An International Journal
computers & mathematics with applications

PERGAMON Computers and Mathematics with Applications 37 (1999) 49–66

Numerical Algorithms for Undamped Gyroscopic Systems

W. R. FERNG
Department of Applied Mathematics, National Chiao Tung University
Hsin-Chu, Taiwan, R.O.C.
ferng@math.nctu.edu.tw

WEN-WEI LIN
Department of Mathematics, National Tsing Hua University
Hsin-Chu, Taiwan, R.O.C.
wwlin@am.nthu.edu.tw

CHERN-SHUH WANG
Department of Mathematics, National Cheng Kung University
Tainan, Taiwan, R.O.C.
cswang@math1.math.ncku.edu.tw

(Received July 1996; revised and accepted April 1998)

Abstract—The solutions of a gyroscopic vibrating system oscillating about an equilibrium position, with no external applied forces and no damping forces, are completely determined by the quadratic eigenvalue problem $(-\lambda_i^2 M + \lambda_i G + K)x_i = 0$, for $i = 1, \dots, 2n$, where M, G, and K are real $n \times n$ matrices, and M is symmetric positive definite (denoted by $M > 0$), G is skew symmetric, and either $K > 0$ or $-K > 0$. Gyroscopic system in motion about a stable equilibrium position (with $-K > 0$) are well understood. Two Lanczos-type algorithms, the pseudo skew symmetric Lanczos algorithm and the J-Lanczos algorithm, are studied for computing some extreme eigenpairs for solving gyroscopic systems in motion about an unstable equilibrium position (with $K > 0$). Shift and invert strategies, error bounds, implementation issues, and numerical results for both algorithms are presented in details. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords—Gyroscopic system, Lanczos algorithm, Hamiltonian matrix, Quadratic eigenvalue problem, Generalized eigenvalue problem.

1. INTRODUCTION

The motion arising in the gyroscopic vibrating systems oscillating about an equilibrium position, with no external applied forces and no damping forces, modeled with the finite element method leads to systems of constant coefficient differential equations

$$-M\ddot{u}(t) + G\dot{u}(t) + Ku(t) = 0, \tag{1}$$

in which M, G, and K are real $n \times n$ matrices. The leading coefficient matrix M is symmetric and positive definite (denoted by $M > 0$) and is generally the mass matrix of the quadratic form

This work was supported in part by the National Science Council, Taiwan.
Part of this research was performed while the authors were visiting the Boeing Computer Services at Bellevue, Washington, U.S.A. This work was inspired by J. Lewis while he was visiting Tsing Hua University, Taiwan. The authors would thank him for his valuable comments and discussions. The authors are also grateful to D. J. Pierce and his family and R. Grimes for their warm hospitality.

0898-1221/98/$ - see front matter © 1998 Elsevier Science Ltd. All rights reserved. Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII: S0898-1221(98)00241-7

determining the kinetic energy of the system. G is skew symmetric ($G^\top = -G$), representing the effect of gyroscopic internal forces. K represents the stiffness matrix and is symmetric and either $K > 0$ or $-K > 0$, when oscillations may be about unstable or stable equilibrium positions, respectively. For example, the Lagrangian function of the motion of a sleeping symmetric top leads to the gyroscopic system (1) (see [1,2]).

A gyroscope can be defined as a rotating body possessing one axis of symmetry and whose rotation about the symmetry axis is relatively large compared with the rotation about any other axes. In modern usage, a gyroscope is a system consisting of a symmetric rotor spinning rapidly about its symmetry axis and free to move about one or two perpendicular axes. A considerable number of spinning bodies can be regarded as gyroscopes, for instance, helicopter rotor blades or spin stabilized satellites with elastic appendages such as solar panels or antennas. It is therefore no surprise that the subject of gyroscopic motion and devices has received so much attention.

The natural vibration frequencies λ_i and the corresponding mode shapes x_i of the system (1) are of prime interest to the designer, and can completely be determined by the quadratic eigenvalue problem

$$\left(-\lambda_i^2 M + \lambda_i G + K\right) x_i = 0, \tag{2}$$

for $i = 1, \dots, 2n$. This problem can be reduced to a linear generalized eigenvalue problem by doubling the order of the system

$$\begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} x_i \\ \lambda_i x_i \end{bmatrix} - \frac{1}{\lambda_i} \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} x_i \\ \lambda_i x_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{3}$$

for $i = 1, 2, \dots, 2n$. For complex structures, the order n of these matrices is usually large and the structure of these matrices is often sparse. Hence, numerical methods based on modifying the entries of the matrices, e.g., QR or QZ method, become less practical. The subspace method, on the other hand, developed originally by Clint and Jennings [3] is well established and is commonly used for such large eigenproblems. However, more recently, the Lanczos type methods have been gaining increasing popularity since from many numerical observations they are considerably more efficient than the subspace iteration method.

When $-K > 0$, the gyroscopic system in motion about a stable equilibrium position is well understood (see, e.g., [4–8]). All nonzero eigenvalues of (2) are known to be necessarily pure imaginary and semisimple (the case when the eigenvalue has only linear elementary divisors). Thus, the system is weakly stable, that is, $\|u(t)\|$ is uniformly bounded for all $t \geq 0$ and all solutions $u(t)$ of (1). Also in this case the matrix $\left[\begin{smallmatrix} -K & 0 \\ 0 & M \end{smallmatrix}\right]$ in (3) is symmetric positive definite. Suitable numerical algorithms can be derived with less difficulty for solving such problem [9].

It is equally important to investigate the gyroscopic system in motion about an unstable equilibrium position with $K > 0$ (see [8,10–12]). A classical treatment of this kind was given by Whittaker [2]. Stability criteria for such system were shown in [10,13]. It is also known that the spectrum of the quadratic eigenproblem is symmetric about the imaginary axis (see, e.g., [11]). However, unlike the previous case, the numerical algorithms for this particular application is relatively limited in literature. The fact that matrix $\left[\begin{smallmatrix} -K & 0 \\ 0 & M \end{smallmatrix}\right]$ in (3) becomes indefinite imposes greater difficulty on the numerical computation. Some standard numerical methods, such as Arnoldi or unsymmetric Lanczos method, may be applied without modification; however, they do not take account of the special structure of the coefficient matrices.

In [14], Parlett and Chen considered the quadratic eigenvalue problem with symmetric positive definite coefficient matrices. They reduced the problem to linear form and derived a two-sided Lanczos algorithm which retains symmetry in the new coefficients but not the positive definiteness. Recently Rajakumar [15] employed the Lanczos two-sided recursion to solve the quadratic eigenvalue problem when M, G, and K are real and unsymmetric. Motivated by their successful implementation of the algorithms, we seek to devise the Lanczos type algorithms that can be directly applied to the quadratic eigenvalue problem (2) arising from the gyroscopic system in motion about an unstable equilibrium position.

Two Lanczos type algorithms are studied in this paper. The first approach, we name it pseudo skew symmetric Lanczos algorithm, extends the idea of Parlett and Chen [14] to skew symmetric coefficient matrix case. As a result, a two-term recurrence formula is derived to produce a skew tridiagonal matrix using an indefinite inner product induced by $\begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix}$. The second approach transforms the quadratic eigenvalue problem to a $2n \times 2n$ linear Hamiltonian eigenvalue problem, then applies the J-Lanczos algorithm [16]. The J-Lanczos algorithm uses a sequence of symplectic transformations to produces a Hamiltonian tridiagonal matrix so that the eigenvalue quartet $\{\mu, \bar{\mu}, -\mu, -\bar{\mu}\}$ always converges at the same time. The symplectic look-ahead Lanczos algorithm proposed by Freund and Mehrmann [17,18] may also be applied here. We emphasize that, with careful implementation, both approaches avoid the need to work explicitly with matrices of order $2n$.

We organize this paper as follows. In Section 2, we derive the two-sided recursion and complex shifts formulations for the pseudo skew symmetric Lanczos algorithm. In Section 3, we show how to transform the quadratic eigenvalue problem (2) to an enlarged linear Hamiltonian eigenvalue problem and apply the J-Lanczos algorithm. Complex shift formulations which preserve the Hamiltonian structure are given. Also included in Sections 2 and 3 are error bounds, convergence behavior, computational cost, and implementation issues for each approach. Numerical experiments are presented in Section 4. Conclusion and remarks follow in Section 5. Throughout this paper, unless otherwise stated, we only consider the gyroscopic system with K positive definite.

2. PSEUDO SKEW SYMMETRIC LANCZOS ALGORITHM

By letting

$$H = \begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix}, \tag{4}$$

equation (3) can simply be expressed as

$$Hz = \frac{1}{\lambda} Az. \tag{5}$$

It is easily seen that H is skew symmetric and A is symmetric but indefinite. In [14] Parlett and Chen presented the pseudo symmetric Lanczos algorithm for the case where H is symmetric and A is symmetric but indefinite. They use a three-term recurrence formulation to produce an unsymmetric tridiagonal matrix. We extend their method to our problem (where H is skew symmetric) and derive a Lanczos type method, called pseudo skew symmetric Lanczos algorithm, with the operator $A^{-1}H$ using an indefinite inner product defined by A, namely, $(u, v)_A = u^\top Av$, for all $u, v \in \mathbf{R}^n$). In contrast to their formulation, the pseudo skew symmetric Lanczos algorithm uses a two-term recurrence formula to produce a skew symmetric tridiagonal matrix.

We start from the theory presented in [19] that there almost always exists a nonsingular $2n \times 2n$ matrix $Q_{2n} \equiv [q_1, \dots, q_{2n}]$ such that

$$Q_{2n}^\top H Q_{2n} = T_{2n} \tag{6}$$

and

$$Q_{2n}^\top A Q_{2n} = \Omega_{2n}, \tag{7}$$

where

$$\Omega_{2n} = \operatorname{diag}(\omega_1, \omega_2, \dots, \omega_{2n}) \tag{8}$$

is diagonal and

$$T_{2n} = \begin{bmatrix} 0 & \beta_2 & & & \\ -\beta_2 & 0 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_{2n} \\ & & & -\beta_{2n} & 0 \end{bmatrix} \tag{9}$$

is skew tridiagonal. Then we have the two-term recurrence in matrix form

$$A^{-1}HQ_{2n} = Q_{2n}\Omega_{2n}^{-1}T_{2n}. \tag{10}$$

As suggested in [14] for the pseudo symmetric Lanczos algorithm, we require the Lanczos vectors q_j to satisfy

$$\|q_k\|_2 = 1, \qquad \text{for } k = 1, 2, \dots, 2n. \tag{11}$$

Equating the k^{th} column on each side of (10) and letting $\beta_1 \equiv 0$ yields the two-term recurrence for our pseudo skew symmetric Lanczos algorithm:

$$A^{-1}Hq_k = \frac{\beta_k}{\omega_{k-1}}q_{k-1} - \frac{\beta_{k+1}}{\omega_{k+1}}q_{k+1}, \tag{12}$$

with

$$\beta_{k+1} = q_k^{\top}Hq_{k+1}, \tag{13}$$

$$\omega_{k+1} = (q_{k+1},\, q_{k+1})_A = q_{k+1}^{\top}Aq_{k+1}, \tag{14}$$

for $k = 1, 2, \dots, 2n-1$. Since in each iteration k there is only one Lanczos vector q_{k+1} generated, we use the term one *Lanczos step* to denote one iteration in the pseudo skew symmetric Lanczos algorithm.

Dangers

Equation (12) above shows the possible danger of small values among the $\{\omega_k\}$. Hence, the pseudo skew symmetric Lanczos algorithm using (11)–(14) may break down and may be unstable when close to breakdown.

In addition, undetected growth of Lanczos vectors in certain directions can occur. Similar to the tactical example shown in [14], we construct the following example. Let

$$q_1 = (\sigma, 0, \sigma, 0, x, \dots, x)^{\top},$$
$$A = \operatorname{diag}(1, 1, -1, -1, x, \dots, x),$$

and

$$H = \operatorname{diag}\left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & x \\ -x & 0 \end{bmatrix}, \dots, \begin{bmatrix} 0 & x \\ -x & 0 \end{bmatrix}\right).$$

Then from (12) it follows that

$$q_2 = (0, \sigma, 0, \sigma, x, \dots, x)^{\top}.$$

From (13) and (14), it is easily seen that ω_{k+1} and β_{k+1} are independent of σ, and consequently both T_k and Ω_k are independent of σ. But the subspace spanned by $u_1 = (1, 0, 1, 0, 0, \dots, 0)^{\top}$ and $u_2 = (0, 1, 0, 1, 0, \dots, 0)^{\top}$ forms an invariant subspace for the matrix pencil (H, A) corresponding to the eigenmatrix $\left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$. When σ is arbitrarily large, the subspace spanned by the Lanczos vectors q_1 and q_2 is close to the invariant subspace of u_1 and u_2. But the pseudo skew symmetric Lanczos algorithm is blind to detect them when $\Omega_k^{-1}T_k$ is used to compute the desired Ritz pairs.

How to Compute $T_k - \Lambda\Omega_k$

At each even step $k = 2j$, the QR-algorithm (in EISPACK, for example) can be used to compute the Ritz values and associated vectors of $\Omega_k^{-1}T_k$. The algorithm is stable but takes no advantage of the compact tridiagonalization form. On the other hand, let

$$\triangle_k = \operatorname{diag}\left(\sqrt{|\omega_1|}, \ldots, \sqrt{|\omega_k|}\right)$$

and

$$\widetilde{I}_k = \operatorname{diag}\left(\frac{\omega_1}{|\omega_1|}, \ldots, \frac{\omega_k}{|\omega_k|}\right).$$

Then the pencil $(\triangle_k^{-1}T_k\triangle_k^{-1}, \widetilde{I}_k)$ is equivalent to (T_k, Ω_k) and the HR algorithm [19], which is regarded as a generalization of the skew symmetric tridiagonal QR algorithm, can be applied to solve this pencil. However, HR algorithm is unstable by using hyperbolic rotations and may break down for certain shift values [19]. Furthermore, since $\Omega_k^{-1}T_k$ is skew symmetric, one can verify that if θ is an eigenvalue, then $-\theta$ is also an eigenvalue in theory. However, neither the QR algorithm nor the HR algorithm can guarantee to find both θ and $-\theta$ at one time.

Error Bounds

Suppose

$$(T_j - \theta\Omega_j)s = 0,$$

with $\|s\|_2 = 1$, defines a typical eigenpair (θ, s) of the reduced problem obtained by applying the pseudo skew symmetric Lanczos algorithm to the operator $A^{-1}H$. Let $y = Q_j s$ be a Ritz vector corresponding to the Ritz value θ. After j iterations, in exact arithmetic, one has

$$A^{-1}Hy - \theta y = -\left(q_{j+1}\frac{\beta_{j+1}}{\omega_{j+1}}e_j^{\top}\right)s. \tag{15}$$

Suppose $(\zeta, \tilde{s})$ is another eigenpair of $\Omega_j^{-1}T_j$ such that ζ is the computed eigenvalue that is closest to $-\theta$. We may argue that $|\zeta + \theta| \le O(\epsilon)$, where ϵ is the machine unit roundoff, and $\zeta = -\theta$ in exact arithmetics. Let $z = Q_j\tilde{s}$. Then, similar to (15), one has

$$A^{-1}Hz - \zeta z = -\left(q_{j+1}\frac{\beta_{j+1}}{\omega_{j+1}}e_j^{\top}\right)\tilde{s}. \tag{16}$$

Since A is symmetric and H is skew symmetric, by adding θz and premultiplying by H to (16), one obtains

$$\left(A^{-1}H\right)^{\top}(Hz) - \theta(Hz) = -\left(Hq_{j+1}\frac{\beta_{j+1}}{\omega_{j+1}}e_j^{\top}\right)\tilde{s} + (\zeta + \theta)(Hz). \tag{17}$$

Thus, Hz can be viewed as an approximated "left Ritz vector". Note that

$$\|y\|_2^2 = \left|s^H Q_j^{\top} Q_j s\right| \le j\left|s^H s\right| = j.$$

Similarly, $\|z\|_2^2 \le j$.

Next we define

$$\delta(j) = \max\left\{\left|e_j^{\top}s\right|,\ \left|e_j^{\top}\tilde{s}\right| + \sqrt{j}\,|\zeta + \theta|\right\}. \tag{18}$$

Then (θ, y, Hz) is an eigentriplet of $A^{-1}H - E$ for some E satisfying

$$\begin{aligned}\|E\|_2 &\le \frac{\beta_{j+1}}{\omega_{j+1}}\max\left\{\frac{|e_j^{\top}s|}{\|y\|_2},\ \frac{|e_j^{\top}\tilde{s}|\,\|Hq_{j+1}\|_2 + |\zeta+\theta|\|Hz\|_2}{\|Hz\|_2}\right\}\\ &\le \delta(j)\,\frac{\beta_{j+1}}{\omega_{j+1}}\max\left\{\frac{1}{\|y\|_2},\ \frac{\|H\|_2}{\|Hz\|_2}\right\}.\end{aligned} \tag{19}$$

When the computation has proceeded enough that $|e_j^\top s|$, $|e_j^\top \tilde{s}|$, and $|\zeta + \theta|$ tend to zero so that $\delta(j)\beta_{j+1}/\omega_{j+1}$ is very small, then first-order perturbation theory may be invoked to obtain an accurate error estimate for θ by regarding $A^{-1}H$ as a perturbation to $A^{-1}H - E$. If λ is the eigenvalue of $A^{-1}H$ closest to θ, we have, in general,

$$|\lambda - \theta| = \frac{\|y\|_2 \cdot \|Hz\|_2}{|z^\top Hy|} \cdot \frac{|z^\top HEy|}{\|y\|_2 \cdot \|Hz\|_2} + O\left(\|E\|_2^2\right). \tag{20}$$

Note that

$$z^\top Hy = \tilde{s}^\top Q_j H Q_j s = \tilde{s} T_j s.$$

Now we can obtain an error bound, as $\|E\|_2 \to 0$, using (19) and (20),

$$\begin{aligned} |\lambda - \theta| &\le \frac{\|y\|_2 \cdot \|Hz\|_2}{|z^\top Hy|} \|E\|_2 + O\left(\|E\|_2^2\right) \\ &\le \frac{\|y\|_2 \cdot \|Hz\|_2}{|z^\top Hy|} \delta(j) \frac{\beta_{j+1}}{\omega_{j+1}} \max\left\{\frac{1}{\|y\|_2}, \frac{\|H\|_2}{\|Hz\|_2}\right\} + O\left(\|E\|_2^2\right) \\ &\le \delta(j) \frac{\beta_{j+1}}{\omega_{j+1}} \frac{\sqrt{j}}{|\tilde{s}^\top T_j s|} \|H\|_2 + O\left(\|E\|_2^2\right). \end{aligned} \tag{21}$$

Similar to the error bound derived for the pseudo symmetric Lanczos algorithm [14], the attraction of the last inequality is that the dominant term can be calculated at step j, without recourse to n-vectors, provided that $\|H\|_2$ is provided along with the subroutine that multiplies vectors by H.

Complex Shifts

Let $H - \mu A$ be the pencil defined in (4). We consider the following complex shifts for the proposed pseudo skew symmetric Lanczos algorithm.

TYPE I. Pure imaginary single shift $\sigma = i\eta$. By taking the real operator with complex shift σ on the pencil we get the shifted problem

$$\operatorname{Re}\left[(H - \sigma A)^{-1}\right] Az = \frac{1}{2}\left(\frac{1}{\mu - \sigma} + \frac{1}{\mu - \bar{\sigma}}\right) z. \tag{22}$$

Notice that the operator $(H - \sigma A)^{-1}$ can be factored into

$$\begin{bmatrix} I & 0 \\ \frac{1}{\sigma} I & I \end{bmatrix} \begin{bmatrix} G_\sigma & 0 \\ 0 & -\sigma M \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\sigma} I \\ 0 & I \end{bmatrix}, \tag{23}$$

where $G_\sigma = G + \sigma K - (1/\sigma)M$. Since $\sigma = i\eta$, $\operatorname{Re}[(H - \sigma A)^{-1}]$ is also skew symmetric. Thus, the proposed pseudo skew symmetric Lanczos algorithm can be applied to the pencil $A(\operatorname{Re}[(H - \sigma A)^{-1}])A - \mu A$.

TYPE II. Complex single shift. If $\operatorname{Re}(\sigma) \neq 0$, then $\operatorname{Re}[(H - \sigma A)^{-1}]$ is neither skew symmetric nor symmetric. The pseudo skew symmetric or symmetric Lanczos algorithm can no longer be applied. For this case, we can only use the unsymmetric Lanczos (two-sided Lanczos) algorithm [14] to the unsymmetric matrix $(\operatorname{Re}[(H - \sigma A)^{-1}])A$.

TYPE III. Pure imaginary double shift $\sigma = i\eta$. We consider the shifted problem

$$A(H - \bar{\sigma} A)^{-1} A (H - \sigma A)^{-1} Az = \frac{1}{(\mu - \sigma)(\mu - \bar{\sigma})} Az. \tag{24}$$

Since $\sigma = i\eta$ is pure imaginary, both $A(H-\bar{\sigma}A)^{-1}A(H-\sigma A)^{-1}A$ and A are symmetric, one can apply the pseudo symmetric Lanczos algorithm [14] to the operator $(H-\bar{\sigma}A)^{-1}A(H-\sigma A)^{-1}A$. Note that, similar to (23), $(H-\bar{\sigma}A)^{-1}$ can be factored as

$$\begin{bmatrix} I & 0 \\ \frac{1}{\bar{\sigma}}I & I \end{bmatrix} \begin{bmatrix} G_{\bar{\sigma}} & 0 \\ 0 & -\bar{\sigma}M \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\bar{\sigma}}I \\ 0 & I \end{bmatrix}, \tag{25}$$

where $G_{\bar{\sigma}} = G + \bar{\sigma}K - (1/\bar{\sigma})M$.

TYPE IV. Complex double shift. When $\mathrm{Re}(\sigma) \neq 0$, we consider the shift matrix $(H-\bar{\sigma}A)^{-1}A(H-\sigma A)^{-1}A$ which is a real matrix but is unsymmetric. Therefore, one has to use the unsymmetric Lanczos algorithm in this case.

Implementation Issues

For clarity, we summarize the pseudo skew symmetric Lanczos method in the following algorithm and point out the implementation details.

Given q_1.

Initialize $q_0 = 0$, $\omega_0 = 1$, and $\beta_1 = 0$.

Compute $\omega_1 = q_1^{\mathsf{T}} A q_1$.

For $k = 1, 2, \ldots,$ until converge.

- $r_k = \dfrac{\beta_k}{\omega_{k-1}} q_{k-1} - A^{-1}Hq_k$
- $q_{k+1} = \dfrac{r_k}{\|r_k\|_2}$
- $\beta_{k+1} = q_k^{\mathsf{T}} H q_{k+1}$
- $\omega_{k+1} = q_{k+1}^{\mathsf{T}} A q_{k+1}$

End

In the computer implementation, matrices A and H are never assembled and $A^{-1}H$ is not formed explicitly. However, one has to factor K (e.g., LU-factorization), and only K, so that the computations involving K^{-1} can be performed efficiently as will be seen in (26) in the following.

For convenience, we partition a $2n$-vector into two n-subvectors. For example, we denote $q_k = \begin{bmatrix} q_k^{(1)} \\ q_k^{(2)} \end{bmatrix}$, where $q_k^{(1)}, q_k^{(2)}$ are n-vectors. At the k^{th} Lanczos step, r_k is formulated as $r_k = (\beta_k/\omega_{k-1})q_{k-1} - A^{-1}Hq_k$, and one can compute r_k by

$$\begin{aligned} \begin{bmatrix} r_k^{(1)} \\ r_k^{(2)} \end{bmatrix} &= \frac{\beta_k}{\omega_{k-1}} \begin{bmatrix} q_{k-1}^{(1)} \\ q_{k-1}^{(2)} \end{bmatrix} - \begin{bmatrix} K^{-1} & 0 \\ 0 & M^{-1} \end{bmatrix} \begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} q_k^{(1)} \\ q_k^{(2)} \end{bmatrix} \\ &= \frac{\beta_k}{\omega_{k-1}} \begin{bmatrix} q_{k-1}^{(1)} \\ q_{k-1}^{(2)} \end{bmatrix} - \begin{bmatrix} K^{-1}\left(Gq_k^{(1)} - Mq_k^{(2)}\right) \\ q_k^{(1)} \end{bmatrix}. \end{aligned} \tag{26}$$

We may assume that the vectors $Gq_k^{(1)}$ and $Mq_k^{(2)}$ have been computed in the last iteration and are available at the present stage. In this iteration, they will be updated to $Gq_{k+1}^{(1)}$ and $Mq_{k+1}^{(2)}$ (see (27) below) for use in the next iteration. Therefore, only $K^{-1}v_k$, where $v_k = Gq_k^{(1)} - Mq_k^{(2)}$, has to be computed at the current step.

Now we examine the computations involved in computing $\beta_{k+1} = q_k^{\mathsf{T}} H q_{k+1}$. Since

$$\begin{aligned} \beta_{k+1} &= \begin{bmatrix} q_k^{(1)} \\ q_k^{(2)} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} q_{k+1}^{(1)} \\ q_{k+1}^{(2)} \end{bmatrix} \\ &= \left(q_k^{(1)}\right)^{\mathsf{T}} \left(Gq_{k+1}^{(1)}\right) - \left(q_k^{(1)}\right)^{\mathsf{T}} \left(Mq_{k+1}^{(2)}\right) + \left(q_{k+1}^{(1)}\right)^{\mathsf{T}} \left(Mq_k^{(2)}\right), \end{aligned} \tag{27}$$

one computes the matrix-vector multiplications $Gq_{k+1}^{(1)}$ and $Mq_{k+1}^{(2)}$. The resulting vectors are stored for computing ω_{k+1} and passed to the next Lanczos step. In computing $\omega_{k+1} = q_{k+1}^\top A q_{k+1}$, we have

$$\begin{aligned} \omega_{k+1} &= \begin{bmatrix} q_{k+1}^{(1)} \\ q_{k+1}^{(2)} \end{bmatrix}^\top \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} q_{k+1}^{(1)} \\ q_{k+1}^{(2)} \end{bmatrix} \\ &= -\left(q_{k+1}^{(1)}\right)^\top \left(Kq_{k+1}^{(1)}\right) + \left(q_{k+1}^{(2)}\right)^\top \left(Mq_{k+1}^{(2)}\right), \end{aligned} \tag{28}$$

hence, only $Kq_{k+1}^{(1)}$ is computed.

In summary, the dominant computations in the pseudo skew symmetric Lanczos algorithm are those matrix-vector operations including $K^{-1}v_k$, $Kq_{k+1}^{(1)}$, $Gq_{k+1}^{(1)}$, and $Mq_{k+1}^{(2)}$.

Next we consider the single complex shift $\sigma = i\eta$. From (23),

$$\begin{aligned} \operatorname{Re}\left[(H-\sigma A)^{-1}\right] &= \operatorname{Re}\left(\begin{bmatrix} I & 0 \\ \frac{1}{\sigma} I & I \end{bmatrix} \begin{bmatrix} G_\sigma & 0 \\ 0 & -\sigma M \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\sigma} I \\ 0 & I \end{bmatrix}\right) \\ &= \begin{bmatrix} \operatorname{Re}\left(G_\sigma^{-1}\right) & \frac{1}{\eta}\operatorname{Im}\left(G_\sigma^{-1}\right) \\ -\frac{1}{\eta}\operatorname{Im}\left(G_\sigma^{-1}\right) & \frac{1}{\eta^2}\operatorname{Re}\left(G_\sigma^{-1}\right) \end{bmatrix}, \end{aligned} \tag{29}$$

and r_k is computed by

$$\begin{aligned} r_k &= \frac{\beta_k}{\omega_{k-1}} q_{k-1} - A^{-1}A \operatorname{Re}\left[(H-\sigma A)^{-1}\right] A q_k \\ &= \frac{\beta_k}{\omega_{k-1}} q_{k-1} - \operatorname{Re}\left[(H-\sigma A)^{-1}\right] A q_k. \end{aligned} \tag{30}$$

We may assume that $f_k \equiv Aq_k$ and $\operatorname{Re}[(H-\sigma A)^{-1}]f_k$ have been computed in the previous Lanczos step and are available. Hence only vector operation is required for computing r_k. For β_{k+1}, since

$$\begin{aligned} \beta_{k+1} &= q_k^\top A \operatorname{Re}\left[(H-\sigma A)^{-1}\right] A q_{k+1} \\ &= (Aq_k)^\top \operatorname{Re}\left[(H-\sigma A)^{-1}\right] A q_{k+1}, \end{aligned} \tag{31}$$

one needs to compute $f_{k+1} \equiv Aq_{k+1}$ and $\operatorname{Re}[(H-\sigma A)^{-1}]f_{k+1}$, store the resulting vectors for computing ω_{k+1}, and pass to the next Lanczos step. Note that

$$f_{k+1} = \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} q_{k+1}^{(1)} \\ q_{k+1}^{(2)} \end{bmatrix} = \begin{bmatrix} -Kq_{k+1}^{(1)} \\ Mq_{k+1}^{(2)} \end{bmatrix} \equiv \begin{bmatrix} f_{k+1}^{(1)} \\ f_{k+1}^{(2)} \end{bmatrix}, \tag{32}$$

hence $Kq_{k+1}^{(1)}$ and $Mq_{k+1}^{(2)}$ have to be performed. And

$$\operatorname{Re}\left[(H-\sigma A)^{-1}\right] f_{k+1} = \begin{bmatrix} \operatorname{Re}\left(G_\sigma^{-1}\right) f_{k+1}^{(1)} + \frac{1}{\eta}\operatorname{Im}\left(G_\sigma^{-1}\right) f_{k+1}^{(2)} \\ -\frac{1}{\eta}\operatorname{Im}\left(G_\sigma^{-1}\right) f_{k+1}^{(1)} + \frac{1}{\eta^2}\operatorname{Re}\left(G_\sigma^{-1}\right) f_{k+1}^{(2)} \end{bmatrix}. \tag{33}$$

The factorization, e.g., LU-factorization, of G_σ is done once at the initialization step, and the solutions $G_\sigma^{-1}f_{k+1}^{(1)}$ and $G_\sigma^{-1}f_{k+1}^{(2)}$ have to be performed in complex arithmetics. Once f_{k+1} is available, $\omega_{k+1} = q_{k+1}^\top A q_{k+1} = q_{k+1}^\top f_{k+1}$ can be computed easily.

For complex shift $\operatorname{Re}(\sigma) \neq 0$, one has to apply the unsymmetric Lanczos algorithm. The dominant computations will be $\operatorname{Re}[(H-\sigma A)^{-1}]Ax$ and $(\operatorname{Re}[(H-\sigma A)^{-1}]A)^\top y$ for single shift and $(H-\bar{\sigma}A)^{-1}A(H-\sigma A)^{-1}Ax$ and $((H-\bar{\sigma}A)^{-1}A(H-\sigma A)^{-1}A)^\top y$ for double shift. The implementations of these computations are similar to those in (30)–(33).

Cost of Computations

Suppose the matrices K, G, and M have the same banded structure. Denote Z as a general matrix having the same banded structure as K. We define a matrix operation "*mop*" as the computational cost of the multiplication between Z and a vector, or the backward and forward substitution for solving the linear system associated with Z. With the discussion above, we summarize the dominant computational cost for the pseudo skew symmetric Lanczos algorithm in each iteration as following:

1. origin shift ($\sigma = 0$): four mops;
2. pure imaginary single shift ($\sigma = i\eta$): six mops;
3. complex shift ($\mathrm{Re}(\sigma) \neq 0$) with unsymmetric Lanczos algorithm: 12 mops;
4. pure imaginary double shift ($\sigma = i\eta$) with pseudo symmetric Lanczos algorithm: 14 mops;
5. complex double shift ($\mathrm{Re}(\sigma) \neq 0$) with unsymmetric Lanczos algorithm: 28 mops.

Here we use the convention that one complex multiplication equals to four real multiplications.

3. THE J-LANCZOS ALGORITHM

In this section we apply the J-Lanczos algorithm proposed by Ferng, Lin and Wang [16] to solve the quadratic eigenvalue problem (2) arisen in the undamped gyroscopic system (1). With the definition of H and A in (4), the quadratic eigenvalue problem (2) can be transformed to the following linear generalized eigenvalue problem:

$$HA^{-1}Hz = \frac{1}{\lambda}Hz \tag{34}$$

or, equivalently,

$$\begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} -K^{-1} & 0 \\ 0 & M^{-1} \end{bmatrix} \begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} G & -M \\ M & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{35}$$

Let

$$X = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \begin{bmatrix} I & -\frac{G}{2} \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & M^{-1} \end{bmatrix} \tag{36}$$

and

$$Y = \begin{bmatrix} I & 0 \\ 0 & -M^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\frac{G}{2} & I \end{bmatrix}. \tag{37}$$

By multiplying X and Y from the left and right, respectively, to the matrix pair $(HA^{-1}H, H)$ we get

$$\begin{aligned} X(HA^{-1}H, H)Y &= \left(\begin{bmatrix} I & 0 \\ \frac{G}{2} & -M \end{bmatrix} \begin{bmatrix} -K^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \frac{G}{2} & I \\ I & 0 \end{bmatrix}, \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \right) \\ &= \left(\begin{bmatrix} -\frac{(K^{-1}G)}{2} & -K^{-1} \\ -M - \frac{(GK^{-1}G)}{4} & -\frac{(GK^{-1})}{2} \end{bmatrix}, \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \right) \\ &\equiv (\mathbf{H}, I_{2n}). \end{aligned} \tag{38}$$

Since G is skew symmetric, it is easy to verify that the matrix $\mathbf{H}$ in (38) is Hamiltonian and nonsingular. Recall that a matrix H is Hamiltonian if $(HJ)^\top = HJ = -JH^\top$, where $J = \left[\begin{smallmatrix} 0 & I \\ -I & 0 \end{smallmatrix}\right]$. The inverse of a Hamiltonian matrix, if it exists, is still Hamiltonian.

From [20] we know that there almost always exists a symplectic matrix, $Q_{2n}^{\mathsf{T}} J Q_{2n} = J$ and $Q_{2n} \equiv [q_1, \dots, q_n | q_{n+1}, \dots, q_{2n}]$, such that

$$\mathbf{H} Q_{2n} = Q_{2n} T_{2n}, \tag{39}$$

where

$$T_{2n} = \left[\begin{array}{cccc|cccc} a_1 & & & & c_1 & b_1 & & \\ & \ddots & & & b_1 & \ddots & \ddots & \\ & & \ddots & & & \ddots & \ddots & b_{n-1} \\ & & & a_n & & & b_{n-1} & c_n \\ \hline k_1 & & & & -a_1 & & & \\ & \ddots & & & & \ddots & & \\ & & \ddots & & & & \ddots & \\ & & & k_n & & & & -a_n \end{array}\right] \tag{40}$$

is J-tridiagonal. If we require the Lanczos vector q_j to satisfy

$$\|q_j\|_2 = 1 \quad \text{and} \quad q_j \perp q_{n+j}, \qquad \forall j = 1, \dots, n, \tag{41}$$

then equating the j^{th} and $(n+j)^{\text{th}}$ columns, respectively, on each side of (39), for $j = 1, \dots, n$, yields the uniquely determined two-four-term recurrence for our J-Lanczos algorithm:

$$a_j = q_j^{\mathsf{T}} \mathbf{H} q_j, \qquad k_j = q_j^{\mathsf{T}} J \mathbf{H} q_j, \tag{42}$$

$$q_{n+j} = \frac{1}{k_j} (\mathbf{H} q_j - a_j q_j), \tag{43}$$

$$c_j = -q_{n+j}^{\mathsf{T}} J \mathbf{H} q_{n+j}, \tag{44}$$

$$r_j = \mathbf{H} q_{n+j} - b_{j-1} q_{j-1} - c_j q_j + a_j q_{n+j}, \tag{45}$$

$$b_j = \|r_j\|_2, \qquad q_{j+1} = \frac{r_j}{b_j}. \tag{46}$$

Since in each iteration j, there are two Lanczos vectors q_{n+j} and q_{j+1} generated by the J-Lanczos algorithm, we define that there are two Lanczos steps in each iteration j in comparison with the pseudo skew symmetric Lanczos algorithm.

Next we show that the linear eigenvalue problem $\mathbf{H} z = (1/\lambda) z$ in (38) is equivalent to the original quadratic eigenvalue problem (2) without inverting M. From (38), one has

$$\begin{bmatrix} I & 0 \\ \dfrac{G}{2} & -M \end{bmatrix} \begin{bmatrix} -K^{-1}\dfrac{G}{2} & -K^{-1} \\ I & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{47}$$

It follows that

$$\frac{G}{2} x + y = -\frac{1}{\lambda} K x \qquad \left(\Rightarrow y = -\frac{G}{2} x - \frac{1}{\lambda} K x \right) \tag{48}$$

and

$$-\frac{G}{2} K^{-1} \left(\frac{G}{2} x + y \right) - M x = \frac{1}{\lambda} y. \tag{49}$$

Substituting (48) into (49), we have

$$-\lambda^2 M x + \lambda G x + K x = 0.$$

Dangers

The J-Lanczos algorithm may also break down if k_j in (42) vanishes. But if b_j in (46) becomes zero, then we get the invariant subspace. Detailed analysis on loss of symplecticity and convergence behavior can be found in [16].

How to Compute $T_{2j} - \Lambda$

The J-Lanczos algorithm preserves the 2×2 block Hamiltonian structure and one can apply the SR-algorithm [20] which takes advantage of the compact J-tridiagonal form to compute the desired Ritz pairs from T_{2j}. The SR-algorithm guarantees to find the quartet of Ritz values $(\theta, \bar{\theta}, -\theta, -\bar{\theta})$ at one time. However, the algorithm uses a Gauss-type elimination without pivoting at some steps. Thus, the SR-algorithm is unstable and may break down in some cases. If breakdown occurs, one can recall the conventional QR-algorithm.

Error Bounds

Suppose that at the $2j^{\text{th}}$ iteration of the J-Lanczos algorithm, T_{2j} is the computed J-tridiagonal matrix and

$$U_{2j}^{-1} T_{2j} U_{2j} = \left[\begin{array}{ccc|ccc} \theta_1 & & & & & \\ & \ddots & & & & \\ & & \theta_j & & & \\ \hline & & & -\theta_1 & & \\ & & & & \ddots & \\ & & & & & -\theta_j \end{array}\right] \tag{50}$$

is the J-diagonalization of T_{2j}, in which θ_i are called the J-Ritz values and $U_j = [u_1, \ldots, u_j \mid u_{j+1}, \ldots, u_{2j}]$, with $\|u_i\|_2 = \|u_{j+i}\|_2 = 1$, $i = 1, \ldots, j$. If $Y_{2j} = [y_1, \ldots, y_j \mid y_{n+1}, \ldots, y_{n+j}] = Q_{2j} U_{2j}$, then y_i and y_{n+i} are called the J-Ritz vectors corresponding to the J-Ritz values θ_i and $-\theta_i$, respectively. Moreover, the following identities hold (see [16, Theorem 3.6]):

$$\|\mathbf{H} y_i - \theta_i y_i\|_2 = |\beta_{j,i}| \tag{51}$$

and

$$\|\mathbf{H} y_{n+i} + \theta_i y_{n+i}\|_2 = |\beta_{j,j+i}|, \tag{52}$$

where $\beta_{j,i} = b_j u_{2j,i}$ and $\beta_{j,j+i} = b_j u_{2j,j+i}$, for $i = 1, \ldots, j$.

Suppose that r_j is the residual vector such that

$$\mathbf{H} y_i - \theta_i y_i = u_{2j,i} r_j \tag{53}$$

and

$$\left(y_{n+i}^{\top} J\right) \mathbf{H} - \theta_i \left(y_{n+i}^{\top} J\right) = u_{2j,j+i} (J r_j)^{\top}. \tag{54}$$

Applying the results in [21], it follows that $(\theta_i, y_i, y_{n+i}^{\top} J)$ is an eigentriplet of $\mathbf{H} - E$. The norm of the perturbation E satisfies

$$\|E\|_2 \leq |b_j| \max_{i,j} \left\{ \frac{|u_{2j,i}|}{\|y_i\|_2}, \frac{|u_{2j,j+i}|}{\|y_{n+i}\|_2} \right\}, \tag{55}$$

and the error bound on the J-Ritz value θ_i is

$$|\lambda^{(i)} - \theta_i| \leq \frac{|b_j| \|Q_{2j}\|_2}{|u_{j+i}^H J_j u_i|} \max \left\{ |u_{2j,i}|, |u_{2j,j+i}| \right\} + O\left(\|E\|_2^2\right). \tag{56}$$

When compared with the error bound (21) for the pseudo skew symmetric Lanczos algorithm, the attraction of the last inequality is that it does not depend on $\|\mathbf{H}\|_2$ and that the $\sqrt{j}$ factor does not appear here. This bound can also be calculated with little cost at each step.

Complex Shifts

We now consider complex shifts on the Hamiltonian matrix $\mathbf{H}$. The detailed shift strategy techniques for Hamiltonian matrices are given in [16]. Two types of shifts are considered here.

TYPE I. Real shift $\pm\sqrt{\eta}$ or pure imaginary shift $\pm i\sqrt{\eta}$. Since from (38),

$$\mathbf{H} = \begin{bmatrix} I & 0 \\ \frac{G}{2} & -M \end{bmatrix} \begin{bmatrix} -K^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \frac{G}{2} & I \\ I & 0 \end{bmatrix},$$

one can factor

$$\mathbf{H}^{-1} = \begin{bmatrix} 0 & I \\ I & -\frac{G}{2} \end{bmatrix} \begin{bmatrix} -K & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ M^{-1}\frac{G}{2} & -M^{-1} \end{bmatrix},$$

let

$$\begin{bmatrix} I & 0 \\ \frac{G}{2} & I \end{bmatrix}, \quad \left(\Rightarrow L^{-1} = \begin{bmatrix} I & 0 \\ -\frac{G}{2} & I \end{bmatrix}\right). \tag{57}$$

Then we have

$$L^{-1}\mathbf{H}L = \begin{bmatrix} K^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} -G & -M \\ -M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & M^{-1} \end{bmatrix} \tag{58}$$

and

$$L^{-1}\mathbf{H}^{-1}L = \begin{bmatrix} K^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} 0 & -K \\ -K & G \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & M^{-1} \end{bmatrix}. \tag{59}$$

Hence,

$$(\mathbf{H} \pm \eta\mathbf{H}^{-1})^{-1} = L \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} -G & -M \mp \eta K \\ -M \mp \eta K & \pm\eta G \end{bmatrix}^{-1} \begin{bmatrix} K & 0 \\ 0 & I \end{bmatrix} L^{-1}. \tag{60}$$

The matrix in (60) is the shifted Hamiltonian matrix of $\mathbf{H}$ with the transformation $\mu \longmapsto \mu/(\mu^2 \pm \eta)$. Note that equation (60) also holds when M is singular.

TYPE II. Complex shift $\pm\alpha \pm \beta i$. For the complex shift $2\sigma = \alpha + \beta i$, we consider solving the shifted Hamiltonian eigenvalue problem derived in [16]:

$$\operatorname{Re}\left[\left(\mathbf{H} - c\mathbf{H}^{-1} + i\beta I\right)^{-1}\right] z = \frac{1}{2}\left(\frac{\mu}{(\mu+\sigma)(\mu-\bar{\sigma})} + \frac{\mu}{(\mu+\bar{\sigma})(\mu-\sigma)}\right) z, \tag{61}$$

where $c = (1/4)(\alpha^2 + \beta^2)$. Using (58) and (59) we can derive the factorization

$$\begin{aligned} &\operatorname{Re}\left[\left(\mathbf{H} - c\mathbf{H}^{-1} + i\beta I\right)^{-1}\right] \\ &= L \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \operatorname{Re}\left(\begin{bmatrix} -G + i\beta K & -M + cK \\ -M + cK & -cG + i\beta M \end{bmatrix}^{-1}\right) \begin{bmatrix} K & 0 \\ 0 & I \end{bmatrix} L^{-1}. \end{aligned} \tag{62}$$

Implementation Issues

In the computer implementation of the J-Lanczos algorithm, the Hamiltonian matrix $\mathbf{H}$ as defined in (38) is never formed explicitly. The computation of $\mathbf{H}q_j$ can be arranged as follows:

$$\begin{aligned} \mathbf{H}q_j &= \begin{bmatrix} I & 0 \\ \frac{G}{2} & -M \end{bmatrix} \begin{bmatrix} -K^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \frac{G}{2} & I \\ I & 0 \end{bmatrix} \begin{bmatrix} q_j^{(1)} \\ q_j^{(2)} \end{bmatrix} \\ &= \begin{bmatrix} -K^{-1}\left(\frac{1}{2}Gq_j^{(1)} + q_j^{(2)}\right) \\ -\frac{1}{2}GK^{-1}\left(\frac{1}{2}Gq_j^{(1)} + q_j^{(2)}\right) - Mq_j^{(1)} \end{bmatrix} \end{aligned} \tag{63}$$

$$= \begin{bmatrix} -h_j^{(1)} \\ -\frac{1}{2}Gh_j^{(1)} - Mq_j^{(1)} \end{bmatrix}, \tag{64}$$

where $h_j^{(1)} = K^{-1}f_j^{(1)}$ and $f_j^{(1)} = (1/2)Gq_j^{(1)} + q_j^{(2)}$. Hence, same as the pseudo skew symmetric Lanczos algorithm, the factorization of K has to be computed at the initialization step. Notice that $J\mathbf{H}q_j$ is just a permutation of entries but no floating-point arithmetics. The computation for $\mathbf{H}q_{n+j}$ is similar.

In summary, the dominant computations in one *Lanczos step* of the J-Lanczos algorithm are those matrix-vector operations involved in computing $Gq_j^{(1)}$, $h_j^{(1)} = K^{-1}f_j^{(1)}$, $Gh_j^{(1)}$, and $Mq_j^{(1)}$.

For the shifted Hamiltonian matrix, (60) or (62), we replace the J-Lanczos vectors q_j with Lq_j and J-orthogonalization with $L^\top JL$-orthogonalization, i.e., $\begin{bmatrix} G & I \\ -I & 0 \end{bmatrix}$-orthogonalization. In the implementation, two extra vectors $\tilde{q}_j$ and $\tilde{q}_{n+j}$, where $L\tilde{q}_j = q_j$ and $L\tilde{q}_{n+j} = q_{n+j}$, are used to reduced the computational cost. The computation of $\mathbf{H}q_j$ is replaced by $(\mathbf{H} \pm \eta\mathbf{H}^{-1})^{-1}q_j$ and $\mathrm{Re}\,[(\mathbf{H} - c\mathbf{H}^{-1} + i\beta I)^{-1}]q_j$ for real shift or pure imaginary shift and complex shift, respectively. Since the implementation for both cases are similar, we denote

$$\widetilde{H} = \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} -G & -M \mp \omega K \\ -M \mp \omega K & \pm\omega G \end{bmatrix}^{-1} \begin{bmatrix} K & 0 \\ 0 & I \end{bmatrix} \tag{65}$$

for the real shift or pure imaginary shift case and

$$\widetilde{H} = \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \mathrm{Re}\left(\begin{bmatrix} -G + i\beta K & -M + cK \\ -M + cK & -cG + i\beta M \end{bmatrix}^{-1} \right) \begin{bmatrix} K & 0 \\ 0 & I \end{bmatrix} \tag{66}$$

for the complex shift case. Then $L\widetilde{H}L^{-1}$ is equivalent to the shift matrices $(\mathbf{H} \pm \omega\mathbf{H}^{-1})^{-1}$ and $\mathrm{Re}\,[(\mathbf{H} - c\mathbf{H}^{-1} + i\beta I)^{-1}]$ for either case. The computations of the shifted J-Lanczos algorithm can then be described in detailed as follows.

Assume that both q_j and $\tilde{q}_j$ are available, then

$$a_j = q_j^\top L\widetilde{H}L^{-1}q_j = q_j^\top L\widetilde{H}\tilde{q}_j \tag{67}$$

can be obtained once we have computed $\widetilde{H}\tilde{q}_j$ and $L(\widetilde{H}\tilde{q}_j)$. The computation for $k_j = q_j^\top JL\widetilde{H}\tilde{q}_j$ is obvious. And q_{n+j} and $\tilde{q}_{n+j}$ can be updated with vector operations by

$$q_{n+j} = \frac{1}{k_j}\left(L\widetilde{H}L^{-1}q_j - a_jq_j\right) = \frac{1}{k_j}\left(L\widetilde{H}\tilde{q}_j - a_jq_j\right) \tag{68}$$

and

$$\tilde{q}_{n+j} = \frac{1}{k_j}\left(\widetilde{H}\tilde{q}_j - a_j\tilde{q}_j\right). \tag{69}$$

One can verify that $q_{n+j} = L\tilde{q}_{n+j}$. Both q_{n+j} and $\tilde{q}_{n+j}$ are stored for later computations.

Moreover, since

$$c_j = -q_{n+j}^\top JL\widetilde{H}L^{-1}q_{n+j} = -q_{n+j}^\top{}^\top JL\widetilde{H}\tilde{q}_{n+j}, \tag{70}$$

so c_j can be obtained after the multiplications of $\widetilde{H}\tilde{q}_{n+j}$ and $L(\widetilde{H}\tilde{q}_{n+j})$. Then one computes

$$r_j = L\widetilde{H}\tilde{q}_{n+j} - b_{j-1}q_j - c_jq_j + a_jq_{n+j} \tag{71}$$

and

$$\tilde{r}_j = \widetilde{H}\tilde{q}_{n+j} - b_{j-1}\tilde{q}_j - c_j\tilde{q}_j + a_j\tilde{q}_{n+j} \tag{72}$$

with only vector operations. Note that $r_j = L\tilde{r}_j$. And the J-Lanczos vectors q_{j+1} and $\tilde{q}_{j+1}$ are updated by

$$q_{j+1} = \frac{r_j}{\|r_j\|_2} \quad \text{and} \quad \tilde{q}_{j+1} = \frac{\tilde{r}_j}{\|r_j\|_2}, \tag{73}$$

respectively.

In summary, one can see that the dominant computations in each Lanczos step are the matrix-vector multiplications with $\mathbf{H}$ for the J-Lanczos algorithm and with L and $\widetilde{H}$ for the shifted J-Lanczos algorithm. We comment that none of the enlarged matrices $\mathbf{H}$, L, or $\widetilde{H}$ need to formed explicitly. All computations can be done in terms of $n \times n$ matrices M, G, and K. Therefore, if there is any special structure or sparsity pattern in theses matrices, one should certainly take the advantage and perform the dominant computations with appropriate efficient routines.

Cost of Computations

We use the same definition for "mop" as in Section 2 and suppose that it costs four mops and 16 mops for solving the linear systems involving the real matrix

$$\begin{bmatrix} G & -M \mp \eta K \\ -M \mp \eta K & \pm \eta G \end{bmatrix}$$

and the complex matrix

$$\begin{bmatrix} -G + i\beta K & -M + cK \\ -M + cK & -cG + i\beta M \end{bmatrix},$$

respectively. With the discussion above, we point out that the dominant computations for our J-Lanczos algorithm in each Lanczos step are those formulated in (64), (67), and (70). Therefore, we summarize the computational cost in the following:

1. origin shift ($\sigma = 0$): four mops,
2. real or pure imaginary shift ($\sigma = \sqrt{\omega}$ or $i\sqrt{\omega}$): seven mops,
3. complex shift ($\sigma = \alpha + \beta i$): 19 mops.

Remarks

By comparing the computational cost, one can see that the J-Lanczos algorithm with zero or pure imaginary shift to the matrix $\mathbf{H}$ in (38) needs about the same number of mops at each Lanczos step as the pseudo skew symmetric Lanczos algorithm with zero or pure imaginary shift to the pencil $H - \mu A$ in (4). For the complex shifted eigenvalue problem (62), the J-Lanczos algorithm can converge to a quartet of eigenvalues $\{\pm\mu, \pm\bar{\mu}\}$ at one time. However, the pseudo skew symmetric Lanczos algorithm is no longer applicable to the complex shift case. The alternative unsymmetric Lanczos (two-side Lanczos) algorithm can only converge to a pair of eigenvalues $\{\mu, \bar{\mu}\}$ at once. Hence, in this case it should count 24 mops for computing $\{\pm\mu, \pm\bar{\mu}\}$ when compared with the 19 mops by the J-Lanczos algorithm.

4. NUMERICAL EXPERIMENTS AND RESULTS

In this section, we use several numerical experiments to assess the viability of the proposed Lanczos-type algorithms to extract the eigenpairs for undamped gyroscopic systems. Based on the numerical results, we compare the convergence behavior and numerical efficiency of J-Lanczos algorithm with the pseudo skew symmetric Lanczos algorithm. In the experiments, we focused on finding the eigenvalues with minimal absolute values and the corresponding eigenvectors. The results reported herein were obtained using Pro-Matlab 4.x on a Sun SPARCstation.

TEST SUITE # 1. Random $n \times n$, $n = 1000$, matrices (M, G, K) with known exact eigenvalues were generated such that the minimal eigenvalues in magnitude to the solutions of the quadratic problem (2) appear on the imaginary axis. Figure 1 plots the reciprocal of the spectrum distributions. Table 1 summarizes the numerical results for pseudo skew symmetric Lanczos (S.S. Lanczos) algorithm and J-Lanczos algorithm. We observed that the J-Lanczos algorithm performs better than the pseudo skew symmetric Lanczos algorithm in terms of number of Lanczos steps and computational cost. The results are typical among all testings we have conducted.

Table 1. Numerical results for Test Suite # 1. SSL denotes the pseudo skew symmetric Lanczos algorithm and J-L denotes the J-Lanczos algorithm.

ϵ	No. of Lanczos Steps		mops		No. of Eigenpairs	
	SSL	J-L	SSL	J-L	SSL	J-L
0.15	51	46	206	184	2	2
0.10	52	42	210	168	4	2
0.05	55	50	222	200	2	4
0.02	84	74	338	296	2	2

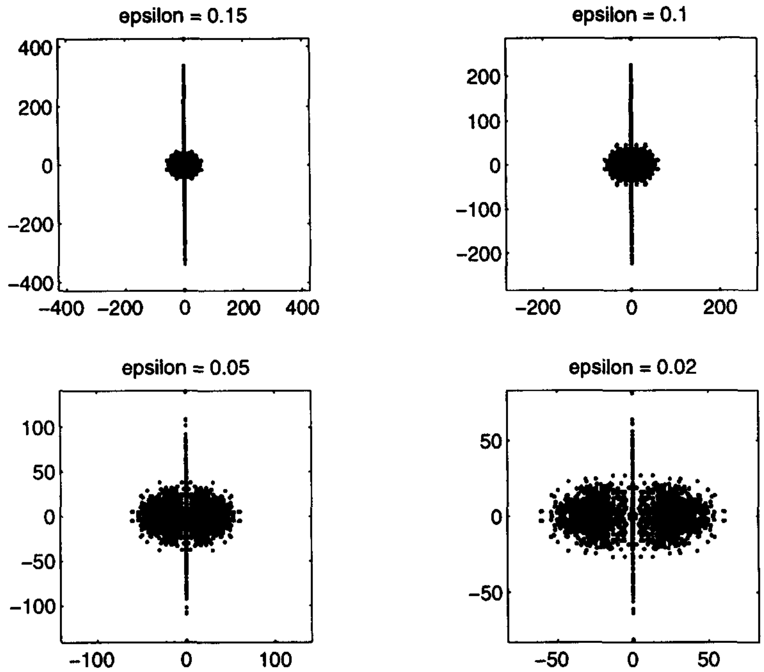

Figure 1. Distribution of reciprocal of all eigenvalues for the associated gyroscopic system in Test Problem # 1 with different ϵ.

In the table, mops denotes the number of n-dimensional matrix-vector multiplications (defined in Section 2) and the last two columns show the number of extreme eigenpairs obtained when the stopping criterion is satisfied and the iteration terminates. We comment that both algorithms require four mops in each Lanczos step; however, pseudo skew symmetric Lanczos algorithm requires four mops for initialization and the J-Lanczos requires two. The J-Lanczos algorithm converges faster to the required accuracy and hence uses less mops.

TEST SUITE # 2. Again, random $n \times n$, $n = 1408$, matrices (M, G, K) with known exact eigenvalues were generated. But in contrast to the previous setting, the desired eigenvalues may not appear on the imaginary axis. Figure 2 shows the reciprocal of the spectrum distributions, and Table 2 summarizes the numerical results. One can see that the pseudo skew symmetric Lanczos algorithm is slightly better than the J-Lanczos algorithm, but the the J-Lanczos algorithm is competitive.

Table 2. Numerical results for Test Suite # 2. SSL denotes the pseudo skew symmetric Lanczos algorithm and J-L denotes the J-Lanczos algorithm.

ϵ	No. of Lanczos Steps		mops		No. of Eigenpairs	
	SSL	J-L	SSL	J-L	SSL	J-L
0.9	63	64	254	256	4	4
0.7	59	60	238	240	4	4
0.5	57	62	230	248	4	4
0.3	59	58	238	232	4	4

Based on our numerical experience, we comment that the J-Lanczos algorithm seems to outperform the pseudo skew symmetric Lanczos algorithm on the first type of problems (Test Suite # 1) and is competitive on the second type of problems.

TEST SUITE # 3. In contrast to the first two settings, we consider band matrices of dimension $n = 100$ in this test. K and M are random band matrices with bandwidth seven. G is chosen to be a random skew symmetric tridiagonal matrix with zeros on the diagonal.

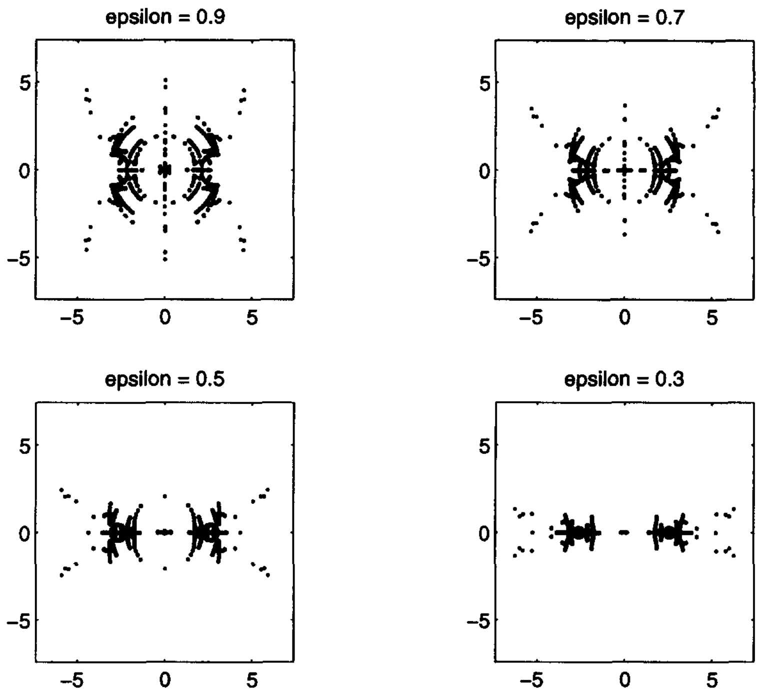

Figure 2. Distribution of reciprocal of all eigenvalues for the associated gyroscopic system in Test Problem # 2 with different ϵ.

Since M, G, and K are all band matrices, the matrix-vector multiplications can be treated as element-wise vector multiplications. We use the same definition for mop as before and define "vec" to to an element-wise product of two vectors. Hence one mop with M or K is equivalent to seven vecs while one mop with G involved is only counted for two vecs. In Table 3 we report the number of operations performed and the number of eigenpairs converged after 12 , 14, 16, and 20 iterations for both algorithms.

Table 3. Numerical results for Test Suite # 3.

No. of Iterations	mops/vecs		No. of Eigenpairs	
	S.S. Lanczos	J-L	S.S. Lanczos	J-L
12	50/290	48/216	4	4
14	58/336	56/252	6	6
16	66/382	64/288	8	8
20	82/474	80/360	10	10

We comment that with the same number of Lanczos steps performed, both algorithm obtain the same number of desired eigenpairs and the same dimension of Krylov subspace, however, the J-Lanczos algorithm requires much less "vecs" operations. Equivalently, more eigenpairs can be obtained by the J-Lanczos if the same amount of operations are spent. Consequently, the J-Lanczos algorithm converges much faster when G has narrow bandwidth. We also comment that breakdown occurred frequently with the pseudo skew symmetric Lanczos algorithm during this test due to the tiny values of $\{\omega_i\}$ as formulated in (12). A new starting vector is chosen to restart the Lanczos iteration when this happens. Break down was not observed for the J-Lanczos algorithm.

5. CONCLUSIONS AND REMARKS

In this paper, we proposed two Lanczos-type approaches, pseudo skew symmetric Lanczos algorithm and J-Lanczos algorithm, for computing a few extreme eigenpairs of the quadratic eigenvalue problem (2) which in turn solves the undamped gyroscopic system (1) in motion about an unstable equilibrium position. Some remarks are summarized in the following.

1. The dimensions of the Krylov subspaces constructed by each algorithm with the same number of Lanczos steps are identical. And the computational costs for both algorithms are about even if the matrices are general dense matrices without any special sparse structures. Hence, the convergence behaviors are similar.
2. If M, G, and K are banded and the bandwidth of G is much less than that of K, then the J-Lanczos algorithm has advantages over the pseudo skew symmetric Lanczos algorithm in terms of computational cost. Consequently, the J-Lanczos algorithm converges much faster.
3. The J-Lanczos algorithm incorporating the SR-algorithm can find a quartet of eigenvalues $\{\pm\mu, \pm\bar{\mu}\}$ at one time. However, the pseudo skew symmetric Lanczos algorithm has to incorporate the QR-algorithm or HR-algorithm that can only find a pair of eigenvalues $\{\mu, \bar{\mu}\}$ at a time.
4. The J-Lanczos algorithm preserves not only the Hamiltonian structure but also the 2×2 block structure of the enlarged matrix form. The pseudo skew symmetric Lanczos algorithm, on the other hand, preserves only the skew symmetry structure, but not the 2×2 block structure of the enlarged matrix.
5. For some complex shifted problems, the pseudo skew symmetric or symmetric Lanczos algorithm can no longer be applied since the shifted matrix is neither symmetric nor skew symmetric. One can only resort to the unsymmetric Lanczos (two-sided Lanczos) algorithm. However, the Hamiltonian structure is preserved in the shifted problems.
6. In the numerical experiments, breakdown is observed more often with the pseudo skew symmetric algorithm than the J-Lanczos algorithm.

Based on the theoretical point of view as well as the numerical observations we may suggest that the J-Lanczos algorithm has to be preferred.

REFERENCES

1. V.I. Arnold, *Mathematical Methods of Classical Mechanics*, Springer-Verlag, New York, (1989).
2. E.T. Whittaker, *Analytical Dynamics*, 4^{th} edition, Macmillan, Cambridge, U.K., (1937).
3. M. Clint and A. Jennings, The evaluation of eigenvalues and eigenvectors of real symmetric matrices by simultaneous iteration, *The Computer J.* **13**, 76–80 (1970).
4. R.J. Duffin, The Rayleigh-Ritz method for dissipative or gyroscopic systems, *Quart. Appl. Math.* **18**, 215–221 (1960).
5. W.M. Greenlee, Lyapunov stability of linear gyroscopic systems, *Utilitas Math.* **8**, 225–231 (1975).
6. K. Huseyin, Standard forms of the eigenvalue problem associated with gyroscopic systems, *J. of Sound and Vibration* **45**, 29–37 (1976).
7. P. Lancaster, *Lambda-Matrices and Vibrating Systems*, Pergamon Press, Oxford, U.K., (1966).
8. R. H. Plaut, Alternative formulations for discrete gyroscopic eigenvalue problems, *American Inst. Aero. and Astro.* **14**, 431–435 (1976).
9. O.A. Bauchau, A solution of the eigenproblem for undamped gyroscopic systems with the Lanczos algorithm, *Internat. J. Numer. Methods Engrg.* **23**, 1705–1713 (1980).
10. L. Barkwell and P. Lancaster, Overdamped and gyroscopic vibrating systems, *J. Appl. Mech.—Trans. of the ASME* **59**, 176–181 (1992).
11. P. Hagedorn, Über die instabilität konservativer systeme mit gyroskopischen Kräften, *Archive for Rational Mechanics and Analysis* **58**, 1–9 (1975).
12. K. Huseyin, P. Hagedorn and W. Tescher, On the stability of linear conservative gyroscopic systems, *J. Appl. Math. Phys.* **34**, 807–815 (1983).
13. L. Barkwell, P. Lancaster and A.S. Markus, Gyroscopically stabilized systems: A class of quadratic eigenvalue problems with real spectrum, *Canadian J. Math.* **44**, 42–53 (1992).
14. B.N. Parlett and H.C. Chen, Use of indefinite pencils for computing damped natural modes, *Lin. Alg. Appl.* **140**, 53–88 (1990).
15. C. Rajakumar, Lanczos algorithm for the quadratic eigenvalue problem in engineering applications, *Comput. Methods Appl. Mech. Engrg.* **105**, 1–22 (1993).
16. W.R. Ferng, W.W. Lin and C.S. Wang, The shift-inverted J-Lanczos algorithm for the numerical solutions of large sparse algebraic Riccati equations, *J. Comput. Appl. Math.* **33**, 23–40 (1997).

17. R.W. Freund, Lanczos-type algorithm for structured non-Hermitian eigenvalue problems, In *Proc. of the Cornelius Lanczos Internat. Centenary Conf.*, pp. 243–245, SIAM, (1994).
18. R.W. Freund and V. Mehrmann, A symplectic look-ahead Lanczos algorithm for the Hamiltonian eigenvalue problem, AT&T Numerical Analysis Manuscript, Bell Laboratories, Murray Hill, NJ, (1994).
19. A. Bunse-Gerstner, An analysis of the HR algorithm for computing eigenvalues of a matrix, *Lin. Alg. Appl.* **35**, 155–173 (1981).
20. A. Bunse-Gerstner and V. Mehrmann, A symplectic QR like algorithm for the solution of the real algebraic Riccati equation, *IEEE Trans. Auto. Control* **31**, 1104–1113 (1986).
21. W. Kahan, B.N. Parlett and E. Jiang, Residual bounds on approximate eigensystems of nonnormal matrices, *SIAM J. Numer. Anal.* **19**, 470–484 (1982).
22. D.J. Inman, *Vibration with Control, Measurement and Stability*, Prentice-Hall, Englewood Cliffs, NJ, (1989).
23. L. Meirovitch, *Computational Methods in Structural Dynamics*, Dijthoft and Noordhoff, Rockville, MD, (1980).
24. B.N. Parlett and Y. Saad, Complex shift and invert strategies for real matrices, *Lin. Alg. Appl.* **88/89**, 575–595 (1987).
25. A. Van der Vorst, A generalized Lanczos scheme, *Math. Comp.* **39**, 559–561 (1982).