Solving the cubic regularization model by a nested restarting Lanczos method

Xiaojing Jia* Xin Liang† Chungen Shen‡ Lei-Hong Zhang§

Abstract

As a variant of the classical trust-region method for unconstrained optimization, the cubic regularization of Newton method introduces a cubic regularization term in the surrogate objective to adaptively adjust the updating step and deals with cases with both indefinite and definite Hessians. It has been demonstrated that the cubic regularization of Newton method enjoys a good global convergence and is an efficient solver for the unconstrained minimization. The main computational cost in each iteration is to solve a cubic regularization subproblem. The Newton iteration is a common and efficient method for this task, especially for small- to medium-size problems. For large size problems, a Lanczos type method was proposed in [Cartis, Gould and Toint, Math. Program., 127:245–295(2011)]. This method relies on a Lanczos procedure to reduce the large-scale cubic regularization subproblem to a small one and solve it by the Newton iteration. For large and ill-conditioned problems, the Lanczos method still needs to produce a large dimensional subspace to achieve a relatively highly accurate approximation, which declines its performance overall. In this paper, we first show that the cubic regularization subproblem can be equivalently transformed into a quadratic eigenvalue problem, which provides an eigensolver alternative to the Newton iteration. We then establish the convergence of the Lanczos method and also propose a nested restarting version for the large scale and ill-conditioned case. By integrating the nested restarting Lanczos iteration into the cubic regularization of Newton method, we verify its efficiency for solving large scale minimization problems in CUTEst collection.

Key words. Unconstrained optimization, Cubic regularization, Newton's method, Eigenvalue problem, Global convergence, Lanczos process, Restarting

AMS subject classifications. 90C30, 90C06, 65K05, 49M15, 65F15

1 Introduction

For the unconstrained minimization

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}), \tag{1.1}$$

where $f:\mathbb{R}^n\to\mathbb{R}$ and its Hessian $\nabla^2 f(\boldsymbol{x})\in\mathbb{R}^{n\times n}$ satisfies

$$\|\nabla^2 f(\boldsymbol{x})-\nabla^2 f(\boldsymbol{y})\|_2\le L\|\boldsymbol{x}-\boldsymbol{y}\|_2,\quad \forall \boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^n,$$

*School of Mathematics, Shanghai University of Finance and Economics, 777 Guoding Road, Shanghai 200433, China. Email: 13127761671@163.com.

†Yau Mathematical Sciences Center, Tsinghua University, Beijing 100084, China, and Yanqi Lake Beijing Institute of Mathematical Sciences and Applications, Beijing 101408, China. Supported in part by the National Natural Science Foundation of China NSFC-11901340 and NSFC-12071332. Email: liangxinslm@tsinghua.edu.cn.

‡College of Science, University of Shanghai for Science and Technology, Shanghai 200093, China. Email: shenchungen@gmail.com.

§Corresponding author. School of Mathematical Sciences, Soochow University, Suzhou 215006, Jiangsu, China. Supported in part by the National Natural Science Foundation of China NSFC-12071332. Email: longzlh@suda.edu.cn.

[27] proposes a variant of the classical trust-region method [9, 35], namely, a cubic regularization of Newton method. Let $0 < L_0 \leq L$. At the current iterate $\boldsymbol{x}_k$, the method solves the following cubic regularization model:

$$\min_{\boldsymbol{h}\in\mathbb{R}^n} \left\{ m(\boldsymbol{h}) := f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^{\mathrm{T}}\boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{h} + \frac{M_k}{6}\|\boldsymbol{h}\|_2^3 \right\}, \tag{1.2}$$

with a properly chosen $M_k \in [L_0, 2L]$; whenever $f(\boldsymbol{x}_k + \boldsymbol{h}) \leq m(\boldsymbol{h})$, the iterate $\boldsymbol{x}_k$ is updated to $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{h}$. The global convergence and local quadratic convergence of this iteration are proved in [27]. Moreover, efficient variants and modifications have been extensively discussed in [7, 8]. In particular, [7, 8] show that there is flexibility in using certain symmetric approximates $H_k \in \mathbb{R}^{n\times n}$ of the Hessian matrix $\nabla^2 f(\boldsymbol{x}_k)$ where the nice global and local convergence can still be guaranteed. Moreover, it is shown in [8] that an adaptive cubic regularization of Newton method needs at most $O(\epsilon^{-3/2})$ function- and gradient-evaluations to achieve an approximation $\boldsymbol{x}_k$ with $\|\nabla f(\boldsymbol{x}_k)\|_2 \leq \epsilon$ for a given accuracy ϵ. This improves the worst-case complexity [19] of the traditional second-order trust-region method where $O(\epsilon^{-2})$ iterations are required to have $\|\nabla f(\boldsymbol{x}_k)\|_2 \leq \epsilon$.

The above cubic regularization of Newton method, as was claimed in [7, 8], can be viewed as an adaptive version of the classical trust-region method [9], where the rules for updating M_k are justified by an analogy to trust-region methods; in particular, M_k might be regarded as the reciprocal of the trust-region radius. Instead of imposing a trust-region $\|\boldsymbol{h}\| \leq \Delta$ for the well-definiteness when $\nabla^2 f(\boldsymbol{x}_k)$ is indefinite, the cubic regularization model (1.2) introduces a regularization term $\frac{M_k}{3}\|\boldsymbol{h}\|_2 I_n$ to $\nabla^2 f(\boldsymbol{x}_k)$ to adaptively adjust the solution $\boldsymbol{h}$ both for the indefinite case and for the definite case. It can be seen that when $\nabla^2 f(\boldsymbol{x}_k)$ is indefinite, the solution $\boldsymbol{h}$ cannot be of infinite norm as the regularization term $\frac{M_k}{3}\|\boldsymbol{h}\|_2 I_n$ will enforce the modified Hessian $\nabla^2 f(\boldsymbol{x}_k) + \frac{M_k}{3}\|\boldsymbol{h}\|_2 I_n$ to be positive definite, and hence the solution of (2.1) is well-defined.

Regarding the update $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{h}$ in the cubic regularization of Newton method, we note that the main computational step lies in solving the cubic regularization model (1.2). This is similar to the trust-region method where the primary computation is to solve the so-called trust-region subproblem (TRS)

$$\min_{\|\boldsymbol{h}\|_W \leq \Delta} \boldsymbol{h}^{\mathrm{T}}\nabla f(\boldsymbol{x}_k) + \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{h} \tag{1.3}$$

where W is a proper positive definite weighted matrix. The TRS has been well-understood in theory (see e.g., [9, 21, 22, 26, 28, 38]) and many efficient numerical methods have been proposed which can be basically grouped into factorization-based algorithms for small-to-medium sized dense problems (see, e.g., [1, 22, 23, 26]) and factorization-free algorithms for large scale and sparse problems (see, e.g., [14, 17, 21, 30, 31, 32, 33, 34, 35, 37, 40, 41]). One of the most widely known factorization-based methods is the Moré-Sorensen method [26], which is a Newton method for solving the associated Lagrange multiplier. Another approach proposed recently in [1] generalizes [14] and translates TRS into certain eigenvalue problems. For large scale TRS or when the Hessian matrix $\nabla^2 f(\boldsymbol{x}_k)$ is only available through its action $\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{z}$ on a vector $\boldsymbol{z}$, a Krylov subspace method, namely the generalized Lanczos Trust-Region method (GLTR), was proposed by Gould *et al.* [15] (see also [9, Chapter 5]). The convergence of GLTR has been recently established in [5, 6, 18, 40, 41] and reveals the linear convergence in the worst case scenario [41, 5].

The purpose of this paper is to develop efficient methods for (1.2). A recent work by Lieder [25] extends [1] for TRS and proposes an equivalent $2(n+1)$ dimensional generalized eigenvalue problem to solve (1.2). In this paper, we shall first introduce an equivalent $(n+1)$ dimensional quadratic eigenvalue problem (QEP) for (1.2). By the new QEP, on the one hand, efficient Krylov subspace methods working on $\mathbb{R}^{n+1}$ such as the second-order Arnoldi process (SOAR) [2] can be directly applied to solve (1.2), and provides new forms of $2(n+1)$ dimensional generalized eigenvalue problems for (1.2) on the other hand. The equivalent reformulations in the form of generalized eigenvalue problem and QEP provide relations of (1.2) with the eigenvalue problem and also offer numerical schemes to solve (1.2), especially for small to- medium size cases. For large size problems or the cases when only the action $\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{z}$ of $\nabla^2 f(\boldsymbol{x}_k)$ on a vector $\boldsymbol{z}$ is available, we discuss Lanczos methods [7, Section 6.2] for (1.2). Previously, the linear convergence of this Lanczos method has been established in [5, 6], and we will sharpen this convergence result

by developing a new convergence analysis; furthermore, we will also design an efficient restarting scheme for this Lanczos method to obtain accurate approximation for ill-conditioned instances[1] of (1.2). The resulting approach consists of a nested restarting procedure and is able to alleviate numerical difficulties of the basic Lanczos method [5, 6] caused by the dimension increment of the underlying Krylov subspace in the Lanczos process. As a practical application, we will integrate the nested restarting Lanczos method for (1.2) to solve large scale minimization problems (1.1) from CUTEst collection [16]. Our numerical experience demonstrates that the nested restarting Lanczos method can be an efficient approach to deal with ill-conditioned inner subproblems (1.2) and improves the overall performance of the cubic regularization of Newton method.

We organize the paper in the following way: in section 2, we first provide basic properties of (1.2). Section 3 then introduces a QEP and establishes the equivalence. The presentation of the Lanczos method of (1.2) is given in section 4 where we shall establish the linear convergence in the worst case, and also propose a nested restarting version for ill-conditioned problems. Numerical verification of our restarting Lanczos method will be carried out in section 5, and final conclusions are drawn in section 6.

Notation. We use the following notation system in this paper. Vectors are generally referred to be column vectors and are typeset in bold lower case letters; in particular, $\boldsymbol{e}_i \in \mathbb{R}^n$ is the ith column of the identity matrix I_n. For a matrix $A \in \mathbb{R}^{m\times n}$, its Moore-Penrose inverse and its column range space of A are presented by $A^\dagger$ and $\mathrm{span}(A)$, respectively. The dimension of $\mathrm{span}(A)$ is given by $\dim(\mathrm{span}(A))$. To facilitate the presentation, we shall conveniently adopt the MATLAB format to access the entries of vectors and matrices: $A_{(i,j)}$ is (i,j)th entry of A, and $A_{(k:\ell,i:j)}$ is the submatrix of A that contains intersections of row k to row ℓ and column i to column j. For a square matrix A, the set of all eigenvalues and the determinant are denoted by $\mathrm{eig}(A)$ and $\det(A)$, respectively. Finally, $\mathcal{K}_\ell(A,\boldsymbol{x})$ stands for the ℓth Krylov subspace and any $\boldsymbol{h} \in \mathcal{K}_\ell(A,\boldsymbol{x})$ can be expressed by $\boldsymbol{h} = p(A)\boldsymbol{x}$, where $p \in \mathbb{P}_\ell$ is a polynomial with degree no higher than ℓ.

2 The cubic regularization model

For the simplicity of presentation, we omit the subscript k in (1.2), and also denote

$$\boldsymbol{g} = \nabla f(\boldsymbol{x}_k),\ \sigma = M_k/2,\ H = H_k \approx \nabla^2 f(\boldsymbol{x}_k).$$

Thus, we focus on the following minimization:

$$\min_{\boldsymbol{h}\in\mathbb{R}^n} \left\{ m(\boldsymbol{h}) := \boldsymbol{g}^{\mathrm{T}}\boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}H\boldsymbol{h} + \frac{\sigma}{3}\|\boldsymbol{h}\|_2^3 \right\}. \tag{2.1}$$

The following result generalizes the well-known sufficient and necessary conditions (Gay [13] and Moré and Sorensen [26]) for the trust-region subproblem to (2.1).

Theorem 2.1. ([7, Theorem 3.1] and [27, Theorem 10]) *Any $\boldsymbol{h}_{\mathrm{opt}}$ is a global minimizer of (2.1) over $\mathbb{R}^n$ if and only if it satisfies the system of equations*

$$(H + \lambda_{\mathrm{opt}} I_n)\boldsymbol{h}_{\mathrm{opt}} = -\boldsymbol{g}, \tag{2.2}$$

where $\lambda_{\mathrm{opt}} = \sigma\|\boldsymbol{h}_{\mathrm{opt}}\|_2$ and $H + \lambda_{\mathrm{opt}} I_n$ is positive semidefinite. If $H + \lambda_{\mathrm{opt}} I_n$ is positive definite, then $\boldsymbol{h}_{\mathrm{opt}}$ is unique.

By Theorem 2.1, we know that $\lambda_{\mathrm{opt}} \geq -\theta_1^+$ where

$$-\theta_1^+ := \max(0, -\theta_1) \ \text{ and } \ \theta_1 = \theta_2 = \cdots = \theta_p < \theta_{p+1} \leq \cdots \leq \theta_n \tag{2.3}$$

are the ordered eigenvalues[2] of H and $H = U\Theta U^{\mathrm{T}}$ is its spectral decomposition with $U = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n]$ orthonormal. Notice that the value λ_{opt} plays a similar role with the Lagrangian

[1] A problem (1.2) is said to be ill-conditioned if the matrix $\nabla^2 f(\boldsymbol{x}_k) + \frac{M_k}{3}\|\boldsymbol{h}_{\mathrm{opt}}\|_2 I_n$ is ill-conditioned, where $\boldsymbol{h}_{\mathrm{opt}}$ is the minimizer of (1.2); see Theorem 4.1.

[2] In our discussion, without loss of generality, we assume that $p < n$. When $p = n$, then Theorem 2.1 implies that $\boldsymbol{h}_{\mathrm{opt}}$ is parallel to $\boldsymbol{g}$, and $\boldsymbol{h}_{\mathrm{opt}}$ can be obtained by solving a one-dimensional minimization.

multiplier for the trust-region subproblem; thus, in what follows, we will also call $\lambda_{\rm opt}$ as the Lagrangian multiplier for the problem (2.1). Now, by the fact $\lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2$, whenever $\lambda_{\rm opt} > -\theta_1$, $\lambda_{\rm opt}$ can be found via the system:

$$\lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2 = \sigma\|(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{g}\|_2,$$

or

$$\lambda_{\rm opt}^2 = \sigma^2 \boldsymbol{g}^{\rm T}(H + \lambda_{\rm opt} I_n)^{-2}\boldsymbol{g} = \boldsymbol{t}^{\rm T}(\Theta + \lambda_{\rm opt} I_n)^{-2}\boldsymbol{t} = \sum_{j=1}^{n} \frac{t_j^2}{(\theta_j + \lambda_{\rm opt})^2}, \tag{2.4}$$

where $\boldsymbol{t} = \sigma U^{\rm T}\boldsymbol{g} = [t_1, \ldots, t_n]^{\rm T}$. Introduce the system

$$q(\lambda) := \lambda^2 - \sum_{j=1}^{n} \frac{t_j^2}{(\theta_j + \lambda)^2}, \tag{2.5}$$

and we have $q(\lambda_{\rm opt}) = 0$.

We next present results for special cases: $\boldsymbol{g} = 0$ and $\lambda_{\rm opt} = \max(0, -\theta_1)$.

Theorem 2.2. *For (2.1) with $\sigma > 0$, we have*

(i) if $\boldsymbol{g} = \boldsymbol{0}$, then $\lambda_{\rm opt} = -\theta_1^+$;

(ii) $\lambda_{\rm opt} = 0$ if and only if $\boldsymbol{g} = \boldsymbol{0}$ and H is positive semidefinite;

(iii) $\lambda_{\rm opt} = -\theta_1$ if and only if

$$\boldsymbol{g} \perp \mathcal{E}_1 = {\rm span}([\boldsymbol{u}_1, \ldots, \boldsymbol{u}_p]) \quad \text{and} \quad -\theta_1 \geq \sigma\|(H - \theta_1 I_n)^{\dagger}\boldsymbol{g}\|_2, \tag{2.6}$$

where $\mathcal{E}_1$ is the eigenspace associated with the smallest eigenvalue $\theta_1 = \cdots = \theta_p$ of H.

Proof. For (i), we know that when $-\theta_1 \leq 0$, then by (2.2), the assumption $\lambda_{\rm opt} > 0$ leads to $\boldsymbol{h}_{\rm opt} = \boldsymbol{0}$, which contradicts with $0 < \lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2 = 0$. Therefore, when $-\theta_1 \leq 0$, we have $\lambda_{\rm opt} = \max(0, -\theta_1) = 0$. For $-\theta_1 > 0$, by a similar argument, we conclude $\lambda_{\rm opt} = -\theta_1^+ = -\theta_1$.

For (ii), if $\lambda_{\rm opt} = 0$, by $0 = \lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2$ and the semidefiniteness of $H + \lambda_{\rm opt} I_n$, we have $\boldsymbol{h}_{\rm opt} = \boldsymbol{0}$, and $\theta_1 \geq 0$; therefore, $\boldsymbol{0} = H\boldsymbol{h}_{\rm opt} = -\boldsymbol{g}$. The converse is from (i).

For (iii), we first consider the necessity for $\lambda_{\rm opt} = -\theta_1$. If $\theta_1 = 0$, then by (ii), we know that (2.6) holds. If $\lambda_{\rm opt} = -\theta_1 > 0$, then optimality condition (2.2) implies that $\boldsymbol{g} \perp \mathcal{E}_1$, and also all the solutions to the system (2.2) can be given by $\boldsymbol{h} = -(H - \theta_1 I_n)^{\dagger}\boldsymbol{g} + \boldsymbol{u}$ where $\boldsymbol{u} \in \mathcal{E}_1$ is arbitrary. Also, the condition $-\theta_1 = \lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2$ gives

$$\theta_1^2 = \sigma^2\left[\|(H - \theta_1 I_n)^{\dagger}\boldsymbol{g}\|_2^2 + \|\boldsymbol{u}\|_2^2\right] = \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i - \theta_1)^2} + \sigma^2\|\boldsymbol{u}\|_2^2, \text{ with } \boldsymbol{t} = \sigma U^{\rm T}\boldsymbol{g} = [t_1, \ldots, t_n]^{\rm T}.$$

Consider the function

$$\widetilde{q}(\lambda) = \lambda^2 - \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i + \lambda)^2}. \tag{2.7}$$

It is easy to see that $\widetilde{q}(\lambda) \to +\infty$ as $\lambda \to +\infty$, and $\widetilde{q}'(\lambda) > 0$ for $\lambda \in [-\theta_1, +\infty)$. Thus, $\widetilde{q}(-\theta_1) - \sigma^2\|\boldsymbol{u}\|_2^2 = 0$ for some $\boldsymbol{u}$ holds only if $-\theta_1 \geq \sigma\|(H - \theta_1 I_n)^{\dagger}\boldsymbol{g}\|_2$, and in this case, furthermore, the optimal solution $\boldsymbol{h}_{\rm opt} = -(H - \theta_1 I_n)^{\dagger}\boldsymbol{g} + \boldsymbol{u}$ where $\boldsymbol{u}$ is any eigenvector of H corresponding to θ_1 with $\|\boldsymbol{u}\|_2^2 = \theta_1^2/\sigma^2 - \|(H + \lambda I_n)^{\dagger}\boldsymbol{g}\|_2^2$.

For the sufficiency, $-\theta_1 \geq \sigma\|(H - \theta_1 I_n)^{\dagger}\boldsymbol{g}\|_2$ first implies that $-\theta_1 \geq 0$. Assume $\lambda_{\rm opt} > -\theta_1$. Then $\boldsymbol{h}_{\rm opt} = -(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{g}$ and $\boldsymbol{g} \perp \mathcal{E}_1$ lead to $\boldsymbol{h}_{\rm opt} \perp \mathcal{E}_1$, and therefore,

$$\theta_1^2 < \lambda_{\rm opt}^2 = \sigma^2\|\boldsymbol{h}_{\rm opt}\|_2^2 = \sigma^2\boldsymbol{g}^{\rm T}(H + \lambda_{\rm opt} I_n)^{-2}\boldsymbol{g} = \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i + \lambda_{\rm opt})^2} \leq \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i - \theta_1)^2}.$$

But $\sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i - \theta_1)^2} = \sigma^2\|(H - \theta_1 I_n)^{-1}\boldsymbol{g}\|_2^2$, and by assumption, we have $\theta_1^2 < \lambda_{\rm opt}^2 \leq \theta_1^2$, a contradiction. Thus we conclude $\lambda_{\rm opt} = -\theta_1$. □

By (iii) of Theorem 2.2, as in the trust-region subproblem [9, 1, 20, 41], we define the "hard" case and "easy" case for the cubic regularization model (2.1).

Definition 2.1. For (2.1) with $\sigma > 0$, we say it is "hard" case (or the degenerate case) if condition (2.6) holds. For the hard case, $\lambda_{\text{opt}} = -\theta_1$ and any optimal solution $\boldsymbol{h}_{\text{opt}}$ is of the form

$$\boldsymbol{h}_{\text{opt}} = -(H - \theta_1 I_n)^{\dagger}\boldsymbol{g} + \boldsymbol{u} \tag{2.8}$$

where $\boldsymbol{u}$ is any eigenvector of H corresponding to θ_1 with $\|\boldsymbol{u}\|_2^2 = \theta_1^2/\sigma^2 - \|(H + \lambda I_n)^{\dagger}\boldsymbol{g}\|_2^2$. The "easy" case (or the non-degenerate case) is characterized by the opposite of the hard case, and $\lambda_{\text{opt}} > \max(0, -\theta_1)$.

3 An associated quadratic eigenvalue problem (QEP)

Apart from Newton's iteration for computing the root λ_{opt} of $q(\lambda) = 0$, analogous to the trust-region subproblem [1], we can also translate (1.2) into an eigenvalue problem, for which accurate solution can be efficiently obtained when n is of small- to medium-size. In particular, by (2.4), we can transform the solution λ_{opt} into the following quadratic eigenvalue problem[3] (QEP):

$$G(\lambda) := \lambda^2 I_{n+1} + 2\lambda \begin{bmatrix} 0 & 0 \\ 0 & H \end{bmatrix} + \begin{bmatrix} 0 & \sigma\boldsymbol{g}^{\mathrm{T}} \\ \sigma\boldsymbol{g} & H^2 \end{bmatrix} = \begin{bmatrix} \lambda^2 & \sigma\boldsymbol{g}^{\mathrm{T}} \\ \sigma\boldsymbol{g} & (\lambda I_n + H)^2 \end{bmatrix}. \tag{3.1}$$

To see the relation more clearly, using again the spectral decomposition $H = U\Theta U^{\mathrm{T}}$ and denoting $K = \begin{bmatrix} 0 & 1 \\ U & 0 \end{bmatrix}$, we have

$$\begin{aligned} \widetilde{G}(\lambda) := K^{\mathrm{T}} G(\lambda) K &= \begin{bmatrix} U^{\mathrm{T}}(\lambda I + H)^2 U & \boldsymbol{t} \\ \boldsymbol{t}^{\mathrm{T}} & \lambda^2 \end{bmatrix} = \begin{bmatrix} (\lambda I + \Theta)^2 & \boldsymbol{t} \\ \boldsymbol{t}^{\mathrm{T}} & \lambda^2 \end{bmatrix} \\ &= \begin{bmatrix} (\lambda+\theta_1)^2 & & & t_1 \\ & \ddots & & \vdots \\ & & (\lambda+\theta_n)^2 & t_n \\ t_1 & \cdots & t_n & \lambda^2 \end{bmatrix}. \end{aligned}$$

Denote the eigenvalues of $G(\lambda)$ by $\text{eig}(G(\lambda))$ and $\text{eig}(H) = \{\theta_1, \ldots, \theta_n\}$. Noting for all $\lambda \notin \text{eig}(-H)$, we have the determinant

$$\begin{aligned} \det G(\lambda) = \det \widetilde{G}(\lambda) &= \det \begin{bmatrix} (\lambda I + \Theta)^2 & \boldsymbol{t} \\ \boldsymbol{t}^{\mathrm{T}} & \lambda^2 \end{bmatrix} \\ &= \det(\lambda I + \Theta)^2 \det(\lambda^2 - \boldsymbol{t}^{\mathrm{T}}(\lambda I + \Theta)^{-2}\boldsymbol{t}) \\ &= \left(\lambda^2 - \sum_{i=1}^{n} \frac{t_i^2}{(\lambda+\theta_i)^2}\right) \prod_{i=1}^{n} (\lambda+\theta_i)^2 \qquad (3.2) \\ &= q(\lambda) \prod_{i=1}^{n} (\lambda+\theta_i)^2 \qquad (3.3) \\ &= \lambda^2 \prod_{i=1}^{n} (\lambda+\theta_i)^2 - \sum_{i=1}^{n} t_i^2 \prod_{j\neq i} (\lambda+\theta_j)^2, \qquad (3.4) \end{aligned}$$

and (3.4) is also valid for all λ because (3.4) is a continuous function of λ. This implies that all the eigenvalues of $G(\lambda)$ which are not in $\text{eig}(-H)$ are the zeros of $q(\lambda) = 0$, i.e., the solutions to (2.4). This gives the connection between (2.4) and the QEP (3.1).

[3] A pair $(\lambda, \boldsymbol{x})$ with $\boldsymbol{x} \neq 0$ is an eigenpair of a polynomial eigenvalue problem $P(\lambda) = P_m\lambda^m + \cdots + P_1\lambda + P_0$ if $P(\lambda)\boldsymbol{x} = 0$. When $m = 2$ and $m = 1$, we say it is a quadratic and generalized eigenvalue problem, respectively.

3.1 The largest real eigenvalue of $G(\lambda)$ and the associated eigenvector

We next show that the Lagrangian multiplier $\lambda_{\rm opt}$ can be found by solving the largest real eigenvalue of the QEP (3.1).

Theorem 3.1. *For (2.1) with $\sigma > 0$, the Lagrangian multiplier $\lambda_{\rm opt}$ associated with the global optimal solution $\boldsymbol{h}_{\rm opt}$ of (2.1) is the largest real eigenvalue of $G(\lambda)$.*

Proof. We consider two cases.

Case I: $\boldsymbol{g} \not\perp \mathcal{E}_1$ where $\mathcal{E}_1$ given in (2.6) is the eigenspace associated with the smallest eigenvalue of H. In this case, we know that there exists at least one $t_i \neq 0$ for some $1 \leq i \leq p$, and thus by (3.4), for $\lambda > -\theta_1^+$, we have $\det(G(\lambda)) = q(\lambda)\prod_{i=1}^n(\lambda+\theta_i)^2$ where $q(\lambda)$ is defined in (2.5) and

$$q(\lambda) = \lambda^2 - \sum_{t_i\neq 0,\ 1\leq i\leq p} \frac{t_i^2}{(\theta_1+\lambda)^2} - \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i+\lambda)^2} \begin{cases} \to +\infty, & \text{as } \lambda \to +\infty \\ < 0 & \text{as } \lambda \to 0 \\ \to -\infty, & \text{as } \lambda \to -\theta_1 \end{cases}. \tag{3.5}$$

Therefore, together with the monotonicity of $q(\lambda)$ on $(-\theta_1^+, +\infty)$, we know that there is a unique solution in $(-\theta_1^+, +\infty)$ for $\det(G(\lambda)) = 0$, which by Theorem 2.2 is $\lambda_{\rm opt}$.

Case II: $\boldsymbol{g} \perp \mathcal{E}_1$. In this case, by (3.4), we know that $-\theta_1 \in \mathrm{eig}(G(\lambda))$, and also $t_i = 0$ for all $1 \leq i \leq p$. If $\boldsymbol{g} = \boldsymbol{0}$, then $\mathrm{eig}(G(\lambda)) = \{-\theta_1, -\theta_1, \ldots, -\theta_n, -\theta_n, 0\}$, and the largest eigenvalue is $-\theta_1^+$. According to Theorem 2.2 (i), the claim of this theorem is true.

For $\boldsymbol{g} \neq \boldsymbol{0}$, by (3.4), for $\lambda > -\theta_1^+$, we have $\det(G(\lambda)) = \widetilde{q}(\lambda)\prod_{i=1}^n(\lambda+\theta_i)^2$ where $\widetilde{q}(\lambda)$ is given by (2.7). Note that $\widetilde{q}(\lambda) \to +\infty$ as $\lambda \to +\infty$, and is monotonically increasing on $[-\theta_1^+, +\infty)$. If $-\theta_1 \leq 0$, then $-\theta_1^+ = 0$ and by noting $\widetilde{q}(0) < 0$, we know that there is a unique solution in $(-\theta_1^+, +\infty)$ for $\det(G(\lambda)) = 0$, which by Theorem 2.2 is $\lambda_{\rm opt}$. Otherwise, $-\theta_1 > 0$ and $-\theta_1^+ = -\theta_1$. Note that

$$\widetilde{q}(-\theta_1) = \theta_1^2 - \sum_{i=p+1}^{n} \frac{t_i^2}{(\theta_i-\theta_1)^2} = \theta_1^2 - \sigma^2\|(H-\theta_1 I_n)^\dagger \boldsymbol{g}\|_2^2.$$

Therefore, if $\widetilde{q}(-\theta_1) < 0$, then there is still a unique solution in $(-\theta_1^+, +\infty)$ for $\det(G(\lambda)) = 0$, which by Theorem 2.2 is just $\lambda_{\rm opt}$. But if $\widetilde{q}(-\theta_1) \geq 0$, then there is no solution in $(-\theta_1^+, +\infty)$ for $\det(G(\lambda)) = 0$, and the largest real eigenvalue of $G(\lambda)$ is therefore $-\theta_1$. The latter case corresponds to the hard case. □

We next show that the solution $\boldsymbol{h}_{\rm opt}$ of (2.1) can be obtained from the eigenpair $(\lambda_{\rm opt}, \boldsymbol{z})$ associated with the largest real eigenvalue $\lambda_{\rm opt}$ of $G(\lambda)$.

Theorem 3.2. *For (2.1) with $\sigma > 0$, suppose $\boldsymbol{z} = [\alpha; \boldsymbol{y}] \in \mathbb{R}^{n+1}$ is the normalized eigenvector of $G(\lambda)$ associated with the largest real eigenvalue $\lambda_{\rm opt}$. Then we have*

(i) if $\alpha \neq 0$, then $\boldsymbol{h}_{\rm opt} = (H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y})/(\sigma\alpha)$ is the solution to (2.1);

(ii) if $\alpha = 0$, then $\lambda_{\rm opt} = -\theta_1$, $\boldsymbol{y}$ is an eigenvector of H associated with $-\lambda_{\rm opt}$ and $\boldsymbol{h}_{\rm opt} = (H+\lambda_{\rm opt}I_n)^\dagger\boldsymbol{g} + \eta\boldsymbol{y}$ is the solution to (2.1), where

$$\eta = \pm\sqrt{\lambda_{\rm opt}^2 - \sigma^2\|(H+\lambda_{\rm opt}I_n)^\dagger\boldsymbol{g}\|_2^2}.$$

Proof. By $G(\lambda_{\rm opt})\boldsymbol{z} = \boldsymbol{0}$, we have

$$\lambda_{\rm opt}^2\alpha + \sigma\boldsymbol{g}^{\rm T}\boldsymbol{y} = 0, \quad \text{and} \quad \sigma\alpha\boldsymbol{g} + (H+\lambda_{\rm opt}I_n)^2\boldsymbol{y} = \boldsymbol{0}. \tag{3.6}$$

For (i), we first consider the case $\lambda_{\rm opt} > -\theta_1$, which by the second equation in (3.6) implies $\boldsymbol{h}_{\rm opt} = (H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y})/(\sigma\alpha)$ satisfies $(H+\lambda_{\rm opt}I_n)\boldsymbol{h}_{\rm opt} = -\boldsymbol{g}$; moreover, by (3.6) again, we know that $\sigma\|\boldsymbol{h}_{\rm opt}\|_2 = \lambda$, and according to Theorem 2.1, the claim is true. For the case $\lambda_{\rm opt} = -\theta_1$, Theorem 2.2 (iii) indicates that $\boldsymbol{g} \perp \mathcal{E}_1$, and thus, the second equation in (3.6) implies

$$-(H+\lambda_{\rm opt}I_n)^\dagger\boldsymbol{g} = (H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y})/(\sigma\alpha).$$

Moreover, by (3.6) and $\alpha \neq 0$, we have $\|(H+\lambda_{\rm opt} I_n)^\dagger \boldsymbol{g}\|_2 = -\theta_1/\sigma$, and hence $\boldsymbol{h}_{\rm opt} = (H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y})/(\sigma\alpha)$ is the solution to (2.1). We remark that this case ($\lambda_{\rm opt} = -\theta_1$ and $\alpha \neq 0$) only happens if $\boldsymbol{g} \perp \mathcal{E}_1$ and $-\theta_1 = \sigma\|(H-\theta_1 I_n)^\dagger \boldsymbol{g}\|_2$.

For (ii), we first know $\|\boldsymbol{y}\|_2 = 1$, and also (3.6) reduces to $\sigma \boldsymbol{g}^{\rm T}\boldsymbol{y} = 0$ and $(H+\lambda I_n)^2\boldsymbol{g} = \boldsymbol{0}$. The largest real eigenvalue of $G(\lambda)$ in this case must be $-\theta_1$ because otherwise $\boldsymbol{y} = \boldsymbol{0}$. Thus, $\boldsymbol{y} \in \mathcal{E}_1$ and by Theorem 2.2 (iii), we know that the solution in this case is $\boldsymbol{h}_{\rm opt} = -(H+\lambda_{\rm opt} I_n)^\dagger \boldsymbol{g} + \eta \boldsymbol{y}$ with η as claimed. □

In practice, similar to [1], when $|\alpha| \neq 0$ is close to zero, we can use

$$\boldsymbol{h}_{\rm opt} = {\rm sign}(\alpha) \left| \frac{\lambda_{\rm opt}}{\sigma} \right| \frac{H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y}}{\|H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y}\|_2}$$

instead of $\boldsymbol{h}_{\rm opt} = (H\boldsymbol{y} + \lambda_{\rm opt}\boldsymbol{y})/(\sigma\alpha)$ to compute the solution $\boldsymbol{h}_{\rm opt}$.

3.2 All the eigenvalues of $G(\lambda)$

We take a close look at the eigenvalues of $G(\lambda)$ in this subsection. We separate our discussion into two scenarios: the generic case when θ_i are distinct and the remaining special case.

3.2.1 The generic case: θ_i are distinct

In this situation, by (3.4), we have $\det(G(-\theta_i)) = -t_i^2 \prod_{j\neq i}(\theta_j - \theta_i)^2$, and it is clear that

$$t_i = 0 \Longleftrightarrow \det(G(-\theta_i)) = 0 \Longleftrightarrow -\theta_i \in {\rm eig}(G(\lambda)).$$

When all $t_i \neq 0$ for all $i = 1, 2, \ldots, n$, then ${\rm eig}(G(\lambda)) \cap {\rm eig}(-H) = \emptyset$, and moreover, $0 \notin {\rm eig}(G(\lambda))$, because, otherwise by (3.4), $\det(G(0)) = -\sum_{i=1}^n t_i^2 \prod_{i=1}^n \theta_i^2 \neq 0$, implying $0 \notin {\rm eig}(G(\lambda))$, a contradiction. Write $q(\lambda) = \lambda^2 \delta(\lambda)$, where

$$\delta(\lambda) := 1 - \sum_{i=1}^n \frac{t_i^2}{\lambda^2(\lambda+\theta_i)^2}, \quad \text{and} \quad \delta'(\lambda) = 2\sum_{i=1}^n \frac{(2\lambda+\theta_i)t_i^2}{\lambda^3(\lambda+\theta_i)^3}.$$

According to (3.3), the zeros of $\delta(\lambda)$ are the eigenvalues of $G(\lambda)$. To have a clear picture of the distribution of the eigenvalues, based on the signs of $\delta(\lambda)$ and $\delta'(\lambda)$, we draw a diagram of the function $\delta(\lambda)$ in Figure 3.1 as an illustration for the case $\theta_1 > 0$. In this illustration, the branches in the intervals $(-\theta_n, -\theta_{n-1}), (-\theta_{n-1}, -\theta_{n-2}), (-\theta_1, 0)$ represent three types of eigenvalues respectively: two different real eigenvalues, two same real eigenvalues, and two conjugate complex eigenvalues. Together with $(-\infty, -\theta_n)$ and $(0, \infty)$, these $n+2$ intervals contain $2n+2$ zeros in total. Clearly there exists unique real eigenvalue of $G(\lambda)$ larger than $\max(0, -\theta_1)$, and according to Theorem 2.1, this real eigenvalue is just $\lambda_{\rm opt}$. We remark that this case, i.e., $t_i \neq 0$ for all i, is the easy case of (2.1) by Definition 2.1.

For the case when $t_i = 0$ for $i \in \mathcal{I} \subset \{1, \ldots, n\}$, we know that $-\theta_i$ is an eigenvalue of $G(\lambda)$. To obtain other eigenvalues, we consider $\widetilde{G}_{\mathcal{I}^c}(\lambda) = I_{\mathcal{I}^c}^{\rm T} \widetilde{G}(\lambda) I_{\mathcal{I}^c}$, where $I_{\mathcal{I}^c}$ is a matrix consisting of $\boldsymbol{e}_j$ for $j \in \{1, \ldots, n\} \setminus \mathcal{I}$. The eigenvalues of $\widetilde{G}_{\mathcal{I}^c}(\lambda)$ can be treated in the case above. If $|\mathcal{I}| = n$, namely $\boldsymbol{g} = \boldsymbol{t} = 0$ (implying that 0 is also an eigenvalue of $G(\lambda)$), there is no real eigenvalue of $G(\lambda)$ larger than $\max(0, -\theta_1)$, and according to Theorem 2.1, $\lambda_{\rm opt} = \max(0, -\theta_1)$. Otherwise, $|\mathcal{I}| < n$, and there exists a unique real eigenvalue, say χ, of $G(\lambda)$ larger than $\max(0, -\theta_i)_{i\notin\mathcal{I}}$. For this case, if $1 \notin \mathcal{I}$, then χ is the largest real eigenvalue of $G(\lambda)$ which is just $\lambda_{\rm opt}$ (the easy case) according to Theorem 2.1, while $1 \in \mathcal{I}$, χ and $-\theta_1$ are two eigenvalues. The latter case leads to the easy case when $\chi > -\theta_1$ and the hard case $\chi \leq -\theta_1$ by Definition 2.1.

3.2.2 The special case

We now assume that the distinct values of θ_i are $\mu_1, \ldots, \mu_m$ with $m < n$. First we consider the subset $\mathcal{I}_1 \subset \{1, \ldots, n\}$ where $\theta_i = \mu_1$ for any $i \in \mathcal{I}_1$.

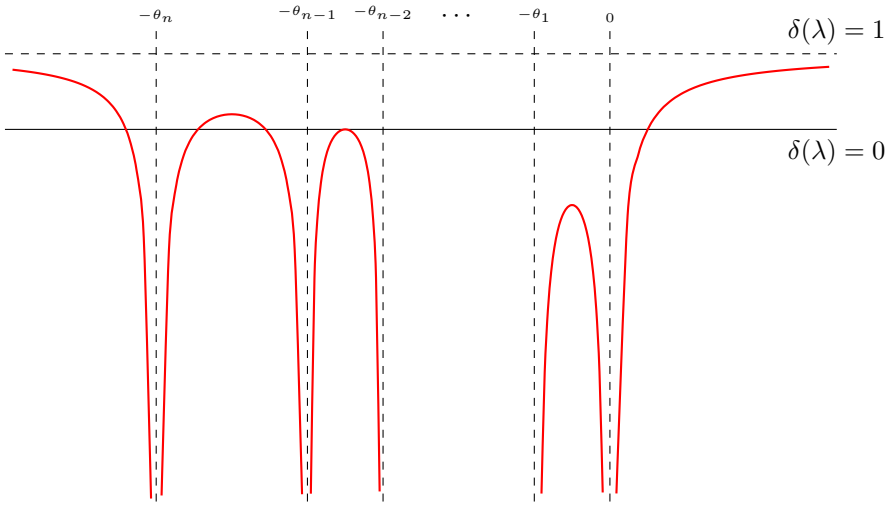

Figure 3.1: $\delta(\lambda)$ on the case $\theta_1 > 0$

When $t_i = 0$ for all $i \in \mathcal{I}_1$, then by (3.2)

$$\det(G(\lambda)) = \left(\lambda^2 - \sum_{i\notin\mathcal{I}_1} \frac{t_i^2}{(\lambda+\theta_i)^2}\right)(\lambda+\mu_1)^{2|\mathcal{I}_1|} \prod_{i\notin\mathcal{I}_1}(\lambda+\theta_i)^2.$$

Thus, $-\mu_1$ is an eigenvalue of $G(\lambda)$ with (algebraic) multiplicity $2|\mathcal{I}_1|$. Analogous to the generic case, the other eigenvalues can be obtained by considering $\widetilde{G}_{\mathcal{I}_1^c}(\lambda) = I_{\mathcal{I}_1^c}^{\mathrm{T}} \widetilde{G}(\lambda) I_{\mathcal{I}_1^c}$.

Otherwise (i.e., $\sum_{i\in\mathcal{I}_1} t_i^2 \neq 0$), we have by (3.2) that

$$\det(G(\lambda)) = \left(\lambda^2(\lambda+\mu_1)^2 - \sum_{i\notin\mathcal{I}_1} \frac{(\lambda+\mu_1)^2 t_i^2}{(\lambda+\theta_i)^2} - \sum_{i\in\mathcal{I}_1} t_i^2\right)(\lambda+\mu_1)^{2(|\mathcal{I}_1|-1)} \prod_{i\notin\mathcal{I}_1}(\lambda+\theta_i)^2.$$

Thus, $-\mu_1$ is an eigenvalue of $G(\lambda)$ with (algebraic) multiplicity $2(|\mathcal{I}_1|-1)$, and the other eigenvalues can be obtained by considering $\widetilde{G}_{\widetilde{\mathcal{I}}_1^c}(\lambda) = \begin{bmatrix} (\lambda+\mu_1)^2 & s_1 \boldsymbol{e}_n^{\mathrm{T}} \\ s_1 \boldsymbol{e}_n & I_{\mathcal{I}_1^c}^{\mathrm{T}} \widetilde{G}(\lambda) I_{\mathcal{I}_1^c} \end{bmatrix}$, where $s_1^2 = \sum_{i\in\mathcal{I}_1} t_i^2$.

The above arguments can be continuously applied to $\mu_2, \ldots, \mu_m$ to obtain all the eigenvalues of $G(\lambda)$ and the details are omitted.

3.3 Associated generalized eigenvalue problems

In a recent work by Lieder [25], it is shown that the optimal λ_{opt} is the largest real eigenvalue of the following generalized eigenvalue problem:

$$\mathcal{M}(\lambda) = \begin{bmatrix} 0 & 0 & 0 & -\boldsymbol{g}^{\mathrm{T}} \\ 0 & \sigma I_n & 0 & -H \\ 0 & 0 & \sigma & 0 \\ -\boldsymbol{g} & -H & 0 & 0 \end{bmatrix} - \lambda \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & I_n \\ 1 & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 \end{bmatrix} \in \mathbb{R}^{2(n+1)\times 2(n+1)}. \tag{3.7}$$

It is known that a QEP can be linearized to various types of generalized eigenvalue problem (see e.g., [36]). Thus, our QEP (3.1) can lead us to other generalized eigenvalue problems. For example,

$$\mathcal{L}(\lambda) = \begin{bmatrix} -A & 0 \\ 0 & I_{n+1} \end{bmatrix} - \lambda \begin{bmatrix} B & I_{n+1} \\ I_{n+1} & 0 \end{bmatrix} =: \mathcal{A} - \lambda\mathcal{B} \in \mathbb{R}^{2(n+1)\times 2(n+1)} \tag{3.8}$$

where

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 2H \end{bmatrix}, \quad A = \begin{bmatrix} 0 & \sigma \boldsymbol{g}^{\mathrm{T}} \\ \sigma \boldsymbol{g} & H^2 \end{bmatrix}.$$

We will see in the next theorem that the eigenpair of $G(\lambda)$ associated with λ_{opt} can be equivalently obtained via a generalized eigenvalue problem.

Theorem 3.3. ([24, Theorem 4.1]) *Let $\mathcal{L}(\lambda)$ be given by (3.8), then we have*

(1) The set of eigenvalues of $G(\lambda)$ is the same as that of the matrix pencil $\mathcal{A} - \lambda\mathcal{B}$.

(2) The inertia (i.e., the number of the positive, zero, and negative eigenvalues respectively) of $\mathcal{B}$ is $(n+1, 0, n+1)$.

(3) If $(\lambda, \boldsymbol{z})$ is an eigenpair of $G(\lambda)$, then $\left(\lambda, \begin{bmatrix} \boldsymbol{z} \\ \lambda \boldsymbol{z} \end{bmatrix}\right)$ is an eigenpair of $\mathcal{L}(\lambda)$.

(4) If $\left(\lambda, \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{t} \end{bmatrix}\right)$ is an eigenpair of $\mathcal{L}(\lambda)$, then $(\lambda, \boldsymbol{z})$ is an eigenpair of $G(\lambda)$ and $\boldsymbol{t} = \lambda \boldsymbol{z}$.

Remark 3.1. It should be pointed out that even though both (3.7) and (3.8) are $2(n+1)$ dimensional generalized eigenvalue problems, (3.7) is preferable as it does not involve H^2. However, our QEP (3.1) has the following advantages.

1) The QEP (3.1) is of dimension $(n+1)$ and efficient Krylov subspace methods working on $\mathbb{R}^{n+1}$ such as the second-order Arnoldi process (SOAR) [2] can be directly applied to solve (1.2).

2) The QEP (3.1) is more flexible. First, there are various types of generalized eigenvalue problems that can be derived from (3.1) by linearization [36]. For instance, by taking the advantage of the coefficient matrix I_{n+1} in the term λ^2 in (3.1), another commonly used linearization in the literature [36] leads to the following standard eigenvalue problem:

$$C\boldsymbol{y} = \lambda \boldsymbol{y}, \quad C = \begin{bmatrix} -B & -A \\ I_{n+1} & 0 \end{bmatrix}, \ \boldsymbol{y} = \begin{bmatrix} \lambda \boldsymbol{x} \\ \boldsymbol{x} \end{bmatrix}.$$

Secondly, in many applications of the cubic regularization of Newton method for (1.1), the BFGS update [28] will be used to approximate the inverse of the Hessian matrix, where only $\widehat{H} = H^{-1}$ is available. In this situation, by noting

$$\det\left(\underbrace{\begin{bmatrix} 1 & \\ & H^{-1} \end{bmatrix} G(\lambda) \begin{bmatrix} 1 & \\ & H^{-1} \end{bmatrix}}_{=:\widehat{G}(\lambda)}\right) = \det(H^{-2}) \cdot \det(G(\lambda)),$$

we have a QEP involving just $\widehat{H}$:

$$\widehat{G}(\lambda) = \lambda^2 \begin{bmatrix} 0 & 0 \\ 0 & \widehat{H}^2 \end{bmatrix} + 2\lambda \begin{bmatrix} 0 & 0 \\ 0 & \widehat{H} \end{bmatrix} + \begin{bmatrix} 0 & \sigma \boldsymbol{g}^{\mathrm{T}} \widehat{H} \\ \sigma \widehat{H} \boldsymbol{g} & I_n \end{bmatrix},$$

and the optimal λ_{opt} can be computed by finding the largest real eigenvalue of $\widehat{G}(\lambda)$.

4 A Lanczos method

We next discuss a Lanczos type procedure introduced in [7, Section 6.2] for solving (1.2). The approach is analogous to the generalized Lanczos trust-region (GLTR) method proposed in [15] for the trust-region problem. GLTR is an efficient Lanczos type method for large-scale minimization

problems and its convergence analysis was recently established in [41, 42] and efficient restarting techniques are developed in [40]. For the cubic regularization model (2.1), the approach begins with forming an ℓ-th Krylov subspace $\mathcal{K}_\ell(H, \boldsymbol{g}) = \text{span}(Q_\ell)$ via the standard Lanczos process (Algorithm 1) for ℓ less than the grade τ of $\boldsymbol{g}$ with respect to H (i.e., τ is the smallest number that the Lanczos process breaks at step 6), and we have

$$HQ_\ell = Q_\ell S_\ell + \gamma_\ell \boldsymbol{q}_{\ell+1} \boldsymbol{e}_\ell^{\mathrm{T}}, \quad Q_\ell \boldsymbol{e}_1 = \boldsymbol{g}/\|\boldsymbol{g}\|_2, \tag{4.1}$$

where

$$S_\ell = Q_\ell^{\mathrm{T}} H Q_\ell = \begin{bmatrix} \delta_0 & \gamma_1 & & \\ \gamma_1 & \delta_1 & \ddots & \\ & \ddots & \ddots & \gamma_\ell \\ & & \gamma_\ell & \delta_\ell \end{bmatrix}$$

is tridiagonal, $Q_\ell = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_\ell]$ is orthogonal.

Algorithm 1 The standard Lanczos Process

It computes an orthogonal basis matrix Q_ℓ of $\mathcal{K}_\ell(H, \boldsymbol{g})$.

1: set $\boldsymbol{q}_0 = 0$, $\gamma_0 = 0$, $\boldsymbol{q}_1 = \boldsymbol{g}/\|\boldsymbol{g}\|_2$;
2: **for** $j = 1, 2, \ldots, \ell$ **do**
3: $\quad \boldsymbol{t} = H\boldsymbol{q}_j$, $\delta_{j-1} = \boldsymbol{q}_j^{\mathrm{T}} \boldsymbol{t}$;
4: $\quad \boldsymbol{t} = \boldsymbol{t} - \delta_{j-1}\boldsymbol{q}_j - \gamma_{j-1}\boldsymbol{q}_{j-1}$, $\gamma_j = \|\boldsymbol{t}\|_2$;
5: $\quad$ **if** $\gamma_j = 0$ **then**
6: $\quad\quad$ **break**;
7: $\quad$ **else**
8: $\quad\quad \boldsymbol{q}_{j+1} = \boldsymbol{t}/\gamma_j$;
9: $\quad$ **end if**
10: **end for**

The approximation $\boldsymbol{h}_\ell \in \mathbb{R}^n$ at the ℓ-th step is obtained by

$$\boldsymbol{h}_\ell = \underset{\boldsymbol{h} \in \mathcal{K}_\ell(H, \boldsymbol{g})}{\text{argmin}} \; m(\boldsymbol{h}), \tag{4.2}$$

which, by denoting $\boldsymbol{h} \in \mathcal{K}_\ell(H, \boldsymbol{g})$ by $\boldsymbol{h} = Q_\ell \boldsymbol{s}$ with $\boldsymbol{s} \in \mathbb{R}^\ell$, can be solved equivalently as the following smaller sized cubic regularization model:

$$\min_{\boldsymbol{s} \in \mathbb{R}^\ell} \{\|\boldsymbol{g}\|_2 \boldsymbol{e}_1^{\mathrm{T}} \boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}} S_\ell \boldsymbol{s} + \frac{\sigma}{3}\|\boldsymbol{s}\|_2^3\}. \tag{4.3}$$

Denote the solution of (4.3) by $\boldsymbol{s}_{\text{opt},\ell}$, which can be obtained by solving the associated eigenvalue problem discussed in Theorem 3.2. Also, we have $\boldsymbol{h}_\ell = Q_\ell \boldsymbol{s}_{\text{opt},\ell}$, and the corresponding Lagrangian multiplier is

$$\lambda_\ell = \sigma\|\boldsymbol{h}_\ell\|_2 = \sigma\|\boldsymbol{s}_{\text{opt},\ell}\|_2. \tag{4.4}$$

4.1 Convergence

To establish the convergence results of $\{m(\boldsymbol{h}_\ell)\}$ and $\{\boldsymbol{h}_\ell\}$, we first have the following lemmas.

Lemma 4.1. *Suppose $\boldsymbol{g} \not\perp \mathcal{E}_1$ where $\mathcal{E}_1$ is the eigenspace of H associated with the smallest eigenvalue θ_1. Then we have*

(i) Before the breakdown at step 6 in Algorithm 1, it holds that $S_\ell + \lambda_\ell I_\ell \succ 0$, where S_ℓ is the tridiagonal matrix given in (4.1) and λ_ℓ is given by (4.4);

(ii) If the standard Lanczos process (Algorithm 1) breaks at the τ-th step, then $\lambda_{\text{opt}} = \lambda_\tau$ and $\boldsymbol{h}_\tau = \boldsymbol{h}_{\text{opt}}$.

Proof. For i), we note that S_ℓ is irreducible, and then by [15, Theorem 5.3] (see also [29, Theorem 7.9.5]), any eigenvector $\boldsymbol{v}$ of S_ℓ satisfies $\boldsymbol{v}^{\mathrm{T}}\boldsymbol{e}_1 \neq 0$. Thus, by Theorem 2.2 (iii), we know that (4.3) is an easy case and the associated λ_ℓ is strictly larger than the smallest eigenvalue of S_ℓ.

We prove ii) by showing that $\lambda_{\mathrm{opt}} = \lambda_\tau$. According to Theorem 2.1, the assumption of $\boldsymbol{g} \not\perp \mathcal{E}_1$ and the arguments in (3.5), it follows that λ_{opt} and λ_τ are the largest real roots to the systems

$$q(\lambda) = \lambda^2 - \sigma^2\|(H+\lambda I_n)^{-1}\boldsymbol{g}\|_2^2 = 0, \quad \text{and} \quad \widehat{q}(\lambda) = \lambda^2 - \sigma^2\|\boldsymbol{g}\|_2\|(S_\tau+\lambda I_\tau)^{-1}\boldsymbol{e}_1\|_2^2 = 0,$$

respectively. Indeed, in this case we can show that $q(\lambda) = \widehat{q}(\lambda)$ for any $\lambda \notin \mathrm{eig}(H)$, and hence $\lambda_{\mathrm{opt}} = \widehat{\lambda}_\tau$. To this end, by assumptions, we know that $\mathrm{span}(Q_\tau) = \mathcal{K}_\tau(H,\boldsymbol{g})$ is an invariant subspace of H (implying $\mathrm{eig}(S_\tau) \subseteq \mathrm{eig}(H)$), which contains the eigenvectors associated with the smallest eigenvalue θ_1 of H. So the smallest eigenvalue of S_τ is θ_1 and $HQ_\tau = Q_\tau S_\tau$ leading to

$$(H+\lambda I_n)Q_\tau = Q_\tau(S_\tau+\lambda I_\tau).$$

Thus for any $\lambda \notin \mathrm{eig}(H)$, we get from $Q_\tau\boldsymbol{e}_1 = \boldsymbol{g}/\|\boldsymbol{g}\|_2$ that

$$(H+\lambda I_n)^{-1}\boldsymbol{g} = (H+\lambda I_n)^{-1}Q_\tau\boldsymbol{e}_1\|\boldsymbol{g}\|_2 = Q_\tau(S_\tau+\lambda I_\tau)^{-1}\boldsymbol{e}_1\|\boldsymbol{g}\|_2$$

which leads to $q(\lambda) = \widehat{q}(\lambda)$, and the conclusion follows. □

The linear convergence of λ_ℓ in (4.4) to λ_{opt} has been previously discussed in [5, 6], in which the proof follows similarly as that for the trust-region subproblem. Here, following the argument in [41] for the trust-region subproblem, we provide a different way which can render sharper bounds (refer to Remark 4.2) for the approximate objective function value as well as the solution.

Lemma 4.2. *Under the assumption of Lemma 4.1, let $\mathcal{L}$ be any subspace of $\mathbb{R}^n$ with $\dim(\mathcal{L}) \geq 1$ and $\boldsymbol{c}$ be the solution to $\min_{\boldsymbol{h}\in\mathcal{L}} m(\boldsymbol{h})$. Then for any nonzero vector $\boldsymbol{h} \in \mathcal{L}$, we have*

$$0 \leq m(\boldsymbol{c}) - m(\boldsymbol{h}_{\mathrm{opt}}) \leq 2\|H_{\mathrm{opt}}\|_2\|\boldsymbol{h}-\boldsymbol{h}_{\mathrm{opt}}\|_2^2, \tag{4.5}$$

$$\|\boldsymbol{c}-\boldsymbol{h}_{\mathrm{opt}}\|_2 \leq 2\sqrt{\kappa}\|\boldsymbol{h}-\boldsymbol{h}_{\mathrm{opt}}\|_2, \tag{4.6}$$

where $H_{\mathrm{opt}} = H + \lambda_{\mathrm{opt}} I_n$ and $\kappa = \frac{\theta_n+\lambda_{\mathrm{opt}}}{\theta_1+\lambda_{\mathrm{opt}}}$.

Proof. Let $\boldsymbol{h} \in \mathcal{L}$ be any nonzero vector and let $\boldsymbol{v} = \boldsymbol{h}\frac{\lambda_{\mathrm{opt}}}{\|\boldsymbol{h}\|_2\sigma} \in \mathcal{L}$. Define $\boldsymbol{m} = \boldsymbol{v} - \boldsymbol{h}_{\mathrm{opt}}$. Since $\boldsymbol{c}$ is the minimizer, we have

$$\begin{aligned}
0 &\leq m(\boldsymbol{c}) - m(\boldsymbol{h}_{\mathrm{opt}}) \leq m(\boldsymbol{v}) - m(\boldsymbol{h}_{\mathrm{opt}}) \\
&= \boldsymbol{g}^{\mathrm{T}}\boldsymbol{m} + \boldsymbol{m}^{\mathrm{T}}H\boldsymbol{h}_{\mathrm{opt}} + \frac{1}{2}\boldsymbol{m}^{\mathrm{T}}H\boldsymbol{m} + \frac{\sigma}{3}(\|\boldsymbol{v}\|_2^3 - \|\boldsymbol{h}_{\mathrm{opt}}\|_2^3) \\
&= -\lambda_{\mathrm{opt}}\boldsymbol{m}^{\mathrm{T}}\boldsymbol{h}_{\mathrm{opt}} + \frac{1}{2}\boldsymbol{m}^{\mathrm{T}}H\boldsymbol{m}, \quad (H\boldsymbol{h}_{\mathrm{opt}} = -\boldsymbol{g} - \lambda_{\mathrm{opt}}\boldsymbol{h}_{\mathrm{opt}},\ \|\boldsymbol{v}\|_2 = \|\boldsymbol{h}_{\mathrm{opt}}\|_2) \\
&= \frac{1}{2}\boldsymbol{m}^{\mathrm{T}}(H+\lambda_{\mathrm{opt}}I_n)\boldsymbol{m} \\
&\leq \frac{1}{2}\|H_{\mathrm{opt}}\|_2\|\boldsymbol{m}\|_2^2,
\end{aligned} \tag{4.7}$$

where the last equality is due to the fact $\boldsymbol{m}^{\mathrm{T}}\boldsymbol{h}_{\mathrm{opt}} = -\|\boldsymbol{m}\|_2^2/2$ which follows from

$$\boldsymbol{h}_{\mathrm{opt}}^{\mathrm{T}}\boldsymbol{h}_{\mathrm{opt}} = \boldsymbol{v}^{\mathrm{T}}\boldsymbol{v} = (\boldsymbol{h}_{\mathrm{opt}}+\boldsymbol{m})^{\mathrm{T}}(\boldsymbol{h}_{\mathrm{opt}}+\boldsymbol{m}) = \boldsymbol{h}_{\mathrm{opt}}^{\mathrm{T}}\boldsymbol{h}_{\mathrm{opt}} + 2\boldsymbol{m}^{\mathrm{T}}\boldsymbol{h}_{\mathrm{opt}} + \|\boldsymbol{m}\|_2^2.$$

Furthermore, $\|\boldsymbol{m}\|_2 = \|\boldsymbol{v}-\boldsymbol{h}_{\mathrm{opt}}\|_2 \leq \|\boldsymbol{v}-\boldsymbol{h}\|_2 + \|\boldsymbol{h}-\boldsymbol{h}_{\mathrm{opt}}\|_2$, and

$$\begin{aligned}
\|\boldsymbol{v}-\boldsymbol{h}\|_2 &= \left\|\boldsymbol{h} - \boldsymbol{h}\frac{\lambda_{\mathrm{opt}}}{\|\boldsymbol{h}\|_2\sigma}\right\|_2 \\
&= \|\boldsymbol{h}\|_2 \cdot \left|1 - \frac{\lambda_{\mathrm{opt}}}{\|\boldsymbol{h}\|_2\sigma}\right|_2
\end{aligned}$$

$$
\begin{aligned}
&= \left\| \|\boldsymbol{h}\|_2 - \frac{\lambda_{\rm opt}}{\sigma} \right\|_2 \\
&= \left\| \|\boldsymbol{h}\|_2 - \|\boldsymbol{h}_{\rm opt}\|_2 \right\|_2, \quad (\|\boldsymbol{h}_{\rm opt}\|_2 = \frac{\lambda_{\rm opt}}{\sigma}) \\
&\le \|\boldsymbol{h} - \boldsymbol{h}_{\rm opt}\|_2.
\end{aligned}
$$

Thus, $\|\boldsymbol{m}\|_2 \le 2\|\boldsymbol{h} - \boldsymbol{h}_{\rm opt}\|_2$, and together with (4.7), the claim (4.5) follows.

Now we consider (4.6). Denote by $Q_{\mathcal{L}}$ an orthonormal basis of $\mathcal{L}$. Let

$$
\boldsymbol{c} = Q_{\mathcal{L}}\widetilde{\boldsymbol{h}}_{\mathcal{L}} = \operatorname*{argmin}_{\boldsymbol{h}\in\mathcal{L}} m(\boldsymbol{h}), \text{ where } \widetilde{\boldsymbol{h}}_{\mathcal{L}} = \operatorname*{argmin}_{\widetilde{\boldsymbol{h}}\in\mathbb{R}^{\dim(\mathcal{L})}} m(Q_{\mathcal{L}}\widetilde{\boldsymbol{h}}).
$$

By Theorem 2.1, we know

$$
(Q_{\mathcal{L}}^{\rm T} H Q_{\mathcal{L}} + \lambda_{\mathcal{L}} I)\widetilde{\boldsymbol{h}}_{\mathcal{L}} = -Q_{\mathcal{L}}^{\rm T}\boldsymbol{g}, \ \lambda_{\mathcal{L}} = \sigma\|\widetilde{\boldsymbol{h}}_{\mathcal{L}}\|_2, \ Q_{\mathcal{L}}^{\rm T} H Q_{\mathcal{L}} + \lambda_{\mathcal{L}} I \succeq 0,
$$

where $\lambda_{\mathcal{L}}$ is the corresponding Lagrangian multiplier satisfying $\lambda_{\mathcal{L}} = \sigma\|\boldsymbol{c}\|_2$. Since

$$
\begin{aligned}
& m(\boldsymbol{c}) - m(\boldsymbol{h}_{\rm opt}) \\
&= \boldsymbol{g}^{\rm T}(\boldsymbol{c} - \boldsymbol{h}_{\rm opt}) + \frac{1}{2}\boldsymbol{c}^{\rm T} H \boldsymbol{c} - \frac{1}{2}\boldsymbol{h}_{\rm opt}^{\rm T} H \boldsymbol{h}_{\rm opt} + \frac{\sigma}{3}\|\boldsymbol{c}\|_2^3 - \frac{\sigma}{3}\|\boldsymbol{h}_{\rm opt}\|_2^3 \\
&= -\boldsymbol{h}_{\rm opt}^{\rm T}(H + \lambda_{\rm opt} I)(\boldsymbol{c} - \boldsymbol{h}_{\rm opt}) + \frac{1}{2}\boldsymbol{c}^{\rm T}(H + \lambda_{\rm opt} I)\boldsymbol{c} - \frac{1}{2}\boldsymbol{h}_{\rm opt}^{\rm T}(H + \lambda_{\rm opt} I)\boldsymbol{h}_{\rm opt} \\
&\qquad - \frac{1}{2}\lambda_{\rm opt}\|\boldsymbol{c}\|_2^2 + \frac{1}{2}\lambda_{\rm opt}\|\boldsymbol{h}_{\rm opt}\|^2 + \frac{\sigma}{3}\|\boldsymbol{c}\|_2^3 - \frac{\sigma}{3}\|\boldsymbol{h}_{\rm opt}\|_2^3 \\
&= \frac{1}{2}(\boldsymbol{c} - \boldsymbol{h}_{\rm opt})^{\rm T}(H + \lambda_{\rm opt} I)(\boldsymbol{c} - \boldsymbol{h}_{\rm opt}) \\
&\qquad - \frac{1}{2}\lambda_{\rm opt}\left(\frac{\lambda_{\mathcal{L}}}{\sigma}\right)^2 + \frac{1}{2}\lambda_{\rm opt}\left(\frac{\lambda_{\rm opt}}{\sigma}\right)^2 + \frac{\sigma}{3}\left(\frac{\lambda_{\mathcal{L}}}{\sigma}\right)^3 - \frac{\sigma}{3}\left(\frac{\lambda_{\rm opt}}{\sigma}\right)^3 \\
&= \frac{1}{2}(\boldsymbol{c} - \boldsymbol{h}_{\rm opt})^{\rm T}(H + \lambda_{\rm opt} I)(\boldsymbol{c} - \boldsymbol{h}_{\rm opt}) + \frac{1}{6\sigma^2}(\lambda_{\rm opt}^3 - 3\lambda_{\rm opt}\lambda_{\mathcal{L}}^2 + 2\lambda_{\mathcal{L}}^3) \\
&= \frac{1}{2}(\boldsymbol{c} - \boldsymbol{h}_{\rm opt})^{\rm T}(H + \lambda_{\rm opt} I)(\boldsymbol{c} - \boldsymbol{h}_{\rm opt}) + \frac{1}{6\sigma^2}(\lambda_{\rm opt} - \lambda_{\mathcal{L}})^2(\lambda_{\rm opt} + 2\lambda_{\mathcal{L}}) \\
&\ge \frac{1}{2}(\theta_1 + \lambda_{\rm opt})\|\boldsymbol{c} - \boldsymbol{h}_{\rm opt}\|_2^2,
\end{aligned}
$$

together with (4.5), we have (4.6). □

Lemma 4.3 (Bernstein [3]). *Given $\phi > 1$, the best approximating polynomial $p_\ell(x) \in \mathbb{P}_\ell$ of $\frac{1}{x-\phi}$ in $[-1, 1]$ satisfies*

$$
\frac{1}{x - \phi} - p_\ell(x) = \frac{\left(\phi + \sqrt{\phi^2 - 1}\right)^{-\ell}}{\phi^2 - 1}\cos(\ell\alpha + \beta),
$$

where $\mathbb{P}_\ell$ denotes the set of all polynomials with degree no higher than ℓ, α and β are such that $x = \cos\alpha$ and $\frac{\phi x - 1}{x - \phi} = \cos\beta$, and moreover,

$$
\epsilon_\ell(\phi) := \min_{\wp\in\mathbb{P}_\ell} \max_{-1\le x\le 1} \left| \wp(x) - \frac{1}{x - \phi} \right| = \frac{\left(\phi + \sqrt{\phi^2 - 1}\right)^{-\ell}}{\phi^2 - 1}. \tag{4.8}
$$

With these preliminary results, we are able to show the convergence of the Lanczos approach (4.2).

Theorem 4.1. *Suppose $\boldsymbol{g} \not\perp \mathcal{E}_1$ where $\mathcal{E}_1$ is the eigenspace of H associated with the smallest eigenvalue θ_1 and $\boldsymbol{h}_{\rm opt}$ is the minimizer of (2.1). Let $\boldsymbol{h}_\ell$ be the solution to (4.2). Then $\boldsymbol{h}_\tau = \boldsymbol{h}_{\rm opt}$ where τ is the grade of $\boldsymbol{g}$ with respect to H, and for any $1 \le \ell < \tau$, we have*

$$
0 \le m(\boldsymbol{h}_\ell) - m(\boldsymbol{h}_{\rm opt}) \le 2\|H_{\rm opt}\|_2\zeta_\ell^2, \tag{4.9}
$$

$$\|\boldsymbol{h}_\ell - \boldsymbol{h}_{\rm opt}\|_2 \le 2\sqrt{\kappa}\zeta_\ell, \tag{4.10}$$

where $H_{\rm opt} = H + \lambda_{\rm opt} I_n$,

$$\zeta_\ell = \frac{2\|\boldsymbol{g}\|_2 \epsilon_\ell(\phi)}{\theta_n - \theta_1}, \tag{4.11}$$

$\epsilon_\ell(\phi)$ *is defined by* (4.8) *and*

$$\phi = \frac{\kappa+1}{\kappa-1} = 1 + 2\frac{\theta_1 + \lambda_{\rm opt}}{\theta_n - \theta_1} > 1 \tag{4.12}$$

with $\kappa = (\theta_n + \lambda_{\rm opt})/(\theta_1 + \lambda_{\rm opt})$ *being the condition number of* $H_{\rm opt}$.

Proof. We apply Lemma 4.2 with $\mathcal{L} = \mathcal{K}_\ell(H, \boldsymbol{g})$ and $\boldsymbol{c} = \boldsymbol{h}_\ell$. In particular, we search a vector $\boldsymbol{h} \in \mathcal{K}_\ell(H, \boldsymbol{g})$ that is closest to $\boldsymbol{h}_{\rm opt}$ in 2-norm:

$$\begin{aligned}
&\min_{\boldsymbol{h}\in\mathcal{K}_\ell(H,\boldsymbol{g})} \|\boldsymbol{h} - \boldsymbol{h}_{\rm opt}\|_2 \\
&= \min_{\wp\in\mathbb{P}_\ell} \|\wp(H)\boldsymbol{g} + U(\Theta + \lambda_{\rm opt} I_n)^{-1} U^{\rm T}\boldsymbol{g}\|_2, \quad \big(\boldsymbol{h} = \wp(H)\boldsymbol{g} \in \mathcal{K}_\ell(H,\boldsymbol{g})\big) \\
&= \min_{\wp\in\mathbb{P}_\ell} \|\wp(\Theta)\widehat{\boldsymbol{g}} + (\Theta + \lambda_{\rm opt} I_n)^{-1}\widehat{\boldsymbol{g}}\|_2 \quad (\widehat{\boldsymbol{g}} = [\hat{g}_1, \ldots, \hat{g}_n]^{\rm T} = U^{\rm T}\boldsymbol{g}) \\
&= \min_{\wp\in\mathbb{P}_\ell} \sqrt{\sum_{i=1}^n \left(\wp(\theta_i) + \frac{1}{\theta_i + \lambda_{\rm opt}}\right)^2 \cdot \hat{g}_i^2} \\
&\le \min_{\wp\in\mathbb{P}_\ell} \max_{\theta_1\le\theta\le\theta_n} \left|\wp(\theta) + \frac{1}{\theta + \lambda_{\rm opt}}\right| \cdot \|\boldsymbol{g}\|_2.
\end{aligned} \tag{4.13}$$

In the following, we seek an optimal polynomial given in (4.13) with the aid of Lemma 4.3. First, note that the linear transformation

$$\theta(x) = -\frac{\theta_n - \theta_1}{2}x + \frac{\theta_1 + \theta_n}{2}$$

maps $x \in [-1, 1]$ one-to-one and onto $\theta \in [\theta_1, \theta_n]$; thus,

$$\begin{aligned}
&\min_{\wp\in\mathbb{P}_\ell} \max_{\theta_1\le\theta\le\theta_n} \left|\wp(\theta) + \frac{1}{\theta + \lambda_{\rm opt}}\right| \\
&= \min_{\wp\in\mathbb{P}_\ell} \max_{-1\le x\le 1} \left|\wp(\theta(x)) - \frac{2}{(\theta_n - \theta_1)(x - \frac{\theta_1+\theta_n+2\lambda_{\rm opt}}{\theta_n-\theta_1})}\right| \\
&= \frac{2}{\theta_n - \theta_1} \times \min_{\wp\in\mathbb{P}_\ell} \max_{-1\le x\le 1} \left|\frac{(\theta_n - \theta_1)\wp(\theta(x))}{2} - \frac{1}{x - \frac{\theta_1+\theta_n+2\lambda_{\rm opt}}{\theta_n-\theta_1}}\right| \\
&= \frac{2}{\theta_n - \theta_1} \times \min_{\psi\in\mathbb{P}_\ell} \max_{-1\le x\le 1} \left|\psi(x) - \frac{1}{x - \phi}\right|, \qquad \left(\text{with } \psi(x) = \tfrac{(\theta_n-\theta_1)\wp(\theta(x))}{2}\right) \\
&= \frac{2\epsilon_\ell(\phi)}{\theta_n - \theta_1}
\end{aligned}$$

with ϕ given by (4.12). Consequently, we can combine the above with (4.5) and (4.13) to get (4.9). The inequality (4.10) also follows directly from (4.6) and and the proof is completed. □

Remark 4.1. It is noted that $\phi + \sqrt{\phi^2 - 1} > \phi > 1$ since $\phi > 1$, and for ϕ given by (4.12),

$$\phi + \sqrt{\phi^2 - 1} = \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}. \tag{4.14}$$

Therefore, $\epsilon_\ell(\phi)$ converges linearly to zero with the linear factor $\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-1}$ as ℓ increases, and consequently, $\|\boldsymbol{h}_\ell - \boldsymbol{h}_{\rm opt}\|_2$ converges to 0 linearly with the linear factor $\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-1}$, while the objective function value $m(\boldsymbol{h}_\ell)$ converges to $m(\boldsymbol{h}_{\rm opt})$ linearly with the linear factor $\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-2}$.

We finally can provide the worst case convergence of the associated λ_ℓ.

Theorem 4.2. *Under the assumptions of Theorem 4.1, it follows that*

$$|\lambda_\ell^3 - \lambda_{\rm opt}^3| \le 12\sigma^2\|H_{\rm opt}\|_2\zeta_\ell^2 + 6\sigma^2\|\boldsymbol{g}\|_2\sqrt{\kappa}\zeta_\ell, \tag{4.15}$$

where ζ_ℓ is given by (4.11).

Proof. By $(H+\lambda_{\rm opt}I_n)\boldsymbol{h}_{\rm opt} = -\boldsymbol{g}$ and $\lambda_{\rm opt} = \sigma\|\boldsymbol{h}_{\rm opt}\|_2$, we have

$$\begin{aligned}\lambda_{\rm opt}^3 &= -\sigma^2(\boldsymbol{g}^{\rm T}\boldsymbol{h}_{\rm opt} + \boldsymbol{h}_{\rm opt}^{\rm T}H\boldsymbol{h}_{\rm opt})\\ &= -\sigma^2(2m(\boldsymbol{h}_{\rm opt}) - \boldsymbol{g}^{\rm T}\boldsymbol{h}_{\rm opt} - \frac{2}{3}\sigma\|\boldsymbol{h}_{\rm opt}\|_2^3)\\ &= -\sigma^2(2m(\boldsymbol{h}_{\rm opt}) - \boldsymbol{g}^{\rm T}\boldsymbol{h}_{\rm opt}) + \frac{2}{3}\lambda_{\rm opt}^3\end{aligned}$$

leading to

$$\lambda_{\rm opt}^3 = -3\sigma^2(2m(\boldsymbol{h}_{\rm opt}) - \boldsymbol{g}^{\rm T}\boldsymbol{h}_{\rm opt}).$$

Similarly, by

$$(S_\ell + \lambda_\ell I_\ell)\boldsymbol{s}_{{\rm opt},\ell} = -\|\boldsymbol{g}\|_2\boldsymbol{e}_1,\ \lambda_\ell = \sigma\|\boldsymbol{s}_{{\rm opt},\ell}\|_2 = \sigma\|\boldsymbol{h}_\ell\|_2,\ \boldsymbol{h}_\ell^{\rm T}H\boldsymbol{h}_\ell = \boldsymbol{s}_{{\rm opt},\ell}^{\rm T}S_\ell\boldsymbol{s}_{{\rm opt},\ell}$$

and $\boldsymbol{h}_\ell^{\rm T}\boldsymbol{g} = \|\boldsymbol{g}\|_2\boldsymbol{s}_{{\rm opt},\ell}^{\rm T}\boldsymbol{e}_1$, it follows that

$$\lambda_\ell^3 = -3\sigma^2(2m(\boldsymbol{h}_\ell) - \boldsymbol{g}^{\rm T}\boldsymbol{h}_\ell).$$

Consequently, the conclusion follows from Theorem 4.1. □

Remark 4.2. In [5, 6], the linear convergence of the Lanczos iteration is also discussed. In particular, it is shown that

$$0 \le m(\boldsymbol{h}_\ell) - m(\boldsymbol{h}_{\rm opt}) \le 36[m(\boldsymbol{0}) - m(\boldsymbol{h}_{\rm opt})]e^{\frac{-4\ell}{\sqrt{\kappa}}}. \tag{4.16}$$

The right-hand size of (4.16) is a very simple form of upper bound in [5, 6]. Ignoring the constants that are independent on ℓ in (4.9) and (4.16), and using (4.8) and (4.14), we note that the ratio factor related with ℓ is

$$[\upsilon(\kappa)]^\ell := \left[e^{\frac{-4}{\sqrt{\kappa}}}\Big/\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2\right]^\ell$$

which satisfies $\upsilon(\kappa) = \frac{e^{\frac{-4}{\sqrt{\kappa}}}}{\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2} > 1$. Figure 4.1 illustrates the values $e^{\frac{-4}{\sqrt{\kappa}}}$, $\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2$ and $\upsilon(\kappa)$ with respect to $0 < \frac{1}{\kappa} < 1$.

For the hard case, we remark that the approximation $\boldsymbol{h}_\ell$ will not provide sufficient accuracy because the condition $\boldsymbol{g} \perp \mathcal{E}_1$ implies $\boldsymbol{h}_\ell \in {\rm span}(Q_\ell) \perp \mathcal{E}_1$, and thus $\boldsymbol{h}_\ell$ will never contain the component $\boldsymbol{u} \in \mathcal{E}_1$ given in (2.8). Similar to the hard case in TRS, the Lanczos procedure should restart after breakdown (i.e., $\ell = \tau$) with new starting vector orthogonal to ${\rm span}(Q_\tau)$ [15]. We omit the further detailed discussions on this situation and refer to [15, 41, 5, 6].

4.2 A nested restarting procedure

Revealed by (4.9) and (4.10), the convergence of $\boldsymbol{h}_\ell$ could be slow when the condition number κ is large. In this case, a large ℓ is required for an accurate approximation $\boldsymbol{h}_\ell$. However, as the dimension ℓ of $\mathcal{K}_\ell(H,\boldsymbol{g})$ continuously gets large, the orthogonality of Q_ℓ deteriorates and memory requirement increases; moreover, the computational costs for solving the reduced problem (4.2) also grow and the numerical stability decreases. An effective treatment for this situation is to restart the Lanczos process, which is the topic of this subsection.

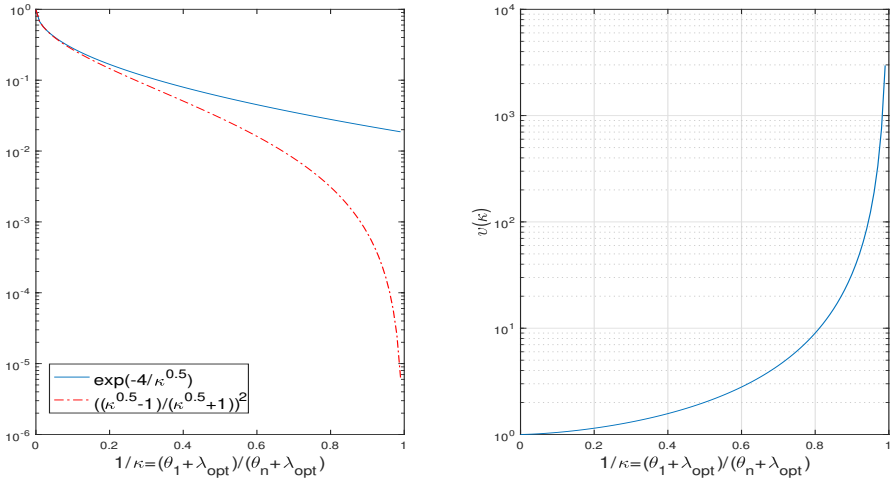

Figure 4.1: The values $e^{\frac{-4}{\sqrt{\kappa}}}$, $\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2$ and $\upsilon(\kappa)$ w.r.t. $\frac{1}{\kappa}$.

4.2.1 Restart the Lanczos procedure

Suppose $(\boldsymbol{h}^{(k)}, \lambda^{(k)})$ is the current approximation pair of $(\boldsymbol{h}_{\rm opt}, \lambda_{\rm opt})$ and define the residual

$$\boldsymbol{r}^{(k)} = H\boldsymbol{h}^{(k)} + \lambda^{(k)}\boldsymbol{h}^{(k)} + \boldsymbol{g}. \tag{4.17}$$

We aim at finding the correction $(\boldsymbol{d}^{(k)}, \rho^{(k)})$ so that $\boldsymbol{h}_{\rm opt} = \boldsymbol{h}^{(k)} + \boldsymbol{d}^{(k)}$ and $\lambda_{\rm opt} = \lambda^{(k)} + \rho^{(k)}$. The conditions (2.2) and (4.17) imply

$$\boldsymbol{d}^{(k)} = -(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{r}^{(k)} - \rho^{(k)}(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{h}^{(k)}. \tag{4.18}$$

The vectors $(H+\lambda_{\rm opt} I_n)^{-1}\boldsymbol{r}^{(k)}$ and $\rho^{(k)}(H+\lambda_{\rm opt} I_n)^{-1}\boldsymbol{h}^{(k)}$ cannot be obtained due to the unknown $\lambda_{\rm opt}$ and $\rho^{(k)}$. However, we can produce a subspace where these vectors lie. Specifically, using the fact $\mathcal{K}(H + \lambda_{\rm opt} I_n, \boldsymbol{r}) = \mathcal{K}(H, \boldsymbol{r})$ for any vector $\boldsymbol{r}$, we have

$$\begin{aligned}
\boldsymbol{d}^{(k)} &= -(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{r}^{(k)} - \rho^{(k)}(H + \lambda_{\rm opt} I_n)^{-1}\boldsymbol{h}^{(k)} \\
&\approx p(H)\boldsymbol{r}^{(k)} + \hat{p}(H)\boldsymbol{h}^{(k)} \\
&\in \mathcal{K}_{k_i}(H, \boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)}),
\end{aligned} \tag{4.19}$$

where p and $\hat{p}$ are certain polynomials of properly chosen degree $k_i - 1$ and $m_i - 1$, respectively. This implies that we can construct the subspace $\mathcal{K}_{k_i}(H, \boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ for the correction $\boldsymbol{d}^{(k)}$ and restart the Lanczos process at $(\boldsymbol{h}^{(k)}, \lambda^{(k)})$. In particular, we solve the following to update $\boldsymbol{h}^{(k)}$ by the solution of

$$\min_{\boldsymbol{h} \in \boldsymbol{h}^{(k)} + \mathcal{K}_{k_i}(H, \boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})} m(\boldsymbol{h}). \tag{4.20}$$

We shall show in our numerical results in section 5 that the second Krylov subspace $\mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ will improve significantly the convergence; in practice, a small dimension m_i is usually sufficient.

An orthonormal basis matrix $U^{(k)}$ of $\mathcal{K}_{k_i}(H, \boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ can be obtained by first computing the orthonormal basis of $\mathcal{K}_{k_i}(H, \boldsymbol{r}^{(k)})$ and then augment it to have $U^{(k)}$ by, for example [40, Algorithm 3.3], the (modified) Gram-Schmidt process.

4.2.2 Acceleration by a nested restarting procedure

The previous restarting procedure continues with $\boldsymbol{h}^{(k+1)}$ as a solution to (4.20) and $\lambda^{(k+1)} = \sigma\|\boldsymbol{h}^{(k+1)}\|_2$. This is the basic restarting Lanczos method for (1.2). For TRS, the convergence of

this version has been established in [40]. Also, it is shown that this basic restarting procedure can be further improved by a nested restarting structure originally proposed in [11] (see also [10]) for the nonsymmetric linear system in the GMRES framework. Recently the nested restarting Lanczos approach is also applied to solve the maximal correlation problem arising in applied statistics [39].

To describe this nested restarting scheme, we denote by $\mathring{\boldsymbol{h}}^{(k+1)}$ the solution of (4.20) and let

$$\mathring{\boldsymbol{h}}^{(k+1)} = \boldsymbol{h}^{(k)} + \mathring{\boldsymbol{d}}^{(k)} := \underset{\boldsymbol{h}\in\boldsymbol{h}^{(k)}+\mathcal{K}_{k_i}(H,\boldsymbol{r}^{(k)})+\mathcal{K}_{m_i}(H,\boldsymbol{h}^{(k)})}{\operatorname{argmin}} m(\boldsymbol{h}) \tag{4.21}$$

where k_i and m_i are preset dimensions of the Krylov subspaces $\mathcal{K}_{k_i}(H,\boldsymbol{r}^{(k)})$ and $\mathcal{K}_{k_i}(H,\boldsymbol{h}^{(k)})$, respectively. The idea of the nested restarting in [11] is to refine $\mathring{\boldsymbol{h}}^{(k+1)}$ by finding an improved approximation $\boldsymbol{h}^{(k+1)}$ in the affine set $\mathring{\boldsymbol{h}}^{(k+1)} + \operatorname{span}(D^{(k)})$ where $D^{(k)} = [\mathring{\boldsymbol{d}}^{(k-p+1)}, \ldots, \mathring{\boldsymbol{d}}^{(k)}] \in \mathbb{R}^{n\times p}$ contains the previous p correction vectors $\mathring{\boldsymbol{d}}^{(j)}$ for $j = k-p+1, \ldots, k$. That is,

$$\boldsymbol{h}^{(k+1)} = \boldsymbol{h}^{(k)} + \boldsymbol{d}^{(k)} := \underset{\boldsymbol{h}\in\boldsymbol{h}^{(k)}+\operatorname{span}(D^{(k)})}{\operatorname{argmin}} m(\boldsymbol{h}). \tag{4.22}$$

One can see that $\boldsymbol{h}^{(k+1)}$ is a better approximation than $\mathring{\boldsymbol{h}}^{(k+1)}$ because $\mathring{\boldsymbol{h}}^{(k+1)} \in \boldsymbol{h}^{(k)} + \operatorname{span}(D^{(k)})$ (corresponding to $p = 1$). Algorithm 2 summarizes the nested restarting Lanczos method for (1.2).

Algorithm 2 A nested restarted Lanczos method for (1.2)

1: Choose $\epsilon > 0$, $p > 0$ and let $\boldsymbol{h}^{(0)} = \boldsymbol{0}$, $\boldsymbol{r}^{(0)} = \boldsymbol{g}$, $D^{(-1)} = [\,]$, $k = 0$;
2: **while** $\|\boldsymbol{r}^{(k)}\|_2 > \epsilon$ and $k < k_{\max}$ **do**
3: Compute $\mathring{\boldsymbol{h}}^{(k+1)}$ by (4.21);
4: Set $D^{(k)} = [D^{(k-1)}, \mathring{\boldsymbol{h}}^{(k+1)} - \boldsymbol{h}^{(k)}]$;
5: Delete the first column of $D^{(k)}$ if $D^{(k)}$ has $p+1$ columns;
6: Compute $\boldsymbol{h}^{(k+1)}$ by (4.22) and $\lambda^{(k+1)} = \sigma\|\boldsymbol{h}^{(k+1)}\|_2$;
7: Set $\boldsymbol{r}^{(k+1)} = H\boldsymbol{h}^{(k+1)} + \lambda^{(k+1)}\boldsymbol{h}^{(k+1)} + \boldsymbol{g}$ and $k = k+1$;
8: **end while**

4.2.3 Solve the inner subproblem

Notice that the two inner subproblems (4.21) and (4.22) in Algorithm 2 have the same formulation:

$$\min_{\boldsymbol{h}\in\boldsymbol{h}^{(k)}+\mathcal{V}} m(\boldsymbol{h}) \tag{4.23}$$

where $\mathcal{V} = \mathcal{K}_{k_i}(H,\boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H,\boldsymbol{h}^{(k)})$ for (4.21) and $\mathcal{V} = \operatorname{span}(D^{(k)})$ for (4.22), respectively. Let $V \in \mathbb{R}^{n\times v}$ be orthonormal basis matrix of $\mathcal{V}$, and represent $\boldsymbol{h} \in \mathcal{V}$ by $\boldsymbol{h} = V\boldsymbol{v}$. Then the minimizer $\widehat{\boldsymbol{h}} = V\widehat{\boldsymbol{v}}$ of (4.23) can be solved by finding

$$\widehat{\boldsymbol{v}} = \underset{\boldsymbol{v}\in\mathbb{R}^v}{\operatorname{argmin}} \left\{ \frac{1}{2}\boldsymbol{v}^{\mathrm{T}}T\boldsymbol{v} + \boldsymbol{n}^{\mathrm{T}}\boldsymbol{v} + \frac{\sigma}{3}\left(\sqrt{\|V^{\mathrm{T}}\boldsymbol{h}^{(k)} + \boldsymbol{v}\|_2^2 + c^2}\right)^3 \right\}$$

where $T = V^{\mathrm{T}}HV \in \mathbb{R}^{v\times v}$, $\boldsymbol{n} = V^{\mathrm{T}}(\boldsymbol{g} + H\boldsymbol{h}^{(k)})$ and $c^2 = \|\boldsymbol{h}^{(k)}\|_2^2 - \|V^{\mathrm{T}}\boldsymbol{h}^{(k)}\|_2^2 \geq 0$. Noting from $\mathcal{V} = \mathcal{K}_{k_i}(H,\boldsymbol{r}^{(k)}) + \mathcal{K}_{m_i}(H,\boldsymbol{h}^{(k)})$ and $\boldsymbol{h}^{(k)} \in \operatorname{span}(V)$, we know the associated $c^2 = \|\boldsymbol{h}^{(k)}\|_2^2 - \|V^{\mathrm{T}}\boldsymbol{h}^{(k)}\|_2^2 = \|\boldsymbol{h}^{(k)}\|_2^2 - \|\boldsymbol{h}^{(k)}\|_2^2 = 0$. Denote $\boldsymbol{z} = V^{\mathrm{T}}\boldsymbol{h}^{(k)} + \boldsymbol{v}$ and $\boldsymbol{b} = \boldsymbol{n} - TV^{\mathrm{T}}\boldsymbol{h}^{(k)}$ to rewrite the above to

$$\widehat{\boldsymbol{z}} = \underset{\boldsymbol{z}\in\mathbb{R}^v}{\operatorname{argmin}} \left\{ \widehat{m}(\boldsymbol{z}) := \boldsymbol{b}^{\mathrm{T}}\boldsymbol{z} + \frac{1}{2}\boldsymbol{z}^{\mathrm{T}}T\boldsymbol{z} + \frac{\sigma}{3}\left(\sqrt{\|\boldsymbol{z}\|_2^2 + c^2}\right)^3 \right\}. \tag{4.24}$$

The resulting problem (4.24) is of a similar form as the original (1.2) except for the new parameter $c^2 \geq 0$. Following the argument for establishing Theorem 2.1 (see [7, Theorem 3.1]), we next provide a necessary and sufficient optimality condition for the global solution of (4.24).

Theorem 4.3. *$\widehat{\boldsymbol{z}} \in \mathbb{R}^v$ is a global minimizer of (4.24) if and only if $(T + \widehat{\lambda} I_v)\widehat{\boldsymbol{z}} + \boldsymbol{b} = \boldsymbol{0}$ and $T + \widehat{\lambda} I_v$ is symmetric positive semi-definite, where $\widehat{\lambda} = \sigma\sqrt{\|\widehat{\boldsymbol{z}}\|_2^2 + c^2}$. Moreover, if $T + \widehat{\lambda} I_v$ is positive definite, then $\widehat{\boldsymbol{z}}$ is unique.*

Proof. The proof follows similarly from that of [7, Theorem 3.1], and thus we omit the details. □

The value $\widehat{\lambda}_{\rm opt}$ can be obtained by expressing $\widehat{\boldsymbol{z}} = -(T + \widehat{\lambda} I_v)^{-1}\boldsymbol{b}$ and solving from $\widehat{\lambda} = \sigma\sqrt{\|(T + \widehat{\lambda} I_v)^{-1}\boldsymbol{b}\|_2^2 + c^2}$ if $\widehat{\lambda} > -\lambda_1(T)$, where $(\lambda_1(T), \boldsymbol{w}_1)$ is an eigenpair of T associated with the smallest eigenvalue $\lambda_1(T)$. The case for $\widehat{\lambda} = -\lambda_1(T)$ implies that $\widehat{\boldsymbol{z}} = -(T + \widehat{\lambda} I_v)^{\dagger}\boldsymbol{b} + \xi \boldsymbol{w}_1$ and ξ can be determined by $\widehat{\lambda} = \sigma\sqrt{\|(T + \widehat{\lambda} I_v)^{\dagger}\boldsymbol{b} + \xi \boldsymbol{w}_1\|_2^2 + c^2} = \sigma\sqrt{\|(T + \widehat{\lambda} I_v)^{\dagger}\boldsymbol{b}\|_2^2 + \xi^2\|\boldsymbol{w}_1\|_2^2 + c^2}$. Let $T = W \Xi W^{\rm T}$ be the eigen-decomposition of T and define

$$\psi(\lambda) = \|(T + \lambda I_v)^{-1}\boldsymbol{b}\|_2^2 = \sum_{i=1}^{\nu} \frac{\varpi_i^2}{(\lambda_i(T) + \lambda)^2}, \quad \varpi_i = \boldsymbol{e}_i^{\rm T} W \boldsymbol{b},$$

then we can apply the Newton iteration to the system (see [7])

$$\phi(\lambda) = \frac{1}{\sqrt{\psi(\lambda)}} - \frac{\sigma}{\sqrt{\lambda^2 - c^2\sigma^2}} = 0, \tag{4.25}$$

for the general case $\widehat{\lambda} > -\lambda_1(T)$ and analogously for $\widehat{\lambda} = -\lambda_1(T)$ by including the eigenpair $(\lambda_1(T), \boldsymbol{w}_1)$. Note that $\widehat{\lambda} = \sigma\sqrt{\|\widehat{\boldsymbol{z}}\|_2^2 + c^2} > \sigma c$ and we use σc as a lower bound for the approximation λ of $\widehat{\lambda}$.

The Newton iteration for the system (4.25) involves the derivative of $\phi(\lambda)$, and for this, we first have

$$\psi'(\lambda) = -2\boldsymbol{z}(\lambda)^{\rm T}(T + \lambda I_v)^{-1}\boldsymbol{z}(\lambda), \quad \text{with} \quad \boldsymbol{z}(\lambda) = (T + \lambda I_v)^{-1}\boldsymbol{b}; \tag{4.26}$$

thus, if $T + \lambda I_v = L(\lambda)L(\lambda)^{\rm T}$ is the Cholesky decomposition of $T + \lambda I_v$ with $\lambda > -\lambda_1(T)$, the derivative $\psi'(\lambda) = -2\|L(\lambda)^{-1}\boldsymbol{z}(\lambda)\|_2^2$ at λ can be computed via first solving $\boldsymbol{z}(\lambda)$ from $L(\lambda)L(\lambda)^{\rm T}\boldsymbol{z}(\lambda) = \boldsymbol{b}$ and then solving $\boldsymbol{l}(\lambda)$ from $L(\lambda)\boldsymbol{l}(\lambda) = \boldsymbol{z}(\lambda)$. This gives $\psi'(\lambda) = -2\|\boldsymbol{l}(\lambda)\|_2^2$. With $\psi'(\lambda)$, we further have

$$\phi'(\lambda) = \frac{\|\boldsymbol{l}(\lambda)\|_2^2}{\|\boldsymbol{z}(\lambda)\|_2^3} + \frac{\sigma\lambda}{\sqrt{(\lambda^2 - c^2\sigma^2)^3}} > 0, \quad \forall \lambda > -\lambda_1(T). \tag{4.27}$$

Therefore, for the current approximation λ of $\widehat{\lambda}_{\rm opt}$, the Newton step computes a correction $\Delta\lambda$ and updates the approximation as $\lambda + \Delta\lambda$ where

$$\Delta\lambda = \frac{\frac{a}{\lambda}\left(\|\boldsymbol{z}(\lambda)\|_2 - \frac{\sqrt{a}}{\sigma}\right)}{\|\boldsymbol{z}(\lambda)\|_2 + \frac{\sigma\sqrt{a^3}}{\lambda}\frac{\|\boldsymbol{l}(\lambda)\|_2^2}{\|\boldsymbol{z}(\lambda)\|_2^2}}, \quad \text{where} \quad a = \lambda^2 - c^2\sigma^2. \tag{4.28}$$

5 Numerical results

In this section, we will report numerical results of the nested restarting Lanczos algorithm to illustrate two aspects: (a) its performance for solving the cubic model (1.2), and (b) the performance for the minimization (1.1) when it serves as an inner solver for the subproblems of the cubic regularization of Newton method.

The MATLAB code of Algorithm 2 is labeled as `nrLan_cubic`. Numerical experiments are conducted in the environment of MATLAB R2015b and Ubuntu 20.04 system on a 64-bit PC with an Intel Core(TM) I5 8550U CPU (3.0GHz) and 8GB of RAM. As a stopping criterion, for the given tolerance $\epsilon = 10^{-6}$ and the maximun number $k_{\max} = 10000$, `nrLan_cubic` terminates whenever the relative residual[4] is no greater than ϵ, or the iteration k exceeds the maximum number, i.e.,

$$res := \frac{\|\boldsymbol{r}^{(k)}\|_\infty}{\|\boldsymbol{g}\|_\infty} = \frac{\|(H + \lambda^{(k)} I_n)\boldsymbol{h}^{(k)} + \boldsymbol{g}\|_\infty}{\|\boldsymbol{g}\|_\infty} \le \epsilon, \quad \text{or} \quad k > k_{\max}.$$

[4] We choose $res := \frac{\|\boldsymbol{r}^{(k)}\|_\infty}{\|\boldsymbol{g}\|_\infty}$ as the relative residual is because approximately we have $(H + \lambda^{(k)} I_n)\boldsymbol{h}^{(k)} \approx -\boldsymbol{g}$.

Table 5.1: Numerical results of `nrLan_cubic` with different values of k_i and m_i

k_i	m_i	$n = 1000$, $\sigma = 0.1$				$n = 1000$, $\sigma = 0.05$			
		Iter_{outer}	Prod	res	CPU	Iter_{outer}	Prod	res	CPU
20	2	33	1115	9.40e-7	0.51	71	1989	1.00e-6	0.68
30	10	21	1181	7.51e-7	0.46	45	2165	8.04e-7	0.71
30	20	20	1320	8.73e-7	0.53	43	2493	9.95e-7	0.83
30	30	19	1439	5.46e-7	0.56	42	2842	9.39e-7	0.94
50	**2**	15	1091	7.14e-7	**0.46**	30	1886	5.82e-7	**0.65**
50	10	14	1134	6.74e-7	0.48	28	1988	7.06e-7	0.73
50	30	13	1293	9.64e-7	0.60	28	2508	4.19e-7	0.93
50	50	13	1513	8.77e-7	0.70	27	2927	3.03e-7	1.15
80	**2**	10	**1066**	4.78e-7	0.49	19	**1813**	8.82e-7	0.73
80	10	10	1130	4.06e-7	0.55	19	1949	7.55e-7	0.78
80	30	10	1290	1.48e-7	0.63	17	2067	9.23e-7	0.88
80	50	10	1450	5.53e-7	0.71	18	2498	9.83e-7	1.09
80	80	9	1529	6.08e-7	0.77	16	2656	8.12e-7	1.30
100	2	9	1123	2.28e-7	0.52	17	1947	5.05e-7	0.80
100	10	9	1179	1.74e-7	0.55	16	1956	9.22e-7	0.88
100	50	9	1459	1.12e-7	0.73	14	2214	8.54e-7	1.04
100	80	8	1488	5.08e-7	0.81	13	2393	8.89e-7	1.24
100	100	8	1608	3.74e-7	0.89	14	2814	3.04e-7	1.51
150	2	7	1167	1.59e-7	0.60	12	1932	8.21e-7	0.93
150	10	7	1207	1.90e-7	0.63	12	2012	3.57e-7	1.01
150	50	7	1407	9.37e-8	0.84	11	2211	8.04e-7	1.22
150	100	7	1657	7.16e-8	1.08	11	2661	2.69e-7	1.65
150	150	6	1606	5.40e-7	1.14	10	2810	7.80e-7	1.96

Before the evaluation of the two aspects of `nrLan_cubic`, we first carry out numerical tests to demonstrate two crucial integrations of `nrLan_cubic`, namely the double Krylov subspaces in (4.21) and the nested structure in (4.22). For this purpose, we choose an appropriate parameter pair (k_i, m_i) for `nrLan_cubic`, and verify the contribution of the second Krylov subspace $\mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ and the nested structure (Lines 4–6 in Algorithm 2). The test problems for this purpose are from randomly generated H and $\boldsymbol{g}$ as used in [40], i.e.,

$$H = GG^T - I_n, \quad G = \mathsf{randn}(n), \ \boldsymbol{g} = \mathsf{randn}(n, 1). \tag{5.1}$$

Also, we choose two values for σ, i.e., $\sigma = 0.1$ and $\sigma = 0.05$.

In Table 5.1, we report the numerical results of `nrLan_cubic` with various parameters, where Iter_{outer}, Prod and CPU stand for the number of iterations, the number of matrix-vector products and the consuming CPU time in second. It tells that the number of iterations decreases as k_i and m_i increase in general. However, as the dimension of the projected subproblems of (4.21) and (4.22) get larger as well, the efficiency overall does not improve consistently. A good choice of parameters indicated by this testing is $k_i = 50$ and $m_i = 2$.

We next show that the additional Krylov subspace $\mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ and the nested structure are two crucial integrations for the performance of the algorithm. First, by setting $m_i = 2$ and $m_i = 0$, we report in Figure 5.1 the average quantities from `nrLan_cubic` over 20 random test problems (with the same settings for H and $\boldsymbol{g}$ as (5.1)) with and without using this additional Krylov subspace $\mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$, respectively. One can clearly see that the additional information from $\mathcal{K}_{m_i}(H, \boldsymbol{h}^{(k)})$ improves the performance substantially.

For the nested structure, we similarly run `nrLan_cubic` by enabling and disabling this nested structure (Lines 4–6 in Algorithm 2) on 20 random instances. The average numerical results are plotted in Figure 5.2, which also demonstrate the importance of the nested structure for `nrLan_cubic`.

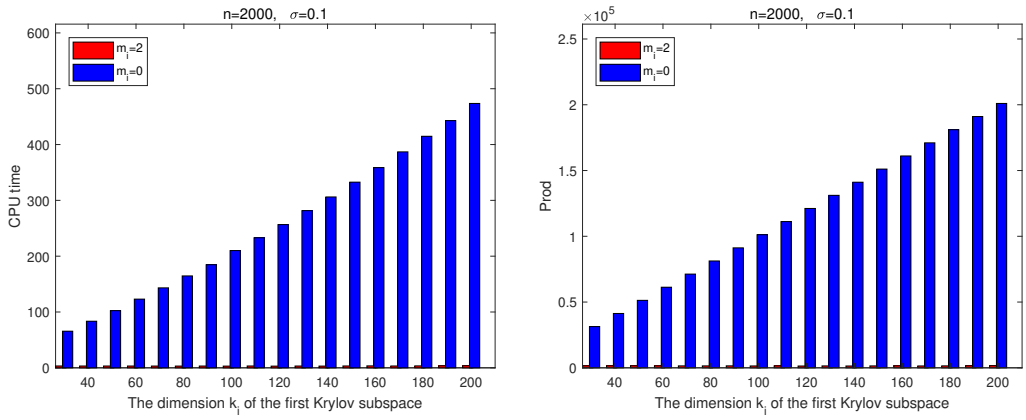

Figure 5.1: CPU time and matrix-vector products for $H = GG^T - I_n$

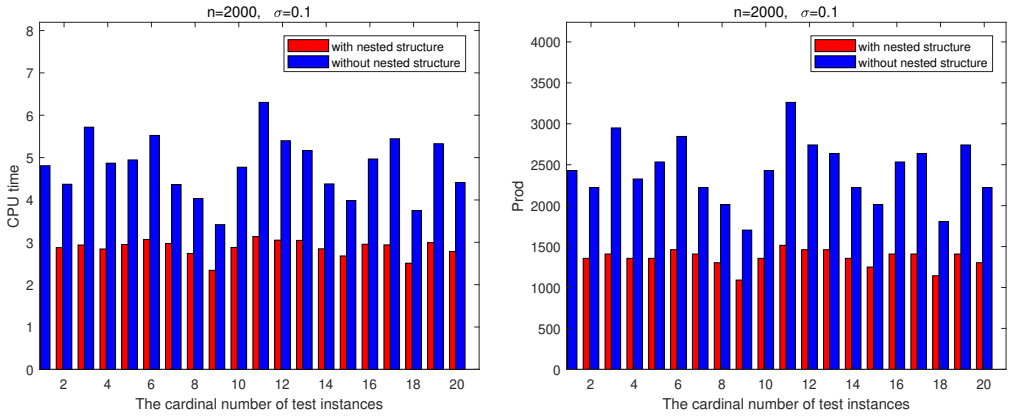

Figure 5.2: CPU time and matrix-vector products for $H = GG^T - I_n$

5.1 Performance for the cubic model (1.2)

Now, we carry out numerical evaluation to illustrate the first aspect of `nrLan_cubic`, namely, its performance for solving the cubic model (1.2). For this purpose, we compare `nrLan_cubic` using the associated parameters

$$p = \min(100, n), \ k_i = \min(50, n), \ m_i = 2.$$

with the basic Lanczos method (labeled as `Lan_cubic`) [7, 8]. We test four instances of model (2.1) on dimension $n = 5000,\ 8000$ and parameter $\sigma = 0.1,\ 0.05$, where H and $\boldsymbol{g}$ are generated randomly as (5.1). In Figures 5.3 and 5.4, we demonstrate how the computed relative residual of each algorithm behaves against the number of matrix-vector products. It can be seen from Figures 5.3 and 5.4 that, for all instances, `nrLan_cubic` consumes less CPU time to reach the stopping criterion than `Lan_cubic`. The saving computational cost is from the requirements in dealing with smaller projected subproblems (4.3) than those of `Lan_cubic`, and this saving compensates the slight increase of the matrix-vector products in `nrLan_cubic`. More numerical comparisons on the two implementations will be reported in the next subsection for solving the minimization problem (1.1).

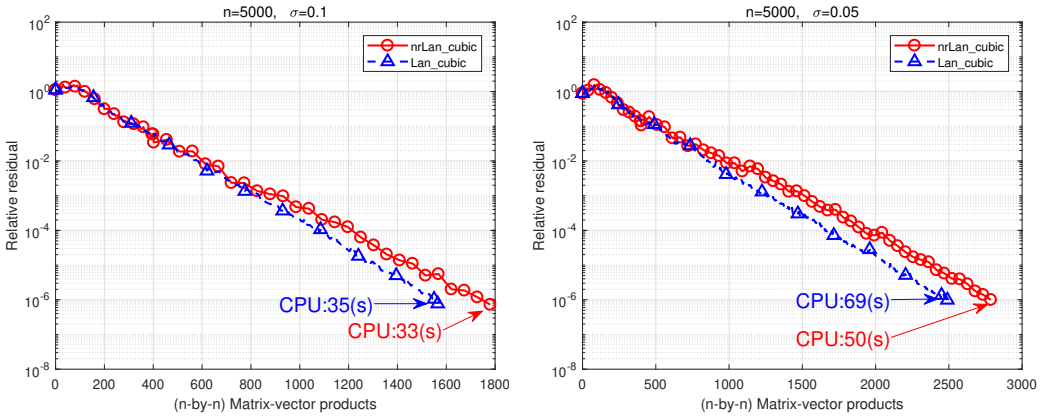

Figure 5.3: Residuals for $H = GG^T - I_n$

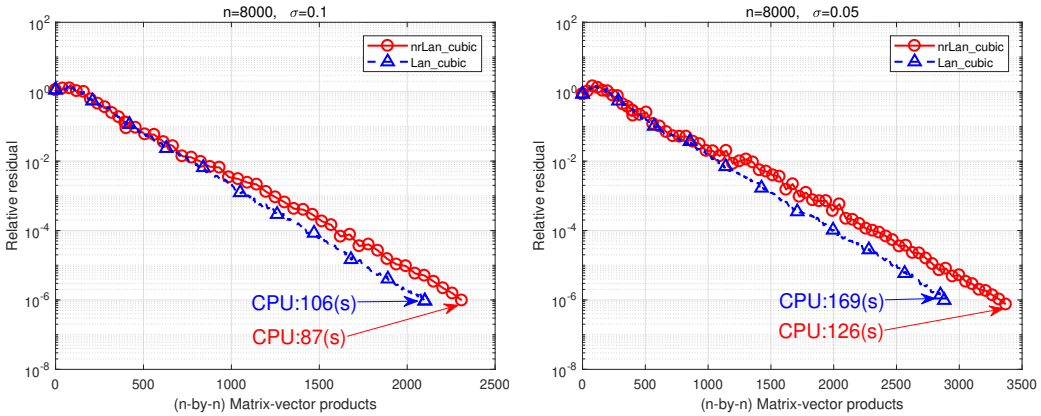

Figure 5.4: Residuals for $H = GG^T - I_n$

5.2 Performance for minimization on the CUTEst collection

In this subsection, we turn to the second aspect of `nrLan_cubic` for solving the minimization problem (1.1). We conduct this testing by choosing unconstrained optimization problems from the CUTEst[5] collection [16], when `nrLan_cubic` is embedded in the cubic regularization framework [7, 8]. In particular, we denote by `Lan_cubic`(500) and `Lan_cubic`(1000) the basic Lanczos method [7, 8] that restarts in every 500 and 1000 steps (i.e., the dimension of the underlying Krylov subspace), respectively. To clearly indicate the minimization solvers with different inner solver for (1.2), we use `min_nrLan_cubic`, `min_Lan_cubic`(500), `min_Lan_cubic`(1000), and `min_Nt_cubic` to represent the overall minimization solver where our `nrLan_cubic`, `Lan_cubic`(500), `Lan_cubic`(1000) and the Newton method [7, 8] are used for solving subproblems (1.2) in the cubic regularization framework [7, 8], respectively.

All algorithms are coded in MATLAB, and the cubic framework as well as `min_Lan_cubic`(500), `min_Lan_cubic`(1000), and Newton iteration `min_Nt_cubic` are translated from Manopt(MATLAB)[6] [4] by removing the manifold structure. Manopt(MATLAB) is a toolbox for optimization on manifolds. Specifically, `min_Lan_cubic`(500) and `min_Lan_cubic`(1000) are modified from **arc_lanczos.m** in Manopt(MATLAB), while `min_Nt_cubic` is from **minimize_cubic_newton.m**. The outer-loop

[5]It is available at https://github.com/ralna/CUTEst.
[6]It is available at https://www.manopt.org/.

iteration of the cubic regularization algorithm terminates if the relative residual res_o is no more than 10^{-6} and CPU time is no greater than the maximum CPU time 3600 seconds, i.e.,

$$res_o := \frac{\|\boldsymbol{g}^{(j)}\|_\infty}{\|\boldsymbol{g}^{(0)}\|_\infty} \leq 10^{-6} \text{ and CPU time} \leq 3600(s),$$

where j denotes the j-th outer-loop iteration in the cubic regularization framework. As the cubic regularization subproblem needs not to be solved accurately at each iteration j, we adaptively tighten the inner tolerance ϵ for all inner solvers `nrLan_cubic`, `Lan_cubic`(500) and `Lan_cubic`(1000), and `Nt_cubic` in a same strategy.

The set of test minimization problems is systematically chosen from the CUTEst collection. Specifically, we set relevant options in the following table for the resulting problems.

Objective function type	: **Q S O**
Constraints type	: *****
Regularity	: ∗
Degree of available derivatives	: 2
Problem interest	: ∗
Explicit internal variables	: ∗
Number of variables	: **in [100, 99999999]**
Number of constraints	: ∗

where **Q** = *quadratic type*, **S** = *sum of square type*, **O** = *other type (nonlinear, non-constant, etc.)*, ∗ = *everything goes*, **in [100, 99999999]** = $100 \leq n \leq 99999999$, and *"Degree of available derivatives =2" means the analytic second-order Hessian is used.*

It should be mentioned that most of the resulting problems from the above choice consist of a particular parameter; in this case, for each problem, we select one value in the given set for this parameter so that the dimension n of the resulting problem is the largest one in $[2000, 15000]$. For those that do not have a value of the parameter corresponding to n in $[2000, 15000]$, we then remove them from the resulting set. As a result, 81 problems are selected for testing, and the detailed information about these problems is listed in Table 6.1 in Appendix.

From the numerical results of the set of 81 test problems, we find that all the test solvers fail to obtain approximations within the given stopping criterion for the problems FLETCBV3, FLETCHBV, GENHUMPS, INDEF, NONMSQRT; therefore, we did not record them in our numerical report in Figure 5.5 where the relative residuals res_o and CPU time are plotted. In particular, the left subfigure in Figure 5.5 demonstrates res_o for each problem (in 76 test problems) indexed by the x-axis. It can be seen then that the number of failure cases for `min_nrLan_cubic`, `min_Lan_cubic`(500), `min_Lan_cubic`(1000), and `min_Nt_cubic` is 6, 8, 10, and 8, respectively. This subfigure also reveals that the Newton iteration `min_Nt_cubic` is able to achieve higher accurate solutions for most test problems. However, in term of efficiency, the right subfigure in Figure 5.5, which is a demonstration of numerical results for the performance in the format of Dolan and Moré [12], implies that `min_nrLan_cubic` in general is more efficient than the other three.

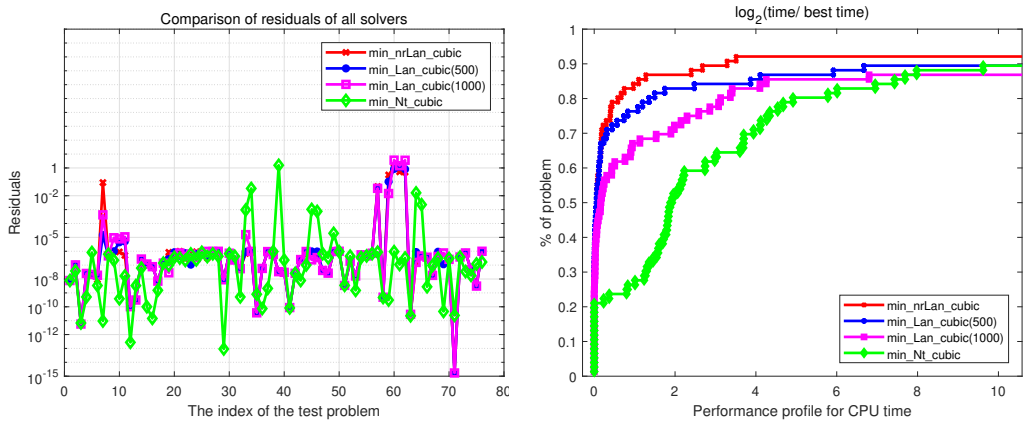

Figure 5.5: Residuals and CPU time for the CUTEst collection

Our final remark on the solvers is from the observation of two test problems SSBRYBND ($n = 5000$) and SSCOSINE ($n = 10000$). Figure 5.6 provides the details on how the relative residual res_o of each algorithm behaves against the number of outer iteration j. In particular, for SSBRYBND and SSCOSINE, we notice that the relative residuals res_o of `min_Nt_cubic` decrease to 10^{-6} rapidly. This is an indicator showing that highly accurate approximations for the subproblems (1.2) can be helpful for the outer-loop convergence. This fact can also be seen by the results from `min_Lan_cubic`(500) and `min_Lan_cubic`(1000), in which the Lanczos process stops earlier and then restarts. This produces approximations of low accuracy for (1.2). In such problems where the subproblems (1.2) encounter ill-conditioned cases but relatively highly accurate approximations are still desired in the outer-loop iteration, our proposed `min_nrLan_cubic` can help as they have demonstrated in SSBRYBND and SSCOSINE.

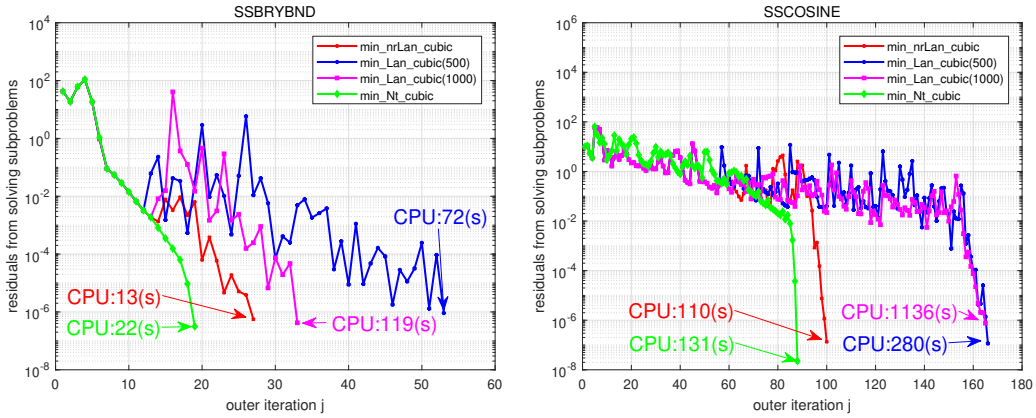

Figure 5.6: Residuals vs. the number of iterations for SSBRYBND and SSCOSINE

6 Conclusions

In this paper, we made two numerical contributions to the cubic regularization of Newton method. We first established an $(n+1)$ dimensional equivalent QEP for the cubic regularization subproblem (1.2) and derive two new $2(n+1)$ dimensional equivalent generalized eigenvalue problems by means of linearization of QEP. Our second contribution is on the Lanczos method for (1.2) for the large scale minimization. A new and sharp convergence result on the basic Lanczos method [7, 8] was established, and a nested restarting version was proposed to handle ill-conditioned cases. Our

numerical experience indicates that the nested restarting Lanczos iteration can be helpful for the large scale minimization problem, especially in which ill-conditioned subproblems may emerge.

Acknowledgments.

The authors are grateful to the anonymous referees for their useful comments and suggestions to improve the presentation of this paper. They also thank Dr. Ren-Cang Li at University of Texas at Arlington for the idea of the proof of Theorem 4.2.

References

[1] S. Adachi, S. Iwata, Y. Nakatsukasa, and A. Takeda. Solving the trust-region subproblem by a generalized eigenvalue problem. *SIAM J. Optim.*, 27(1):269–291, 2017.

[2] Z. Bai and Y. Su. SOAR: A second-order Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 26(3):640–659, 2005.

[3] S. N. Bernstein. Sur l'ordre de la meilleure approximation des fonctions continues par les polynômes de degré donné. *Mém. acad. royale Belg.*, 4:1–104, 1912.

[4] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre. Manopt, a Matlab Toolbox for Optimization on Manifolds. *J. Mach. Learn. Res.*, 15(42):1455–1459, 2014.

[5] Y. Carmon and J. Duchi. Analysis of Krylov subspace solutions of regularized nonconvex quadratic problems. In *Neural Information Processing Systems (NIPS), 2018, Selected for oral presentation*, 2018.

[6] Y. Carmon and J. Duchi. First-order methods for nonconvex quadratic minimization. *SIAM Rev.*, 62(2):395–436, 2020.

[7] C. Cartis, N. I. M. Gould, and P. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. *Math. Program.*, 127:245–295, 2011.

[8] C. Cartis, N. I. M. Gould, and P. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function- and derivative-evaluation complexity. *Math. Program.*, 130:295–319, 2011.

[9] A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust-Region Methods*. SIAM, Philadelphia, PA, 2000.

[10] E. de Sturler. Nested Krylov methods based on GCR. *J. Comput. Appl. Math.*, 67:15–41, 1996.

[11] H. A. Van der Vorst and C. Vuik. GMRESR: A family of nested GMRES methods. *Numer. Linear Algebra Appl.*, 1:369–386, 1994.

[12] E. D. Dolan and J. Moré. Benchmarking optimization software with performance profiles. *Math. Program.*, 91:201–213, 2002.

[13] D. M. Gay. Computing optimal locally constrained steps. *SIAM J. Sci. Statist. Comput.*, 2(1):186–197, 1981.

[14] G. H. Golub and U. von Matt. Quadratically constrained least squares and quadratic problems. *Numer. Math.*, 59:561–580, 1991.

[15] N. I. M. Gould, S. Lucidi, M. Roma, and P. L. Toint. Solving the trust-region subproblem using the Lanczos method. *SIAM J. Optim.*, 9:504–525, 1999.

[16] N. I. M. Gould, D. Orban, and P. L. Toint. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Comput. Optim. Appl.*, 60:545–557, 2015.

[17] N. I. M. Gould, D. P. Robinson, and H. S. Thorne. On solving trust-region and other regularised subproblems in optimization. *Math. Program. Comput.*, 2(1):21–57, 2010.

[18] N. I. M. Gould and V. Simoncini. Error estimates for iterative algorithms for minimizing regularized quadratic subproblems. *Optim. Methods Softw.*, 35(2):304–328, 2019.

[19] S. Gratton, A. Sartenaer, and P. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization. *SIAM J. Optim.*, 19(8):414–444, 2008.

[20] A. Griewank. The modification of NewtonÕs method for unconstrained optimization by bounding cubic terms. Technical Report NA/12, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, UK, 1981.

[21] W. W. Hager. Minimizing a quadratic over a sphere. *SIAM J. Optim.*, 12:188–208, 2001.

[22] E. Hazan and T. Koren. A linear-time algorithm for trust region problems. *Math. Program., Ser. A*, 158(1):363–381, 2016.

[23] N. Ho-Nguyen and F. Kilinç-Karzan. A second-order cone based approach for solving the trust-region subproblem and its variants. *SIAM J. Optim.*, 27(3):1485–1512, 2017.

[24] X. Liang and R.-C. Li. The hyperbolic quadratic eigenvalue problem. *Forum of Mathematics, Sigma*, 3(e13):1–93, 2015.

[25] F. Lieder. Solving large-scale cubic regularization by a generalized eigenvalue problem. *SIAM J. Optim.*, 30(4):3345–3358, 2020.

[26] J. J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. Statist. Comput.*, 4(3):553–572, 1983.

[27] Y. Nesterov and B.T. Polyak. Cubic regularization of Newton method and its global performance. *Math. Program.*, 108:177–205, 2006.

[28] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, New York, 2nd ed. edition, 2006.

[29] B. N. Parlett. *The Symmetric Eigenvalue Problem*. SIAM, Philadelphia, 1998.

[30] R. Rendl and H. Wolkowicz. A semidefinite framework for trust region subproblems with applications to large scale minimization. *Math. Program.*, 77(2):273–299, 1997.

[31] M. Rojas, S. A. Santos, and D. C. Sorensen. A new matrix-free algorithm for the large-scale trust-region subproblem. *SIAM J. Optim.*, 11:611–646, 2000.

[32] M. Rojas, S. A. Santos, and D. C. Sorensen. Algorithm 873: LSTRS: MATLAB software for large-scale trust-region subproblems and regularization. *ACM Trans. Math. Software*, 34(2):11:1–28, 2008.

[33] M. Rojas and D. C. Sorensen. A trust-region approach to the regularization of large-scale discrete forms of ill-posed problems. *SIAM J. Sci. Comput.*, 23:1842–1860, 2002.

[34] D. C. Sorensen. Minimization of a large-scale quadratic function subject to a spherical constraint. *SIAM J. Optim.*, 7:141–161, 1997.

[35] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, 20:626–637, 1983.

[36] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Rev.*, 43:235–286, 2001.

[37] P. L. Toint. *Towards an efficient sparsity exploiting Newton method for minimization*. In: Sparse Matrices and Their Uses, Academic Press, London, 57-88, 1981.

[38] Y. Yuan. On the truncated conjugate gradient method. *Math. Program.*, 87:561–573, 2000.

[39] L.-H. Zhang, X. Ma, and C. Shen. A structure-exploiting nested Lanczos-type iteration for the multi-view canonical correlation analysis. *SIAM J. Sci. Comput.*, 43(4):A2685–A2713, 2021.

[40] L.-H. Zhang and C. Shen. A nested Lanczos method for the trust-region subproblem. *SIAM J. Sci. Comput.*, 40(4):A2005–A2032, 2018.

[41] L.-H. Zhang, C. Shen, and R.-C. Li. On the generalized Lanczos trust-region method. *SIAM J. Optim.*, 27(3):2110–2142, 2017.

[42] L.-H. Zhang, W. H. Yang, C. Shen, and J. Feng. Error bounds of the Lanczos approach for the trust-region subproblem. *Front. Math. China*, 13(2):459–481, 2018.

Appendix

Table 6.1: Information on test problems selected from the `CUTEst` collection

Problem	Parameter	n	Problem	Parameter	n	Problem	Parameter	n
ARWHEAD	N=5000	5000	BDQRTIC	N=5000	5000	BOX	N=10000	10000
BROYDN7D	N/2=5000	10000	BRYBND	UB=1	5000	CHAINWOO	NS=4999	10000
COSINE	N=10000	10000	CRAGGLVY	M=2499	5000	CURLY10	N=10000	10000
CURLY20	N=10000	10000	CURLY30	N=10000	10000	DIXMAANA	M=3000	9000
DIXMAANB	M=3000	9000	DIXMAANC	M=3000	9000	DIXMAAND	M=3000	9000
DIXMAANE	M=3000	9000	DIXMAANF	M=3000	9000	DIXMAANG	M=3000	9000
DIXMAANH	M=3000	9000	DIXMAANI	M=3000	9000	DIXMAANJ	M=3000	9000
DIXMAANK	M=3000	9000	DIXMAANL	M=3000	9000	DIXMAANM	M=3000	9000
DIXMAANN	M=3000	9000	DIXMAANO	M=3000	9000	DIXMAANP	M=3000	9000
DIXON3DQ	N=10000	10000	DQDRTIC	N=5000	5000	DQRTIC	N=5000	5000
EDENSCH	N=2000	2000	EIGENALS	N=50	2550	EIGENBLS	N=50	2550
EIGENCLS	M=25	2652	ENGVAL1	N=5000	5000	FLETBV3M	KAPPA=0.0	5000
FLETCBV2	KAPPA=0.0	5000	FLETCBV3	KAPPA=0.0	5000	FLETCHBV	KAPPA=0.0	5000
FMINSRF2	P=100	10000	FMINSURF	P=75	5625	FREUROTH	N=5000	5000
GENHUMPS	ZETA=20.0	5000	INDEF	ALPHA=1000.0	5000	INDEFM	N=10000	10000
LIARWHD	N=10000	10000	MODBEALE	N/2=1000	2000	MOREBV	N=5000	5000
MSQRTALS	P=70	4900	MSQRTBLS	P=70	4900	NCB20	N=5000	5010
NCB20B	N=5000	5000	NONCVXU2	N=10000	10000	NONCVXUN	N=10000	10000
NONDIA	N=10000	10000	NONDQUAR	N=10000	10000	NONMSQRT	P=70	4900
OSCIGRAD	N=10000	10000	POWELLSG	N=10000	10000	POWER	N=5000	5000
QUARTC	N=10000	10000	SBRYBND	N=5000	5000	SCHMVETT	N=10000	10000
SCOSINE	N=10000	10000	SCURLY10	N=10000	10000	SCURLY20	N=10000	10000
SCURLY30	N=10000	10000	SINQUAD	N=10000	10000	SPARSINE	N=10000	10000
SPARSQUR	N=10000	10000	SPMSRTLS	M=3334	10000	SROSENBR	N/2=5000	10000
SSBRYBND	N=5000	5000	SSCOSINE	N=10000	10000	TESTQUAD	N=5000	5000
TOINTGSS	N=10000	10000	TQUARTIC	N=10000	10000	TRIDIA	DELTA=1.0	5000
WOODS	NS=2500	10000	YATP1LS	N=100	10200	YATP2LS	N=100	10200