NORTH-HOLLAND

An Implicitly Restarted Symplectic Lanczos Method for the Hamiltonian Eigenvalue Problem

Peter Benner*
Technische Universität Chemnitz-Zwickau
Fakultät für Mathematik
09107 Chemnitz, FRG

and

Heike Faßbender†
Universität Bremen
Fachbereich 3—Mathematik und Informatik
28357 Bremen, FRG

Submitted by Ludwig Elsner

ABSTRACT

An implicitly restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem is presented. The Lanczos vectors are constructed to form a symplectic basis. The inherent numerical difficulties of the symplectic Lanczos method are addressed by inexpensive implicit restarts. The method is used to compute eigenvalues, eigenvectors, and invariant subspaces of large and sparse Hamiltonian matrices and low-rank approximations to the solution of continuous-time algebraic Riccati equations with large and sparse coefficient matrices. © 1997 Elsevier Science Inc.

[*] E-mail: benner@mathematik.tu-chemnitz.de. This author was supported by Deutsche Forschungsgemeinschaft, research grant Me 790/7-1 Singuläre Steuerungsprobleme.

[†] E-mail: heike@mathematik.uni-bremen.de.

LINEAR ALGEBRA AND ITS APPLICATIONS 263:75–111 (1997)

655 Avenue of the Americas, New York, NY 10010

0024-3795/97/$17.00
PII S0024-3795(96)00524-1

1. INTRODUCTION

Many applications require the numerical solution of the real Hamiltonian eigenvalue problem

$$Hx = \lambda x, \tag{1}$$

where

$$H = \begin{bmatrix} A & G \\ Q & -A^T \end{bmatrix} \in \mathbb{R}^{2n \times 2n}$$

is large and sparse, and

$$A, G = G^T, Q = Q^T \in \mathbb{R}^{n \times n}.$$

The eigenvalues of Hamiltonian matrices are used in algorithms to compute the real and complex stability radius of matrices (see, e.g., [12, 16]) and the $\mathcal{H}_\infty$-norm of transfer matrices (see, e.g., [17]). In computational chemistry, the problem of finding some eigenvalues of largest modulus and the corresponding eigenvectors of a Hamiltonian matrix arises in linear response theory; see, e.g., [19, 39].

The essential role of the continuous-time algebraic Riccati equation (CARE) in control theory,

$$Q + A^TX + XA - XGX = 0, \tag{2}$$

and its connection to the Hamiltonian eigenproblem (1) is well known; see, e.g., [33, 35, 37] and the references given therein. The solution of the CARE (2) with small and dense coefficient matrices (say $n \leqslant 100$) has been the topic of numerous publications during the last 30 years. Even for these problems a numerically sound method, i.e., a strongly backward stable method in the sense of [5], is yet not known. Only a few attempts have been made to solve (1) for large and sparse matrices, e.g., [29, 31, 42]. In order to reduce both computational cost and workspace, it is crucial to use the Hamiltonian structure.

It is well known that for each *Hamiltonian* matrix H, we have

$$(JH)^T = JH$$

where

$$J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \tag{3}$$

and I_n is the $n \times n$ identity matrix. The eigenvalues of a Hamiltonian matrix H occur in pairs $\lambda, -\lambda$ and, if they are complex with nonzero real part, even in quadruples $\lambda, \bar{\lambda}, -\lambda, -\bar{\lambda}$. *Symplectic* matrices S are defined by the property $S^TJS = J$ for $S \in \mathbb{R}^{2n \times 2n}$ (this property is also called *J-orthogonality*). If H is Hamiltonian and S is symplectic, then $S^{-1}HS$ is Hamiltonian. Thus, a structure-preserving and numerically stable algorithm for the eigenproblem (1) should consist only of orthogonal symplectic similarity transformations. An algorithm with this property was proposed in [11] for the case that rank $G = 1$ or rank $Q = 1$. To the best of our knowledge, the only existing algorithm for the general case satisfying this demand was proposed in [1]. But for growing dimension n, this method suffers from convergence problems. The Lanczos method proposed here for the large-scale problem exploits the structure of the problem by weakening orthogonality to J-orthogonality. In exact arithmetic, the method would compute a symplectic (nonorthogonal) matrix S and a *Hamiltonian J-Hessenberg matrix* $\tilde{H}$ such that

$$\tilde{H} = S^{-1}HS = \left[\begin{array}{ccccc|ccccc} \delta_1 & & & & & \beta_1 & \zeta_2 & & & \\ & \delta_2 & & & & \zeta_2 & \beta_2 & \zeta_3 & & \\ & & \delta_3 & & & & \zeta_3 & \ddots & \ddots & \\ & & & \ddots & & & & \ddots & \ddots & \zeta_n \\ & & & & \delta_n & & & & \zeta_n & \beta_n \\ \hline \nu_1 & & & & & -\delta_1 & & & & \\ & \nu_2 & & & & & -\delta_2 & & & \\ & & \nu_3 & & & & & -\delta_3 & & \\ & & & \ddots & & & & & \ddots & \\ & & & & \nu_n & & & & & -\delta_n \end{array}\right]. \tag{4}$$

The reduction of Hamiltonian matrices to Hamiltonian J-Hessenberg form serves as initial step in the Hamiltonian SR algorithm proposed by Bunse-Gerstner and Mehrmann [9]. This algorithm is a QR-like method for the Hamiltonian eigenproblem based on the SR decomposition. There, $\tilde{H}$ is computed by an elimination process. During this elimination process the use of very badly conditioned matrices cannot always be circumvented. It is shown that the reduction of a Hamiltonian matrix to such a Hamiltonian J-Hessenberg form does not always exist. The existence of this reduction and also the existence of a numerically stable method to compute this reduction is strongly dependent on the first column of the transformation matrix that carries out the reduction.

A few attempts have been made to create structure-preserving methods using a symplectic Lanczos method. The symplectic Lanczos method proposed by Mei [38] works with the squared Hamiltonian matrix and suffers from stability problems as well as from breakdown. The structure-preserving symplectic Lanczos method considered here creates a Hamiltonian J-Hessenberg matrix as in (4) if no breakdowns or near-breakdowns occur. Eigenvalue methods for such matrices and the application to the solution of algebraic Riccati equations (2) are examined in [8, 9, 36, 37, 43]. In [23], Freund and Mehrmann present a symplectic look-ahead Lanczos algorithm which overcomes breakdown by giving up the strict Hamiltonian J-Hessenberg form (4). In this paper we combine the ideas of implicitly restarted Lanczos methods [13, 26, 44] with ideas to reflect the Hamiltonian structure and present a restarted symplectic Lanczos algorithm for the Hamiltonian eigenvalue problem. Implicitly restarted Lanczos methods typically have a higher numerical accuracy than explicit restarts, and moreover they are more economical to implement [26].

In Section 2 the implicitly restarted Lanczos method for nonsymmetric matrices is reviewed. Section 3 describes the symplectic Lanczos method for Hamiltonian matrices. In order to preserve the Hamiltonian J-Hessenberg form obtained from the symplectic Lanczos method, an SR decomposition has to be employed in an implicitly restarted symplectic Lanczos method. Thus in Section 4 we briefly present those details of the SR decomposition necessary for the restart. The implicitly restarted symplectic Lanczos method itself is derived in Section 5. Numerical properties of the proposed algorithm are discussed in Section 6. Section 7 gives a survey over applications of the method, and in Section 8 we present some numerical examples.

2. THE IMPLICITLY RESTARTED LANCZOS METHOD

The Lanczos process is an effective method for computing a few eigenvalues and associated eigenvectors of a large and sparse matrix A. Given $p_1, q_1 \in \mathbb{R}^n$ and a nonsymmetric matrix $A \in \mathbb{R}^{n\times n}$, the standard nonsymmetric Lanczos algorithm [34] produces matrices $P_k = [p_1, \dots, p_k] \in \mathbb{R}^{n\times k}$

and $Q_k = [q_1, \ldots, q_k] \in \mathbb{R}^{n \times k}$ which satisfy the recursive identities

$$AP_k = P_k T_k + \beta_{k+1} p_{k+1} e_k^T, \tag{5}$$

$$A^T Q_k = Q_k T_k^T + \gamma_{k+1} q_{k+1} e_k^T. \tag{6}$$

The vector e_k is the kth unit vector, and

$$T_k = \begin{bmatrix} \alpha_1 & \gamma_2 & & \\ \beta_2 & \ddots & \ddots & \\ & \ddots & \ddots & \gamma_k \\ & & \beta_k & \alpha_k \end{bmatrix}$$

is a truncated reduction of A. Generally, the elements β_j and γ_j are chosen so that $|\beta_j| = |\gamma_j|$ and $Q_k^T P_k = I_k$ (biorthogonality). One pleasing result of this biorthogonality condition is that multiplying (5) on the left by Q_k^T yields the relationship $Q_k^T A P_k = T_k$.

The Lanczos method only requires the computation of matrix-vector products, which often can be accomplished without explicit storage of the matrix. Unfortunately the Lanczos method can suffer from large storage requirements (for the columns of Q_k, P_k) and numerical difficulties.

In theory, the three-term recurrences in (5) and (6) are sufficient to guarantee $Q_k^T P_k = I_k$. Yet in practice, it is known [40] that biorthogonality will in fact be lost when at least one of the eigenvalues of T_k converges to an eigenvalue of A. (See also [25] and the references therein.) Thus reorthogonalization is necessary, increasing the computational cost significantly.

At each step of the nonsymmetric Lanczos tridiagonalization, an orthogonalization is performed, which requires a division by the inner product of (multiples of) the vectors produced at the previous step. Thus the algorithm suffers from breakdown and instability if any of these inner products is zero or close to zero. If $\beta_{j+1} = 0$, breakdown is welcome, as in this case the columns of P_j define an invariant subspace for A. This event is called a *benign breakdown*. Unfortunately, the iteration can break down without any invariant-subspace information. In this case one speaks of a *serious breakdown*. It is known [30] that vectors q_1 and p_1 exist such that the Lanczos process with them as starting vectors does not encounter (serious) breakdown. However, determining these vectors requires knowledge of the minimal polynomial of A. Further, there are no theoretical results showing that

p_1 and q_1 can be chosen such that small inner products can be avoided. Thus, no algorithm for successfully choosing p_1 and q_1 at the start of the computation yet exists.

It is possible to modify the Lanczos process so that it skips over exact breakdowns. A complete treatment of the modified Lanczos algorithm and its intimate connection with orthogonal polynomials and Padé approximation was presented by Gutknecht [27, 28]. Taylor [45], and Parlett, Taylor, and Liu [41] were the first to propose a look-ahead Lanczos algorithm that skips over breakdowns and near-breakdowns. The price paid is that the resulting matrix is no longer tridiagonal, but has a small bulge in the tridiagonal form to mark each occurrence of a (near) breakdown. Freund, Gutknecht, and Nachtigal presented in [24] a look-ahead Lanczos code that can handle look-ahead steps of any length.

A different approach to deal with the numerical difficulties of the Lanczos process is to modify the starting vectors by an implicitly restarted Lanczos process. The problems are addressed by fixing the number of steps in the Lanczos process at a prescribed value k which is dependent on the required number of approximate eigenvalues. Orthogonality of the k Lanczos vectors is secured by reorthogonalizing these vectors when necessary. The purpose of the implicit restart is to determine initial vectors such that the associated residual vectors are tiny. Given that P_k and Q_k from (5) and (6) are known, an implicit Lanczos restart computes the Lanczos factorization

$$A\tilde{P}_k = \tilde{P}_k\tilde{T}_k + \tilde{r}_k e_k^T, \tag{7}$$

$$A^T\tilde{Q}_k = \tilde{Q}_k\tilde{T}_k^T + \tilde{s}_k e_k^T, \tag{8}$$

which corresponds to the starting vectors

$$\tilde{p}_1 = \rho_p(A - \mu I)p_1, \qquad \tilde{q}_1 = \rho_q(A^T - \mu I)q_1, \tag{9}$$

without explicitly restarting the Lanczos process with the vectors in (9). For a detailed derivation see [26] and the fundamental work in [13, 44].

In Section 5 we show how to use these ideas to deal with the numerical difficulties of the symplectic Lanczos algorithm discussed in the next section. Another application of the restart idea will be given in Section 7, where the symplectic Lanczos method is used to find low-rank approximations to the solution of algebraic Riccati equations.

3. A SYMPLECTIC LANCZOS METHOD FOR HAMILTONIAN MATRICES

In this section, we describe a symplectic Lanczos method to compute the reduced Hamiltonian J-Hessenberg form (4) for a Hamiltonian matrix H similar to the one proposed in [22, 23]. The usual nonsymmetric Lanczos algorithm generates two sequences of vectors. Due to the Hamiltonian structure of H, it is easily seen that one of the two sequences can be eliminated here and thus the work and storage can essentially be halved. (This property is valid for a broader class of matrices; see [21].)

In order to simplify the notation in the following we use, as in [9, 23], a permuted version of H and $\tilde{H}$. Let

$$H_P = PHP^T, \qquad \tilde{H}_P = P\tilde{H}P^T, \qquad S_P = PSP^T, \qquad J_P = PJP^T$$

with the permutation matrix $P = P^n$, where

$$P^n = [e_1, e_3, \dots, e_{2n-1}, e_2, e_4, \dots, e_{2n}] \in \mathbb{R}^{2n\times 2n}.$$

If the dimension of P^n is clear from the context, we leave off the superscript.

From $S^TJS = J$ we obtain

$$S_P^T J_P S_P = J_P = \mathrm{diag}\left(\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \dots, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\right),$$

while $S^{-1}HS = \tilde{H}$ yields

$$H_P S_P = S_P \tilde{H}_P$$

$$= S_P \left[\begin{array}{cc|cc|cc|cc|cc}
\delta_1 & \beta_1 & 0 & \zeta_2 & & & & & & \\
\nu_1 & -\delta_1 & 0 & 0 & & & & & & \\
\hline
0 & \zeta_2 & \delta_2 & \beta_2 & 0 & \zeta_3 & & & & \\
0 & 0 & \nu_2 & -\delta_2 & 0 & 0 & & & & \\
\hline
 & & 0 & \zeta_3 & \ddots & & \ddots & & & \\
 & & 0 & 0 & & \ddots & & \ddots & & \\
\hline
 & & & & \ddots & & \ddots & & 0 & \zeta_n \\
 & & & & & \ddots & & \ddots & 0 & 0 \\
\hline
 & & & & & & 0 & \zeta_n & \delta_n & \beta_n \\
 & & & & & & 0 & 0 & \nu_n & -\delta_n
\end{array}\right]. \tag{10}$$

The structure-preserving Lanczos method generates a sequence of matrices

$$S_P^{2k} = [v_1, w_1, v_2, w_2, \dots, v_k, w_k] \in \mathbb{R}^{2n \times 2k}$$

satisfying

$$H_P S_P^{2k} = S_P^{2k} \tilde{H}_P^{2k} + \zeta_{k+1} v_{k+1} e_{2k}^T, \tag{11}$$

where $\tilde{H}_P^{2k} = P^k \tilde{H}^{2k} (P^k)^T$ is a permuted $2k \times 2k$ Hamiltonian J-Hessenberg matrix $\tilde{H}^{2k}$ of the form (10). The space spanned by the columns of $S^{2k} = (P^n)^T S_P^{2k} P^k$ is symplectic, since $(S_P^{2k})^T J_P^n S_P^{2k} = J_P^k$, where $P^j J^j (P^j)^T = J_P^j$ and J^j is a $2j \times 2j$ matrix of the form (3).

As this reduction is strongly dependent on the first column of the transformation matrix that carries out the reduction, we must expect breakdown or near-breakdown in the Lanczos process, as they also occur in the reduction to Hamiltonian J-Hessenberg form (e.g., [9]). Assume that no such breakdowns occur, and let $S_P = [v_1, w_1, v_2, w_2, \dots, v_n, w_n]$. For a given v_1, a Lanczos method constructs the matrix S_P columnwise from the equations

$$H_P S_P e_j = S_P \tilde{H}_P e_j, \qquad j = 1, 2, \dots .$$

From this we obtain the algorithm given in Table 1.

Note that only one matrix-vector product is required for each computed Lanczos vector w_m or v_m. Thus an efficient implementation of this algorithm requires $6n + (4nz + 32n)k$ flops[1] where nz is the number of nonzero elements in H_P and $2k$ is the number of Lanczos vectors computed (that is, the loop is executed k times). The algorithm as given in Table 1 computes an odd number of Lanczos vectors; for a practical implementation one has to omit the computation of the last vector v_{k+1} (or one has to compute an additional vector w_{k+1}).

There is still some freedom in the choice of the parameters that occur in this algorithm. Possibilities to remove these ambiguities have been discussed in [36]. Essentially, the parameters δ_m can be chosen freely. Here we set $\delta_m = 1$. Likewise a different choice of the parameters ζ_m, ν_m is possible.

[1] Following [25], we define each floating-point arithmetic operation together with the associated integer indexing as a flop.

TABLE 1
ALGORITHM: SYMPLECTIC LANCZOS METHOD

```
Choose an initial vector $\tilde{v}_1 \in \mathbb{R}^{2n}$, $\tilde{v}_1 \neq 0$.
Set $v_0 = 0 \in \mathbb{R}^{2n}$.
Set $\zeta_1 = \|\tilde{v}_1\|_2$ and $v_1 = \frac{1}{\zeta_1}\tilde{v}_1$.
for $m = 1, 2, \ldots$ do
  (update of $w_m$)
    set
    $\tilde{w}_m = H_P v_m - \delta_m v_m$
    $\nu_m = v_m^T J_P H_P v_m$
    $w_m = \frac{1}{\nu_m}\tilde{w}_m$
  (computation of $\beta_m$)
    $\beta_m = -w_m^T J_P H_P w_m$
  (update of $v_{m+1}$)
    $\tilde{v}_{m+1} = H_P w_m - \zeta_m v_{m-1} - \beta_m v_m + \delta_m w_m$
    $\zeta_{m+1} = \|\tilde{v}_{m+1}\|_2$
    $v_{m+1} = \frac{1}{\zeta_{m+1}}\tilde{v}_{m+1}$
```

In the symplectic Lanczos method as given above we have to divide by a parameter that may be zero or close to zero. If such a case occurs for the normalization parameter ζ_{m+1}, the corresponding vector $\tilde{v}_{m+1}$ is zero or close to the zero vector. In this case, a symplectic invariant subspace of H (or a good approximation to such a subspace) is detected. By redefining $\tilde{v}_{m+1}$ to be any vector satisfying

$$v_j^T J_P \tilde{v}_{m+1} = 0,$$

$$w_j^T J_P \tilde{v}_{m+1} = 0$$

for $j = 1, \ldots, m$, the algorithm can be continued. The resulting Hamiltonian J-Hessenberg matrix is no longer unreduced; the eigenproblem decouples into two smaller subproblems. In case $\tilde{w}_m$ is zero (or close to zero), an invariant subspace of H_P with dimension $2m - 1$ is found (or a good approximation to such a subspace). It is easy to see that in this case the parameter ν_m will be zero (or close to zero). Two eigenvalues and one right and one left eigenvector can be read off directly from the reduced matrix $\tilde{H}^{2m-2}$ as in (4).

Thus if either v_{m+1} or w_{m+1} vanishes, the breakdown is benign. If $v_{m+1} \neq 0$ and $w_{m+1} \neq 0$ but $\nu_{m+1} = 0$, then the breakdown is serious. No reduction of the Hamiltonian matrix to a Hamiltonian J-Hessenberg matrix with v_1 as first column of the transformation matrix exists.

Without some form of re-J-orthogonalization the symplectic Lanczos method is numerically unstable (see Section 6.1 and the discussion there). Thus, the symplectic Lanczos method suffers from the same numerical difficulties as any other Lanczos-like algorithm.

These issues are addressed here by employing an implicit restart. Before discussing this approach in Section 5, we need to introduce the SR decomposition, which will turn out to be fundamental in the restart process.

4. THE SR DECOMPOSITION

In [13, 44], the decomposition $T_k - \mu I = QR$ and the corresponding QR step, $\tilde{T}_k = Q^T T_k Q$, play a key role in implicit restarts for the symmetric Lanczos method. These transformations preserve the symmetry and tridiagonality of T_k as well as the orthogonality of the updated Lanczos basis vectors. In the implicitly restarted Lanczos method for nonsymmetric matrices [26], the HR decomposition and a corresponding HR step [7] are used, as this transformation preserves sign symmetry along with the tridiagonality of the T_k and the biorthogonality of the basis vectors.

Although symmetry is lacking in the symplectic Lanczos process defined above, the resulting matrix $\tilde{H}_p^{2k}$ is a permuted Hamiltonian J-Hessenberg matrix as in (10). In order to preserve this structure and the J-orthogonality of the basis vectors we will employ an SR decomposition of $\tilde{H}_p^{2k} - \mu I$, $\mu \in \mathbb{R}$. Besides this single shift, we study double shifts $(\tilde{H}_p^{2k} - \mu I)(\tilde{H}_p^{2k} + \mu I)$ where $\mu \in \mathbb{R}$ or $\mu \in \imath\mathbb{R}$ ($\imath = \sqrt{-1}$). Double shifts with purely imaginary values turn out to be useful in connection with the computation of low-rank approximations to the solution of the continuous-time algebraic Riccati equation, as will be shown in Section 7.2.

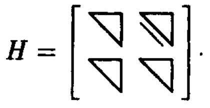
The SR decomposition has been studied in, e.g., [9, 15]. A slightly modified version of the notation of [9] will be employed here.

Definition 4.1.

(a) A matrix

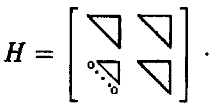
$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

where $H_{ij} \in \mathbb{R}^{n\times n}$ is called a *J-Hessenberg matrix* if H_{11}, H_{21}, H_{22} are upper triangular matrices and H_{12} is an upper Hessenberg matrix, i.e.,

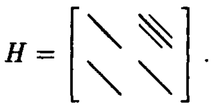


H is called *unreduced* if H_{21} is nonsingular and the upper Hessenberg matrix H_{12} is unreduced, i.e., has no zero entry in its first subdiagonal.

(b) H is called a *J-triangular matrix* if $H_{11}, H_{12}, H_{21}, H_{22}$ are upper triangular matrices and H_{21} has a zero main diagonal, i.e.,



(c) H is called a *J-tridiagonal matrix* if H_{11}, H_{21}, H_{22} are diagonal matrices and H_{12} is a tridiagonal matrix, i.e.,



REMARK 4.1. A Hamiltonian J-Hessenberg matrix $\tilde{H} \in \mathbb{R}^{2n\times 2n}$ is J-tridiagonal and Hamiltonian.

THEOREM 4.1. *Let X be a $2k \times 2k$ nonsingular matrix. Then:*

(a) There exists a symplectic $2k \times 2k$ matrix S and a J-triangular matrix R such that $X = SR$ if and only if all leading principal minors of even dimension of PX^TJXP^T are nonzero.

(*b*) *Let* $X = SR$ *and* $X = \tilde{S}\tilde{R}$ *be SR factorizations of* X. *Then there exists a matrix*

$$D = \begin{bmatrix} C & F \\ 0 & C^{-1} \end{bmatrix},$$

where $C = \operatorname{diag}(c_1, \dots, c_n)$, $F = \operatorname{diag}(f_1, \dots, f_n)$, *such that* $\tilde{S} = SD^{-1}$ *and* $\tilde{R} = DR$.

(*c*) *Let* $X = \tilde{H}$ *be an unreduced Hamiltonian J-Hessenberg matrix. If* $p(\tilde{H})$ *is a real polynomial and* $p(\tilde{H}) = SR$ *exists with* S *and* R *satisfying* (*a*), *then* $\hat{H} = S^{-1}\tilde{H}S$ *is a Hamiltonian J-Hessenberg matrix.*

(*d*) *If* $p(\tilde{H}) = \tilde{H} - \mu I$ *in* (*c*) *and* $\mu \in \mathbb{R}$ *is an eigenvalue of* $\tilde{H}$, *then* $\hat{h}_{2k,2k} = \mu$, $\hat{h}_{k,k} = -\mu$, *and* $\hat{h}_{2k,k} = 0$.

(*e*) *If* $p(\tilde{H}) = (\tilde{H} - \mu I)(\tilde{H} + \mu I)$ *in* (*c*) *and* $\mu \in \mathbb{R}$ *or* $\mu \in \imath\mathbb{R}$ *is an eigenvalue of* $\tilde{H}$, *then* $\hat{h}_{k,2k-1} = \hat{h}_{k-1,2k} = 0$ *and the* 2×2 *submatrix*

$$\begin{bmatrix} \hat{h}_{kk} & \hat{h}_{k,2k} \\ \hat{h}_{2k,k} & \hat{h}_{2k,2k} \end{bmatrix}$$

has the eigenvalues μ *and* $-\mu$.

Proof. For the original statement and proof of (a) see Theorem 11 in [18].

For the original statement and proof of (b) see Proposition 3.3 in [9].

For the original statement and proof of (c) see Remark 4.1 in [9].

The proof of (d) and (e) follows the lines of [26, Theorem 2(iii)]. A detailed derivation is given in [3]. ■

Assuming their existence, the SR decomposition and SR step (that is, $\hat{H} = S^{-1}\tilde{H}S$) possess many of the desirable properties of the QR method. For the moment, it will be assumed that the SR decomposition always exists. A discussion of the existence and stability of the SR step in the context of the Lanczos algorithm is provided in Section 6.

An algorithm for explicitly computing S and R is presented in [9]. As with explicit QR steps, the expense of explicit SR steps comes from the fact that both S^{-1} and S have to be computed explicitly. A preferred alternative is the implicit SR step, an analogue to the Francis QR step [20, 25, 32]. The first implicit rotation is selected so that the first columns of the implicit and the explicit S are equivalent. The remaining implicit rotations perform a bulge-chasing sweep down the subdiagonal to restore the J-Hessenberg form. As the implicit SR step is analogous to the implicit QR step, this technique will not be discussed here. For a detailed derivation of an implicit single- and double-shift SR step see [3].

The algorithm for the implicit double shift uses $4k - 3$ transformations, the algorithm for the implicit single shift $2k - 1$. In the double-shift case, $3k - 2$ of these transformation are orthogonal (k in the single-shift case). These are known to be numerically stable. Thus, in both algorithms $k - 1$ permuted symplectic Gaussian transformation matrices have to be used. Problems can arise here because of breakdown or near-breakdown. Since the condition number of these matrices is not bounded, such a transformation can cause a dramatic growth of rounding errors. We come back to this problem in Section 6.

5. A RESTARTED SYMPLECTIC LANCZOS METHOD

Given that a $2n \times 2k$ matrix S_P^{2k} is known such that

$$H_P S_P^{2k} = S_P^{2k}\tilde{H}_P^{2k} + \zeta_{k+1}v_{k+1}e_{2k}^T \tag{12}$$

as in (11), an implicit Lanczos restart computes the Lanczos factorization

$$H_P \breve{S}_P^{2k} = \breve{S}_P^{2k}\breve{H}_P^{2k} + \breve{\zeta}_{k+1}\breve{v}_{k+1}e_{2k}^T, \tag{13}$$

which corresponds to the starting vector

$$\breve{v}_1 = \rho(H_P - \mu I)v_1$$

without having to explicitly restart the Lanczos process with the vector $\breve{v}_1$. Such an implicit restarting mechanism will now be derived, analogous to the technique introduced in [26, 44].

For any permuted symplectic $2k \times 2k$ matrix S_P, (12) can be reexpressed as

$$H_P\left(S_P^{2k}S_P\right) = \left(S_P^{2k}S_P\right)\left(S_P^{-1}\tilde{H}_P^{2k}S_P\right) + \zeta_{k+1}v_{k+1}e_{2k}^T S_P.$$

Defining $\breve{S}_P^{2k} = S_P^{2k} S_P$, $\breve{H}_P^{2k} = S_P^{-1} \tilde{H}_P^{2k} S_P$, this yields

$$H_P \breve{S}_P^{2k} = \breve{S}_P^{2k} \breve{H}_P^{2k} + \zeta_{k+1} v_{k+1} e_{2k}^T S_P. \tag{14}$$

Let s_{ij} be the (i, j)th entry of S_P. If we choose S_P from the permuted SR decomposition $\tilde{H}_P^{2k} - \mu I = S_P R_P$, then it is easy to see that S_P is an upper Hessenberg matrix. Thus the residual term in (14) is

$$\zeta_{k+1} v_{k+1} \left(s_{2k,2k-1} e_{2k-1}^T + s_{2k,2k} e_{2k}^T \right).$$

In order to obtain a residual term of the desired form "vector times e_{2k}^T" we have to truncate off a portion of (14). Rewriting (14) as

$$H_P \breve{S}_P^{2k} = \left[\breve{S}_P^{2k-2}, \breve{v}_k, \breve{w}_k, v_{k+1} \right] \left[\begin{array}{c|cc} \breve{H}_P^{2k-2} & 0 & \breve{\zeta}_k e_{2k-3} \\ \hline \breve{\zeta}_k e_{2k-2}^T & \breve{\delta}_k & \breve{\beta}_k \\ 0 & \breve{\nu}_k & -\breve{\delta}_k \\ \hline 0 & \zeta_{k+1} s_{2k,2k-1} & \zeta_{k+1} s_{2k,2k} \end{array} \right],$$

we obtain as a new Lanczos identity

$$H_P \breve{S}_P^{2k-2} = \breve{S}_P^{2k-2} \breve{H}_P^{2k-2} + \breve{\zeta}_k \breve{v}_k e_{2k-2}^T. \tag{15}$$

Here, $\breve{\zeta}_k$, $\breve{\delta}_k$, $\breve{\beta}_k$, $\breve{\nu}_k$ denote the parameters of $\breve{H}_P^{2k}$, and ζ_{k+1} a parameter of $\tilde{H}_P^{2k}$. In addition, $\breve{v}_k, \breve{w}_k$ are the last two column vectors from $\breve{S}_P^{2k}$, while v_{k+1} is the next to last column vector of S_P^{2k}.

As the space spanned by the columns of $S^{2k} = (P^n)^T S_P^{2k} P^k$ is symplectic, and S_P is a permuted symplectic matrix, the space spanned by the columns of $\breve{S}^{2k-2} = (P^n)^T \breve{S}_P^{2k-2} P^{k-1}$ is symplectic. Thus, (15) is a valid Lanczos factorization for the new starting vector $\breve{v}_1 = \rho(H_P - \mu I) v_1$. Only one additional step of the symplectic Lanczos algorithm is required to obtain (13) from (12).

Note that in the symplectic Lanczos process the vectors v_j of S_P^{2k} satisfy the condition $\|v_j\|_2 = 1$ and the parameters δ_j are chosen to be one. Due to the multiplication by S_P, in general, this is no longer true for the parameters $\breve{\delta}_j$ from $\breve{H}_P^{2k}$ and for the odd-numbered column vectors of $\breve{S}_P^{2k}$ and thus for the new Lanczos factorization (15).

The extension of this technique to the double- or multiple-shift case is straightforward.

The implicitly restarted symplectic Lanczos method will be used to compute a few eigenvalues and associated eigenvectors. For this, we fix the number of steps in the Lanczos process at a prescribed value k of modest size. An attempt will be made to iteratively update the starting vector v_1 by implicit restarts in order to force the residual vector $\zeta_{k+1}v_{k+1}e_{2k}^T$ to zero. That is, we propose the k-step restarted symplectic Lanczos method as given in Table 2 (analogous to [44]).

A detailed discussion of this approach along the lines of the discussion in [44] can be given. The approach has several advantages over the standard symplectic Lanczos method. J-orthogonality can be maintained at reasonable computational costs. There is fixed storage requirement, and we can use deflation techniques similar to those associated with the SR iteration.

Other applications of the restart are discussed in the following sections.

6. NUMERICAL PROPERTIES OF THE IMPLICITLY RESTARTED SYMPLECTIC LANCZOS METHOD

6.1. *Stability Issues*

It is well known that for general Lanczos-like methods the stability of the overall process is improved when the norm of the Lanczos vectors is chosen to be equal to 1 [41, 45]. Thus, Freund and Mehrmann propose in [23] to modify the prerequisite $S_P^T J_P S_P = J_P$ of the symplectic Lanczos method to

$$S_P^T J_P S_P = \operatorname{diag}\left(\begin{bmatrix} 0 & \sigma_1 \\ -\sigma_1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & \sigma_2 \\ -\sigma_2 & 0 \end{bmatrix}, \dots, \begin{bmatrix} 0 & \sigma_n \\ -\sigma_n & 0 \end{bmatrix}\right) =: \Sigma$$

and

$$\|v_j\|_2 = \|w_j\|_2 = 1, \qquad j = 1, \dots, n.$$

TABLE 2

ALGORITHM: k-STEP RESTARTED SYMPLECTIC LANCZOS METHOD

```
Perform $k$ steps of the symplectic Lanczos algorithm to compute $S_P^{2k}$ and $\tilde{H}_P^{2k}$.
while $\|\zeta_{k+1}v_{k+1}\| > tol$
  perform $q$ additional steps of the symplectic Lanczos method to
    compute $S_P^{2(k+q)}$ and $\tilde{H}_P^{2(k+q)}$
  select $q$ shifts
  obtain $S_P^{2k}$ and $\tilde{H}_P^{2k}$ from $S_P^{2(k+q)}$ and $\tilde{H}_P^{2(k+q)}$ by implicit restarts
end while
```

For the resulting algorithm and a discussion of it we refer to [23]. It is easy to see that $\tilde{H}_P = S_P^{-1} H_P S_P$ is no longer a permuted Hamiltonian J-Hessenberg matrix, as S is only "almost" symplectic, but

$$\Sigma \tilde{H}_P = \left(\Sigma \tilde{H}_P \right)^T.$$

Thus $\tilde{H} = P^T \tilde{H}_P P$ still has the desired form of a Hamiltonian J-Hessenberg matrix, but the upper right $n \times n$ block is no longer symmetric. Therefore, H is diagonally similar to a Hamiltonian J-Hessenberg matrix.

Unfortunately an SR step does not preserve this structure, and thus this modified version of the symplectic Lanczos method cannot be used in connection with our restart approaches.

Without some form of reorthogonalization any Lanczos algorithm is numerically unstable. Hence we re-J-orthogonalize each Lanczos vector as soon as it is computed against the previous ones via

$$w_m = w_m + S_P^{2m-2} J_P^{m-1} S_P^{2m-2^T} J_P^n w_m,$$

$$v_{m+1} = v_{m+1} + S_P^{2m} J_P^m S_P^{2m^T} J_P^n v_{m+1}.$$

This re-J-orthogonalization is costly: it requires $16n(m-1)$ flops for the vector w_m and $16nm$ flops for v_{m+1}. Thus, if $2k$ Lanczos vectors $v_1, w_1, \ldots, v_k, w_k$ are computed, the re-J-orthogonalization adds $16n(k+1)k - 32n$ flops to the overall cost of the symplectic Lanczos method.

For standard Lanczos algorithms, different reorthogonalization techniques have been studied (for references see, e.g., [25]). Those ideas can be used to design analogous re-J-orthogonalizations for the symplectic Lanczos method.

Another important issue is the numerical stability of the SR step employed in the restart. During the SR step on the $2k \times 2k$ shifted Hamiltonian J-Hessenberg matrix, all but $k-1$ transformations are orthogonal. These are known to be numerically stable. For the $k-1$ nonorthogonal symplectic transformations that have to be used, we choose among all possible transformations the ones with optimal (smallest possible) condition numbers as proposed in [9].

6.2. *Why Implicit Restarts?*

Implicit restarts have some advantages over explicit restarts, as will be discussed in this section. First of all, implicit restarts are more economical to implement. Assume we have to employ a restart after k steps of the symplectic Lanczos method. An implicit single-shift restart requires

$28n \cdot k + 16n + (100k - 65)$	flops for the implicit SR step,
$38n + 4nz$	flops for one additional Lanczos step, and
$32n \cdot k - 16n$	flops for re-J-orthogonalization.

That is a total of $4nz + 60n \cdot k + 38n + 100k - 65$ flops. An explicit restart requires

$4nz \cdot k + 32n \cdot k + 6n$	flops for k Lanczos steps, and
$16n \cdot (k + 1)k - 32n$	flops for re-J-orthogonalization.

This sums to $4nz \cdot k + 16n \cdot k^2 + 48n \cdot k - 26n$ flops. If an explicit restart with the starting vector $\breve{v}_1 = (H_P - \mu I)v_1$ were performed, that would add another $8n^2 + 2n$ to this flop count.

From these numbers we can conclude that performing an implicit restart is significantly cheaper than explicitly restarting the Lanczos iteration. This is due to the fact that an implicit SR step is usually cheaper than k Lanczos steps. Besides, we have to re-J-orthogonalize only once, while an explicit restart would require a re-J-orthogonalization in each iteration step. For more economical re-J-orthogonalization techniques implicit restarts are also advantageous. For double-shifted or multishifted restarts the implicit technique is still favorable, although the difference in the flop count becomes smaller.

Performing an explicit restart with $(H_P - \mu I)v_1$ or $(H_P - \mu I)(H_P + \mu I)v_1$ as the new starting vector, one is forced to directly multiply the old starting vector by matrices of the form $H_P - \mu I$. This can be avoided by the implicit method.

Note that the starting vector v_1 can be expressed as a linear combination of the eigenvectors y_i of H_P (assuming for simplicity that H_P is diagonalizable):

$$v_1 = \sum_{i=1}^{2n} \alpha_i y_i.$$

Then a single-shifted starting vector takes the form

$$\breve{v}_1 = \rho(H_P - \mu I)v_1 = \rho \sum_{i=1}^{2n} \alpha_i(\lambda_i - \mu) y_i,$$

where the λ_i are the eigenvalues corresponding to y_i. As the single shift selected will be real, applying such a modification to v_1 tends to emphasize those eigenvalues of H_P in $\breve{v}_1$ which correspond to eigenvalues λ_i with the largest positive or negative real part (depending on whether the chosen shift is positive or negative). Thus it is possible that the vector $\breve{v}_1$ will be dominated by information only from a few of the eigenvalues with largest real part. An implicit restart directly forms $\breve{S}_P^{2k}$ from a wide range of information available in S_P^{2k} and this should give better numerical results than the explicit computation of $\breve{v}_1$.

As an example consider

$$H = U\begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix}U^T,$$

where

$$A = \operatorname{diag}\left(-10^6, 9, 8, 7, 6, 5, 4, 3, \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix}\right)$$

is a block-diagonal matrix and U is the product of randomly generated symplectic Householder and Givens matrices. The eigenvalues of H can be read off directly. The following computations were done using MATLAB[2] on a Sun SPARC10. The starting vector v_1 is chosen randomly. After four steps of the symplectic Lanczos method the resulting 8 × 8 Hamiltonian J-Hessenberg matrix $\tilde{H}^8$ has the eigenvalues (computed by the MATLAB function `eig`)

$$\lambda(\tilde{H}^8) = \begin{Bmatrix} 9.999999999999997\text{E} + 05 \\ -9.999999999999997\text{E} + 05 \\ 3.040728370123861\text{E} + 00 \\ -3.040728370123995\text{E} + 00 \\ 9.200627380564711\text{E} + 00 \\ -9.200627380564642\text{E} + 00 \\ 9.477682371618508\text{E} + 00 \\ -9.477682371618551\text{E} + 00 \end{Bmatrix}.$$

To remove an eigenvalue pair from $\tilde{H}^8$ one can perform an implicitly double-shifted restart analogous to the single-shift restart described in Sec-

[2] MATLAB is a trademark of The MathWorks, Inc.

tion 5 (for a detailed derivation see [3]). Removing the two eigenvalues of smallest absolute value from $\tilde{H}^8$, we obtain a Hamiltonian J-Hessenberg matrix $\breve{H}^6_{\text{impl}}$ whose eigenvalues are

$$\lambda\left(\breve{H}^6_{\text{impl}}\right) = \left\{\begin{array}{r} 9.999999999999994\text{E} + 05 \\ -9.999999999999994\text{E} + 05 \\ 9.200627382497721\text{E} + 00 \\ -9.200627382497721\text{E} + 00 \\ 9.477682372414739\text{E} + 00 \\ -9.477682372414737\text{E} + 00 \end{array}\right\}.$$

From Theorem 4.1(e) it follows that these have to be the six eigenvalues of $\tilde{H}^8$ which have not been removed. As can be seen, we lose 4–5 digits during the implicit restart. Performing an explicit restart with the explicitly computed new starting vector $\breve{v}_1 = (H - \mu I)(H + \mu I)v_1$ yields a Hamiltonian J-Hessenberg matrix $\breve{H}^6_{\text{expl}}$ with eigenvalues

$$\lambda\left(\breve{H}^6_{\text{expl}}\right) = \left\{\begin{array}{r} 9.999999999999999\text{E} + 05 \\ -9.999999999999999\text{E} + 05 \\ 9.200679454660859\text{E} + 00 \\ -9.200679454660861\text{E} + 00 \\ 9.477559041923007\text{E} + 00 \\ -9.477559041923007\text{E} + 00 \end{array}\right\}.$$

This time we lost up to 10 digits. As a general observation from a wide range of numerical tests, the explicit restart loses at least 2 digits more than the implicit restart.

6.3. *Breakdowns in the SR Factorization*

So far we have assumed that the SR decomposition always exists. Unfortunately this assumption does not always hold. If there is a starting vector $\breve{v}_1$ for which the explicitly restarted symplectic Lanczos method breaks down, then it is impossible to reduce the Hamiltonian matrix H to Hamiltonian J-Hessenberg form with a transformation matrix whose first column is $\breve{v}_1$. Thus, in this situation the SR decomposition of $(H - \mu I)$ or $(H - \mu I)(H + \mu I)$ cannot exist.

As will be shown in this section, this is the only way that breakdowns in the SR decomposition can occur. In the single-shift SR step, only two types of elementary transformations are used. Most of them are orthogonal symplectic Givens rotations; their computation cannot break down. The other type of transformations used are symplectic Gaussian eliminations. These are the only possible source of breakdown.

THEOREM 6.1. *Suppose the Hamiltonian J-Hessenberg matrix $\tilde{H}^{2k}$ corresponding to* (11) *is unreduced, and let $\mu \in \mathbb{R}$. Let $G_P(j, y)$ be the jth permuted symplectic Gauss transformation required in the SR step on $\tilde{H}_P^{2k} - \mu I$. If the first $j-1$ permuted symplectic Gauss transformations of this SR step exist, then $G_P(j, y)$ fails to exist if and only if $\check{v}_j^T J_P H_P \check{v}_j = 0$ with $\check{v}_j$ as in* (15).

Proof. The proof follows the lines of [26, Theorem 3].

A symplectic Gauss transformation is defined such that

$$G(k, y)(ae_k + be_{n+k-1}) = \beta e_{n+k-1},$$

where $y = -b/a$. $G_P(k, y)$ denotes the permuted version of $G(k, y)$, i.e., $G_P(k, y) = PG(k, y)P^T$.

In the implicit SR decomposition, the first implicit rotation is selected so that the first columns of the implicit and the explicit S are equivalent. The remaining implicit rotations perform a bulge-chasing sweep down the subdiagonal to restore the J-Hessenberg form.

Assume that we have computed a symplectic matrix $\hat{S}_P^{2j}$ such that

$$\begin{bmatrix} \hat{S}_P^{2j} & 0 \\ 0 & I \end{bmatrix}^{-1} \tilde{H}_P^{2k} \begin{bmatrix} \hat{S}_P^{2j} & 0 \\ 0 & I \end{bmatrix}$$

has the desired permuted J-Hessenberg form in the first $2j$ columns. Then from (11), i.e.,

$$H_P S_P^{2j} = S_P^{2j} \tilde{H}_P^{2j} + \zeta_{j+1} v_{j+1} e_{2j}^T,$$

we obtain

$$H_P \check{S}_P^{2j} = \check{S}_P^{2j} \check{H}_P^{2j} + \zeta_{j+1} v_{j+1} e_{2j}^T \hat{S}_P^{2j},$$

where $\check{S}_P^{2j} = S_P^{2j} \hat{S}_P^{2j}$ and $\check{H}_P^{2j} = (\hat{S}_P^{2j})^{-1} \tilde{H}_P^{2j} \hat{S}_P^{2j}$. Since

$$\left(\check{S}_P^{2j}\right)^T J_P^n \check{S}_P^{2j} = J_P^j,$$

it follows that

$$-J_P^j\left(\breve{S}_P^{2j}\right)^T J_P^n H_P \breve{S}_P^{2j} = \breve{H}_P^{2j}. \tag{16}$$

The leading $(2j+2) \times (2j+2)$ principal submatrix of

$$\begin{bmatrix} \hat{S}_P^{2j} & 0 \\ 0 & I \end{bmatrix}^{-1} \tilde{H}_P^{2k} \begin{bmatrix} \hat{S}_P^{2j} & 0 \\ 0 & I \end{bmatrix}$$

is

$$\left[\begin{array}{cc|cc|cc|cc}
\breve{\delta}_1 & \breve{\beta}_1 & 0 & \breve{\zeta}_2 & & & & \\
\breve{\nu}_1 & -\breve{\delta}_1 & 0 & 0 & & & & \\
\hline
0 & \breve{\zeta}_2 & \ddots & & \ddots & & & \\
0 & 0 & & \ddots & & \ddots & & \\
\hline
 & & \ddots & & \breve{\delta}_j & \breve{\beta}_j & 0 & x_2 \\
 & & & \ddots & \breve{\nu}_j & -\breve{\delta}_j & 0 & x_1 \\
\hline
 & & & & x_1 & x_2 & \delta_{j+1} & \beta_{j+1} \\
 & & & & 0 & 0 & \nu_{j+1} & -\delta_{j+1}
\end{array}\right],$$

as $\zeta_{j+1} e_{2j}^T \hat{S}_P^{2j} = [0, \ldots, 0, x_1, x_2]^T$ because $\hat{S}_P^{2j}$ is an upper Hessenberg matrix. On the other hand, this leading principal submatrix can be expressed as

$$-J_P^{j+1}\left[\breve{S}_P^{2j} \;\middle|\; v_{j+1} \;\middle|\; w_{j+1}\right]^T J_P^n H_P \left[\breve{S}_P^{2j} \;\middle|\; v_{j+1} \;\middle|\; w_{j+1}\right]$$

using (16). That is,

$$\left[\begin{array}{c|cc}
\breve{H}_P^{2j} & -J_P^j\left(\breve{S}_P^{2j}\right)^T J_P^n H_P v_{j+1} & -J_P^j\left(\breve{S}_P^{2j}\right)^T J_P^n H_P w_{j+1} \\
\hline
-w_{j+1}^T J_P^n H_P \breve{S}_P^{2j} & \delta_{j+1} & \beta_{j+1} \\
v_{j+1}^T J_P^n H_P \breve{S}_P^{2j} & \nu_{j+1} & -\delta_{j+1}
\end{array}\right].$$

Thus we have

$$x_1 = -w_{j+1}^T J_P^n H_P \breve{v}_j,$$

$$x_2 = -w_{j+1}^T J_P^n H_P \breve{w}_j.$$

The next step in the implicit SR step eliminates x_1 using a transformation of type G_P. This can be done if $\breve{v}_j$ is nonzero. Hence, the SR step breaks down if $\breve{v}_j = 0$ and thus implies a breakdown in the symplectic Lanczos method.

The opposite implication follows from the uniqueness of the symplectic Lanczos method. ■

A similar theorem can be shown for the double-shift case considered in Section 7.2.

7. APPLICATIONS

7.1. *Approximating Eigenvalues and Eigenvectors of Hamiltonian Matrices*

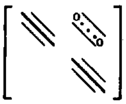
Lanczos-type algorithms are especially well suited for computing some of the extremal eigenvalues of a matrix. As a well-known fact, Lanczos algorithms usually produce *Ritz values* (i.e., eigenvalues of the reduced matrix) which converge very fast to the extremal eigenvalues of the original matrix (see, e.g., [25]).

The computed Ritz values can also be used as shifts either in the restart process (Section 7.2) or to accelerate convergence in the SR algorithm (see [43]). Besides, purely imaginary Ritz values signal that a stable k-dimensional invariant subspace of the computed $\tilde{H}^{2k}$ does not exist. This will be considered in Section 7.2.

Computing the Ritz values after the kth symplectic Lanczos step requires the computation of the eigenvalues of a $2k \times 2k$ Hamiltonian J-Hessenberg matrix as in (4). This can be done using the standard Hessenberg QR algorithm, which requires $O(k^3)$ flops. We present two different approaches which require only $O(k^2)$ flops.

7.1.1. Approximating the Eigenvalues of a Hamiltonian J-Hessenberg Matrix Using a Square Reduced Method. Squaring $\tilde{H}^{2k}$, we obtain a matrix of the following structure:

$$\left(\widetilde{H}^{2k}\right)^2 = M^{2k} = \begin{bmatrix} M_1^k & M_2^k \\ 0 & M_1^{kT} \end{bmatrix} = \qquad (17)$$



where

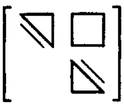
$$M_1^k = \begin{bmatrix} \mu_1 & \psi_2 & & & \\ \rho_2 & \mu_2 & \psi_3 & & \\ & \rho_3 & \ddots & \ddots & \\ & & \ddots & \ddots & \psi_k \\ & & & \rho_k & \mu_k \end{bmatrix},$$

$$\begin{aligned} \mu_j &= \delta_j^2 + \beta_j \nu_j, & j &= 1, \ldots, k, \\ \rho_j &= \gamma_j \nu_{j-1}, & j &= 2, \ldots, k, \\ \psi_j &= \gamma_j \nu_j, & j &= 2, \ldots, k. \end{aligned}$$

Hence the eigenvalues of M^{2k} may be obtained by computing the eigenvalues $\{\hat{\lambda}_1, \ldots, \hat{\lambda}_k\}$ of the nonsymmetric tridiagonal matrix M_1^k. Therefore, $\sigma(\tilde{H}^{2k}) = \{\pm\sqrt{\hat{\lambda}_1}, \ldots, \pm\sqrt{\hat{\lambda}_k}\}$, which reflects the structure of the spectrum of the Hamiltonian matrix $\tilde{H}^{2k}$.

This approach is similar to Van Loan's square-reduced algorithm [46]. There, a general Hamiltonian matrix H is first reduced to the so-called *square-reduced* form, i.e., a symplectic orthogonal matrix U is computed such that

$$(U^T H U)^2 = \begin{bmatrix} N_1 & N_2 \\ 0 & N_1^T \end{bmatrix} = \qquad (18)$$



Then the eigenvalues of H are computed by taking the square roots of the eigenvalues of the upper Hessenberg matrix N_1. Since Hamiltonian J-

Hessenberg matrices are already square-reduced, the reduction process (18) can be skipped in our case. Besides, M_1 is tridiagonal, whereas in the general case the corresponding block N_1 is an upper Hessenberg matrix.

For a detailed discussion of Van Loan's algorithm see [10, 46]. Squaring the Hamiltonian matrix may cause a loss of accuracy. A worst-case bound for the eigenvalues computed by Van Loan's method indicates that one may lose essentially half of the significant digits compared to eigenvalues computed by the QR algorithm. This is observed rather seldom in practice, though. On the other hand, this method reflects the structure of the spectrum of Hamiltonian matrices, whereas the standard QR algorithm often does not find exactly k eigenvalues in each half plane, since small perturbations may cause the computed eigenvalues to cross the imaginary axis.

7.1.2. *Computing Eigenvalues and Eigenvectors by the SR Algorithm.* Given a Hamiltonian J-Hessenberg matrix $\tilde{H} = \tilde{H}_0 \in \mathbb{R}^{2k \times 2k}$ as in (4), the SR algorithm computes a sequence of orthogonal and nonorthogonal symplectic similarity transformation matrices $\tilde{S}_j$, $j = 0, 1, \ldots,$ that preserve this structure, i.e., $\tilde{H}_{j+1} = S_j^{-1}\tilde{H}_j\tilde{S}_j$ is a Hamiltonian J-Hessenberg matrix for all $j = 0, 1, \ldots$. The sequence $\tilde{H}_j$ converges to a Hamiltonian matrix

$$\tilde{H}^{SR} = \left(S^{SR}\right)^{-1}\tilde{H}S^{SR} = \begin{bmatrix} D_1 & D_2 \\ 0 & -D_1^T \end{bmatrix} \tag{19}$$

where D_1, D_2 are block diagonal $k \times k$ matrices with blocks of size 1×1 or 2×2 and all transformations $\tilde{S}_j$ are accumulated in the symplectic matrix S^{SR}. The eigenvalues of $\tilde{H}$ are thus given by D_1 and their counterparts in $-D_1^T$. The eigenvectors corresponding to the eigenvalues contained in D_1 are given by the first k columns of S^{SR}. If (λ_i, s_i) represents such a *right* eigenpair, then because of the Hamiltonian structure, the corresponding *left* eigenpair is $(-\lambda_i, s_i^TJ)$. If only eigenvalues are desired, the SR algorithm is an $O(k^2)$ algorithm. If eigenvectors and/or invariant subspaces are required, S^{SR} has to be formed explicitly, which requires $O(k^3)$ flops. For a detailed discussion of QR-type algorithms based on SR decompositions see, e.g., [8, 9, 15, 36, 43].

Now assume that we have performed k steps of the symplectic Lanczos procedure and thus obtained the identity (after permuting back)

$$HS^{2k} = S^{2k}\tilde{H}^{2k} + \zeta_{k+1}\tilde{v}_{k+1}e_{2k}^T. \tag{20}$$

We can use the SR algorithm to compute eigenvalues and eigenvectors of $\tilde{H}^{2k}$. Setting $\tilde{H} = \tilde{H}^{2k}$ and $D_i = D_i^k$, $i = 1, 2$, in (19) and multiplying (20) from the right by S^{SR} yields

$$HS^{2k}S^{\mathrm{SR}} = S^{2k}S^{\mathrm{SR}}\begin{bmatrix} D_1^k & D_2^k \\ 0 & -D_1^{k^T} \end{bmatrix} + \zeta_{k+1}\bar{v}_{k+1}e_{2k}^T S^{\mathrm{SR}}. \tag{21}$$

Thus the Ritz values are the eigenvalues λ_j of D_1^k and their counterparts $-\lambda_j$. Define the vector $y_j = S^{2k}S^{\mathrm{SR}}e_j$ to be a *Ritz vector* corresponding to the Ritz value λ_j. Then from (21) with $s_j = S^{\mathrm{SR}}e_j$,

$$\|Hy_j - \lambda_j y_j\| = \|\zeta_{k+1}\bar{v}_{k+1}e_{2k}^T s_j\| = \left|\zeta_{k+1}(s_j)_{2k}\right|\|\bar{v}_{k+1}\| \tag{22}$$

indicates that if the last component of an eigenvector of $\tilde{H}^{2k}$ is small, then the Ritz pair (λ_j, y_j) is an approximation to an eigenpair of H (see, e.g., [6]). Thus the last row of S^{SR} shows which Ritz values and Ritz vectors yield good approximations to eigenvalues and eigenvectors of H.

Another application of the SR algorithm and of (21) is described in the next section.

7.2. *Low-Rank Approximations to Invariant Subspaces of Hamiltonian Matrices and Solutions of Algebraic Riccati Equations*

It is well known that the solution of the CARE (2),

$$Q + A^TX + XA - XGX = 0,$$

is connected with the invariant subspaces of the corresponding Hamiltonian matrix H (1). If the columns of

$$\begin{bmatrix} V \\ W \end{bmatrix} \in \mathbb{R}^{2n\times n}$$

span an invariant subspace of H, and $V \in \mathbb{R}^{n\times n}$ is invertible, then $X = -WV^{-1}$ solves (2). For discussion of existence and uniqueness of such solutions and further issues such as symmetry, see, e.g., [33, 37] and the references therein.

In control theory one is usually concerned with the symmetric (positive semidefinite) *stabilizing* solution of (2), i.e., a solution $\hat{X}$ such that $A - G\hat{X}$ is stable. Under the conditions that (A, G) is stabilizable and (Q, A) is detectable, such a solution exists, is unique, and may be determined by computing the stable invariant subspace of H. For simplification we will in the following assume that these conditions hold. Note that under these conditions, the Hamiltonian matrix does not have any purely imaginary eigenvalues.

Now suppose we have computed k steps of the symplectic Lanczos algorithm. Thus we obtain the $2k \times 2k$ Hamiltonian J-Hessenberg matrix $\tilde{H}^{2k}$. For a moment we will assume that $\tilde{H}^{2k}$ has no purely imaginary eigenvalues. Hence we can compute an invariant subspace of $\tilde{H}^{2k}$ by the SR algorithm as in (19). In [9] it is described how to separate the stable invariant subspace from (19) by symplectic similarity transformation which preserve the structure of (20). We can thus assume that D_1^k is stable and that the first k columns of S^{SR} span the stable invariant subspace of $\tilde{H}^{2k}$. Combining with the Lanczos factorization, we again obtain (21). If

$$Y^k = S^{2k} S^{SR} = \begin{bmatrix} Y_1^k & Y_2^k \end{bmatrix}, \qquad Y_1^k, Y_2^k \in \mathbb{R}^{2n \times k}, \tag{23}$$

we can conclude that the columns of Y_1^k span an approximate stable H-invariant subspace of dimension k if

$$\|HY_1^k - Y_1^k D_1^k\| = |\zeta_{k+1}| \, \|\bar{v}_{k+1} e_{2k}^T S_1^{\mathrm{SR}}\| \tag{24}$$

is sufficiently small, where $S^{\mathrm{SR}} = [S_1^{\mathrm{SR}} \ S_2^{\mathrm{SR}}]$ with $S_1^{\mathrm{SR}}, S_2^{\mathrm{SR}} \in \mathbb{R}^{2k \times k}$.

We want to use this low-rank approximate stable H-invariant subspace to compute a low-rank approximation to the solution of the CARE (2). So far it is not clear what is the best way to obtain such a solution, especially because there may be different interpretations of what is the "best" low-rank approximation. One possible strategy for constructing a low-rank approximate Riccati solution from a low-rank approximate stable H-invariant subspace is considered in [23]. In the following we will describe another possibility which differs slightly from the one given in [23].

Since $(S^{2k})^T J^n \bar{v}_{k+1} = 0$ and Y^k satisfies the symplecticity property

$$(Y^k)^T J^n Y^k = J^k, \tag{25}$$

we obtain from (21)

$$(J^k)^T (Y^k)^T J^n H Y^k = \begin{bmatrix} D_1^k & D_2^k \\ 0 & -(D_1^k)^T \end{bmatrix}, \tag{26}$$

and from the lower left block of this equation

$$-(Y_{21}^k)^T A Y_{11}^k + (Y_{11}^k)^T Q Y_{11}^k - (Y_{21}^k)^T G Y_{21}^k - (Y_{11}^k)^T A^T Y_{21}^k = 0, \tag{27}$$

where

$$Y_1^k = \begin{bmatrix} Y_{11}^k \\ Y_{21}^k \end{bmatrix}.$$

Let $Y_{11}^k = Z^k R^k$ be an "economy size" QR factorization, i.e., $Z^k \in \mathbb{R}^{n \times k}$ has orthonormal columns and $R^k \in \mathbb{R}^{k \times k}$ is an upper triangular matrix. Assuming that Y_{11}^k has full column rank, R^k is invertible. Premultiplying (27) by $(R^k)^{-T}$ and postmultiplying by $(R^k)^{-1}$ yields

$$\begin{aligned} -(R^k)^{-T}(Y_{21}^k)^T A Z^k + (Z^k)^T Q Z^k - (R^k)^{-T}(Y_{21}^k)^T G Y_{21}^k (R^k)^{-1} \\ -(Z^k)^T A^T Y_{21}^k (R^k)^{-1} = 0. \end{aligned} \tag{28}$$

Setting $X^k = -Y_{21}^k (R^k)^{-1} (Z^k)^T$, we obtain

$$(Z^k)^T \big((X^k)^T A + Q - (X^k)^T G X^k + A^T X^k \big) Z^k = 0. \tag{29}$$

The computed X^k may now be considered as a low-rank approximation to the solution of (2). From (26) we obtain

$$(Z^k)^T (A - G X^k) Z^k = R^k D_1^k (R^k)^{-1} + E_1^k, \tag{30}$$

where E_1^k is the upper left $k \times k$ block of $(Z^k)^T(\zeta_{k+1} \bar{v}_{k+1} e_{2k}^T S^{\mathrm{SR}})$. From (29) and (30) it is clear that in exact arithmetic for $k = n$, X^k is the required stabilizing solution of (2). The properties of this low-rank approximation are under current investigation and will be described elsewhere.

So far, we have assumed that $\tilde{H}^{2k}$ has no eigenvalues on the imaginary axis. Under the above assumptions, H has no purely imaginary eigenvalues.

But for $\tilde{H}^{2k}$, $k < n$, computed by the Lanczos process, in general this property (and also the stabilizability-detectability condition) does not hold. Thus we may expect $\tilde{H}^{2k}$ to have purely imaginary eigenvalues for some k. If this happens, H^{2k} does not have a stable, k-dimensional invariant subspace.

One way to remove these eigenvalues is to employ a double-shifted implicit restart. Suppose $\tilde{H}^{2k}$ has l pairs of purely imaginary eigenvalues denoted by $\imath\mu_1, -\imath\mu_1, \ldots, \imath\mu_l, -\imath\mu_l$. We can then perform a double-shifted implicit restart corresponding to the starting vector $\breve{v}_1 = \rho(H - \imath\mu_1 I)(H + \imath\mu_1 I)v_1$ to obtain a new Lanczos identity which, after permuting back, reads

$$H\breve{S}^{2k-2} = \breve{S}^{2k-2}\breve{H}^{2k-2} + \bar{r}_k e_{2k-2}^T. \tag{31}$$

Because of Theorem 4.1 the Hamiltonian J-Hessenberg matrix $\breve{H}^{2k-2}$ has the same eigenvalues as $\tilde{H}^{2k}$ besides the removed pair $\pm\imath\mu_1$. The remaining pairs of purely imaginary eigenvalues can be removed with another $l - 1$ double-shifted implicit restarts to obtain a new Lanczos factorization

$$H\breve{S}^{2(k-l)} = \breve{S}^{2(k-l)}\breve{H}^{2(k-l)} + \bar{r}_{k-l+1}e_{2(k-l)}^T, \tag{32}$$

where the eigenvalues of $\breve{H}^{2(k-l)}$ are those eigenvalues of $\tilde{H}^{2k}$ having nonzero real parts. The starting vector corresponding to the Lanczos factorization (32) is the multishift vector

$$\breve{v}_1 = \rho(H - \imath\mu_l I)(H + \imath\mu_l I)\cdots(H - \imath\mu_1 I)(H + \imath\mu_1 I)v_1.$$

Thus it is possible to compute a low-rank approximate stable H-invariant subspace of dimension $k - l$ and the corresponding Riccati solution. If an approximation of dimension k is required, we may use the same approach as in [26], where restarts are used to obtain a stable reduced-order system: performing l symplectic Lanczos steps, we obtain from $\breve{H}^{2(k-l)}$ a new Hamiltonian J-Hessenberg matrix $\breve{H}^{2k}$ with hopefully no eigenvalues on the imaginary axis. If there are again purely imaginary eigenvalues, we have to repeat the restart process. In preliminary numerical experiments, this never produced an $\breve{H}^{2k}$ having again l (or even more) pairs of purely imaginary Ritz values. With this method we obtain, after a finite number of restarts, a

Hamiltonian J-Hessenberg matrix of required dimension having only eigenvalues with nonzero real part. This approach is under current investigation, and a detailed analysis will be given elsewhere.

8. NUMERICAL RESULTS

In this section we present some examples to demonstrate the ability of the proposed algorithm to deal with the numerical difficulties of the symplectic Lanczos method. As our results concerning convergence and choice of the shifts in the k-step restarted symplectic Lanczos method are still preliminary, we will give a detailed discussion of algorithmic issues and on numerical results in a different paper. But we would like to remark that these computational results are quite promising. Here we will demonstrate the typical behavior of the symplectic Lanczos method. Especially we concentrate on the potential ability to overcome (near) breakdown by an implicit restart. Typically a serious breakdown after j steps of the symplectic Lanczos method can be overcome by a restart. But the change in the starting vector is limited, so that another serious breakdown might occur after $j + l$ steps, l small. Hence there is hope that in the rare case a serious breakdown occurs during a k-step symplectic Lanczos algorithm, this breakdown can be overcome by a restart if $k \ll n$.

An example where the restart process is used to remove eigenvalues was already given in Section 6.2.

All computations were done using MATLAB Version 4.2c on a Sun SPARC10 with IEEE double-precision arithmetic and machine precision $\varepsilon = 2.2204 \times 10^{-16}$. In case the symplectic Lanczos method encounters a serious breakdown (or near-breakdown), that is, if $\nu_j = 0$ for some j (or $|\nu_j| < tol$, where tol is an appropriately chosen tolerance), then an implicit single-shifted restart as discussed in Section 5 is employed. If breakdown occurs during the restart or if the original breakdown condition persists after the restart, the implicit restart is repeated at most three times with a different randomly chosen shift. After three consecutive unsuccessful recovery attempts, the restart attempts are stopped and an explicit restart with a new random starting vector is initiated.

We tested the restarted symplectic Lanczos method for the Hamiltonian matrices corresponding to the continuous-time algebraic Riccati equations given in the benchmark collection [4]. Restarts due to a serious breakdown were only encountered in a very few cases, and we never had to perform an explicit restart when choosing a random starting vector.

To demonstrate the restart process we report the two most intriguing of those examples. Due to a special starting vector, the implicit restart fails for the first example and an explicit restart has to be performed. The second example demonstrates a serious breakdown overcome by an implicit restart.

EXAMPLE 1 (See [2, Example 1] and [4, Example 7]). The first example shows that a serious breakdown cannot always be overcome by employing an implicit restart. Let

$$H = \begin{bmatrix} 1 & 0 & \epsilon & 0 \\ 0 & -2 & 0 & 0 \\ 1 & 1 & -1 & 0 \\ 1 & 1 & 0 & 2 \end{bmatrix}.$$

As a starting vector v_1 for the symplectic Lanczos method we choose e_1. During the first step of the symplectic Lanczos algorithm the following quantities are computed:

$$\zeta_1 = 1, \qquad \nu_1 = 1, \qquad w_1 = e_2 + e_4$$

$$\beta_1 = \epsilon, \qquad \zeta_2 = 3, \qquad v_2 = e_4.$$

For the second step, $\tilde{w}_2$ and ν_2 have to be computed:

$$\tilde{w}_2 = e_4, \qquad \nu_2 = 0.$$

A serious breakdown is encountered. An implicit restart with the new starting vector

$$v_1 = (H_P - \mu I)e_1 = [1 - \mu, 1, 0, 1]^T$$

will break down at the same step, as any further restart will. In fact, any restart with a starting vector v_1 of the form $[a, b, 0, c]^T$ will break down, as this implies that

$$w_1 = \frac{1}{\nu_1}\begin{bmatrix} \epsilon b \\ a - 2b \\ 0 \\ a + \epsilon \end{bmatrix}, \qquad \nu_1 = a^2 - 2ab - \epsilon b^2, \qquad \beta_1 = \frac{\epsilon}{\nu_1}$$

and

$$v_2 = e_4$$

as before. For any vector of the form $v = [0, 0, 0, x]^T$ we have $v^T J_P H_P v = 0$ and thus a serious breakdown. If our starting vector is of the form $[a, b, 0, c]^T$, then the new starting vector in the single-shifted restart is of the same form, and thus the serious breakdown cannot be overcome by implicit single-shifted restarts. An explicit restart with a random starting vector is successful.

EXAMPLE 2 (See [14] and [4, Example 13]). The second example demonstrates a serious breakdown overcome by an implicit single-shifted restart. Let

$$H = \begin{bmatrix} 0 & 0.4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.345 & 0 & 0 & 0 & 0 & 0 \\ 0 & -524000 & -465000 & 262000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -10^6 & 0 & 0 & 0 & 10^{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.4 & 0 & 524000 & 0 \\ 0 & 0 & 1 & 0 & 0 & -0.345 & 465000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -262000 & 10^6 \end{bmatrix}.$$

As a starting vector v_1 for the symplectic Lanczos method we choose e_1. During the first step of the symplectic Lanczos algorithm the following quantities are computed:

$$\zeta_1 = 1, \qquad \nu_1 = 1, \qquad w_1 = e_2 - e_1,$$

$$\beta_1 = -1, \qquad \zeta_2 = 0.4, \qquad v_2 = -e_4.$$

A serious breakdown is encountered as $\nu_2 = 0$. After an implicit restart with the new starting vector $v_1 = (H_P - \mu I)e_1 = [-\mu, 1, 0, 0, 0, 0, 0, 0]^T$, the breakdown condition $\nu_2 = 0$ persists. Thus the restart is repeated with a different shift $\tilde{\mu}$ yielding the new starting vector $v_1 = (H_P - \tilde{\mu} I)(H_P - \mu I)e_1 = [\tilde{\mu}\mu, -\mu - \tilde{\mu}, 0, -0.4, 0, 0, 0, 0]^T$. This restart is successful.

EXAMPLE 3. In computational chemistry, large eigenvalue problems arise for example in linear response theory. The simplest model of a response function for the response of a single self-consistent-field state to an external perturbation is realized by the time-dependent Hartree-Fock model. This leads to the generalized eigenvalue problem (see [39])

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} x = \lambda \begin{bmatrix} \Sigma & \Delta \\ -\Delta & -\Sigma \end{bmatrix} x. \tag{33}$$

Here, $A, B, \Sigma \in \mathbb{R}^{n \times n}$ are symmetric and $\Delta \in \mathbb{R}^{n \times n}$ is skew symmetric. For a closed-shell Hartree-Fock wave function we have $\Sigma = I_n$ and $\Delta = 0$. Thus, the generalized eigenvalue problem (33) reduces to the standard Hamiltonian eigenvalue problem

$$\begin{bmatrix} A & B \\ -B & -A \end{bmatrix} x = \lambda x.$$

The order of the matrices considered in linear response theory can easily reach $n = 10^6, 10^7$. Computations with such models require a thorough implementation as well as adequate data structures and are planned for the future. Here we want to present only a simple model and the results obtained by the symplectic Lanczos process. The chosen example is similar to an example presented in [19], where special versions of the Lanczos algorithm for matrices as given in (33) are examined.

Let $n = 100$, $D = \operatorname{diag}(d_1, \dots, d_n)$, and $\hat{D} = \operatorname{diag}(\hat{d}_1, \dots, \hat{d}_n)$, where $d_1 = 200.0$, $d_2 = 100.0$, $d_3 = 50.0$, $d_i = (i - 1) \times 0.001$ for $i = 4, \dots, n$, and $\hat{d}_1 = \hat{d}_2 = \hat{d}_3 = 0.0$, $\hat{d}_i = i \times 0.0001$. Now set $A = U^T \hat{D} U$ and $B = U^T \hat{D} U$ with a Householder matrix $U = I_n - 2ww^T / w^T w$, where $w = [1, 2, \dots, 100]$. The resulting matrix

$$H = \begin{bmatrix} A & B \\ -B & -A \end{bmatrix}$$

is Hamiltonian and has eigenvalues

$$\{ \pm 200.0, \pm 100.0, \pm 50.0, \pm \lambda_4, \dots, \pm \lambda_n \},$$

where $0.001 < |\lambda_i| < 0.1$ for $i = 4, \dots, n$.

After three steps of the symplectic Lanczos algorithm (without re-J-orthogonalization) we obtain the Ritz values (computed by the method suggested in Section 7.1.1):

$$\pm 1.9999914572790830\text{E} + 02, \qquad \pm 9.931554785773068\text{E} + 01,$$

$$\pm 3.371968773385778\text{E} + 01.$$

That is, the largest eigenvalue is approximated with a relative accuracy of order 10^{-5}. The next Lanczos step yields the Ritz values

$$\pm 1.999999999999998\mathrm{E}+02, \qquad \pm 9.999999999999989\mathrm{E}+01,$$

$$\pm 4.999999999997731\mathrm{E}+01, \qquad \pm 8.451080813545205\mathrm{E}-02,$$

i.e., the three largest Ritz values have (almost) converged to the three largest eigenvalues of H. Thus, one can expect a loss of symplecticity (J-orthogonality) in the Lanczos vectors and, as in standard Lanczos algorithms, that the converged eigenvalues will be duplicated (*ghost eigenvalue problem*; see, e.g., [25]). In fact, after 9 iterations we have Ritz values

$$\pm 1.999999999999999\mathrm{E}+02, \qquad \pm 1.999999999999997\mathrm{E}+02,$$

$$\pm 9.999999999985583\mathrm{E}+01, \qquad \pm 9.999999999999999\mathrm{E}+01,$$

$$\pm 5.000000000000038\mathrm{E}+01, \qquad \pm 4.999999974747666\mathrm{E}+01,$$

$$\pm 9.524662688488485\mathrm{E}-02, \qquad \pm 7.720710855953188\mathrm{E}-02,$$

$$\pm 3.757475009324353\mathrm{E}-02.$$

Using complete re-J-orthogonalization, this effect is avoided and we obtain after 9 steps the following Ritz values:

$$\pm 1.999999999999999\mathrm{E}+02, \qquad \pm 9.999999999999993\mathrm{E}+01,$$

$$\pm 4.999999999999997\mathrm{E}+01, \qquad \pm 9.754957790699192\mathrm{E}-02,$$

$$\pm 9.154380154101090\mathrm{E}-02, \qquad \pm 8.237785481069571\mathrm{E}-02,$$

$$\pm 6.786890886560507\mathrm{E}-02, \qquad \pm 4.923341543122169\mathrm{E}-02,$$

$$\pm 1.448276946901055\mathrm{E}-02.$$

These first results give rise to the hope that the (implicitly restarted) symplectic Lanczos algorithm is an efficient tool for the numerical solution of large scale Hartree–Fock problems.

9. CONCLUDING REMARKS

We have presented a symplectic Lanczos method for the Hamiltonian eigenproblem. This method is used to approximate a few eigenvalues and associated eigenvectors and to compute a low-rank approximate stable invariant subspace of a Hamiltonian matrix which can be used to approximate the stabilizing solution of continuous-time algebraic Riccati equations. Unfortunately, the symplectic Lanczos process has the same inherent numerical difficulties as any Lanczos-like method. When used to compute a low-rank approximation to the solution of continuous-time algebraic Riccati equations, there is no guarantee that the symplectic Lanczos process yields a reduced Hamiltonian matrix $\tilde{H}^{2k}$ having a stable k-dimensional invariant subspace due to purely imaginary Ritz values. Inexpensive implicit restarts are developed which can be used to deal with the numerical difficulties of the symplectic Lanczos process and to remove the undesirable purely imaginary Ritz values.

As in the standard nonsymmetric Lanczos method, one can expect convergence of eigenvalues after a small number of steps.

Our analysis indicates that the implicitly restarted symplectic Lanczos method is an efficient tool for extracting a few eigenvalues of large Hamiltonian matrices. Nevertheless the method needs to be tested on a broader range of problems.

We have presented a possible way to use the method to approximate the solution of algebraic Riccati equations. But it is yet not clear what is the best way to form an approximate solution X from a low-rank approximation to the stable invariant subspace of the Hamiltonian matrix. This will be the topic of further studies. Future work will also include the study of symplectic Lanczos methods for the (generalized) symplectic eigenvalue problem and the related discrete-time algebraic Riccati equation as well as possible combinations of the restart process with look-ahead approaches.

REFERENCES

1 G. Ammar, P. Benner, and V. Mehrmann, A multishift algorithm for the numerical solution of algebraic Riccati equations, *Electron. Trans. Numer. Anal.* 1:33–48 (1993).

2 W. Arnold, III, and A. Laub, Generalized eigenproblem algorithms and software for algebraic Riccati equations, *Proc. IEEE* 72:1746–1754 (1984).

3 P. Benner and H. Faβbender, A Restarted Symplectic Lanczos Method for the Hamiltonian Eigenvalue Problem, Tech. Rep. SPC 95_28, Fak f. Mathematik, TU Chemnitz-Zwickau, 09107 Chemnitz, FRG, 1995.

4 P. Benner, A. Laub, and V. Mehrmann, A Collection of Benchmark Examples for the Numerical Solution of Algebraic Riccati Equation I: Continuous-Time Case, Tech. Rep. SPC 95_22, Fak. f. Mathematik, TU Chemnitz-Zwickau, 09107 Chemnitz, FRG, 1995.

5 J. Bunch, The weak and strong stability of algorithms in numerical algebra, *Linear Algebra Appl.* 88:49–66 (1987).

6 W. Bunse and A. Bunse-Gerstner, *Numerische Lineare Algebra*, Teubner, Stuttgart, 1985.

7 A. Bunse-Gerstner, An analysis of the HR algorithm for computing the eigenvalues of a matrix, *Linear Algebra Appl.* 35:155–173 (1981).

8 A. Bunse-Gerstner, Matrix factorization for symplectic QR-like methods, *Linear Algebra Appl.* 83:49–77 (1986).

9 A. Bunse-Gerstner and V. Mehrmann, A symplectic QR-like algorithm for the solution of the real algebraic Riccati equation, *IEEE Trans. Automat. Control* AC-31:1104–1113 (1986).

10 R. Byers, Hamiltonian and Symplectic Algorithms for the Algebraic Riccati Equation, Ph.D. Thesis, Dept. of Computer Science, Cornell Univ., Ithaca, N.Y., 1983.

11 ____, A Hamiltonian QR-algorithm, *SIAM J. Sci. Statist. Comput.* 7:212–229 (1986).

12 ____, A bisection method for measuring the distance of a stable to unstable matrices, *SIAM J. Sci. Statist. Comput.* 9:875–881 (1988).

13 D. Calvetti, L. Reichel, and D. C. Sorensen, An implicitly restarted Lanczos method for large symmetric eigenvalue problems, *Electron. Trans. Numer. Anal.* 2:1–21 (1994).

14 J. Chow and P. Kokotovic, A decomposition of near-optimum regulators for systems with slow and fast modes, *IEEE Trans. Automat. Control* AC-21:701–705 (1976).

15 J. Della-Dora, Sur Quelques Algorithmes de Recherche de Valeurs Propres, Thèse, Univ. Scientifique et Medicale de Grenoble, 1973.

16 P. Van Dooren, Talk given at IMA Conference on Linear Algebra and Applications, Manchester, U.K., July 1995.

17 J. Doyle, K. Glover, P. Khargonekar, and B. Francis, State-space solutions to standard H_2 and H_∞ control problems, *IEEE Trans. Automat. Control* AC-34:831–847 (1989).

18 L. Elsner, On some algebraic problems in connection with general eigenvalue algorithms, *Linear Algebra Appl.* 26:123–138 (1979).

19 U. Flaschka, Eine Variante des Lanczos-Algorithmus für große, dünn besetzte symmetrische Matrizen mit Blockstruktur, Preprint 92-057, SFB 343, "Diskrete Strukturen in der Mathematik," Univ. Bielefeld, Bielefeld, FRG, 1992.

20 J. G. F. Francis, The QR transformation, Parts I and II, *Comput. J.* 4:265–271, 332–345 (1961).

21 R. Freund, Transpose-Free Quasi-Minimal Residual Methods for Non-Hermitian Linear Systems, Numerical Analysis Manuscript 92-97, AT & T Bell Labs., 1992.

22 R. Freund, Lanczos-type algorithms for structured non-Hermitian eigenvalue problems, in *Proceedings of the Cornelius Lanczos International Centenary Conference* (J. Brown, M. Chu, D. Ellison, and R. Plemmons, Eds.), SIAM, 1994, pp. 243–245.

23 R. Freund and V. Mehrmann, A Symplectic Look-ahead Lanczos Algorithm for the Hamiltonian Eigenvalue Problem, Manuscript.

24 R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal, An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices, *SIAM J. Sci. Comput.* 14:137–158 (1993).

25 G. Golub and C. Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins U.P., Baltimore, 1989.

26 E. Grimme, D. Sorensen, and P. Van Dooren, Model reduction of state space systems via an implicitly restarted Lanczos method, *Numer. Algorithms* 12:1–31 (1996).

27 M. Gutknecht, A completed theory of the unsymmetric Lanczos process and related algorithms, Part I, *SIAM J. Matrix Anal. Appl.* 13:594–639 (1992).

28 ____, A completed theory of the unsymmetric Lanczos process and related algorithms, Part II, *SIAM J. Matrix Anal. Appl.* 15:15–58 (1994).

29 A. Hodel and K. Poolla, Heuristic approaches to the solution of very large sparse Lyapunov and algebraic Riccati equations, in *Proceedings of the 27th IEEE Conference on Decision and Control*, Austin, Tex., 1988, pp. 2217–2222.

30 A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964.

31 I. M. Jaimoukha and E. M. Kasenally, Krylov subspace methods for solving large Lyapunov equations, *SIAM J. Numer. Anal.* 31:227–251 (1994).

32 V. N. Kublanoskaja, On some algorithms for the solution of the complete eigenvalue problem, *U.S.S.R. Comput. Math. and Math. Phys.* 3:637–657 (1961).

33 P. Lancaster and L. Rodman, *The Algebraic Riccati Equation*, Oxford U.P., Oxford, 1995.

34 C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, *J. Res. Nat. Bur. Standards* 45:255–282 (1950).

35 A. Laub, Invariant subspace methods for the numerical solution of Riccati equations, in *The Riccati Equation* (S. Bittanti, A. Laub, and J. Willems, Eds.), Springer-Verlag, Berlin, 1991, pp. 163–196.

36 V. Mehrmann, Der SR-Algorithmus zur Berechnung der Eigenwerte einer Matrix, Diplomarbeit, Univ. Bielefeld, Bielefeld, FRG, 1979.

37 ____, The *Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution*, Lecture Notes in Control and Inform. Sci. 163, Springer-Verlag, Heidelberg, July 1991.

38 G. Mei, A New Method for Solving the Algebraic Riccati Equation, Master's Thesis, Nanjing Aeronautical Inst., Campus P.O. Box 245, Nanjing, P.R. China, 1986.

39 J. Olson, H. Jensen, and P. Jørgensen, Solution of large matrix equations which occur in response theory, *J. Comput. Phys.* 74:265–282 (1988).

40 C. Paige, The Computation of Eigenvalues and Eigenvectors of Very Large Sparse Matrices, Ph.D. Thesis, Univ. of London (U.K.), 1971.

41 B. N. Parlett, D. R. Taylor, and Z. A. Liu, A look-ahead Lanczos algorithm for unsymmetric matrices, *Math. Comp.* 44:105–124 (1985).

42 I. Rosen and C. Wang, A multi-level technique for the approximate solution of operator Lyapunov and algebraic Riccati equations, *SIAM J. Numer. Anal.* 32:514–541 (1995).

43 V. Sima, Algorithm GRICSR solving continuous-time algebraic Riccati equations using Gaussian symplectic transformations, *Stud. Res. Comp. Inf.* 1:237–254 (1992).

44 D. C. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, *SIAM J. Matrix Anal. Appl.* 13:357–385 (1992).

45 D. R. Taylor, Analysis of the Look-ahead Lanczos Algorithm, Ph.D. Thesis, Center for Pure and Applied Mathematics, Univ. of California, Berkeley, 1982.

46 C. Van Loan, A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix, *Linear Algebra Appl.* 16:233–251 (1984).

Received 7 November 1995; final manuscript accepted 3 September 1996