Kuwait J. Sci. Eng. **28**(1) 2001

Rational-Lanczos technique for solving total least squares problems

MOHAMMEDI R. ABDEL-AZIZ

Department of Mathematics and Computer Science, Faculty of Science, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait
(On leave from Department of Mathematics, Faculty of Science, Alexandria University, Egypt.)

ABSTRACT

This paper aims to introduce an iterative algorithm for solving large scale total least squares problems. The algorithm is based on solving a sequence of linear systems and adjusting the minimum eigenvalue. The implicitly restarted Lanczos method is determined to be well suited for solving linear systems arising during the iterations. A rational interpolation scheme is developed for updating the minimum eigenvalue. A local convergence theory for this algorithm is presented. It is shown that this algorithm is faster than Newton's method.

Keywords: Convergence analysis; implicitly restarted Lanczos method; rational interpolation; total least squares problem.

AMS(MOS) Subject classification: *65F15, 65G05*

1. INTRODUCTION

The total least squares (TLS) technique is a generalization of the least squares technique (LS) for an overdetermined system of linear equations, $Ax \approx b$, where $A \in \Re^{m \times n}$ with $m > n$. In the LS framework, it is assumed that the matrix A is known exactly, but the vector b is corrupted by random errors. The LS solution (x_{LS}) is determined so that $\|b - Ax_{LS}\|_2 = min.$ (Björck 1996). Sometimes, a constraint is added to provide a balance between a smooth solution and a small residual (Barrlund 1998). However, if A is also corrupted by errors, then the TLS technique may be more appropriate. The basic formulation of the TLS problem is to

$$\begin{aligned} &\underset{[\hat{A}\hat{b}] \in \Re^{m\times(n+1)}}{\text{minimize}} \quad \|[A\ b] - [\hat{A}\ \hat{b}]\| \\ &\text{subject to } \hat{b} \in \text{Range}(\hat{A}), \end{aligned} \tag{1}$$

where the norm is the matrix Frobenius norm. Once a minimizing matrix $[\hat{A}\ \hat{b}]$ is found, then any x satisfying $\hat{A}x = \hat{b}$ is called a TLS solution (x_{TLS}).

A complete description of the TLS technique is given in Van Huffel and Vandewalle (1991), where many applications to system identification, signal processing and system response prediction are described. In many of these applications the matrix A has a special structure (Toeplitz) or errors occur only in a small number of its elements.

Many direct and iterative methods have been proposed to solve TLS problems. The generally used direct methods for solving TLS are based on the singular value decomposition (SVD) of A and $[A\ b]$. If the SVD of A and $[A\ b]$ are given by

$$
\begin{aligned}
A &= \hat{U}\hat{\Sigma}\hat{V}^t, & [A\ b] &= U\Sigma V^t \\
\hat{U} &= [\hat{u}_1\hat{u}_2\cdots\hat{u}_m], & U &= [u_1u_2\cdots u_m], \\
\hat{V} &= [\hat{v}_1\hat{v}_2\cdots\hat{v}_n], & V &= [v_1v_2\cdots v_{n+1}], \\
\hat{\Sigma} &= diag(\hat{\sigma}_1\hat{\sigma}_2\cdots\hat{\sigma}_n), & \Sigma &= diag(\sigma_1,\sigma_2,\ldots,\sigma_{n+1}), \\
\hat{U}^t\hat{U} &= I_m,\ \ \hat{V}^t\hat{V} = I_n, & U^tU &= I_m,\ \ V^tV = I_{n+1},
\end{aligned}
$$

where $\hat{\sigma}_n$ and $\sigma_{n+1}(\hat{\sigma}_n > \sigma_{n+1})$ are the smallest singular values of the matrices A and $[A\ b]$, then there exists a unique TLS solution (Golub & Van Loan 1980).

For applications where the matrix A has a special structure, the SVD-based methods may not always be appropriate, since they do not preserve the special structure of the matrix. Moreover, when A is large it may require a prohibitive amount of storage.

Iterative methods are appropriate techniques for solving large-scaled problems. Some of these techniques have been proposed in Kamm & Nagy (1998) and Björck (1997). In Kamm & Nagy (1998), a technique for solving TLS problem (1) with an $m \times n$ Toeplitz matrix A, has been developed through a modification of a method suggested in Cybenko & Van Loan (1986) for computing the smallest eigenvalue of a symmetric positive definite Toeplitz matrix.

The purpose of this paper is to describe and analyze an algorithm for solving large scale TLS problems. This algorithm is based on the Implicitly Restarted Lanczos method (IRL) to construct a basis for the Krylov subspace in conjunction with a rational interpolating scheme to update the smallest eigenvalue on that subspace. Our algorithm requires a fixed-size limited storage proportional to the size of the problem and relies only upon matrix-vector multiplications.

The scope of this work is as follows. In Section 2 we investigate the solution of problem (1) and the related results in terms of eigenproblems. Section 3 involves a variant of the k-step Arnoldi (IRL) method for solving large symmetric linear systems. A rational interpolating scheme and the statement of the algorithm are introduced in Section 4. Section 5 involves convergence properties of the introduced interpolating scheme. Finally, concluding remarks and further ideas are given in the last Section.

2. STRUCTURE OF THE TLS SOLUTION

Studying TLS problems in numerical analysis was started by Golub & Van Loan (1980). In 1991 Van Huffel and Vandewalle introduced their TLS book. Several authors proposed their direct and iterative algorithms for solving general and special structured TLS problems. The solution of the TLS problem (1) can be represented as follows:

- $[x^t_{TLS}, -1]^t = (-1/v_{n+1,n+1})v_{n+1}$, where $v_{n+1,n+1} = e^t_{n+1}Ve_{n+1}$, $v_{n+1} = Ve_{n+1}$, and $e_j = (0, 0, \ldots, 1)^t \in \Re^j$.
- the solution of the normal equations

  $$[H - \lambda_{n+1}I]x = h,$$

  where $H = A^tA$, $h = A^tb$, $\lambda_{n+1} = \sigma^2_{n+1}$ and λ_{n+1} is the smallest eigenvalue of the matrix $[A\ b]^t[A\ b]$.
  For simplicity we shall use $\{\lambda_j = \sigma^2_{n-j+2}\}_{j=1,2,,n+1}$ and $\{v_j = \hat{\sigma}^2_{n-i+1}\}_{i=1,2,\ldots,n}$, where $\{\lambda_j\}_{j=1,2,\ldots,n+1}$ and $\{v_j\}_{j=1,2,\ldots,n}$ are the increasing order of the eigenvalues of the matrices $[A\ b]^t[A\ b]$ and H and $\{\sigma_i\}_{i=n+1,\ldots,2,1}$ and $\{\hat{\sigma}_i\}_{i=n,\ldots,2,1}$ are the increasing order of the singular values of the matrices $[A\ b]$ and A.
- the eigenvector that has its last component equals -1 and associated with smallest eigenvalues λ_1 of the eigenproblem

  $$F(\lambda)z = O, \tag{2}$$

  where $F(\lambda) = \begin{bmatrix} H - \lambda I & h \\ h^t & \alpha - \lambda \end{bmatrix}$, $z = [x^t - 1]^t$ and $\alpha = \|b\|^2_2$.

For more details about these forms of x_{TLS}, we refer the reader to Van Huffle & Vanderwalle (1991).

The eigenproblem (2) can be written as a system of $(n + 1)$ nonlinear equations in x and λ:

$$\begin{bmatrix} f(x, \lambda) \\ g(x, \lambda) \end{bmatrix} = \begin{bmatrix} Hx - x - h \\ h^tx + \lambda - \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{3}$$

This system (3) is solved using Newton's method with x_{LS} as an initial guess of x_{TLS} (Björck 1997). Assuming that λ_0 is within $[\lambda_1, v_1)$, then the matrix $[H - v_jI]$ is always positive definite. Also (Björck 1997), the Rayleigh quotient $\rho(x) = (z^tF(0)z)/(z^tz)$ for the eigenvector z is introduced, where $F(0)$ is the matrix defined in (2). Quadratic and cubic convergences are provided using Newton and Rayleigh quotient methods, respectively. But, each of these methods requires two linear systems to be solved for each iteration.

The eigenproblem (2) is equivalent to

$$[H - \lambda I]x = h, \quad \text{and} \tag{4}$$

$$h^Tx - \alpha = -\lambda. \tag{5}$$

Combining formulas (4) and (5), we obtain the rational function

$$\omega(\lambda) = -\alpha + \lambda + h^t[H - \lambda I]^{-1}h. \tag{6}$$

Therefore, λ_1 is the smallest root of the nonlinear equation $\omega(\lambda) = 0$. Differentiating (6) with respect to $\lambda \in (0, \nu_1)$, we obtain

$$\omega'(\lambda) = 1 + h^t[H - \lambda I]^{-2}h > 1 \quad \text{and} \quad \omega''(\lambda) = 2h^t[H - \lambda I]^{-3}h > 0.$$

Hence, ω is strictly monotonically increasing and strictly convex on the interval $(0, \nu_1)$.

Using the same assumptions that have been used to solve system (3), Newton's iteration

$$\lambda_{p+1} = \lambda_p + \frac{-\alpha + h^t x + \lambda_p}{1 + x^t x}, \tag{7}$$

where $[H - \lambda_p I]x = h$ can be used to find λ_1. An analysis given by Cybenko & Van Loan (1986) shows that the sequences of the Newton's iteration (7) will remain within the interval $[\lambda_1, \nu_1)$ and will converge from right to λ. Therefore, the matrix $[H - \lambda_p I]$ is always positive definite.

Recasting the eigenproblem (2) in terms of problems (4) and (5) provides that the solution of the TLS problem (1) is the solution of the parameterized system of equations (4). In fact, a sequence of linear systems of equations has to be solved in order to iteratively adjust the smallest eigenvalue λ_1. Thus, a rapidly convergent interpolating scheme must be used to update λ_1, and an efficient linear-solver for computing x at certain values of λ_1, must be available.

3. THE LINEAR SOLVER

Krylov methods generate a sequence of iterates converging to the solution of the linear system

$$Gx = h, \tag{8}$$

where $G \in \Re^{n \times n}$ is nonsingular and $h \in \Re^n$. If x_0 is an initial guess for the true solution of (8), then letting $x = x_0 + s$, we have the equivalent system

$$Gs = r_0, \tag{9}$$

where $r_0 = h - Gx_0$ is the initial residual.

Let $\mathcal{K}^k = \mathcal{K}(G, r_0, k)$ be the Krylov subspace

$$\mathcal{K}(G, r_0, k) \equiv span\{r_0, Gr_0, G^2 r_0, \ldots, G^{k-1} r_0\}.$$

Then Krylov methods find an approximate solution

$$x_k = x_0 + s_k \quad \text{with} \quad s_k \in \mathcal{K}^k \quad \text{such that} \quad (h - Gx_k) \perp \mathcal{K}^k \equiv (r_0 - Gx_k) \perp \mathcal{K}^k.$$

The IRL method is one of the Krylov methods for large symmetric sparse matrices. It is based on Lanczos process to project the matrix onto a Krylov subspace in conjunction with a variant of the Graham–Schmidt QR factorization for small dense problems to maintain orthogonality between the Lanczos vectors (Golub & Van Loan 1996). The IRL method has been successfully used for solving nonlinear eigenvalue and optimization problems (Abdel-Aziz 1994, 2000b). Thus, we have determined this method to be well-suited for solving the linear systems that arise during the iterations in (4).

IRL begins with the specification of a starting vector $v_1 \in \Re^n$ and a real symmetric matrix $G \in \Re^{n\times n}$. In k-steps, the Lanczos process produces a decomposition of G in the form

$$GV_k = V_k T_k + r_k e_k^t, \tag{10}$$

where $V_k \in \Re^{n\times k}$, $V_k^t V_k = I_k$, $T_k \in \Re^{k\times k}$ is a symmetric tridiagonal matrix, $r_k \in \Re^n$ with $V_k^t r_k = 0$ is the residual vector and $e_k = (0, 0, \ldots, 1)^t \in \Re^k$. It is desirable for (10) to result in a zero residual, for $r_k = 0$ will imply that the columns of V_k form an orthonormal basis for an invariant subspace of G and that the eigenvalues of T_k are exact eigenvalues of G.

If we have selected $v_1 = V_k e_1$ in (10) to satisfy $\theta v_1 = h - Gs$, where s is a guess at the solution to (9) and e_1 is the first column of the identity matrix I_n, then solving $T_k y = \theta e_1$ and putting $s_+ = s + V_k y$ gives

$$\|h - Gs_+\| = \|h - Gs - GV_k y\| = \|GV_k y - \theta V_k e_1\|. \tag{11}$$

From decomposition (10), we obtain

$$GV_k y - V_k T_k y = r_k e_k^t y. \tag{12}$$

Combining formulas (12) and (11), we obtain

$$\|h - Gs_+\| = \|GV_k y - V_k T_k y\| = \|r_k\| |e_k^t y|. \tag{13}$$

Note that if $\|r_k\| = 0$ then s_+ will be an exact solution to the linear system (9). Moreover, if it is possible to iteratively force $\|r_k\| \to 0$ it will be possible to construct increasingly accurate approximate solutions.

IRL iteration involves repeated applications of polynomial filters to the starting vector and an in-place updating of the k-step Lanczos factorization. The approach repeatedly updates the starting vector, $\Phi(G)v_i \to v_1$, where the polynomial Φ is applied implicitly through a mechanism directly related to the implicitly shifted QR procedure. The polynomial is constructed to damp undesirable solution components from the starting vector forcing it into an invariant subspace. This leads to termination of the Lanczos sequence which begins with this starting vector in precisely k steps with $r_k = 0$. The construction and application of these polynomials and other related details are explained in Sorensen (1990).

With respect to the subject of this work, the major advantages of the IRL method are:

- Orthogonality of Lanczos vectors: It is computationally feasible to maintain orthogonality among the columns of V_k since the value of k is modest.
- Fixed space: In the IRL process, the number of Lanczos basis vectors never exceeds a pre-specified bound that is proportional to the dimension of the Krylov subspace. Moreover, as in the basic Lanczos process, only matrix-vectors multiplications are required with G. Peripheral storage of Lanczos vectors is not required.

4. RATIONAL INTERPOLATION SCHEME

Rational approximations have been used by the author to compute the smallest eigenvalue in solving constrained least squares and nonlinear eigenvalue problems (Abdel-Aziz 1994, 2000a).

In this section, we introduce an interpolating scheme that is based on the previous iterate and the first order information. Although Newton's method for solving $\omega(\lambda) = 0$ is q-quadratic convergent for an initial guess $\lambda_0 \in [\lambda_1, \nu_1)$, the global convergence behavior usually is not guaranteed. This is because ω is a rational function, where the root λ_1 and the pole ν_1 can be very close to each other. If λ_0 is close to ν_1 then the first steps of Newton's method can be extremely slow or it may not converge.

The convergence can be improved if an iterative method is based on a better model of the rational function ω than its tangent in Newton's method. Thus, the constructed function (6) can be written as

$$\omega(\lambda) = \beta_1 + \beta_2\lambda + \lambda^2\mu(\lambda) \quad \text{and} \quad \mu(\lambda) = \sum_{j=1}^{n} \frac{\gamma_j^2}{\nu_j - \lambda}, \tag{14}$$

where $\beta_1 = \omega(0) = -\alpha + h^T H^{-1} h$, $\beta_2 = \omega'(0) = 1 + \|H^{-1}h\|_2^2$ and γ_j are the components of h in the direction of the eigenvectors of H.

Our goal is to introduce an iterative method to update λ so that $\omega(\lambda) = 0$. To construct this iterative method, consider an interpolant of the form

$$\phi(\xi) = \beta_1 + \beta_2\xi + \xi^2 \frac{\beta_3}{\beta_4 - \xi}, \quad \text{where } \beta_4 \neq \xi. \tag{15}$$

Let x_{l+1} be the solution of the linear system $[\eta_l I - H]x = h$, where $\eta_l \in (0, \nu_1)$ is a given approximation to λ_1. It is natural to develop a method based on the interpolant (15) and the values x_{l+1} and η_l. This leads us to determine expressions for the coefficients β_3 and β_4 using the Hermitian interpolation conditions

$$\phi(\eta_l) = \omega(\eta_l) \quad \text{and} \quad \phi'(\eta_l) = \omega'(\eta_l). \tag{16}$$

Using formulas (16), (15) and (14), we obtain

$$\beta_3 = \frac{\mu^2(\eta_l)}{\mu'(\eta_l)} = \frac{(h^t x_{l+1})^2}{x_{l+1}^t x_{l+1}} \quad \text{and} \quad \beta_4 = \eta_l + \frac{\mu(\eta_l)}{\mu'(\eta_l)} = \eta_l + \frac{h^t x_{l+1}}{x_{l+1}^t x_{l+1}}. \tag{17}$$

Defining $\hat{\eta}$ such that $\hat{\eta} = \alpha - h^t x_{l+1}$, the updating formula for λ can be expressed by

$$\phi(\xi) = \omega(\hat{\eta}). \tag{18}$$

Substituting from formulas (15) and (14) in formula (18), we obtain

$$(\beta_3 - \beta_2)\xi^2 + (\Lambda + \beta_2\beta_4)\xi - \Lambda\beta_4 = 0, \quad \text{where } \Lambda = \hat{\eta}\beta_2 + \hat{\eta}^2\mu(\hat{\eta}). \tag{19}$$

Solving the quadratic equation (19), we obtain

$$\xi_1 = \frac{1}{\delta}(\Gamma + \sqrt{\Delta}) \text{ or } \xi_2 = \frac{1}{\delta}(\Gamma - \sqrt{\Delta}), \text{ where} \tag{20}$$

$$\delta = \beta_3 - \beta_2, \; \Gamma = -(\beta_2\beta_4 + \Delta),$$

$$\Delta = (\Lambda - \beta_2\beta_4)^2 + 4\Lambda\beta_3\beta_4. \tag{21}$$

The updated value η_{l+1} of λ will be the smallest positive root ξ_1 or ξ_2. If ξ_1 and ξ_2 are negative, then we take $\eta_{l+1} = \eta_l$. We have to make sure that $\eta_{l+1} < \nu_1$. If $\eta_{l+1} \geq \nu_1$, then we safeguard the iteration using the bisection method.

4.1. Statement of the Algorithm

The algorithm we shall introduce here is an iterative algorithm for solving large scale TLS problems (1). This algorithm involves two levels, In the first level, we use Lanczos factorization (10) to reduce the matrix $G = H - \lambda I$ of the linear system (4) into a tridiagonal matrix T and construct an orthonormal basis for the invariant subspace of G. We then solve the reduced system on that subspace. In the second level, we construct the rational interpolant (15) for the function (6). Then we use this interpolant to update the parameter λ.

The following describes a full iteration of this iterative algorithm.

Algorithm

(1) Initialization:
- (1.1) Choose: x_0, ε and $\eta_0 \in (0, \nu_1)$.
- (1.2) Compute: $H = A^t A$, $h = A^t b$ and $\alpha = \|b\|_2^2$.
- (1.3) Compute: $G = H - \eta_0 I$, β_1, β_2 and set $l = 0$.

(2) IRL Process:
- (2.1) Choose: s_0 (which can be chosen to be zero).
- (2.2) Form: $r_0 = h - Gs_0$, where $G = G(\eta_l)$.
- (2.3) Compute: $\|r_0\|_2$ and $v_1 = r_0/\|r_0\|_2$.
- (2.4) For $j = 1, 2, \ldots, k$ **Do**
  - Form Gv_j and orthogonalize it against $\{v_1, v_2, \ldots, v_j\}$ then
  - $t(i,j) = (Gv_j, v_i)$, $i = 1, 2, \ldots, j$
  - $\hat{v}_{j+1} = Gv_j - \sum_{i=1}^{j} t(i,j)v_1$
  - $t(j+1, j) = \|\hat{v}_{j+1}\|_2$. If $\|\hat{v}_{j+1}\|_2 = 0$, set $j = k$ and go to (3).
  - $v(j+1) = \hat{v}_{j+1}/t(j+1, j)$.
  - end Do.

(3) Working on the subspace:
- (3.1) Form: the tridiagonal matrix $T_k = [t(i,j)] \in \Re^{k\times k}$, where $1 \leq i \leq j$, $1 \leq j \leq k$ and the orthonormal matrix $V_k = [v_1, v_2, , \ldots, v_k]$.
- (3.2) Solve: $T_k y = \|r_0\|e_1$, for $y \in \Re^k$ where, $e_1 = I_k e_1$.
- (3.3) Compute: $x_{l+1} = x_l + s_k$, where $s_k = V_k y$.

(4) Updating $\lambda_{\min}$ using rational interpolation:
- (4.1) Compute: β_3 and β_4 using formula (17).
- (4.2) Compute: δ, Γ, Δ and Λ using formulas (19) and (21).

(4.3) Compute: ξ_1 and ξ_2 using formulas (20).
If $\xi_1 > 0$ and $\xi_2 > 0$, then $\eta_{l+1} = \min(\xi_1, \xi_2)$,
else if $\xi_1\xi_2 < 0$, then $\eta_{l+1} = \text{positive}(\xi_1\xi_2)$,
else $\eta_{l+1} = h^t x_l - \alpha$.
End if

(5) Safeguarding:
(5.1) If $\eta_{l+1} \geq \nu_1$, then $\eta_{l+1} = \frac{1}{2}(\eta_l + \eta_{l+1})$.

(6) Stopping rule:
(6.1) If $|\eta_{l+1} - \eta_l| \leq \epsilon\eta_l$,
then stop and set $x_{TLS} = x_{l+1}$
else set $\eta_l \leftarrow \eta_{l+1}$, $x_l \leftarrow x_{l+1}$,
$l \leftarrow l+1$ and go to 2.
End if

Remark. We left unspecified some computational issues such as the initial guesses x_0 and η_0. Details of these issues should be covered in the implementation of this work.

5. CONVERGENCE PROPERTIES

In this section, we introduce some basic properties of the rational interpolant $\phi(\xi)$.

Lemma 1. *Let $\eta \in (0, \nu_1)$ be an approximation of λ_1 computed by the proposed algorithm and the interpolant $\phi(\xi)$ as defined in (15). Then, $\beta_3 > 0$ and $\beta_4 - \eta > 0$.*

Proof. Using formulas (14) and (15) and the interpolation conditions (16), we obtain

$$\mu(\eta) = \frac{\beta_3}{\beta_4 - \eta} \quad \text{and} \quad \mu'(\eta) = \frac{\beta_3}{(\beta_4 - \eta)^2}. \tag{22}$$

Since $\eta < \nu_j$ for every j, we have

$$\mu(\eta) > 0 \quad \text{and} \quad \mu'(\eta) = \sum_{j=1}^{n} \frac{\gamma_j^2}{(\nu_j - \eta)^2} > 0. \tag{23}$$

The proof is completed by using formulas (22) and (23).

Lemma 2. *Let the assumptions of Lemma 1 hold. Then the $\phi(\varepsilon)$ interpolant is strictly monotonically increasing and strictly convex for every $\xi \in (0, \beta_4)$.*

Proof. Differentiating $\phi(\xi)$ with respect to ξ, we obtain

$$\phi'(\xi) = \beta_2 + 2\xi\frac{\beta_3}{\beta_4 - \xi} + \xi^2\frac{\beta_3}{(\beta_4 - \xi)^2},$$

$$\phi''(\xi) = \frac{2\beta_3}{(\beta_4 - \xi)} + 4\xi\frac{\beta_3}{(\beta_4 - \xi)^2} + 2\xi^2\frac{\beta_3}{(\beta_4 - \xi)^3}$$
$$= \frac{2\beta_3\beta_4^2}{(\beta_4 - \xi)^3}.$$

Using Lemma (1) and the value of β_2 we obtain

$$\phi'(\xi) > 0 \quad \text{and} \quad \phi''(\xi) > 0$$

which completes the proof.

Lemma 3. *Let the assumptions of Lemma 1 hold. Then the interpolant* $\phi(\xi) < 0$ *for all* $\eta \neq \lambda_1$.

Proof. Since λ_1 is a root of $\omega(\lambda)$, from (14) we obtain

$$\omega(\lambda_1) = \beta_1 + \beta_2\lambda_1 + \lambda_1^2\mu(\lambda_1) = 0. \tag{24}$$

Using (24), the evaluated interpolant ϕ at λ_1 can be written as

$$\phi(\lambda_1) = \beta_1 + \beta_2\lambda_1 + \lambda_1^2\frac{\beta_3}{\beta_4 - \lambda_1} = \lambda_1^2\left[\frac{\beta_3}{\beta_4 - \lambda_1} - \mu(\lambda_1)\right]. \tag{25}$$

Using formulas (22), we obtain

$$\beta_3 = \frac{\mu^2(\eta)}{\mu'(\eta)}, \quad \beta_4 - \eta = \frac{\mu(\eta)}{\mu'(\eta)}.$$

Therefore,

$$\frac{\beta_3}{\beta_4 - \lambda_1} = \frac{\beta_3}{(\beta_4 - \eta) + (\eta - \lambda_1)} = \frac{\mu^2(\eta)}{\mu(\eta) + (\eta - \lambda_1)\mu'(\eta)}. \tag{26}$$

But, for any $\eta \in (0, v_1)$, we have

$$\begin{aligned} \mu(\eta) + (\eta - \lambda_1)\mu'(\eta) &= \sum_{j=1}^{n}\frac{\gamma_j^2}{v_j - \eta} + \sum_{j=1}^{n}\frac{\gamma_j^2(\eta - \lambda_1)}{(v_j - \eta)^2} \\ &= \sum_{j=1}^{n}\frac{\gamma_j^2(v_j - \lambda_1)}{(v_j - \eta)^2} > 0. \end{aligned} \tag{27}$$

Substituting from (26) into (25), we deduce that the inequality $\phi(\eta) < 0$ is equivalent to the inequality

$$\mu(\lambda_1)[\mu(\eta) + (\eta - \lambda_1)\mu'(\eta)] - \mu^2(\eta) > 0. \tag{28}$$

From formulas (27) and (14), we obtain

$$\begin{aligned} \mu(\lambda_1)[\mu(\eta) + (\eta - \lambda_1)\mu'(\eta)] &= \left[\sum_{j=1}^{n}\frac{\gamma_j^2}{v_i - \lambda_1}\right]\sum_{j=1}^{n}\frac{\gamma_j^2(v_j - \lambda_1)}{(v_j - \eta)^2} \\ &= \sum_{i,j=1}^{n}\frac{\gamma_i^2\gamma_j^2(v_j - \lambda_1)}{(v_j - \lambda_1)(v_j - \eta)^2}. \end{aligned} \tag{29}$$

Using formula (29) in inequality (28), we obtain

$$\begin{aligned}\mu(\lambda_1)[\mu(\eta) + (\eta - \lambda_1)\mu'(\eta)] - \mu^2(\eta) &= \sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2(\nu_j - \lambda_1)}{(\nu_i - \lambda_1)(\nu_j - \eta)^2} - \sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2}{(\nu_i - \eta)(\nu_j - \eta)} \\ &= \sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2}{\nu_j - \eta}\left[\frac{\nu_j - \lambda_1}{(\nu_i - \lambda_1)(\nu_j - \eta)} - \frac{1}{\nu_i - \eta}\right] \\ &= \sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2(\nu_i - \nu_j)(\eta - \lambda_1)}{(\nu_i - \eta)(\nu_i - \lambda_1)(\nu_j - \eta)^2}.\end{aligned}$$

The sign of the last summation is the same as the sign of

$$\sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2(\eta - \lambda_1)}{(\nu_i - \eta)(\nu_j - \eta)}\left[\frac{\nu_i - \nu_j}{(\nu_i - \lambda_1)(\nu_j - \eta)} + \frac{\nu_j - \nu_i}{(\nu_j - \lambda_1)(\nu_i - \eta)}\right]. \tag{30}$$

But, $\eta < \nu_i$ for $i = 1, 2, \ldots, n$ and $\eta \neq \lambda_1$. Hence,

$$\sum_{i,j=1}^{n} \frac{\gamma_i^2\gamma_j^2(\eta - \lambda_1)^2(\nu_i - \nu_j)^2}{(\nu_i - \eta)^2(\nu_j - \eta)^2(\nu_i - \lambda_1)(\nu_j - \lambda_1)} > 0. \tag{31}$$

Since formulas (30) and (31) are equal, the proof is completed.

Lemma 4. *Let $\eta_l \in (\lambda_1, \nu_1)$ be a given approximation to λ_1 computed by the proposed algorithm. Then the rational function $\phi(\eta_l)$ has exactly one solution $\eta_{l+1} \in (\lambda_1, \eta_l)$.*

Proof. Since λ_1 is a root of $\omega(\lambda)$, from Lemma 3 we obtain

$$\phi(\xi) < 0 = \omega(\lambda_1) < \omega(\eta_l) = \phi(\eta_l).$$

The proof is completed from the fact that ϕ is strictly convex on the interval $(0, \eta_l)$.

Lemma 5. *Let the assumptions of Lemma 4 hold. Then the exact solution $\eta_{l+1} \in (\lambda_1, \eta_l)$ is always a better approximation to λ_1, than Newton's iterate with initial guess η_l.*

Proof. From the convexity of the rational function ϕ, we have

$$\begin{aligned}\phi(\xi) &> \phi(\eta_l) + \phi'(\eta_l)(\xi - \eta_l) \\ &= \omega(\eta_l) + \omega'(\eta_l)(\xi - \eta_l) \quad \text{for every } \xi \in (\lambda_1, \eta_l).\end{aligned}$$

Thus, η_{l+1} is always a better approximation to λ_1 than Newton's iterate.

The previous discussion together with the above lemmas establish the following:

Theorem: *For $\eta_0 \in (\lambda_1, \nu_1)$, the method which finds η_{l+1}, as a unique root of $\phi(\xi)$ in $(0, \eta_l)$ converges monotonically decreasing to λ_1 and it is guaranteed to be faster than Newton's method.*

6. CONCLUDING REMARKS

We have presented an algorithm for solving large-scale TLS problems. The algorithm is based on solving a sequence of linear systems and updating the smallest eigenvalues.

Convergence properties have been introduced for this algorithm. It is proved that this algorithm is convergent and it is guaranteed to be faster than Newton's method. Implementation and extension of this algorithm for solving some optimization problems will be an important topic for research.

7. ACKNOWLEDGEMENTS

This work is supported by Kuwait University grant number SM188.

REFERENCES

Abdel-Aziz, M.R. 1994. Safeguarded use of implicit restarted Lanczos technique for solving nonlinear structural eigensystems. International Journal of Numerical Methods in Engineering **37**: 3117–3133.

Abdel-Aziz, M.R. 2000a. Parameterized eigensolution technique for solving constrained least squares problems. International Journal of Computer Mathematics **75**: 481–495.

Abdel-Aziz, M.R. 2000b. Implicitly restarted projection algorithm for solving optimization problems. Journal of Numerical Functional Analysis and Optimization **21(3&4)**: 319–336.

Barrlund, A. 1998. Efficient solution of constrained least squares problems with kronecker product structure. SIAM Journal on Matrix Analysis and Applications **19(l)**: 154–160.

Björck, A. 1996. Numerical Methods for Least Squares Problems. SIAM, Philadelphia, PA, U.S.A.

Björck, A. 1997. Newton and Rayleigh quotient methods for total least squares problems. In: **Van Huffel, S. (Ed.).** Proceedings of the second international workshop on total least squares and errors-in-variables modeling. SIAM, Philadelphia, PA, U.S.A. Pp. 149–160.

Cybenko, C. & Van Loan, C. 1986. Computing the minimum eigenvalue of a symmetric positive definite Toeplitz matrix. SIAM Journal of Scientific Computing **7(l)**: 123–131.

Golub, G.H. & Van Loan, C.F. 1980. An analysis of the total least squares problem. SIAM Journal of Numerical Analysis **17**: 883–893.

Golub, G.H. & Van Loan, C.F. 1996. Matrix Computations. Third edition. The Johns Hopkins University Press, Baltimore, MD, U.S.A.

Kamm, J. & Nagy, J.G. 1998. A total least squares method for Toeplitz systems of equations. BIT **38(3)**: 560–582.

Sorensen, D.C. 1990. Implicit application of polynomial filters in a k-step Arnoldi method. SIAM Journal on Matrix Analysis and Applications **13(l)**: 357–385.

Van Huffel, S. & Vandewalle, J. 1991. The Total Least Squares Problem: Computational Aspects and Analysis. SIAM, Philadelphia, PA, U.S.A.

(Accepted 5 August 2000)

تقنية لانشوس النسبية لحل مشكلات المربعات الصغرى الكلية

محمدي رضوان عبدالعزيز
قسم الرياضيات وعلم الحاسوب -جامعة الكويت
ص.ب. 5969 - الصفاة 13060- الكويت

خلاصة

ترجع أهمية مشكلات المربعات الصغرى الكلية إلى ثبوت تأثيراتها في العديد من التطبيقات . تطبق هذه الطريقة على أي مجموعة من المعادلات الخطية حيث أن كل من مصفوفة المعاملات والمتجه الواقع في الطرف الأيمن معلومين ويحتويان على نسبة خطأ . في هذه الطريقة نحن لا نفكر في استخدام الطرق العادية الموجودة في التحليل العددي وذلك لأن الأخطاء المتراكمة هي المنبع الوحيد لعدم الدقة . إننا نفكر في عدم الدقة الموروثة من خلال عملية قياس البيانات . لذلك فإن الهدف الأساسي من هذا البحث هو دراسة مشكلة المربعات الصغرى الكلية ذات الحجم الكبير . ويتضمن البحث إعادة صياغة المشكلة لتحويلها إلى مشكلة قيمة مميزة بطريقة متقاربة . لإنجاز الحل الأمثل تم التعديل على طريقة لانشوس لاستخدامها في حل مجموعات المعادلات الخطية التي تظهر ولحساب أصغر قيمة مميزة للمصفوفة استخدمنا الاستكمال النسبي في تقريب القيمة المميزة الصغرى وأثبتنا أن هذا التقريب متقارب.

تتطلب هذه الطريقة مكان ثابت للتخزين في الحاسوب على أن يتناسب هذا المكان مع حجم مشكلة المربعات الصغرى الكلية التي يجري حلها وتعتمد على عمليات ضرب المصفوفات في متجه.