\le \mathcal{G}^{\epsilon}(\alpha_i) + \frac{\epsilon}{2} + \frac{8U(\alpha_{i+1} + \alpha_i)^2(\alpha_{i+1} - \alpha_i)}{3\epsilon_2\alpha_i^2}\\ &\le LB + \frac{32U\lambda_n^2(\alpha_{i+1} - \alpha_i)}{3\epsilon_2\lambda_1^2}.\end{aligned}$$

Consequently, the stopping criterion $UB^* < LB + \epsilon$ is reached if

$$\alpha_{i+1} - \alpha_i < \frac{3\epsilon_2\lambda_1^2}{32U\lambda_n^2} \cdot \epsilon.$$

Therefore, the number of the iterations of our new algorithm cannot exceed the upper bound (5.37). The proof is complete. ∎

According to Theorems 4.2 and 5.3, in order to find a global ϵ-approximation solution of (P_α), one must solve λ_1^{ϵ}, λ_n^{ϵ}, and

$$\left\lceil \frac{32U\lambda_n^2(\lambda_n - \lambda_1)}{3\lambda_1^2\ \epsilon_2\epsilon} \right\rceil \cdot \left(\left\lceil 4\log_2\left(\frac{4}{3}\epsilon_2^{-1}\right) + \log_2\left(2\epsilon^{-1}\right) + c \right\rceil\right)$$

maximum eigenvalue problems. Notice that there is a linear-time algorithm to approximate the maximum eigenvalue.

THEOREM 5.4 (see [14]). *Let $A \in \mathbb{R}^{n\times n}$ be symmetric with at most N nonzero entries and $\|A\| \le \rho$. For given parameters $\epsilon, \delta > 0$, the Lanczos method* [11] *runs in time $O(\frac{N\sqrt{\rho}}{\sqrt{\epsilon}} \log \frac{n}{\delta})$ and returns a unit vector $x \in \mathbb{R}^n$ such that*

$$x^T A x \ge \lambda_{\max}(A) - \epsilon$$

with probability at least $1 - \delta$.

Then we obtain the linear-time complexity of the new algorithm.

COROLLARY 5.1. *Given the parameters $\epsilon > 0$, $\delta > 0$, and ϵ_2 (see* (5.25)*), with probability at least $1-\delta$ over the randomization of an approximate eigenvector oracle, our branch-and-bound algorithm returns a global ϵ-approximation solution of* (P_α) *in total time*

$$O\left(N\sqrt{\rho}\log\frac{n}{\delta}\left(\frac{1}{\sqrt{\epsilon_2}} + \frac{U\lambda_n^2(\lambda_n-\lambda_1)}{\lambda_1^2\ \epsilon_2\epsilon\sqrt{\epsilon}}\left(4\log_2\left(\frac{4}{3}\epsilon_2^{-1}\right) + \log_2\left(2\epsilon^{-1}\right) + c\right)\right)\right),$$

where U is defined in (5.38) *and N is an upper bound for the number of all nonzero entries in B, D, and W.*

Remark 5.1. We have shown that the optimal value of (P) can be approximated in linear time in terms of the number of nonzero elements of the input matrices. This does not mean that we can recover the approximate solution x^* in linear time. As shown at the end of section 4, in order to recover the optimal solution x^*, one has to find the whole eigenspace of the final matrix. Except for the rare case where the maximum eigenvalue of the matrix pencil $D + B/\alpha - \nu_1 W$ is simple, it is unknown whether the recovery procedure can be done in linear time.

6. Numerical experiments. In this section, we numerically compare our new branch-and-bound algorithm with the approach calling the software BARON [21] to solve the equivalent (QCQP) and the existing Algorithm L-BB [24] presented in section 3, where the SDP subproblems are solved by SDPT3 within CVX [10]. The tolerance is set as $\epsilon = 10^{-6}$ for BARON and the two algorithms. The maximum CPU time limit is set equal to 3,600 seconds. All the experiments are carried out in MATLAB R2014a and run on a server with 2.6 GHz dual-core processor and 32 GB RAM.

Notice that the maximum dimension of the tested examples that the existing global optimization algorithms in comparison can solve in 3,600 seconds is only 320. We simply used the function *"eig"* in MATLAB to compute the eigenvalues with high precision (i.e., the unit roundoff $u \approx 1.1 \times 10^{-16}$). When the matrix A is a real symmetric matrix, $eig(A)$ applies the symmetric QR algorithm for small matrices and Cuppen's divide and conquer algorithm for large matrices; see [1] for more details. As shown in [11, section 8.3], the standard symmetric QR algorithm needs about $\frac{4}{3}n^3 + O(n^2)$ floating-point operations to compute the eigenvalues of an nth order symmetric matrix.

First, we test the following five examples, which are taken from [17, 29].

Example 6.1 (Example 3.2 of [29]). Let $n = 3$,

$$B = \begin{pmatrix} 2.3969 & 0.4651 & 4.6392 \\ 0.4651 & 5.4401 & 0.7838 \\ 4.6392 & 0.7838 & 10.1741 \end{pmatrix},$$

$$W = \begin{pmatrix} 0.8077 & 0.8163 & 1.0970 \\ 0.8163 & 4.1942 & 0.8457 \\ 1.0970 & 0.8457 & 1.8810 \end{pmatrix}, \qquad D = \begin{pmatrix} 3.9104 & -0.9011 & -2.0128 \\ -0.9011 & 0.9636 & 0.6102 \\ -2.0128 & 0.6102 & 1.0908 \end{pmatrix}.$$

Example 6.2 (Example 3.1 of [29]). Let $B = \mathrm{Diag}(1, 9, 2)$, $W = D = \mathrm{Diag}(5, 2, 3)$.

Example 6.3 (Example 3 of [17]). Let $n = 4$,

$$B = \begin{pmatrix} 1 & 2 & 3 & 1 \\ 2 & 5 & 4 & -1 \\ 3 & 4 & 0 & 1 \\ 1 & -1 & 1 & 6 \end{pmatrix}, \quad W = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 10 \end{pmatrix}, \quad D = \begin{pmatrix} 5 & -1 & 0 & 3 \\ -1 & 9 & 1 & 0 \\ 0 & 1 & -2 & 0 \\ 3 & 0 & 0 & 8 \end{pmatrix}.$$

Example 6.4 (Example 4 of [17]). Let $n = 10$,

$$B = \mathrm{Diag}(1, 2, 8, 7, 9, 3, 10, 2, -1, 6),$$
$$W = \mathrm{Diag}(9, 8, 7, 6, 5, 4, 3, 2, 1, 10),$$
$$D = \mathrm{Diag}(5, 20, 3, 4, 8, -1, 0, 6, 32, 10).$$

Example 6.5 (Example 5 of [17]). Let $n = 20$,

$$B = \mathrm{Diag}(1, 2, 20, 3, 50, 4, 6, 7, 8, 9, 100, 2, 3, 4, 5, 6, 7, 0, 10, 9),$$
$$W = \mathrm{Diag}(100, 1, 2, 30, 5, 7, 9, 7, 8, 9, 1, 2, 30, 1, 50, 8, 1, 10, 10, 9),$$
$$D = \mathrm{Diag}(0, 1000, 20, 2, 5, 6, 7, 9, 50, 3, 4, 5, 100, 5, 2, 200, 4, 5, 9, 21).$$

In Figures 6.1 and 6.2 we plot the objective functions $\mathcal{G}(\alpha)$ for Examples 6.1–6.5. It is observed that $\mathcal{G}(\alpha)$ is convex for Example 6.2 and concave (though nonsmooth) for Example 6.4. For Examples 6.2, 6.4, and 6.5, one can see that their optimal values are all attained at the left-hand endpoints. Actually, since their input matrices are all diagonal, according to Theorem 2.1, they belong to easy-to-solve cases. However, the nondiagonal case seems to be more complicated, as shown by Figure 6.1 where $\mathcal{G}(\alpha)$ is neither convex nor concave.

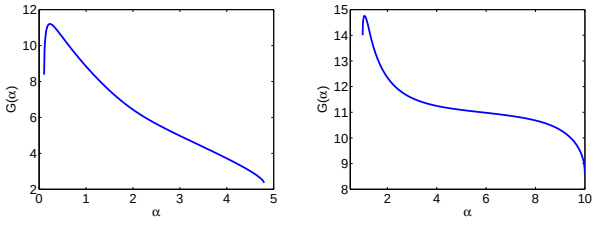

FIG. 6.1. *The variations of* $\mathcal{G}(\alpha)$ *for Examples* 6.1 *and* 6.3.

We use BARON, Algorithm L-BB, and our new branch-and-bound algorithm to solve Examples 6.1–6.5. We report in Table 6.1 the CPU time (recorded in seconds), the number of iterations (denoted by "#iter.") and the returned optimal values

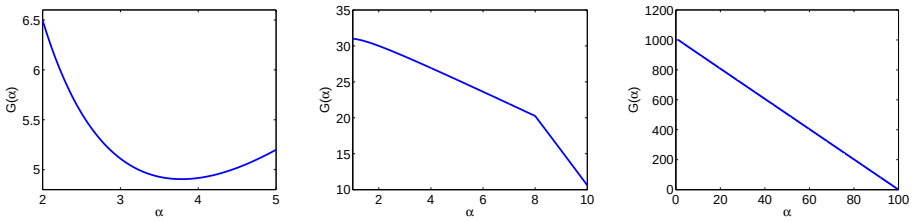

FIG. 6.2. *The variations of* $\mathcal{G}(\alpha)$ *for Examples* 6.2, 6.4, *and* 6.5.

(v(P)) for Algorithm L-BB and our new algorithm. One can observe that Examples 6.2, 6.4, and 6.5 are relatively easy to solve, which is demonstrated by Theorem 2.1. Moreover, according to Figure 6.2 and Table 6.1, for the objective function $\mathcal{G}(\alpha)$, the convex case seems to be much easier than the concave one. Comparing with Algorithm L-BB, BARON is more efficient for these small-size problems. For each example, our new branch-and-bound algorithm is the most efficient among the three approaches. Notice that the main computation in each iteration of Algorithm L-BB and ours is to solve one SDP subproblem. We can observe from the ratios between the CPU time and the number of iterations that our bisection algorithm proposed in section 4 highly outperforms the solver SDPT3.

TABLE 6.1
Numerical comparison between BARON, Algorithm L-BB, and our new algorithm for solving Examples 6.1–6.5.

	BARON		L-BB			Ours		
Ex.	Time	v(P)	Time	# iter.	v(P)	Time	# iter.	v(P)
1	2.21	11.2008	337.66	1463	11.2008	0.05	28	11.2008
2	0.07	6.5000	0.59	4	6.5000	0.02	2	6.5000
3	1.42	14.7550	50.69	229	14.7550	0.05	33	14.7550
4	0.07	31.0000	3.09	18	31.0000	0.05	17	31.0000
5	0.17	1002.0000	5.33	29	1002.0000	0.02	5	1002.0000

Second, we test more randomly generalized examples in high dimension. We generate symmetric matrices B, D and a lower bidiagonal matrix L, whose (nonzero) elements are independently and uniformly distributed in $[-\eta, \eta]$ ($\eta = 1$ or 10), and then let $W = 4LL^T + I$ to keep the denominator far away from zero. For each dimension n, we independently and randomly generate five instances and then run BARON, Algorithm L-BB, and our new branch-and-bound algorithm. We report in Tables 6.2–6.5 the average CPU time (recorded in seconds) and average numbers of iterations (denoted by "#iter."), where "—" indicates that the time limit of 3,600 seconds was reached. For the problems in low dimension ($n \leq 5$), BARON performs more efficiently than Algorithm L-BB. Since BARON failed to solve problems of large size in 3,600 seconds, for the cases in which $n \geq 30$, we only report the numerical comparison between Algorithm L-BB and our new branch-and-bound algorithm. All instances we have tested demonstrate the very high efficiency of our new algorithm. Finally, since (P) is a simultaneous optimization of an absolute and a relative term, distinct choices of η lead to different levels of difficulty. From Tables 6.2–6.5, we observe that (P) with $\eta = 1$ is much harder to solve than the $\eta = 10$ case when calling BARON and Algorithm L-BB. However, our new branch-and-bound algorithm seems to be insensitive to these different choices.

TABLE 6.2
Numerical comparison between Algorithm L-BB, and our new algorithm for solving (P) *with* $\eta = 1$.

	L-BB		Ours	
n	Time	# iter.	Time	# iter.
30	683.88	2704.0	0.22	39.8
50	1034.22	3390.4	0.44	40.4
80	1825.18	3778.2	0.94	40.1
100	—	—	1.88	43.2

TABLE 6.3
Numerical comparison between Algorithm L-BB, and our new algorithm for solving (P) *with* $\eta = 10$.

	L-BB		Ours	
n	Time	# iter.	Time	# iter.
30	31.01	124.4	0.22	42.2
50	53.63	182.1	0.45	42.6
80	77.67	167.2	1.06	44.8
100	137.11	217.3	1.87	44.7
120	171.04	193.1	2.68	44.5
150	270.26	185.7	4.13	45.0
180	528.61	226.8	5.87	45.9
200	667.49	216.6	7.03	45.2
220	874.41	217.4	8.56	46.0
250	1439.84	245.2	11.13	46.1
280	2186.28	263.9	14.02	46.1
300	2941.02	284.2	15.95	46.8
320	—	—	17.90	45.9

Appendix A. We show how to construct a perturbation version of (P) so that Assumption 5.1 is satisfied. We assume $n \geq 2$ and $\epsilon \in (0,1)$ throughout this section. Let $(0 <)\lambda_1 \leq \cdots \leq \lambda_n$ be n eigenvalues of W and $u_1, \ldots, u_n$ be the corresponding unit eigenvectors. For any $\epsilon \in (0,1)$, let u_1^ϵ be an approximation of u_1 satisfying $\|u_1^\epsilon\| = 1$ and

$$\lambda_1 \leq (u_1^\epsilon)^T W u_1^\epsilon \leq \lambda_1 + \epsilon. \tag{A.1}$$

We first present two technical lemmas which will be used later.

LEMMA A.1. *Let $W \succeq 0$. For any $v \in \mathbb{R}^n$ with $\|v\| = 1$, $v^T W v \leq \mathrm{tr}(W)$.*

Proof. Since $\|v\| = 1$ and $W \succeq 0$, it holds that $v^T W v \leq \lambda_{\max}(W) \leq \mathrm{tr}(W)$. ∎

LEMMA A.2. *Let $\lambda_1 \leq \cdots \leq \lambda_n$ be n eigenvalues of W ($W \succ 0$) and u_1^ϵ be an approximation eigenvector of λ_1 satisfying $\|u_1^\epsilon\| = 1$ and* (A.1). *Then, we have*

$$(u_1^\epsilon)^T W^2 u_1^\epsilon \leq \lambda_1^2 + \epsilon(\mathrm{tr}(W) + \epsilon).$$

Proof. According to the definition of u_1^ϵ, it is sufficient to overestimate the optimal value of the following optimization problem:

$$\max \left\{ (u_1^\epsilon)^T W^2 u_1^\epsilon : \ (u_1^\epsilon)^T W u_1^\epsilon \leq \lambda_1 + \epsilon, \ (u_1^\epsilon)^T u_1^\epsilon = 1 \right\}. \tag{UB}$$

Let $W = \sum_{i=1}^n \lambda_i u_i u_i^T$. We have $W^2 = \sum_{i=1}^n \lambda_i^2 u_i u_i^T$. By introducing $t_i = (u_i^T u_1^\epsilon)^2$, we can reformulate (UB) as the following linear programming problem:

$$\max \left\{ \sum_{i=1}^n \lambda_i^2 t_i : \ \sum_{i=1}^n \lambda_i t_i \leq \lambda_1 + \epsilon, \ \sum_{i=1}^n t_i = 1, \ t \geq 0 \right\}. \tag{ULP}$$

TABLE 6.4
Numerical comparison between BARON, Algorithm L-BB, and our new algorithm for solving small-size (P) *with* $\eta = 1$.

	BARON	L-BB		Ours	
n	Time	Time	# iter.	Time	# iter.
3	1.61	102.32	478.1	0.03	34.6
4	5.82	131.96	627.6	0.02	31.1
5	29.46	163.23	777.4	0.04	33.1
6	628.11	117.44	557.3	0.04	35.2
7	—	272.90	1286.8	0.03	32.6

TABLE 6.5
Numerical comparison among BARON, Algorithm L-BB, and our new algorithm for solving small-size (P) *with* $\eta = 10$.

	BARON	L-BB		Ours	
n	Time	Time	# iter.	Time	# iter.
3	1.33	28.78	134.4	0.03	38.3
4	3.54	28.62	133.2	0.03	37.5
5	22.48	29.96	139.0	0.03	36.3
6	121.16	27.05	123.2	0.03	38.8
7	—	30.49	135.0	0.04	38.4

Notice that (ULP) attains its optimal value at an extreme point of its feasible region and there are at most two nonzero elements for each extreme point. Let t^* be an optimal solution to (ULP). We have the following two cases.

(a) Suppose $t^*_{\bar{i}} > 0$ and $t^*_j = 0$ for $j \neq \bar{i}$. It follows that $t^*_{\bar{i}} = 1$ and $\lambda_{\bar{i}} \leq \lambda_1 + \epsilon$. Then, we have

$$v(\text{ULP}) = \lambda^2_{\bar{i}} \leq (\lambda_1 + \epsilon)^2 = \lambda^2_1 + \epsilon(2\lambda_1 + \epsilon). \tag{A.2}$$

(b) Suppose $t^*_{\bar{i}} > 0$, $t^*_{\bar{j}} > 0$ and $t^*_k = 0$ for $k \neq \bar{i}, \bar{j}$. Without loss of generality, we assume $\lambda_{\bar{i}} < \lambda_{\bar{j}}$. Then it holds that

$$\begin{cases} \lambda_{\bar{i}} t^*_{\bar{i}} + \lambda_{\bar{j}} t^*_{\bar{j}} = \lambda_1 + \epsilon, \\ t^*_{\bar{i}} + t^*_{\bar{j}} = 1,\ t^*_{\bar{i}} > 0,\ t^*_{\bar{j}} > 0. \end{cases}$$

It is easy to verify that $\lambda_{\bar{i}} < \lambda_1 + \epsilon < \lambda_{\bar{j}}$, and hence

$$\begin{aligned} v(\text{ULP}) &= \lambda_{\bar{j}}(\lambda_1 + \epsilon - \lambda_{\bar{i}}) + \lambda_{\bar{i}}(\lambda_1 + \epsilon) \leq \lambda_n(\lambda_1 + \epsilon - \lambda_{\bar{i}}) + \lambda_{\bar{i}}(\lambda_1 + \epsilon) \\ &= (\lambda_1 + \epsilon - \lambda_n)\lambda_{\bar{i}} + \lambda_n(\lambda_1 + \epsilon) \leq (\lambda_1 + \epsilon - \lambda_n)\lambda_1 + \lambda_n(\lambda_1 + \epsilon) \\ &= \lambda^2_1 + \epsilon(\lambda_1 + \lambda_n). \end{aligned} \tag{A.3}$$

Since $2\lambda_1 \leq \lambda_1 + \lambda_n \leq \text{tr}(W)$, combining (A.2) with (A.3) completes the proof. ∎

Define

$$\widetilde{\lambda}_1 = (u^\epsilon_1)^T W u^\epsilon_1 + \sqrt{\epsilon}, \tag{A.4}$$

$$R = \frac{1}{\sqrt{\epsilon}} \left(\widetilde{\lambda}_1 u^\epsilon_1 - W u^\epsilon_1\right) \left(\widetilde{\lambda}_1 u^\epsilon_1 - W u^\epsilon_1\right)^T,$$

$$\widetilde{W} = W + R + (2\epsilon + \sqrt{\epsilon})(I_n - u^\epsilon_1 (u^\epsilon_1)^T). \tag{A.5}$$

It is not difficult to verify that

$$\widetilde{W}u_1^\epsilon = (W+R)u_1^\epsilon = \widetilde{\lambda}_1 u_1^\epsilon.$$

That is, $\widetilde{\lambda}_1$ is an eigenvalue of $\widetilde{W}$ and u_1^ϵ is the corresponding unit eigenvector. For any v such that $v^Tv = 1$ and $v^Tu_1^\epsilon = 0$, we have

$$v^T\widetilde{W}v \geq v^TWv + 2\epsilon + \sqrt{\epsilon} \geq \lambda_1 + 2\epsilon + \sqrt{\epsilon}. \tag{A.6}$$

That is, except for $\widetilde{\lambda}_1$, every eigenvalue of $\widetilde{W}$ is larger than or equal to $\lambda_1 + 2\epsilon + \sqrt{\epsilon}$.

Notice that it implies from (A.1) and (A.4) that

$$\widetilde{\lambda}_1 \leq \lambda_1 + \epsilon + \sqrt{\epsilon}. \tag{A.7}$$

Combining (A.6) with (A.7) yields that $\widetilde{\lambda}_1$ is the minimum eigenvalue of $\widetilde{W}$.

Let

$$\widehat{\epsilon} := \frac{1}{2}\min\left\{1/\widetilde{\lambda}_1, \epsilon\right\} \tag{A.8}$$

and $\widehat{\lambda}_n$ be the maximum eigenvalue of $\widetilde{W} - \frac{1}{\widehat{\epsilon}}u_1^\epsilon(u_1^\epsilon)^T$. Let $\widehat{u}_n$ be an approximation of the unit eigenvector corresponding to $\widehat{\lambda}_n$ satisfying $\|\widehat{u}_n\| = 1$ and

$$\widehat{\lambda}_n - \widehat{\epsilon} \leq \widehat{u}_n^T\left(\widetilde{W} - \frac{1}{\widehat{\epsilon}}u_1^\epsilon(u_1^\epsilon)^T\right)\widehat{u}_n \leq \widehat{\lambda}_n. \tag{A.9}$$

Since $n \geq 2$ and u_1^ϵ is an eigenvector corresponding to the minimum eigenvalue of $\widetilde{W}$, $\widehat{\lambda}_n$ remains the maximum eigenvalue of $\widetilde{W}$. Then it implies from (A.9) that

$$\left(\widehat{u}_n^Tu_1^\epsilon\right)^2 \leq \widehat{\epsilon}\left(\widehat{u}_n^T\widetilde{W}\widehat{u}_n - \widehat{\lambda}_n + \widehat{\epsilon}\right) \leq \widehat{\epsilon}\left(\widehat{\lambda}_n - \widehat{\lambda}_n + \widehat{\epsilon}\right) = \widehat{\epsilon}^2. \tag{A.10}$$

Define

$$u_n^\epsilon = \frac{1}{\sqrt{1-(\widehat{u}_n^Tu_1^\epsilon)^2}}\left(\widehat{u}_n - (\widehat{u}_n^Tu_1^\epsilon)u_1^\epsilon\right). \tag{A.11}$$

It is not difficult to verify that $(u_n^\epsilon)^Tu_1^\epsilon = 0$, $\|u_n^\epsilon\| = 1$, and

$$(u_n^\epsilon)^T\widetilde{W}u_n^\epsilon = \frac{1}{1-(\widehat{u}_n^Tu_1^\epsilon)^2}\left(\widehat{u}_n^T\widetilde{W}\widehat{u}_n - 2(\widehat{u}_n^Tu_1^\epsilon)\widehat{u}_n^T\widetilde{W}u_1^\epsilon + \left(\widehat{u}_n^Tu_1^\epsilon\right)^2(u_1^\epsilon)^T\widetilde{W}u_1^\epsilon\right)$$

$$\geq \widehat{u}_n^T\widetilde{W}\widehat{u}_n - 2(\widehat{u}_n^Tu_1^\epsilon)\widehat{u}_n^T\widetilde{W}u_1^\epsilon = \widehat{u}_n^T\widetilde{W}\widehat{u}_n - 2\widetilde{\lambda}_1(\widehat{u}_n^Tu_1^\epsilon)^2 \tag{A.12}$$

$$\geq \widehat{\lambda}_n - \widehat{\epsilon} + \frac{1}{\widehat{\epsilon}}(\widehat{u}_n^Tu_1^\epsilon)^2 - 2\widetilde{\lambda}_1(\widehat{u}_n^Tu_1^\epsilon)^2 \geq \widehat{\lambda}_n - \widehat{\epsilon} - 2\widetilde{\lambda}_1\widehat{\epsilon}^2 \geq \widehat{\lambda}_n - \epsilon, \tag{A.13}$$

where the equality in (A.12) holds as $\widetilde{W}u_1^\epsilon = \widetilde{\lambda}_1u_1^\epsilon$, and the three inequalities in (A.13) follow from (A.9), (A.10), and (A.8), respectively. One can observe from the definition of $\widetilde{W}$ (see (A.5)) that $\widehat{\lambda}_n > 2\epsilon + \sqrt{\epsilon} > \epsilon$.

Define

$$\overline{\lambda}_n = (u_n^\epsilon)^T\widetilde{W}u_n^\epsilon + \sqrt{\epsilon}, \tag{A.14}$$

$$S = \frac{1}{\sqrt{\epsilon}}\left(\overline{\lambda}_nu_n^\epsilon - \widetilde{W}u_n^\epsilon\right)\left(\overline{\lambda}_nu_n^\epsilon - \widetilde{W}u_n^\epsilon\right)^T,$$

$$\overline{W} = \widetilde{W} + S + \left(2\epsilon + (4\mathrm{tr}(W)+9)\sqrt{\epsilon}\right)u_n^\epsilon(u_n^\epsilon)^T. \tag{A.15}$$

Since $(u_n^\epsilon)^T u_1^\epsilon = 0$ and $\widetilde{W} u_1^\epsilon = \widetilde{\lambda}_1 u_1^\epsilon$, we can verify that

$$\overline{W} u_1^\epsilon = \widetilde{\lambda}_1 u_1^\epsilon,\ \overline{W} u_n^\epsilon = \left(\overline{\lambda}_n + 2\epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon}\right) u_n^\epsilon. \tag{A.16}$$

If $n = 2$, combining (A.14) with (A.13) yields that

$$\overline{\lambda}_n \geq \widehat{\lambda}_n - \epsilon + \sqrt{\epsilon} \geq \widetilde{\lambda}_1 - \epsilon + \sqrt{\epsilon}.$$

Therefore, $\widetilde{\lambda}_1$ and $\overline{\lambda}_n + 2\epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon}$ are the minimal and maximum eigenvalues of $\overline{W}$, respectively, and Assumption 5.1 holds for $\overline{W}$. Now we assume $n \geq 3$. Let $v \notin \{0, \pm u_1^\epsilon, \pm u_n^\epsilon\}$ be any other unit eigenvector of $\overline{W}$. Then $v^T v = 1$ and $v^T u_1^\epsilon = v^T u_n^\epsilon = 0$. It follows that

$$v^T \overline{W} v \geq v^T \widetilde{W} v \geq \lambda_1 + 2\epsilon + \sqrt{\epsilon} \geq \widetilde{\lambda}_1 + \epsilon, \tag{A.17}$$

$$v^T \widetilde{W} v + v^T S v \leq \widehat{\lambda}_n + v^T S v \leq \overline{\lambda}_n + \epsilon - \sqrt{\epsilon} + v^T S v, \tag{A.18}$$

where the last two inequalities in (A.17) hold true due to (A.6) and (A.7), respectively, and the two inequalities in (A.18) follow from the fact that $\widehat{\lambda}_n$ is the maximum eigenvalue of $\widetilde{W}$ and (A.13), (A.14), respectively. Consequently, this implies from (A.17) that $\widetilde{\lambda}_1$ is the minimum eigenvalue of $\overline{W}$ and the second minimal eigenvalue satisfies that

$$\lambda_2(\overline{W}) \geq \lambda_{\min}(\overline{W}) + \epsilon. \tag{A.19}$$

According to Lemma A.2 and the definitions of R and $\widetilde{\lambda}_1$, we have

$$\begin{aligned}
\mathrm{tr}(R) &= \frac{1}{\sqrt{\epsilon}} \left(\widetilde{\lambda}_1^2 - 2\widetilde{\lambda}_1 (u_1^\epsilon)^T W u_1^\epsilon + (u_1^\epsilon)^T W^2 u_1^\epsilon \right) \\
&= \frac{1}{\sqrt{\epsilon}} \left(\left(\widetilde{\lambda}_1 - (u_1^\epsilon)^T W u_1^\epsilon \right)^2 - \left((u_1^\epsilon)^T W u_1^\epsilon \right)^2 + (u_1^\epsilon)^T W^2 u_1^\epsilon \right) \\
&\leq \frac{1}{\sqrt{\epsilon}} \left(\left(\sqrt{\epsilon}\right)^2 - \lambda_1^2 + \lambda_1^2 + \epsilon(\mathrm{tr}(W) + \epsilon) \right) \\
&= \sqrt{\epsilon}(1 + \mathrm{tr}(W) + \epsilon). \qquad \text{(A.20)}
\end{aligned}$$

Therefore, it follows from the definition of $\widetilde{W}$ (see (A.5)) and the assumption $\epsilon < 1$ that

$$\widehat{\lambda}_n = \lambda_{\max}(\widetilde{W}) \leq \mathrm{tr}(W) + \mathrm{tr}(R) + 2\epsilon + \sqrt{\epsilon} \leq 2 + 3\epsilon + 2\mathrm{tr}(W). \tag{A.21}$$

According to Lemma A.1, for $\|v\| = 1$, we have

$$\begin{aligned}
v^T S v \leq \mathrm{tr}(S) &= \frac{1}{\sqrt{\epsilon}} \left(\overline{\lambda}_n u_n^\epsilon - \widetilde{W} u_n^\epsilon \right)^T \left(\overline{\lambda}_n u_n^\epsilon - \widetilde{W} u_n^\epsilon \right) \\
&= \frac{1}{\sqrt{\epsilon}} \left(\left(\overline{\lambda}_n - (u_n^\epsilon)^T \widetilde{W} u_n^\epsilon \right)^2 - \left((u_n^\epsilon)^T \widetilde{W} u_n^\epsilon \right)^2 + (u_n^\epsilon)^T \widetilde{W}^2 u_n^\epsilon \right) \\
&\leq \frac{1}{\sqrt{\epsilon}} \left((\sqrt{\epsilon})^2 - (\widehat{\lambda}_n - \epsilon)^2 + \widehat{\lambda}_n^2 \right) \qquad \text{(A.22)} \\
&= \sqrt{\epsilon} \left(1 + 2\widehat{\lambda}_n - \epsilon \right) \\
&\leq \sqrt{\epsilon} \left(4\mathrm{tr}(W) + 10 \right), \qquad \text{(A.23)}
\end{aligned}$$

where (A.22) follows from (A.14), (A.13), and the fact that $\lambda_{\max}(\widetilde{W}^2) = \lambda^2_{\max}(\widetilde{W}) = \widehat{\lambda}^2_n$, and (A.23) holds due to (A.21). Substituting (A.23) into (A.18) yields that

$$\lambda_i(\overline{W}) \le \overline{\lambda}_n + \epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon}. \tag{A.24}$$

We conclude from (A.16) and (A.24) that $\overline{\lambda}_n + 2\epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon}$ is the maximum eigenvalue of $\overline{W}$ and the second maximal eigenvalue $\lambda_{n-1}(\overline{W})$ satisfies

$$\lambda_{n-1}(\overline{W}) \le \overline{\lambda}_n + \epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon} = \lambda_{\max}(\overline{W}) - \epsilon. \tag{A.25}$$

Define a perturbation version of (P),

$$\max_{x\in S} \left\{ f^\epsilon(x) := \frac{x^T(B + \bar{t}W)x}{x^T\overline{W}x} + x^T D x - \bar{t} \right\}, \tag{P$^\epsilon$}$$

where the parameter $\bar{t}$ is chosen in the following to guarantee that $B + \bar{t}W \succeq 0$:

$$\bar{t} \ge \frac{\max\{0, -\lambda_{\min}(B)\}}{\lambda_{\min}(W)}.$$

According to (A.19) and (A.25), Assumption 5.1 is now satisfied for (P$^\epsilon$). The following result shows that one can obtain an approximation solution to (P) by solving (P$^\epsilon$).

THEOREM A.1. *For any given $\epsilon \in (0,1)$, any $\sqrt{\epsilon}$-approximately optimal solution of* (P$^\epsilon$) *remains a $\rho\sqrt{\epsilon}$-approximately optimal solution of* (P)*, where*

$$\rho = \frac{9\mathrm{tr}(W) + 3n + 20}{\underline{\lambda}} \left(\frac{\mathrm{tr}(B)}{\underline{\lambda}} + \bar{t} \right) + 1$$

and $\underline{\lambda} > 0$ is any underestimate of λ_1.

Proof. Let $M = 9\mathrm{tr}(W) + 3n + 20$. According to the definition of $\overline{W}$, for any $x \in S$, it follows from Lemma A.1 that

$$\begin{aligned} x^T(\overline{W} - W)x &\le \mathrm{tr}(R) + \mathrm{tr}(S) + (2\epsilon + \sqrt{\epsilon})(n-1) + 2\epsilon + (4\mathrm{tr}(W) + 9)\sqrt{\epsilon} \\ &\le \sqrt{\epsilon}\,(\epsilon + 9\mathrm{tr}(W) + n + 19) + 2n\epsilon \le M\sqrt{\epsilon}, \end{aligned}$$

where the second inequality follows from (A.20), (A.23), and the assumption $\epsilon < 1$. Therefore, we have

$$\begin{aligned} v(\mathrm{P}^\epsilon) &\ge \max_{x\in S} \left\{ \frac{x^T(B + \bar{t}W)x}{x^TWx + M\sqrt{\epsilon}} + x^T D x - \bar{t} \right\} \\ &= \max_{x\in S} \left\{ \frac{x^TWx}{x^TWx + M\sqrt{\epsilon}} \cdot \frac{x^T(B + \bar{t}W)x}{x^TWx} + x^T D x - \bar{t} \right\} \\ &\ge \max_{x\in S} \left\{ \frac{\underline{\lambda}}{\underline{\lambda} + M\sqrt{\epsilon}} \cdot \frac{x^T(B + \bar{t}W)x}{x^TWx} + x^T D x - \bar{t} \right\} \qquad \text{(A.26)} \\ &\ge v(\mathrm{P}) - \frac{M\sqrt{\epsilon}}{\underline{\lambda} + M\sqrt{\epsilon}} \cdot \max_{x\in S} \frac{x^T(B + \bar{t}W)x}{x^TWx} \\ &\ge v(\mathrm{P}) - \frac{M\sqrt{\epsilon}}{\underline{\lambda}} \left(\frac{\mathrm{tr}(B)}{\underline{\lambda}} + \bar{t} \right), \qquad \text{(A.27)} \end{aligned}$$

where (A.26) holds since $x^TWx \ge \lambda_1 \ge \underline{\lambda}$.

Let $x^* \in S$ be a $\sqrt{\epsilon}$-approximately optimal solution of (P$^\epsilon$). We have

$$v(\mathrm{P}^\epsilon) - \sqrt{\epsilon} \le f^\epsilon(x^*) \le \frac{x^{*T}(B+\bar{t}W)x^*}{x^{*T}Wx^*} + x^{*T}Dx^* - \bar{t} = f(x^*), \tag{A.28}$$

where the second inequality holds as $\overline{W} \succeq W$. Substituting (A.27) into (A.28) yields

$$f(x^*) \ge v(\mathrm{P}) - \frac{M\sqrt{\epsilon}}{\underline{\lambda}}\left(\frac{\mathrm{tr}(B)}{\underline{\lambda}} + \bar{t}\right) - \sqrt{\epsilon} = v(\mathrm{P}) - \left(\frac{M}{\underline{\lambda}}\left(\frac{\mathrm{tr}(B)}{\underline{\lambda}} + \bar{t}\right) + 1\right)\sqrt{\epsilon}.$$

The proof is complete. ∎

Remark A.1. According to (A.1), $\underline{\lambda}$ can be chosen as $(u_1^{\epsilon'})^T W u_1^{\epsilon'} - \epsilon'$, where

$$\epsilon' = \arg\max\{\epsilon_i : \ (u_1^{\epsilon_i})^T W u_1^{\epsilon_i} - \epsilon_i > 0, \ \epsilon_i = 1/2^i, \ i = 1, 2, \dots\}.$$

The above procedure for finding ϵ' stops in at most $K = \lceil \log_2(1/\lambda_1) \rceil + 1$ steps since $\epsilon_K = 1/2^K < \lambda_1 \le (u_1^{\epsilon_K})^T W u_1^{\epsilon_K}$.

As in Corollary 5.1, the optimal value of (P) can be approximated in linear time in terms of N, which denotes an upper bound for the number of all nonzero entries in B, D, and W. We finally show that either constructing or approximately solving (P$^\epsilon$) can be done in linear time.

We first notice that there is no need to explicitly store $\overline{W}$ as it can be generated from W, u_1^ϵ, and u_n^ϵ according to (A.15) and (A.5). It follows from Theorem 5.4 that u_1^ϵ can be obtained in linear time. According to (A.11) and (A.9), calculating u_n^ϵ relies on approximating $\lambda_{\max}(W+R+\gamma I_n - T)$, where $\gamma = 2\epsilon + \sqrt{\epsilon}$ and $T = (\gamma + \frac{1}{\epsilon})u_1^\epsilon (u_1^\epsilon)^T$. Notice that when applying the Lanczos method to approximate $\lambda_{\max}(A)$, the main complexity relies on providing the Krylov information $[b, Ab, \dots, A^k b]$ (b is a unit vector), which consists of k matrix-vector multiplications; see [14] for more details. Since both R and T are rank-one matrices, the complexity of the matrix-vector multiplication $(W + R + \gamma I_n - T)b$ is linear time with respect to $\max\{N, n\}$. We notice that $\max\{N, n\} = N$ as $W \succ 0$.

In solving (P$^\epsilon$), the calculation at each iteration mainly relies on approximating $\lambda_{\max}(D + \frac{B+\bar{t}W}{\alpha} - \nu_1(\alpha)\overline{W})$. Though $\overline{W}$ may no longer be sparse, this maximum eigenvalue can be approximated in linear time since $\overline{W}$ is the sum of W, a scalar matrix, and four rank-one matrices, and hence $\overline{W}$ has the same computational complexity on matrix-vector multiplication as W.

Acknowledgments. The authors are grateful to two anonymous referees for very helpful comments and suggestions.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK User's Guide*, 3rd ed., SIAM, Philadelphia, PA, 1999.

[2] A. I. Barvinok, *Feasibility testing for systems of real quadratic equations*, Discrete Comput. Geom., 10 (1993), pp. 1–13.

[3] D. Bienstock, *A note on polynomial solvability of the CDT problem*, SIAM J. Optim., 26 (2016), pp. 488–498.

[4] L. Brickman, *On the field of values of a matrix*, Proc. Amer. Math. Soc., 12 (1961), pp. 61–66.

[5] A. R. Conn, N. I. Gould, and P. L. Toint, *Trust Region Methods*, MOS-SIAM Ser. Optim. 1, SIAM, Philadelphia, PA, 2000.

[6] M. M. Dundar, G. Fung, J. Bi, S. Sathyakama, and B. Rao, *Sparse Fisher discriminant analysis for computer aided detection*, in Proceedings of the 2005 SIAM International Conference on Data Mining, SIAM, Philadelphia, PA, 2005, pp. 476–480.
[7] A. Fakhri and M. Ghatee, *Minimizing the sum of a linear and a linear fractional function applying conic quadratic representation: Continuous and discrete problems*, Optimization, 65 (2016), pp. 1023–1038.
[8] A. Eriksson, C. Olsson, and F. Kahl, *Normalized cuts revisited: A reformulation for segmentation with linear grouping constraints*, J. Math. Imaging Vision, 39 (2011), pp. 45–61.
[9] E. Fung and K. N. Michael, *On sparse Fisher discriminant method for microarray data analysis*, Bioinformation, 2 (2007), pp. 230–234.
[10] M. Grant and S. Boyd, *CVX: MATLAB Software for Disciplined Convex Programming*, Version 2.1, http://cvxr.com/cvx, 2015.
[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.
[12] E. Hazan and T. Koren, *A linear-time algorithm for trust region problems*, Math. Program., 158 (2016), pp. 363–381.
[13] H. Jiao and S. Liu, *Range division and compression algorithm for quadratically constrained sum of quadratic ratios*, Comput. Appl. Math., 36 (2017), pp. 225–247.
[14] J. Kuczynski and H. Wozniakowski, *Estimating the largest eigenvalue by the power and Lanczos algorithms with a random start*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 1094–1122.
[15] M. W. Mahoney and L. Orecchia, *Implementing regularization implicitly via approximate eigenvector computation*, in Proceedings of the 28th International Conference on Machine Learning, Omnipress, Madison, WI, 2011, pp. 121–128.
[16] T. Matsui, *NP-hardness of linear multiplicative programming and related problems*, J. Global Optim., 9 (1996), pp. 113–119.
[17] V. B. Nguyen, R. L. Sheu, and Y. Xia, *Maximizing the sum of a generalized Rayleigh quotient and another Rayleigh quotient on the unit sphere via semidefinite programming*, J. Global Optim., 64 (2016), pp. 399–416.
[18] V. B. Nguyen, R. L. Sheu, and Y. Xia, *An SDP approach for quadratic fractional problems with a two-sided quadratic constraint*, Optim. Methods Softw., 31 (2016), pp. 701–719.
[19] B. T. Polyak, *Convexity of quadratic transformations and its use in control and optimization*, J. Optim. Theory Appl., 99 (1998), pp. 553–583.
[20] G. Primolevo, O. Simeone, and U. Spagnolini, *Towards a joint optimization of scheduling and beamforming for MIMO downlink*, in Proceedings of the 2006 IEEE Ninth International Symposium on Spread Spectrum Techniques and Applications, IEEE Press, Piscataway, NJ, 2006, pp. 493–497.
[21] N. V. Sahinidis, *BARON: A general purpose global optimization software package*, J. Global Optim., 8 (1996), pp. 201–205.
[22] S. Schaible, *Simultaneous optimization of absolute and relative terms*, ZAMM Z. Angew. Math. Mech., 64 (1984), pp. 363–364.
[23] J. Wang and Y. Xia, *A linear-time algorithm for the trust region subproblem based on hidden convexity*, Optim. Lett., 11 (2017), pp. 1639–1646.
[24] X. H. Wang, L. F. Wang, and Y. Xia, *An efficient global optimization algorithm for maximizing the sum of two generalized Rayleigh quotients*, Comput. Appl. Math., 37 (2018), pp. 4412–4422.
[25] M. C. Wu, L. S. Zhang, Z. X. Wang, D. C. Christiani, and X. H. Lin, *Sparse linear discriminant analysis for simultaneous testing for the significance of a gene set/pathway and gene selection*, Bioinformation, 25 (2009), pp. 1145–1151.
[26] Y. Xia, L. F. Wang, and S. Wang, *Minimizing the sum of linear fractional functions over the cone of positive semidefinite matrices: Approximation and applications*, Oper. Res. Lett., 46 (2018), pp. 76–80.
[27] Y. Ye and S. Zhang, *New results on quadratic minimization*, SIAM J. Optim., 14 (2003), pp. 245–267.
[28] A. Zhang and S. Hayashi, *Celis–Dennis–Tapia based approach to quadratic fractional programming problems with two quadratic constraints*, Numer. Algebra Control Optim., 1 (2011), pp. 83–98.
[29] L. H. Zhang, *On optimizing the sum of the Rayleigh quotient and the generalized Rayleigh quotient on the unit sphere*, Comput. Optim. Appl., 54 (2013), pp. 111–139.
[30] L. H. Zhang, *On a self-consistent-field-like iteration for maximizing the sum of the Rayleigh quotients*, J. Comput. Appl. Math., 257 (2014), pp. 14–28.
[31] L. H. Zhang, W. H. Yang and L. Z. Liao, *A note on the trace quotient problem*, Optim. Lett., 8 (2014), pp. 1637–1645.