Technische Universität Chemnitz-Zwickau

DFG-Forschergruppe "SPC" · Fakultät für Mathematik

Peter Benner Heike Faßbender

A Restarted Symplectic Lanczos Method for the Hamiltonian Eigenvalue Problem

Preprint-Reihe der Chemnitzer DFG-Forschergruppe
"Scientific Parallel Computing"

SPC 95_28 Oktober 1995

A Restarted Symplectic Lanczos Method for the Hamiltonian Eigenvalue Problem

Peter Benner * Heike Faßbender †

October 26, 1995

Abstract

A restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem is presented. The Lanczos vectors are constructed to form a symplectic basis. Breakdowns and near-breakdowns are overcome 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.

Key words : symplectic Lanczos method, implicit restarting, Hamiltonian matrix, eigenvalues, low rank approximate solution, algebraic Riccati equation.
AMS(MOS) subject classifications : 65F15, 65F50

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 [11, 15]) and the $\mathcal{H}_\infty$–norm of transfer matrices (see [16]). 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. [38].

*Technische Universität Chemnitz–Zwickau, Fakultät für Mathematik, 09107 Chemnitz, FRG, E-mail: benner@mathematik.tu-chemnitz.de. This author was supported by *Deutsche Forschungsgemeinschaft*, research grant *Me 790/7–1 Singuläre Steuerungsprobleme*.

†Universität Bremen, Fachbereich 3 - Mathematik und Informatik, 28357 Bremen, FRG. E-mail: heike@mathematik.uni-bremen.de

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

$$Q + A^T X + XA - XGX = 0 \tag{2}$$

and its connection to the Hamiltonian eigenproblem (1) is well known, see e.g. [32, 34, 36] and the references given therein. The solution of CARE (2) with small and dense coefficient matrices (say $n \leq 100$) has been the topic of numerous publications during the last 30 years. Even for these problems a numerically sound method, i.e., a strongly backwards stable method in the sense of [4], is yet not known. Only a few attempts have been made to solve (1) for large and sparse matrices, e.g. [28, 30, 43]. 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 $\lambda, \overline{\lambda}, -\lambda, -\overline{\lambda}$. *Symplectic* matrices S are defined by the property $S^T JS = J$ 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 [10] for the case that rank $G = 1$ or rank $Q = 1$. To the best of our knowledge, the only exisiting 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 $\widetilde{H}$ such that

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

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 [8]. This algorithm is a QR-like method for the Hamiltonian eigenproblem based on the SR decomposition. There, $\widetilde{H}$ is computed by an elimination process. During this elimination process the use of very badly conditioned matrices can not 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 [37] 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 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 [7, 8, 35, 36, 45]. In [22] 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 restarted Lanczos methods [12, 25, 46] together 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.

In Section 2 the implictly 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 all details of the SR decomposition necessary for the restart are presented. 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

Given $v_1, w_1 \in \mathbb{R}^n$ and a nonsymmetric matrix $A \in \mathbb{R}^{n\times n}$, the standard nonsymmetric Lanczos algorithm [33] produces matrices $P_k = [p_1, \ldots, p_k] \in \mathbb{R}^{n\times k}$ 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 $Q_k^T P_k = I_k$ (bi-orthogonality). One pleasing result of this bi-orthogonality condition is that multiplying (5) on the left by Q_k^T yields the relationship $Q_k^T A P_k = T_k$.

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 [39] that bi-orthogonality will in fact be lost when at least one of the eigenvalues of T_k converges to an eigenvalue of A. (See also [24] and the references therein.)

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. It is known [29] that vectors q_1 and p_1 exist so that the Lanczos process with these as starting vectors does not encounter 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 exakt breakdowns. A complete treatment of the modified Lanczos algorithm and its intimate connection with orthogonal polynomials and Padé approximation was presented by Gutknecht [26, 27]. Taylor [47] 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 occurence of a (near) breakdown. Freund, Gutknecht, and Nachtigal presented in [23] a look-ahead Lanczos code that can handle look-ahead steps of any length.

A different approach to overcome breakdowns and near-breakdowns is to modify the starting vectors by an implicitly restarted Lanczos process. Given that P_k and Q_k from (5) and (6) are known, an implicit Lanczos restart computes the Lanczos factorization

$$A\widetilde{P}_k = \widetilde{P}_k\widetilde{T}_k + \widetilde{r}_k e_k^T \tag{7}$$
$$A^T\widetilde{Q}_k = \widetilde{Q}_k\widetilde{T}_k^T + \widetilde{q}_k e_k^T \tag{8}$$

which corresponds to the starting vectors

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

without explicitly restarting the Lanczos process with the vectors (9). For a detailed derivation see [25] and the related work in [12, 46].

In Section 5 we show how to use this approach to overcome (near) breakdown in 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]. 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 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 we use in the following a permuted version of H and $\widetilde{H}$. Let

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

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

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

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

From $S^TJS = J$ we obtain

$$S_P^T J_P S_P = J_P = \begin{bmatrix} 0 & 1 & & & & & \\ -1 & 0 & & & & & \\ & & 0 & 1 & & & \\ & & -1 & 0 & & & \\ & & & & \ddots & & \\ & & & & & 0 & 1 \\ & & & & & -1 & 0 \end{bmatrix}$$

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

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

The structure preserving Lanczos method generates a sequence of matrices

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

satisfying

$$(11)\qquad H_P S_P^{2k} = S_P^{2k} \widetilde{H}_P^{2k} + \zeta_{k+1} v_{k+1} e_{2k}^T$$

where $\widetilde{H}_P^{2k} = P^k \widetilde{H}^{2k} P^{k^T}$ is a permuted $2k \times 2k$ Hamiltonian J-Hessenberg matrix $\widetilde{H}^{2k}$ 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 process to Hamiltonian J-Hessenberg form, e.g. [8]. Assuming that no such breakdowns occur, a symplectic Lanczos method can be derived as follows.

Let $S_P = [v_1, w_1, v_2, w_2, \ldots, v_n, w_n]$. For a given v_1, a Lanczos method constructs the matrix S_P columnwise from the equations

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

That is, for odd numbered columns

$$\begin{aligned} H_P v_{m+1} &= \delta_{m+1} v_{m+1} + \nu_{m+1} w_{m+1} \\ \Longleftrightarrow \nu_{m+1} w_{m+1} &= H_P v_{m+1} - \delta_{m+1} v_{m+1} \\ &=: \widetilde{w}_{m+1} \end{aligned} \tag{12}$$

and for even numbered columns

$$\begin{aligned} H_P w_m &= \zeta_m v_{m-1} + \beta_m v_m - \delta_m w_m + \zeta_{m+1} v_{m+1} \\ \Longleftrightarrow \zeta_{m+1} v_{m+1} &= H_P w_m - \zeta_m v_{m-1} - \beta_m v_m + \delta_m w_m \\ &=: \widetilde{v}_{m+1}. \end{aligned} \tag{13}$$

Now we have to choose ν_{m+1}, ζ_{m+1} such that $S_P^T J_P S_P = J_P$ is satisfied, that is we have to choose ν_{m+1}, ζ_{m+1} such that $v_{m+1}^T J_P w_{m+1} = 1$. One possibility is to choose

$$\zeta_{m+1} = ||\widetilde{v}_{m+1}||_2, \qquad \nu_{m+1} = v_{m+1}^T J_P H_P v_{m+1}.$$

Premultiplying $\widetilde{v}_{m+1}$ by $w_m^T J_P$ and using $S_P^T J_P S_P = J_P$ yields

$$\beta_m = -w_m^T J_P H_P w_m.$$

Thus 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}).

[1](Following [24], we define each floating point arithmetic operation together with the associated integer indexing as a flop.)

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

Table 1: Symplectic Lanczos Method

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 [35]. 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.

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 $\widetilde{v}_{m+1}$ 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 $\widetilde{v}_{m+1}$ to be any vector satisfying

$$\begin{aligned} v_j^T J_P \widetilde{v}_{m+1} &= 0 \\ w_j^T J_P \widetilde{v}_{m+1} &= 0 \end{aligned}$$

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 $\widetilde{w}_m$ 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). From (12) 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 $\widetilde{H}^{2m-2}$ 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. In this case we propose to use an implicit restart technique to overcome the breakdown by changing the starting vector. 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 [12, 46] the decomposition $T_k - \mu I = QR$ and the corresponding QR step, $T_k = Q^T T_k Q$, 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 implictly restarted Lanczos method for nonsymmetric matrices [25], the HR decomposition and a corresponding HR step [6] is used, as this transformation preserves sign-symmetry along with the tridiagonality of the T_k and the bi-orthogonality of the basis vectors.

Although symmetry is lacking in the symplectic Lanczos process defined above, the resulting matrix $\widetilde{H}_P^{2k}$ is a permuted Hamiltonian J-Hessenberg matrix as in (10). In order to preserve this structure and the J-orthogonality of the basis vectors it turns out to be useful to employ an SR decompositon of $\widetilde{H}_P^{2k} - \mu I, \mu \in \mathbb{R}$. Besides this single shift we study double shifts $\left(\widetilde{H}_P^{2k} - \mu I\right)\left(\widetilde{H}_P^{2k} + \mu I\right)$ 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.

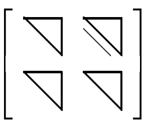
The SR decomposition has been studied in e.g. [8, 14]. A slightly modified version of the notation of [8] will be employed here.

DEFINTION 4.1.

a) A matrix

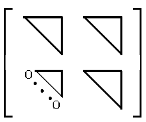
$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

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.,*

$H =$  .

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.,*

$H =$  .

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.,*

$$H = \begin{bmatrix} \diagdown & \diagdown\!\!\diagdown\!\!\diagdown \\ \diagdown & \diagdown \end{bmatrix}.$$

REMARK 4.1. *A Hamiltonian J-Hessenberg matrix $\widetilde{H} \in \mathbb{R}^{2n\times 2n}$ 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 $X = \widetilde{S}\widetilde{R}$ be SR factorizations of X. Then there exists a matrix*

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

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

c) *Let $X = \widetilde{H}$ be an unreduced Hamiltonian J-Hessenberg matrix. If $\widetilde{H} - \mu I = SR$, $\mu \in \mathbb{R}$, with S and R satisfying a) exists, then $\widehat{H} = S^{-1}\widetilde{H}S = RS + \mu I$ is a Hamiltonian J-Hessenberg matrix.*

d) *If μ in c) is an eigenvalue of $\widetilde{H}$, then $\widehat{h}_{2k,2k} = \mu$, $\widehat{h}_{k,k} = -\mu$ and $\widehat{h}_{2k,k} = 0$.*

e) *Let $X = \widetilde{H}$ be an unreduced Hamiltonian J-Hessenberg matrix. If the decomposition $(\widetilde{H} - \mu I)(\widetilde{H} + \mu I) = SR$, $\mu \in \mathbb{R}$ or $\mu \in \imath\mathbb{R}$, with S and R satisfying a) exists, then $\widehat{H} = S^{-1}\widetilde{H}S$ is a Hamiltonian J-Hessenberg matrix.*

f) *If μ in e) is an eigenvalue of $\widetilde{H}$, then $\widehat{h}_{k,2k-1} = \widehat{h}_{k-1,2k} = 0$ and the 2×2 submatrix $\begin{bmatrix} \widehat{h}_{kk} & \widehat{h}_{k,2k} \\ \widehat{h}_{2k,k} & \widehat{h}_{2k,2k} \end{bmatrix}$ has the eigenvalues μ and $-\mu$.*

Proof:

For the original statement and proof of a) see Theorem 11 in [17].

For the original statement and proof of b) see Proposition 3.3 in [8].

For the original statement and proof of c) and e) see Remark 4.1 in [8].

The proof of d) and f) follows the lines of [25, Theorem 2 (iii)].

For d) assume that a symplectic matrix S and a J-triangular matrix R exist such that $\widetilde{H} - \mu I = SR$. In order to simplify the notation we use in the following (as before) the permuted versions of $\widetilde{H}$, S and R :

$$\widetilde{H}_P = P\widetilde{H}P^T, \qquad S_P = PSP^T, \qquad R_P = PRP^T.$$

Since $\widetilde{H}$ is an unreduced Hamiltonian J-Hessenberg matrix, $\widetilde{H}_P$ is an unreduced upper Hessenberg matrix. Since R is a J-triangular matrix, R_P is an upper triangular matrix. With $I^{2k,2k-2}$ we denote the first $2k-2$ columns of the $2k \times 2k$ identity matrix.

Now partition the permuted matrices as follows

$$\begin{array}{rcl}
\widetilde{H}_P & = & [\widetilde{H}_P^{2k,2k-2} \quad | \quad \tilde{h}_P^{2k-1} \quad \tilde{h}_P^{2k}] \\
S_P & = & [S_P^{2k,2k-2} \quad | \quad s_P^{2k-1} \quad s_P^{2k}] \\
R_P & = & \left[\begin{array}{c|cc} R_P^{2k-2,2k-2} & r_P^{2k-1} & r_P^{2k-2} \\ \hline 0 & (r_P)_{2k-1,2k-1} & (r_P)_{2k-1,2k} \\ 0 & 0 & (r_P)_{2k,2k} \end{array}\right]
\end{array}$$

where the matrix blocks and entries are defined as before. The columns of $\widetilde{H}_P^{2k,2k-2} - \mu I^{2k,2k-2}$ are linear independent since $\widetilde{H}_P$ is unreduced. The columns of $S_P^{2k,2k-2}$ are linear independent since S_P is nonsingular. Hence the matrix $R_P^{2k-2,2k-2}$ is nonsingular since

$$\widetilde{H}_P^{2k,2k-2} - \mu I^{2k,2k-2} = S_P^{2k,2k-2} R_P^{2k-2,2k-2}.$$

It follows that

$$S_P^{2k,2k-2} = (\widetilde{H}_P^{2k,2k-2} - \mu I^{2k,2k-2})(R_P^{2k-2,2k-2})^{-1}$$

is an upper Hessenberg matrix and that $\widetilde{H}_P - \mu I$ is singular if and only if $(r_P)_{2k,2k} = 0$.

(c) follows directly from the above. Since $S_P^{2k,2k-2}$ is upper Hessenberg, S_P is upper Hessenberg and S is a symplectic J-Hessenberg matrix and thus $R_P S_P$ is upper Hessenberg and RS is a J-Hessenberg matrix. Hence $\widehat{H} = RS - \mu I = S^{-1}\widetilde{H}S$ is a Hamiltonian J-Hessenberg matrix.)

If $\widetilde{H}_P - \mu I$ is singular, then $(r_P)_{2k,2k} = 0$ and thus the $(2k-1, 2k)$ element of $\widehat{H}_P$ is zero. Let the parameters of $\widehat{H} = RS + \mu I$ be denoted by $\widehat{\delta}_n, \widehat{\beta}_n, \widehat{\gamma}_n, \widehat{\zeta}_n, \widehat{\nu}_n$. Then we have $\widehat{\delta}_k = -\mu$ and $\widehat{\nu}_k = 0$, i.e.,

$$\widehat{H} = \left[\begin{array}{cccc|cccc}
\widehat{\delta}_1 & & & & \widehat{\beta}_1 & \widehat{\zeta}_1 & & \\
& \ddots & & & \widehat{\zeta}_1 & \ddots & \ddots & \\
& & \widehat{\delta}_{k-1} & & & \ddots & \ddots & \widehat{\zeta}_k \\
& & & -\mu & & & \widehat{\zeta}_k & \widehat{\beta}_k \\ \hline
\widehat{\nu}_1 & & & & -\widehat{\delta}_1 & & & \\
& \ddots & & & & \ddots & & \\
& & \widehat{\nu}_{k-1} & & & & -\widehat{\delta}_{k-1} & \\
& & & 0 & & & & \mu
\end{array}\right].$$

$f)$ follows analogous to $d)$. Assume that a symplectic matrix S and a J-triangular matrix R exist such that $(\widetilde{H} - \mu I)(\widetilde{H} + \mu I) = SR$. As before we will use the permuted versions of $\widetilde{H}$, S and R. Since $\widetilde{H}$ is an unreduced Hamiltonian J-Hessenberg matrix, $\widetilde{H}_P$ is an unreduced upper Hessenberg matrix. Thus $\widetilde{H}_P^2$ is, as the product of two unreduced upper Hessenberg matrices, no longer upper Hessenberg, but has an additional second lower subdiagonal with nonzero entries. Since R is a J-triangular matrix, R_P is an upper triangular matrix.

Now partition the permuted matrices as follows :

$$\begin{aligned} \widetilde{H}_P^2 &= [(\widetilde{H}_P^2)^{2k,2k-2} \quad | \quad (\tilde{h}_P^2)^{2k-1} \quad (\tilde{h}_P^2)^{2k}], \\ S_P &= [S_P^{2k,2k-2} \quad | \quad s_P^{2k-1} \quad s_P^{2k}], \\ R_P &= \left[\begin{array}{c|cc} R_P^{2k-2,2k-2} & r_P^{2k-1} & r_P^{2k-2} \\ \hline 0 & (r_P)_{2k-1,2k-1} & (r_P)_{2k-1,2k} \\ 0 & 0 & (r_P)_{2k,2k} \end{array}\right], \end{aligned}$$

where $(\widetilde{H}_P^2)^{2k,2k-2}$ and $S_P^{2k,2k-2}$ are the first $2k-2$ columns of $\widetilde{H}_P^2$ and S_P, respectively, and $R_P^{2k-2,2k-2}$ is the leading $(2k-2)\times(2k-2)$ principal submatrix of R_P. The columns of $(\widetilde{H}_P^2)^{2k,2k-2} - \mu^2 I^{2k,2k-2}$ are linear independent since $\widetilde{H}_P$ is unreduced. The columns of $S_P^{2k,2k-2}$ are linear independent since S_P is nonsingular. Hence the matrix $R_P^{2k-2,2k-2}$ is nonsingular since

$$(\widetilde{H}_P^2)^{2k,2k-2} - \mu^2 I^{2k,2k-2} = S_P^{2k,2k-2} R_P^{2k-2,2k-2}.$$

It follows that

$$S_P^{2k,2k-2} = \left((\widetilde{H}_P^2)^{2k,2k-2} - \mu^2 I^{2k,2k-2}\right)(R_P^{2k-2,2k-2})^{-1}$$

is upper Hessenberg with an additional nonzero second lower subdiagonal and that $\widetilde{H}_P^2 - \mu^2 I$ is singular if and only if the trailing 2×2 principal submatrix of R_P is zero.

Observe that $\widehat{H}^2 = (S^{-1}\widetilde{H}S)^2 = S^{-1}\widetilde{H}^2 S = RS + \mu^2 I$. If μ is an eigenvalue of $\widetilde{H}$, then the kth and $2k$th row and column of R are zero. Statement $f)$ follows from a comparison of the coefficients in $\widehat{H}^2$ and $RS + \mu^2 I$, noting that $\nu_{k-1} \neq 0$ as the second lower subdiagonal of S_P is nonzero and $R_P^{2k-2,2k-2}$ is nonsingular. $\surd$

Assuming its existence, the SR decomposition and SR step (that is, $\widehat{H} = S^{-1}\widetilde{H}S$) possesses many of the desirable properties of the QR method. For the remainder of this section, 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 [8]. 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 [19, 20, 24, 31]. 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 only be sketched here.

As shown in [7], a necessary and sufficient condition for the existence of an orthogonal SR decomposition $M = SR$ (S symplectic and orthogonal) is that M is of the form

$$M = \begin{bmatrix} A & B \\ -B & A \end{bmatrix} \tilde{R}$$

where $A, B \in \mathbb{R}^{n\times n}$ and $\tilde{R}$ is an upper J-triangular matrix. Therefore in general we have to employ nonorthogonal symplectic elimination matrices, too.

For the computation of an SR decomposition we use three types of elementary symplectic matrices (for a detailed discussion see [8, 40]).

- For $k \in \{1, \ldots, n\}$ and $c, s \in \mathbb{R}$ with $c^2 + s^2 = 1$ define a *symplectic Givens (Jacobi) matrix (of type I)* by

$$J(k, c, s) = \begin{bmatrix} C & S \\ -S & C \end{bmatrix}$$

  where $C, S \in \mathbb{R}^{n\times n}$ are diagonal matrices $C = I_n + (c-1)e_k e_k^T$ and $S = s e_k e_k^T$.

- For $k \in \{2, \ldots, n\}$ and $y \in \mathbb{R}$ define a *symplectic Gauss(ian elimination) matrix* by

$$G(k, y) = \begin{bmatrix} D & Y \\ 0 & D^{-1} \end{bmatrix}$$

  where Y is the $n \times n$ matrix

$$Y = \left(\frac{y}{(1+y^2)^{\frac{1}{4}}} \right) (e_{k-1}e_k^T + e_k e_{k-1}^T)$$

  with only two nonzero entries in the positions $(k, k-1)$ and $(k-1, k)$ and D is the $n \times n$ diagonal matrix

$$D = I_n + \left(\frac{1}{(1+y^2)^{\frac{1}{4}}} - 1 \right) (e_{k-1}e_{k-1}^T + e_k e_k^T).$$

- For $k \in \{1, \ldots, n\}$ and $c, s \in \mathbb{R}$ with $c^2 + s^2 = 1$ define a *symplectic Givens matrix of type II* by

$$R(k, c, s) = \begin{bmatrix} U(k, c, s) & 0 \\ 0 & U(k, c, s) \end{bmatrix}$$

  where $U(k, c, s)$ is the $n \times n$ Givens matrix

$$U(k, c, s) = diag(I_{k-1}, \begin{bmatrix} c & s \\ -s & c \end{bmatrix}, I_{n-k-1}).$$

REMARK 4.2.

a) $J(k,c,s)$ *is orthogonal and symplectic.*

b) $J(k,c,s)$ *is a* $2n \times 2n$ *Givens rotation in planes* k *and* $n+k$*.*

c) $G(k,y)$ *is a nonorthogonal symplectic matrix.*

d) $G(k,y)^{-1} = \begin{bmatrix} D^{-1} & -Y \\ 0 & D \end{bmatrix}$.

e) $cond_2(G(k,y)) = (1+y^2)^{\frac{1}{2}} + |y|$, *where* $cond_2(A)$ *is the condition number of a matrix* A *with respect to the spectral norm, i.e.,* $cond_2(A) = ||A||_2||A^{-1}||_2$.

f) Among all possible symplectic nonorthogonal elimination matrices which serve the same purpose as $G(k,y)$*, the symplectic Gauss matrices are optimally conditioned [8].*

g) $R(k,c,s)$ *is orthogonal and symplectic.*

h) $R(k,c,s)$ *is the direct sum of two* $n \times n$ *Givens matrices.*

i) Replacing $U(k,c,s)$ *by a Householder matrix and defining analogously a block diagonal matrix* R *yields a* symplectic Householder matrix *(see [40]). This kind of orthogonal symplectic matrices will not be used here.*

We will use the following notation

- $J(k,c,s) = sgivens(k,a,b)$,
  where $sgivens$ generates a symplectic Givens rotation in planes k and $n+k$ such that
  $$J(k,c,s)(ae_k + be_{n+k}) = \alpha e_k, \qquad 1 \le k \le n.$$
- $G(k,y) = sgauss(k,a,b)$,
  where $sgauss$ generates a symplectic Gaussian elimination matrix such that
  $$G(k,y)(ae_k + be_{n+k-1}) = \beta e_{n+k-1}, \qquad 1 \le k \le n.$$
- $R(k,c,s) = sgivens2(k,a,b)$,
  where $sgivens2$ first generates an orthogonal Givens rotation $U(k,c,s)$ such that
  $$U(k,c,s)(ae_k + be_{k+1}) = \gamma e_k, \qquad 1 \le k \le n-1$$
  and then sets
  $$R(k,c,s) = \begin{bmatrix} U(k,c,s) & 0 \\ 0 & U(k,c,s) \end{bmatrix}.$$

For a presentation of algorithms for the computation of the symplectic reductions see e.g. [8, 40].

Now we illustrate the construction of implicit SR steps. First we will describe the single shift $\widetilde{H} - \mu I$, $\mu \in \mathbb{R}$, on a 6×6 example. As before we will consider the permuted case $\widetilde{H}_P - \mu I = S_P R_P$. Thus we have to use permuted versions of the elementary symplectic rotations

$$G_P(k,y) = PG(k,y)P^T = diag(I_{2k-4}, \left[\begin{array}{cc|cc} r & & t & \\ & \frac{1}{r} & & \\ \hline & t & r & \\ & & & \frac{1}{r} \end{array}\right], I_{2n-2k})$$

with $r = (1+y^2)^{-\frac{1}{4}}$, $t = yr$,

$$R_P(k,c,s) = PR(k,c,s)P^T = diag(I_{2k-1}, \left[\begin{array}{cc|cc} c & & s & \\ & c & & s \\ \hline -s & & c & \\ & -s & & c \end{array}\right], I_{2n-2k-3}),$$

and a $2n \times 2n$ Givens rotation in planes $2k-1$ and $2k$,

$$J_P(k,c,s) = PJ(k,c,s)P^T.$$

Because of the uniqueness of the reduction to J-Hessenberg form, the first rotation of the implicit SR step has to be selected so that the first columns of the implicit and explicit S_P are equivalent (as in the implicit QR step). Thus we have to transform $x = (\widetilde{H}_P - \mu I)e_1 = [\delta_1 - \mu, \nu_1, 0, \ldots, 0]^T$ to a multiple of e_1. That is, we have to annihilate the second element of x, i.e., ν_1, while preserving all existing zeros. This can be done by a transformation with a matrix of type $J_P(1,c,s)$. Computing the similarity transformation $\widetilde{H}_P^{(1)} = J_P(1,c_1,s_1)\widetilde{H}_P J_P^T(1,c_1,s_1)$ we obtain

$$\widetilde{H}_P^{(1)} = \left[\begin{array}{cc|cc|cc} x & x & 0 & x & 0 & 0 \\ x & x & 0 & \otimes & 0 & 0 \\ \hline \otimes & x & x & x & 0 & x \\ 0 & 0 & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

Here x denotes an arbitrary matrix element, $\otimes$ denotes an additional matrix element.

Now we have to perform a bulge-chasing sweep down the diagonal to restore the desired permuted J-tridiagonal form. This can be done by the algorithm JHESS given in [8] which reduces an (arbitrary) $2n \times 2n$ matrix to J-Hessenberg form. If the algorithm is applied to a Hamiltonian matrix, then the resulting condensed form will be a J-triangular form. Due to the special structure of $\widetilde{H}_P^{(1)}$ the algorithm greatly simplifies :

To preserve the zeros already present in the first column, we have to apply a matrix of type $G_P(2,y)$ to annihilate the $(3,1)$ entry. This can be done if the $(2,1)$ entry is nonzero, for a discussion of a breakdown or near-breakdown see Section 6.

Then

$$\widetilde{H}_P^{(2)} = G_P(2,y)\widetilde{H}_P^{(1)}G_P^{-1}(2,y) = \left[\begin{array}{cc|cc|cc} x & x & \otimes & x & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ \hline 0 & x & x & x & 0 & x \\ 0 & \otimes & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

The additional zero in position $(2,4)$ is achieved because $\widetilde{H}_P^{(2)}$ is a permuted Hamiltonian matrix. Now the entry in position $(4,2)$ is eliminated by applying a matrix of type $J(2,c,s)$ giving

$$\widetilde{H}_P^{(3)} = J_P(2,c,s)\widetilde{H}_P^{(2)}J_P^T(2,c,s) = \left[\begin{array}{cc|cc|cc} x & x & 0 & x & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ \hline 0 & x & x & x & 0 & x \\ 0 & 0 & x & x & 0 & \otimes \\ \hline 0 & 0 & \otimes & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

The additional zero in position $(1,3)$ is achieved again because $\widetilde{H}_P^{(3)}$ is a permuted Hamiltonian matrix. We have the same situation as after the construction of $\widetilde{H}_P^{(1)}$, but the bulge has moved 2 rows and columns further down. Therefore these additional elements can be chased down along the diagonal analogous to the last two steps. This gives rise to the sequence of similarity transformations to perform an implicit single-shifted SR step as given in Table 2.

Note that $sgauss_p$ and $sgivens_P$ are the permuted versions of $sgauss$ and $sgivens$. An efficient implementation of this algorithm requires $100k-65$ flops for the similarity transformations and $28kn - 16n$ flops for the update of S_P. All transformation matrices in the loop have as a first column a multiple of e_1 which reflects the fact that the SR decomposition is essentially determined by the first column of S and thus by $J_P(1,c,s)$.

Next we will illustrate the double shift case $(\widetilde{H}-\mu I)(\widetilde{H}+\mu I)$, $\mu \in \mathbb{R}$ or $\mu \in \imath\mathbb{R}$ on an 8×8 example. As before the first rotation of the implicit SR step has to be selected so that the first columns of the implicit and explicit S_P are equivalent. Thus we have to transform $x = (\widetilde{H}-\mu I)(\widetilde{H}+\mu I)e_1 = [\delta_1^2-\mu^2+\beta_1\nu_1, 0, \nu_1\zeta_2, 0, \ldots, 0]^T$ (with $\widetilde{H}$ as in (10)) to a multiple of e_1. Therefore we have to eliminate the third entry of x, i.e. $\nu_1\zeta_2$, while preserving all existing zeros. This can be done by a transformation of type $R_P(1,c,s)$. A similarity transformation of $\widetilde{H}_P$ with R_P yields

$$\widetilde{H}_P^{(1)} = R_P(1,c,s)\widetilde{H}_P R_P^T(1,c,s) = \left[\begin{array}{cc|cc|cc|cc} x & x & \otimes & x & 0 & \otimes & 0 & 0 \\ x & x & \otimes & \otimes & 0 & 0 & 0 & 0 \\ \hline \otimes & x & x & x & 0 & x & 0 & 0 \\ \otimes & \otimes & x & x & 0 & 0 & 0 & 0 \\ \hline 0 & \otimes & 0 & x & x & x & 0 & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

<u>Algorithm : Implicit SR step with single shift</u>

Given a permuted Hamiltonian J-Hessenberg matrix $\widetilde{H}_P \in \mathbb{R}^{2k\times 2k}$
and $S_P \in \mathbb{R}^{2n\times 2k}$ with $S_P^T J_P^n S_P = J_P^k$.
Choose a shift $\mu \in \mathbb{R}$.
Set $\widetilde{H}_P^{(0)} = \widetilde{H}_P$.
(Compute first column of S)
Set $J_P(1,c,s) = sgivens_P(1, \delta_1 - \mu, \nu_1)$
$\widetilde{H}_P^{(1)} = J_P(1,c,s)\widetilde{H}_P^{(0)} J_P^T(1,c,s)$.
$S_P = S_P J_P^T(1,c,s)$.
(Chase the bulge)
for $i = 3,5,\ldots,2k-1$
Set $G_P(\frac{i+1}{2}, y) = sgauss_P(\frac{i+1}{2}, (\tilde{h}_P^{(i-2)})_{i-1,i-2}, (\tilde{h}_P^{(i-2)})_{i,i-2})$
$\widetilde{H}_P^{(i-1)} = G_P(\frac{i+1}{2}, y)\widetilde{H}_P^{(i-2)} G_P^{-1}(\frac{i+1}{2}, y)$
$S_P = S_P G_P^{-1}(\frac{i+1}{2}, y)$.
$J_P(\frac{i+1}{2}, c, s) = sgivens_P(\frac{i+1}{2}, (\tilde{h}_P^{(i-1)})_{i,i-1}, (\tilde{h}_P^{(i-1)})_{i+1,i-1})$
$\widetilde{H}_P^{(i)} = J_P(\frac{i+1}{2}, c, s)\widetilde{H}_P^{(i-1)} J_P^T(\frac{i+1}{2}, c, s)$
$S_P = S_P J_P^T(\frac{i+1}{2}, c, s)$.

Table 2: Implicit SR Step — Single Shift Case

Now we have to perform a bulge-chasing sweep down the diagonal to restore the desired permuted J-tridiagonal form. This can again be done by the algorithm JHESS given in [8] which reduces a $2n \times 2n$ Hamiltonian matrix to J-triangular form. Due to the special structure of $\widetilde{H}_P^{(1)}$ we can use a simplified version of this algorithm :

As before, in each step we will obtain additional zeros because the iterates $\widetilde{H}_P^{(i)}$ are permuted Hamiltonian matrices. Using a matrix of type $J_P(3,c,s)$ we eliminate the element in position $(4,1)$ and obtain

$$\widetilde{H}_P^{(2)} = J_P(3,c,s)\widetilde{H}_P^{(1)} J_P^T(3,c,s) = \left[\begin{array}{cc|cc|cc|cc} x & x & \otimes & x & 0 & \otimes & 0 & 0 \\ x & x & 0 & \otimes & 0 & 0 & 0 & 0 \\ \hline \otimes & x & x & x & 0 & x & 0 & 0 \\ 0 & \otimes & x & x & 0 & \otimes & 0 & 0 \\ \hline 0 & \otimes & \otimes & x & x & x & 0 & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

To preserve the zeros already present in the first column, we have to apply a matrix of type $G_P(2,y)$ to annihilate the $(3,1)$ entry. This can be done if the $(2,1)$ entry is nonzero. Then

$$\widetilde{H}_P^{(3)} = G_P(2,y)\widetilde{H}_P^{(2)}G_P^{-1}(2,y) = \left[\begin{array}{cc|cc|cc|cc} x & x & \otimes & x & 0 & \otimes & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & x & x & x & 0 & x & 0 & 0 \\ 0 & \otimes & x & x & 0 & \otimes & 0 & 0 \\ \hline 0 & \otimes & \otimes & x & x & x & 0 & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

Now the entry in position $(4,2)$ is eliminated by applying a matrix of type $J(2,c,s)$ giving

$$\widetilde{H}_P^{(4)} = J_P(2,c,s)\widetilde{H}_P^{(3)}J_P^T(2,c,s) = \left[\begin{array}{cc|cc|cc|cc} x & x & 0 & x & 0 & \otimes & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & x & x & x & 0 & x & 0 & 0 \\ 0 & 0 & x & x & 0 & \otimes & 0 & 0 \\ \hline 0 & \otimes & \otimes & x & x & x & 0 & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

Eliminating the entry in position $(5,2)$ with a matrix of type $R_P(2,c,s)$ yields

$$\widetilde{H}_P^{(5)} = R_P(2,c,s)\widetilde{H}_P^{(4)}R_P^T(2,c,s) = \left[\begin{array}{cc|cc|cc|cc} x & x & 0 & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & x & x & x & \otimes & x & 0 & \otimes \\ 0 & 0 & x & x & \otimes & \otimes & 0 & 0 \\ \hline 0 & 0 & \otimes & x & x & x & 0 & x \\ 0 & 0 & \otimes & \otimes & x & x & 0 & 0 \\ \hline 0 & 0 & 0 & \otimes & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

We have the same situation as after the construction of $\widetilde{H}_P^{(1)}$, but the bulge has moved 2 rows and columns further down. Therefore these additional elements can be chased down along the diagonal analogous to the last four steps. This gives rise to the sequence of similarity transformations to perform an implicit double-shifted SR step as given in Table 3.

Note that $sgauss_p$, $sgivens_P$, and $sgivens2_P$ are the permuted versions of $sgauss$, $sgivens$, and $sgivens2$, respectively. An efficient implementation of this algorithm requires $247k - 167$ flops for the similarity transformations and $54nk - 30n$ flops for the update of S_P. As before, the first column of the S factor of the SR decomposition is determined by the first column of $R_P(1,c,s)$ which is reflected by the fact that the first column of all transformation matrices in the loop is a multiple of e_1.

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, $3k-2$

Algorithm : Implicit SR step with double shift

Given a permuted Hamiltonian J-Hessenberg matrix $\widetilde{H}_P \in \mathbb{R}^{2k\times 2k}$
and $S_P \in \mathbb{R}^{2n\times 2k}$ with $S_P^T J_P^n S_P = J_P^k$.
Choose a shift $\mu \in \mathbb{R}$.

(Compute first column of S)

Set $R_P(1,c,s) = sgivens2_P(1, \delta_1^2 + \mu^2 + \beta_1\nu_1, \nu_1\zeta_2)$

$\widetilde{H}_P = R_P(1,c,s)\widetilde{H}_P R_P^T(1,c,s)$.

$S_P = S_P R_P^T(1,c,s)$.

(Chase the bulge)

for $i = 3,5,\ldots,2k-1$

Set $J_P(\frac{i+1}{2}+1,c,s) = sgivens_P(i, (\tilde{h}_P)_{i,i-2}, (\tilde{h}_P)_{i+1,i-2})$

$\widetilde{H}_P = J_P(\frac{i+1}{2}+1,c,s)\widetilde{H}_P J_P^T(\frac{i+1}{2}+1,c,s)$

$S_P = S_P J_P^T(\frac{i+1}{2}+1,c,s)$.

$G_P(\frac{i+1}{2},y) = sgauss_P(\frac{i+1}{2}, (\tilde{h}_P)_{i-1,i-2}, (\tilde{h}_P)_{i,i-2})$

$\widetilde{H}_P = G_P(\frac{i+1}{2},y)\widetilde{H}_P G_P^{-1}(\frac{i+1}{2},y)$

$S_P = S_P G_P^{-1}(\frac{i+1}{2},y)$.

$J_P(\frac{i+1}{2},c,s) = sgivens_P(\frac{i+1}{2}, (\tilde{h}_P)_{i,i-1}, (\tilde{h}_P)_{i+1,i-1})$

$\widetilde{H}_P = J_P(\frac{i+1}{2},c,s)\widetilde{H}_P J_P^T(\frac{i+1}{2},c,s)$

$S_P = S_P J_P^T(\frac{i+1}{2},c,s)$.

$R_P(\frac{i+1}{2},c,s) = sgivens2_P(\frac{i+1}{2}, (\tilde{h}_P)_{i+2,i-1}, (\tilde{h}_P)_{i,i-1})$

$\widetilde{H}_P = R_P(\frac{i+1}{2},c,s)\widetilde{H}_P R_P^T(\frac{i+1}{2},c,s)$

$S_P = S_P R_P^T(\frac{i+1}{2},c,s)$.

Table 3: Implicit SR Step — Double Shift Case

of these transformations are orthogonal (k in the single shift case). These are known to be numerically stable. Thus, in both algorithms $(k-1)$ transformation of type G_P have to be used. Problems can arise here because of breakdown or near breakdown. If we eliminate the jth nonzero entry of a vector x with $G_P(j,y)$ and x_{j-1} is very small relative to x_j, then $y = -x_j/x_{j-1}$, and therefore the condition number $||G_P(j,y)||_2 = (1+y^2)^{\frac{1}{2}} + |y|$ will be very large. A transformation with $G_P(j,y)$ will then 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

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

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

$$H_P \breve{S}_P^{2k} = \breve{S}_P^{2k} \breve{H}_P^{2k} + \breve{\zeta}_{k+1} \breve{v}_{k+1} e_{2k}^T \tag{15}$$

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 [25, 46].

For any permuted symplectic $2k \times 2k$ matrix S_P, (14) can be reexpressed as

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

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

$$H_P \breve{S}_P^{2k} = \breve{S}_P^{2k} \breve{H}_P^{2k} + \zeta_{k+1} v_{k+1} e_{2k}^T S_P. \tag{16}$$

Let s_{ij} be the (i,j)th entry of S_P. If we choose S_P from the permuted SR decomposition $\widetilde{H}_P^{2k} - \mu I = S_P R_P$, then from the proof of Theorem 4.1 we know that S_P is an upper Hessenberg matrix. Thus the residual term in (16) is

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

In order to obtain a residual term of the desired form vector times e_{2k}^T we have to truncate off a portion of (16). Rewriting (16) as

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

we obtain as a new Lanczos identity

$$H_P \breve{S}_P^{2k-2} = \breve{S}_P^{2k-2} \breve{H}_P^{2k-2} + \breve{\zeta}_k \breve{v}_k e_{2k-2}^T. \tag{17}$$

Here, $\breve{\zeta}_k$, $\breve{\delta}_k$, $\breve{\beta}_k$, $\breve{\nu}_k$ denote parameters of $\breve{H}_P^{2k}$, ζ_{k+1} a parameter of $\widetilde{H}_P^{2k}$. 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 (17) 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 (15) from (14).

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. This is no longer true for the odd numbered column vectors of S_P generated by the SR decomposition and the parameters $\breve{\delta}_j$ from $\breve{H}_P^{2k}$ and thus for the new Lanczos factorization (17).

In our applications we have to compute a truncated reduction $\widetilde{H}_P^{2j}$ of H_P with $j \ll n$. In case the symplectic Lanczos method breaks down before $\widetilde{H}^{2j}$ can be computed, we propose to employ a single shifted implicit restart as described above to overcome the breakdown.

In connection with the computation of low rank approximations to the solution of continuous-time algebraic Riccati equations we will use a double shifted restarted Lanczos method to remove a pair of purely imaginary eigenvalues from $\widetilde{H}_P^{2k}$. Therefore here we will give the derivation of the corresponding formulas. Using the decomposition $(\widetilde{H}_P^{2k} - \imath\mu I)(\widetilde{H}_P^{2k} + \imath\mu I) = S_P R_P$, we obtain as before from (14)

$$H_P \breve{S}_P^{2k} = \breve{S}_P^{2k} \breve{H}_P^{2k} + \zeta_{k+1} v_{k+1} e_{2k}^T S_P \tag{18}$$

with $\breve{S}_P^{2k} = S_P^{2k} S_P$, $\breve{H}_P^{2k} = S_P^{-1} \widetilde{H}_P^{2k} S_P$. Just the matrix S_P is now different from above. As it is the S-factor of the permuted SR decomposition of $(\widetilde{H}_P^{2k} - \imath\mu I)(\widetilde{H}_P^{2k} + \imath\mu I)$, S_P is no longer an upper Hessenberg matrix, but has an additional lower subdiagonal. Denoting the (i,j)th entry of S_P by s_{ij}, the residual term in (18) is

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

In order to obtain a residual term of the desired form vector times e_{2k}^T we have to truncate off a portion of (18). Rewriting (18) as

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

we obtain as a new Lanczos identity

$$H_P \breve{S}_P^{2k-2} = \breve{S}_P^{2k-2} \breve{H}_P^{2k-2} + r_k e_{2k-2}^T. \tag{19}$$

The new residual vector is given by

$$r_k = \breve{\zeta}_k \breve{v}_k + \zeta_{k+1} s_{2k,2k-2} v_{k+1}.$$

As before we can argue that (19) is a valid Lanczos factorization for the new starting vector $\breve{v}_1 = \rho(H_P - \imath\mu I)(H_P + \imath\mu I)v_1$. Only one additional step of the symplectic Lanczos algorithm is required to obtain (15) from (14).

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

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, 47]. Thus, Freund and Mehrmann propose in [22] to modify the prerequisite $S_P^T J_P S_P = J_P$ of our symplectic Lanczos method to

$$S_P^T J_P S_P = \begin{bmatrix} 0 & \sigma_1 & & & & & \\ -\sigma_1 & 0 & & & & & \\ & & 0 & \sigma_2 & & & \\ & & -\sigma_2 & 0 & & & \\ & & & & \ddots & & \\ & & & & & 0 & \sigma_n \\ & & & & & -\sigma_n & 0 \end{bmatrix} =: \Sigma$$

and

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

For the resulting algorithm and a discussion of it we refer to [22]. It is easy to see that $\widetilde{H}_P = S_P^{-1} H_P S_P$ is no longer a permuted Hamiltonian J-Hessenberg matrix, as S is only almost symplectic, but

$$\Sigma \widetilde{H}_P = (\Sigma \widetilde{H}_P)^T.$$

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

Unfortunately an SR step does not preserve the structure of $\widetilde{H}$ and thus this modified version of the symplectic Lanczos method can not 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

$$\begin{aligned} w_m &= w_m + S_P^{2m-2} J_P^{m-1} S_P^{2m-2^T} J_P^n w_m, \\ v_{m+1} &= v_{m+1} + S_P^{2m} J_P^m S_P^{2m^T} J_P^n v_{m+1}. \end{aligned}$$

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 $v_1, w_1, \dots, v_k, w_k$ 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 [24]). 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. As pointed out before, during the SR step on the $2k \times 2k$ 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 number.

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
and	$38n + 4nz$	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 up to $4nz \cdot k + 16n \cdot k^2 + 48n \cdot k - 26n$ flops. If an explicit restart with the starting vector $\breve{v}_1 = (H_P - \mu I)v_1$ would be performed, this 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 (4nz + $28n \cdot k$ + $54n$ + $(100k - 65)$ flops vs. $4nz \cdot k + 32n \cdot k + 6n$ flops). Besides we have to re-J-orthogonalize only once while an explicit restart would require a re-J-orthogonalization in each iteration step. For different re-J-orthogonalization techniques implicit restarts are also advantageous. For double shifted or multishifted restarts the implicit technique is still favourable 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 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 :

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

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 $A = \operatorname{diag}(-10^5, 9, 8, 7, 6, 5, 4, 3, \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix})$ 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 4 steps of the symplectic Lanczos method the resulting 8×8 Hamiltonian J-Hessenberg matrix $\widetilde{H}^8$ has the eigenvalues

$$\lambda(\widetilde{H}^8) = \left\{ \begin{array}{r} 9.999999999999997e+05 \\ -9.999999999999997e+05 \\ 3.040728370123861e+00 \\ -3.040728370123995e+00 \\ 9.200627380564711e+00 \\ -9.200627380564642e+00 \\ 9.477682371618508e+00 \\ -9.477682371618551e+00 \end{array} \right\}.$$

To remove an eigenvalue pair from $\widetilde{H}^8$ one can perform an implicit double shift restart as described in Section 5. Removing the two eigenvalues of smallest absolute value from $\widetilde{H}^8$, we obtain a Hamiltonian J-Hessenberg matrix $\breve{H}^6_{impl}$ whose eigenvalues are

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

From Theorem 4.1 f) it follows that these have to be the 6 eigenvalues of $\widetilde{H}^8$ which have not been removed. As can be seen, we loose $4-5$ digits during the implicit restart. Performing an explicit restart with the explicitly computed new starting

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

vector $\breve{v}_1 = (H - \mu I)(H + \mu I)v_1$ yields a Hamiltonian J-Hessenberg matrix $\breve{H}^6_{expl}$ with eigenvalues

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

This time we lost up to 10 digits. As a general observation from a wide range of numerical tests, the explicit restart looses 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 $\tilde{v}_1$ 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 $\tilde{v}_1$. Thus, in this situation the SR decomposition of $(H - \mu I)$ or $(H - \mu I)(H + \mu I)$ can not 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 transformations of the type G_P and J_P are used. As the latter ones are orthogonal symplectic Givens rotations, their computation can not break down. Thus the only source of breakdown can be one of the symplectic Gaussian eliminations G_P.

THEOREM 6.1. *Suppose the Hamiltonian J-Hessenberg matrix $\widetilde{H}^{2k}$ 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 $(\widetilde{H}^{2k}_P - \mu I)$. 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 $\breve{v}_j^T J_P H_P \breve{v}_j = 0$ with $\breve{v}_j$ as in (17).*

Proof:

The proof follows the lines of [25, Theorem 3].

Assume that the first $j-1$ permuted symplectic Gauss transformations $G_P(\frac{i+1}{2}, y_i), i = 3, 5, \ldots, 2j-1$ exist and let

$$\begin{bmatrix} \widehat{S}^{2j}_P & 0 \\ 0 & I \end{bmatrix} = J_P(1, c_1, s_1) \prod_{i=2}^{j} G_P(i, y_i) J_P(i, c_i, s_i).$$

Then from (11),

$$H_P S^{2j}_P = S^{2j}_P \widetilde{H}^{2j}_P + \zeta_{j+1} v_{j+1} e^T_{2j},$$

we obtain

$$H_P \breve{S}^{2j}_P = \breve{S}^{2j}_P \breve{H}^{2j}_P + \zeta_{j+1} v_{j+1} e^T_{2j} \widehat{S}^{2j}_P$$

where $\breve{S}^{2j}_P = S^{2j}_P \widehat{S}^{2j}_P$ and $\breve{H}^{2j}_P = (\widehat{S}^{2j}_P)^{-1} \widetilde{H}^{2j}_P \widehat{S}^{2j}_P$.

Since

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

it follows that

$$-J_P^j (\breve{S}_P^{2j})^T J_P^n H_P \breve{S}_P^{2j} = \breve{H}_P^{2j}. \tag{20}$$

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

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

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

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

using (20). That is

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

Thus we have

$$\begin{aligned} x_1 &= -w_{j+1}^T J_P^n H_P \breve{v}_j \\ x_2 &= -w_{j+1}^T J_P^n H_P \breve{w}_j. \end{aligned}$$

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

The opposite implication follows from the uniqueness of the symplectic Lanczos method. $\surd$

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

Lanzcos-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. [24]).

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 for computing a low rank approximation of the corresponding algebraic Riccati equation (see [45]). Besides, purely imaginary Ritz values of odd multiplicity signal that a stable k-dimensional invariant subspace of the computed $\widetilde{H}^{2k}$ does not exist. This will be considered in Section 7.2.

Computing the Ritz values after the k–th 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 $\widetilde{H}^{2k}$, we obtain a matrix of the following structure :

$$\left(\widetilde{H}^{2k}\right)^2 = M^{2k} = \begin{bmatrix} M_1^k & M_2^k \\ 0 & M_1^{kT} \end{bmatrix} = \qquad (21)$$

where

$$\begin{aligned}
M_1^k &= \begin{bmatrix} \mu_1 & \psi_2 & & & \\ \rho_2 & \mu_2 & \psi_3 & & \\ & \rho_3 & \ddots & \ddots & \\ & & \ddots & \ddots & \psi_k \\ & & & \rho_k & \mu_k \end{bmatrix}, \\
\mu_j &= \delta_j^2 + \beta_j \nu_j, \qquad j = 1, \ldots, k, \\
\rho_j &= \gamma_j \nu_{j-1}, \qquad j = 2, \ldots, k, \\
\psi_j &= \gamma_j \nu_j, \qquad j = 2, \ldots, k.
\end{aligned}$$

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

This approach is similar to Van Loan's square reduced algorithm [49]. 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

$$(U^THU)^2 = \begin{bmatrix} N_1 & N_2 \\ 0 & N_1^T \end{bmatrix} = \begin{bmatrix} \diagdown & \square \\ & \diagdown \end{bmatrix}. \tag{22}$$

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 (22) 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. Unfortunately, the tridiagonal matrix M_1 is nonsymmetric such that we either have to give up numerical stability or preservation of the tridiagonal structure when computing the eigenvalues.

Structure preserving methods for computing eigenvalues of unsymmetric tridiagonal matrices include the LR algorithm [44] and the recently proposed DQR algorithm [48]. All these methods require only $O(k^2)$ flops, but may suffer from numerical instabilities. For a discussion of these methods we refer to the references given above and the references therein.

For a detailed discussion of Van Loan's algorithm see [9, 49]. 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 loose 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 $\widetilde{H} = \widetilde{H}_0 \in \mathbb{R}^{2k\times 2k}$ as in (4), the SR algorithm computes a sequence of orthogonal and nonorthogonal symplectic similarity transformation matrices S_j, $j = 0, 1, \ldots$, that preserve this structure, i.e., $\widetilde{H}_{j+1} = S_j^{-1}\widetilde{H}_jS_j$ is a Hamiltonian J-Hessenberg matrix for all $j = 0, 1, \ldots$. The sequence $\widetilde{H}_j$ converges to a Hamiltonian matrix

$$\widetilde{H}^{SR} = (S^{SR})^{-1}\widetilde{H}S^{SR} = \begin{bmatrix} D_1 & D_2 \\ 0 & -D_1^T \end{bmatrix} \tag{23}$$

where D_1, D_2 are block diagonal $k\times k$ matrices with blocks of size 1×1 or 2×2 and all transformations S_j are accumulated in the symplectic matrix S^{SR}. The eigenvalues of $\widetilde{H}$ 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, 14, 35, 45].

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

$$HS^{2k} = S^{2k}\widetilde{H}^{2k} + \zeta_{k+1}\bar{v}_{k+1}e_{2k}^T. \tag{24}$$

We can use the SR algorithm to compute eigenvalues and eigenvectors of $\widetilde{H}^{2k}$. Setting $\widetilde{H} = \widetilde{H}^{2k}$ and $D_i = D_i^k$, $i = 1, 2$, in (23) and multiplying (24) from the right by S^{SR} yields

$$HS^{2k}S^{SR} = S^{2k}S^{SR}\begin{bmatrix} D_1^k & D_2^k \\ 0 & -D_1^{kT} \end{bmatrix} + \zeta_{k+1}\bar{v}_{k+1}e_{2k}^T S^{SR}. \tag{25}$$

Thus the Ritz values are the eigenvalues λ_j of D_1^k and their counterparts $-\lambda_j$. Now assume λ_j is a converged Ritz value, i.e., a sufficient approximation to an eigenvalue of H. As in standard Lanczos type algorithms, an approximation to the (right) eigenvector corresponding to λ_j can be read off from (25) if

$$\|Hy_j - \lambda_j y_j\| = \left\|\zeta_{k+1}\bar{v}_{k+1}e_{2k}^T s_j\right\| = |\zeta_{k+1}(s_j)_{2k}|\,\|\bar{v}_{k+1}\| \tag{26}$$

is sufficiently small (see e.g. [5]), here $y_j = S^{2k}S^{SR}e_j$ and $s_j = S^{SR}e_j$. 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 (25) 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 to the invariant subspaces of the corresponding Hamiltonian matrix. 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 like symmetry see e.g. [32, 36, 42].

In control theory one is usually concerned with the symmetric (positive semidefinite) *stabilizing* solution of (2), i.e., a solution $\widehat{X}$ such that $A - G\widehat{X}$ is stable. Under the conditions that (A, G) is stabilizable, (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 $\widetilde{H}^{2k}$. For a moment we will assume that $\widetilde{H}^{2k}$ has no purely imaginary eigenvalues. Hence we can compute an invariant subspace of $\widetilde{H}^{2k}$ by the SR algorithm as in (23). In [8] it is described how to separate the stable invariant subspace from (23) by symplectic similarity transformations which preserve the structure of (24). 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 $\widetilde{H}^{2k}$. Combined with the Lanczos factorization we again obtain (25). If

$$Y^k = S^{2k}S^{SR} = [\,Y_1^k \;\; Y_2^k\,], \qquad Y_1^k,\, Y_2^k \in \mathbb{R}^{2n\times k}, \tag{27}$$

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

$$\left\| HY_1^k - Y_1^k D_1^k \right\| = |\zeta_{k+1}| \left\| \bar{v}_{k+1} e_{2k}^T Y_1^k \right\| \tag{28}$$

is sufficiently small.

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. In the following we will describe one possibility to construct such a low rank approximation.

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

$$Y^{k^T} J^n Y^k = J^k \tag{29}$$

we obtain from (25)

$$J^{k^T} Y^{k^T} J^n H Y^k = \begin{bmatrix} D_1^k & D_2^k \\ 0 & -D_1^{k^T} \end{bmatrix} \tag{30}$$

and from the lower left block of this equation

$$-Y_{21}^{k^T} A Y_{11}^k + Y_{11}^{k^T} Q Y_{11}^k - Y_{21}^{k^T} G Y_{21}^k - Y_{11}^{k^T} A^T Y_{21}^k = 0 \tag{31}$$

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. If Y_{11}^k has full column rank, R^k is invertible. Premultiplying (31) by $R^{k^{-T}}$ and postmultiplying by $R^{k^{-1}}$ yields

$$-R^{k^{-T}} Y_{21}^{k^T} A Z^k + Z^{k^T} Q Z^k - R^{k^{-T}} Y_{21}^{k^T} G Y_{21}^k R^{k^{-1}} - Z^{k^T} A^T Y_{21}^k R^{k^{-1}} = 0. \tag{32}$$

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

$$Z^{k^T} \left(X^k A + Q - X^k G X^k + A^T X^k \right) Z^k = 0. \tag{33}$$

The computed X^k may now be considered as a low rank approximation to the solution of (2). From the symplecticity property (29) it is easy to verify that X^k is symmetric and from (30) we obtain

$$Z^{k^T}(A - GX^k)Z^k = R^k D_1^k R^{k^{-1}} + E_1^k \tag{34}$$

where E_1^k is the upper left $k \times k$ block of $Z^{k^T}(\zeta_{k+1}\bar{v}_{k+1}e_{2k}^T S^{SR})$. From (33) and (34) it is clear that in exact arithmetic for $k = n$, X^k is the required stabilizing solution of (2).

By now, we have assumed that $\widetilde{H}^{2k}$ has no eigenvalues on the imaginary axis. Under the above assumptions, H has no purely imaginary eigenvalues. But for $\widetilde{H}^{2k}$, $k < n$, computed by the Lanczos process, in general this property (and also the stabilizability–detectability condition) does not hold. Thus we may expect $\widetilde{H}^{2k}$ to have purely imaginary eigenvalues for some k. If this happens, $\widetilde{H}^{2k}$ does not have a stable, k-dimensional invariant subspace.

One way to remove these eigenvalues is to employ a double shifted restart as in (18). Suppose $\widetilde{H}^{2k}$ has ℓ pairs of purely imaginary eigenvalues denoted by $\imath\mu_1, -\imath\mu_1, \ldots, \imath\mu_\ell, -\imath\mu_\ell$. 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 the new Lanczos identity (19) which after permuting back reads

$$H\breve{S}^{2k-2} = \breve{S}^{2k-2}\breve{H}^{2k-2} + \bar{r}_k e_{2k-2}^T. \tag{35}$$

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

$$H\breve{S}^{2(k-\ell)} = \breve{S}^{2(k-\ell)}\breve{H}^{2(k-\ell)} + \bar{r}_{k-\ell+1}e_{2(k-\ell)}^T \tag{36}$$

where the eigenvalues of $\breve{H}^{2(k-\ell)}$ are those eigenvalues of $\widetilde{H}^{2k}$ having nonzero real parts. The starting vector corresponding to the Lanczos factorization (36) is the multishift vector

$$\breve{v}_1 = \rho(H - \imath\mu_\ell I)(H + \imath\mu_\ell I) \cdot \ldots \cdot (H - \imath\mu_1 I)(H + \imath\mu_1 I)v_1.$$

Thus it is possible to compute a low rank approximate stable H-invariant subspace of dimension $k - \ell$ and the corresponding Riccati solution. If an approximation of dimension k is required, we may use the same approach as in [25] where restarts are used to obtain a stable reduced order system. Performing ℓ symplectic Lanczos steps, we obtain from $\breve{H}^{2(k-\ell)}$ 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 our numerical experiments, this never produced an $\breve{H}^{2k}$ having again ℓ (or even more) pairs of purely imaginary Ritz values. With this approach we obtain after a finite number of restarts a Hamiltonian J-Hessenberg matrix of required dimension having only eigenvalues with nonzero real part.

8 Numerical Results

In this section we present some examples to demonstrate the ability of the proposed algorithm to overcome (near) breakdown and one example to show the typical behaviour of the symplectic Lanczos method. An example where the restart process is used to remove eigenvalues was already given in Section 6.2.

All computations were done using MATLAB[3] 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 value), 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 3 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 [3]. Restarts were only encountered in 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 [3, Example 7].) The first example shows that a serious breakdown can not 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:

$$\begin{aligned} \zeta_1 &= 1, \quad \nu_1 = 1, \quad w_1 = e_2 + e_4 \\ \beta_1 &= \epsilon, \quad \zeta_2 = 3, \quad v_2 = e_4. \end{aligned}$$

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

$$\widetilde{w}_2 = e_4, \quad \nu_2 = 0.$$

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

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

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

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

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

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 can not be overcome by implicit single shifted restarts. An explicit restart with a random starting vector is successful.

Example 2 : (See [13] and [3, 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:

$$\begin{aligned} \zeta_1 = 1, \quad \nu_1 = 1, \quad w_1 = e_2 - e_1 \\ \beta_1 = -1, \quad \zeta_2 = 0.4, \quad v_2 = -e_4. \end{aligned}$$

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 : We did a vast number of test runs using randomly chosen Hamiltonian matrices and randomly chosen starting vectors (as well as the starting vector e_1). The occurence of a serious breakdown is very unlikely here as these test examples typically have nice properties. Table 4 reports the distribution of the values of ν_i for 2000 randomly chosen 100×100 Hamiltonian matrices and randomly chosen starting vectors as the symplectic Lanczos method was used to compute 20 Lanczos vectors, that is the algorithm ran for 10 steps.

interval for ν_i	number of occurences
$\|\nu_i\| < 10^{-6}$	0
$10^{-6} \leq \|\nu_i\| < 10^{-5}$	2
$10^{-5} \leq \|\nu_i\| < 10^{-4}$	9
$10^{-4} \leq \|\nu_i\| < 10^{-3}$	113
$10^{-3} \leq \|\nu_i\| < 10^{-2}$	1010
$10^{-2} \leq \|\nu_i\| < 10^{-1}$	7717
$10^{-1} \leq \|\nu_i\| < 10^{0}$	10123
$10^{0} \leq \|\nu_i\| < 10^{1}$	26
$10^{1} \leq \|\nu_i\| < 10^{2}$	1000
$10^{2} \leq \|\nu_i\|$	0

Table 4: Distribution of ν_i

The occurence of a near breakdown is dependent on the value chosen for tol. Choosing tol too small (like $tol = \sqrt{\varepsilon}$ where ε is the floating point relative accuracy) results in almost no breakdown, choosing tol too large in too many. A good choice is dependent on the desired goals : the desired accuracy, the desired speed, etc. A breakdown during the implicit SR step was never encountered during these test runs.

As expected from a Lanczos method, the Ritz values converge to the eigenvalues of largest modulus after a small number of steps.

Example 4 : 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 [38])

$$\left[\begin{array}{cc} A & B \\ B & A \end{array}\right] x = \lambda \left[\begin{array}{cc} \Sigma & \Delta \\ -\Delta & -\Sigma \end{array}\right] x. \tag{37}$$

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 (37) reduces to the standard Hamiltonian eigenvalue problem

$$\left[\begin{array}{cc} A & B \\ -B & -A \end{array}\right] x = \lambda x.$$

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 [18] where special versions of the Lanczos algorithm for matrices as given in (37) are examined.

Let $n = 100$, $D = \operatorname{diag}(d_1, \ldots, d_n)$ and $\hat{D} = \operatorname{diag}(\hat{d}_1, \ldots, \hat{d}_n)$, where $d_1 = 200.0$, $d_2 = 100.0$, $d_3 = 50.0$, $d_i = (i-1) * 0.001$ for $i = 4, \ldots, n$ and $\hat{d}_1 = \hat{d}_2 = \hat{d}_3 = 0.0$,

$\hat{d}_i = i * 0.0001$. Now set $A = U^T D_1 U$ and $B = U^T D_2 U$ with a Householder matrix $U = I_n - 2\frac{ww^T}{w^T w}$ where $w = [1, 2, \ldots, 100]$. The resulting matrix

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

is Hamiltonian and has eigenvalues

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

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

After three steps of the symplectic Lanczos algorithm (without re-J-orthogonalization) we obtain the Ritz values

$$\pm 1.999991457279083\text{e}{+}02, \quad \pm 9.931554785773068\text{e}{+}01, \quad \pm 3.371968773385778\text{e}{+}01.$$

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

$$\begin{array}{lll} \pm 1.999999999999998\text{e}{+}02, & \pm 9.999999999999989\text{e}{+}01, & \pm 4.999999999997731\text{e}{+}01, \\ \pm 8.451080813545205\text{e}{-}02, & & \end{array}$$

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. In fact, after 9 iterations we have Ritz values

$$\begin{array}{lll} \pm 1.999999999999999\text{e}{+}02, & \pm 9.999999999999999\text{e}{+}01, & \pm 5.000000000000038\text{e}{+}01, \\ \pm 1.999999999999997\text{e}{+}02, & \pm 9.999999999985583\text{e}{+}01, & \pm 4.999999974747666\text{e}{+}01, \\ \pm 9.524662688488485\text{e}{-}02, & \pm 7.720710855953188\text{e}{-}02, & \pm 3.757475009324353\text{e}{-}02. \end{array}$$

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

$$\begin{array}{lll} \pm 1.999999999999999\text{e}{+}02, & \pm 9.999999999999993\text{e}{+}01, & \pm 4.999999999999997\text{e}{+}01, \\ \pm 9.754957790699192\text{e}{-}02, & \pm 9.154380154101090\text{e}{-}02, & \pm 8.237785481069571\text{e}{-}02, \\ \pm 6.786890886560507\text{e}{-}02, & \pm 4.923341543122169\text{e}{-}02, & \pm 1.448276946901055\text{e}{-}02. \end{array}$$

These first results give rise to the hope that the (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 which is used to approximate a few eigenvalues and associated eigenvectors and to compute a low rank approximation to the 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 can break down before the desired number of eigenvalues is computed. 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 $\widetilde{H}^{2k}$ having a stable k–dimensional invariant subspace due to purely imaginary Ritz values. Inexpensive implicit restarts are developed which can be used to overcome (near) breakdowns in 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. A restarted symplectic Arnoldi method can be formulated along the lines of our restarted symplectic Lanczos method. But as stated in [41] : *When both the column and the row subspaces contain, respectively, $\sqrt{\epsilon}$ approximations to the eigenvectors of λ then the Ritz values will be an ϵ-approximation to λ. This can not happen with one-sided approximations* (as the Arnoldi method yields) *unless the matrix is normal.*

Our analysis shows 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 possibility how the method can be used 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 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*, Electr. Trans. Num. Anal., 1 (1993), pp. 33–48.

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

[3] P. Benner, A. Laub, and V. Mehrmann, *A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case*, Tech. Report SPC 95_22, Fak. f. Mathematik, TU Chemnitz–Zwickau, 09107 Chemnitz, FRG, 1995.

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

[5] W. Bunse and A. Bunse-Gerstner, *Numerische Lineare Algebra*, Teubner, Stuttgart, 1985.

[6] A. Bunse-Gerstner, *An analysis of the HR algorithm for computing the eigenvalues of a matrix*, Linear Algebra Appl., 35 (19841), pp. 155–173.

[7] A. Bunse-Gerstner, *Matrix factorizations for symplectic QR-like methods*, Linear Algebra Appl., 83 (1986), pp. 49–77.

[8] 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 (1986), pp. 1104–1113.

[9] R. BYERS, *Hamiltonian and Symplectic Algorithms for the Algebraic Riccati Equation*, PhD thesis, Cornell University, Dept. Comp. Sci., Ithaca, NY, 1983.

[10] ——, *A Hamiltonian QR-algorithm*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 212–229.

[11] ——, *A bisection method for measuring the distance of a stable to unstable matrices*, SIAM J. Sci. Statist. Comput., (1988), pp. 875–881.

[12] D. CALVETTI, L. REICHEL, AND D. C. SORENSEN, *An implicitly restarted Lanczos method for large symmetric eigenvalue problems*, Electr. Trans. Num. Anal., 2 (1994), pp. 1 – 21.

[13] J. CHOW AND P. KOKOTOVIC, *A decomposition of near–optimum regulators for systems with slow and fast modes*, IEEE Trans. Automat. Control, AC-21 (1976), pp. 701–705.

[14] J. DELLA-DORA, *Sur quelques Algorithmes de recherche de valeurs propres*, Thése, L'Université Scientifique es Medicale de Grenoble, 1973.

[15] P. V. DOOREN. Talk given at IMA Conference on *Linear Algebra and Applications*, Manchester, UK, July 1995.

[16] 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 (1989), pp. 831–847.

[17] L. ELSNER, *On some algebraic problems in connection with general eigenvalue algorithms*, Linear Algebra Appl., 26 (1979), pp. 123 – 138.

[18] 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", Universität Bielefeld, Bielefeld, FRG, 1992.

[19] J. G. F. FRANCIS, *The QR transformation Part I*, Comput. J., 4 (1961), pp. 265 – 271.

[20] ——, *The QR transformation Part II*, Comput. J., 4 (1961), pp. 332 – 345.

[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 AND V. MEHRMANN, *A symplectic look-ahead Lanczos algorithm for the Hamiltonian eigenvalue problem.*

[23] 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 (1993), pp. 137 – 158.

[24] G. GOLUB AND C. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 2nd ed., 1989.

[25] E. J. GRIMME, D. C. SORENSEN, AND P. V. DOOREN, *Model reduction of state space systems via an implicitly restarted Lanczos method*, Technical Report TR94-21, Department of Computational & Applied Mathematics, Rice University, Houston, Texas, USA, 1994.

[26] M. GUTKNECHT, *A completed theory of the unsymmetric Lanczos process and related algorithms, Part I*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 594 – 639.

[27] ——, *A completed theory of the unsymmetric Lanczos process and related algorithms, Part II*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 15 – 58.

[28] A. HODEL AND K. POOLLA, *Heuristic approaches to the solution of very large sparse Lyapunov and algebraic Riccati equations*, in Proc. 27th IEEE Conf. Decis. Cont., Austin, TX, 1988, pp. 2217–2222.

[29] A. S. HOUSEHOLDER, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New

York, 1964.
[30] I. M. Jaimoukha and E. M. Kasenally, *Krylov subspace methods for solving large Lyapunov equations*, SIAM J. Numer. Anal., 31 (1994), pp. 227–251.
[31] V. N. Kublanoskaja, *On some algorithms for the solution of the complete eigenvalue problem*, U.S.S.R. Comput. Math. and Math. Phys., 3 (1961), pp. 637 – 657.
[32] P. Lancaster and L. Rodman, *The Algebraic Riccati Equation*, Oxford University Press, Oxford, 1995.
[33] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards, 45 (1950), pp. 255 – 282.
[34] 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.
[35] V. Mehrmann, *Der SR-Algorithmus zur Berechnung der Eigenwerte einer Matrix*, Diplomarbeit, Universität Bielefeld, Bielefeld, FRG, 1979.
[36] ——, *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution*, no. 163 in Lecture Notes in Control and Information Sciences, Springer-Verlag, Heidelberg, July 1991.
[37] G. Mei, *A new method for solving the algebraic Riccati equation*, master's thesis, Nanjing Aeronautical Institute, Campus P.O. Box 245, Nanjing, P.R. China, 1986.
[38] J. Olson, H. Jensen, and P. Jørgensen, *Solution of large matrix equations which occur in response theory*, J. Comput. Phys., 74 (1988), pp. 265–282.
[39] C. Paige, *The Computation of Eigenvalues and eigenvectors of Very Large Sparse Matrices*, PhD thesis, University of London (UK), 1971.
[40] C. Paige and C. Van Loan, *A Schur decomposition for Hamiltonian matrices*, Linear Algebra Appl., 14 (1981), pp. 11–32.
[41] B. N. Parlett, D. R. Taylor, and Z. A. Liu, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp., 44 (1985), pp. 105 – 124.
[42] J. Potter, *Matrix quadratic solutions*, SIAM J. Appl. Math., 14 (1966), pp. 496–501.
[43] 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 (1995), pp. 514–541.
[44] H. Rutishauser, *Solution of eigenvalue problems with the LR transformation*, Nat. Bur. Standards. Appl. Math. Ser., 49 (1958), pp. 47–81.
[45] V. Sima, *Algorithm GRICSR solving continuous-time algebraic Riccati equations using Gaussian symplectic transformations*, Stud. Res. Comp. Inf., 1 (1992), pp. 237–254.
[46] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357 – 385.
[47] D. R. Taylor, *Analysis of the look ahead Lanczos algorithm*, PhD thesis, Center for Pure and Applied Mathematics, University of California, Berkley, CA, 1982.
[48] F. Uhlig, *The DQR algorithm, Part I & II.* preprint, 1995.
[49] C. Van Loan, *A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix*, Linear Algebra Appl., 16 (1984), pp. 233–251.

Other titles in the SPC series:

95_1 T. Apel, G. Lube. Anisotropic mesh refinement in stabilized Galerkin methods Januar 1995.

95_2 M. Meisel, A. Meyer. Implementierung eines parallelen vorkonditionierten Schur-Komplement CG-Verfahrens in das Programmpaket FEAP. Januar 1995.

95_3 S. V. Nepomnyaschikh. Optimal multilevel extension operators. January 1995

95_4 M. Meyer. Grafik-Ausgabe vom Parallelrechner für 3D-Gebiete. Januar 1995

95_5 T. Apel, G. Haase, A. Meyer, M. Pester. Parallel solution of finite element equation systems: efficient inter-processor communication. Februar 1995

95_6 U. Groh. Ein technologisches Konzept zur Erzeugung adaptiver hierarchischer Netze für FEM-Schemata. Mai 1995

95_7 M. Bollhöfer, C. He, V. Mehrmann. Modified block Jacobi preconditioners for the conjugate gradient method. Part I: The positive definit case. January 1995

95_8 P. Kunkel, V. Mehrmann, W. Rath, J. Weickert. GELDA: A Software Package for the Solution of General Linear Differential Algebraic Equation. February 1995

95_9 H. Matthes. A DD preconditioner for the clamped plate problem. February 1995

95_10 G. Kunert. Ein Residuenfehlerschätzer für anisotrope Tetraedernetze und Dreiecksnetze in der Finite-Elemente-Methode. März 1995

95_11 M. Bollhöfer. Algebraic Domain Decomposition. March 1995

95_12 B. Nkemzi. Partielle Fourierdekomposition für das lineare Elastizitätsproblem in rotationssymmetrischen Gebieten. März 1995

95_13 A. Meyer, D. Michael. Some remarks on the simulation of elasto-plastic problems on parallel computers. March 1995

95_14 B. Heinrich, S. Nicaise, B. Weber. Elliptic interface problems in axisymmetric domains. Part I: Singular functions of non-tensorial type. April 1995

95_15 B. Heinrich, B. Lang, B. Weber. Parallel computation of Fourier-finite-element approximations and some experiments. May 1995

95_16 W. Rath. Canonical forms for linear descriptor systems with variable coefficients. May 1995

95_17 C. He, A. J. Laub, V. Mehrmann. Placing plenty of poles is pretty preposterous. May 1995

95_18 J. J. Hench, C. He, V. Kučera, V. Mehrmann. Dampening controllers via a Riccati equation approach. May 1995

95_19 M. Meisel, A. Meyer. Kommunikationstechnologien beim parallelen vorkonditionierten Schur-Komplement CG-Verfahren. Juni 1995

95_20 G. Haase, T. Hommel, A. Meyer and M. Pester. Bibliotheken zur Entwicklung paralleler Algorithmen. Juni 1995.

95_21 A. Vogel. Solvers for Lamé equations with Poisson ratio near 0.5. June 1995.

95_22 P. Benner, A. J. Laub, V. Mehrmann. A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case. October 1995.

95_23 P. Benner, A. J. Laub, V. Mehrmann. A collection of benchmark examples for the numerical solution of algebraic Riccati equations II: Discrete-time case. to appear: December 1995.

95_24 P. Benner, R. Byers. Newton's method with exact line search for solving the algebraic Riccati equation. October 1995.

95_25 P. Kunkel, V. Mehrmann. Local and Global Invariants of Linear Differential-Algebraic Equations and their Relation. July 1995.

95_26 C. Israel. NETGEN69 - Ein hierarchischer paralleler Netzgenerator. August 1995.

95_27 M. Jung. Parallelization of multigrid methods based on domain decomposition ideas. September 1995.

95_28 P. Benner, H. Faßbender. A restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem. October 1995.

95_29 G. Windisch. Exact discretizations of two-point boundary value problems. October 1995.

Some papers can be accessed via anonymous ftp from server `ftp.tu-chemnitz.de`, directory `pub/Local/mathematik/SPC`. (Note the capital L in Local!)
The complete list of current and former preprints is available via
`http://www.tu-chemnitz.de/~pester/sfb/spc95pr.html`.