Report Number 09/32

A Hamiltonian Krylov-Schur-type method based on the symplectic Lanczos process

by

Peter Benner, Heike Faßbender, Martin Stoll

Oxford Centre for Collaborative Applied Mathematics
Mathematical Institute
24 - 29 St Giles'
Oxford
OX1 3LB
England

A Hamiltonian Krylov-Schur-type method based on the symplectic Lanczos process

Peter Benner*,[a], Heike Faßbender[b], Martin Stoll[c]

[a] *TU Chemnitz, Fakultät für Mathematik, Mathematik in Industrie und Technik, 09107 Chemnitz, Germany*
[b] *AG Numerik, Institut* Computational Mathematics, *TU Braunschweig, D-38092 Braunschweig, Germany*
[c] *Oxford Centre for Collaborative Applied Mathematics, Mathematical Institute, 24 – 29 St Giles', Oxford, OX1 3LB, United Kingdom*

Abstract

We discuss a Krylov-Schur like restarting technique applied within the symplectic Lanczos algorithm for the Hamiltonian eigenvalue problem. This allows to easily implement a purging and locking strategy in order to improve the convergence properties of the symplectic Lanczos algorithm. The Krylov-Schur-like restarting is based on the SR algorithm. Some ingredients of the latter need to be adapted to the structure of the symplectic Lanczos recursion. We demonstrate the efficiency of the new method for several Hamiltonian eigenproblems.

Key words: Hamiltonian eigenproblem, symplectic Lanczos method, Krylov-Schur method, implicit restarting, SR algorithm.
2000 MSC: 65F15, 65F50, 15A18

1. Introduction

Hamiltonian matrices $H \in \mathbb{R}^{2n\times 2n}$ have the explicit block structure

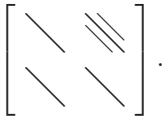
$$H = \begin{bmatrix} A & G \\ Q & -A^T \end{bmatrix}, \qquad G = G^T, \quad Q = Q^T, \tag{1}$$

where A, G, Q are real $n \times n$ matrices. Hamiltonian matrices and eigenproblems arise in a variety of applications. They are ubiquitous in control theory, where they play an important role in various control design procedures (linear-quadratic optimal control, Kalman filtering, H_2- and H_∞-control, etc., see, e.g., [1, 2, 3, 4, 5] and most textbooks on control theory), system analysis problems like stability radius, pseudo-spectra, and H_∞-norm computations [6, 7, 8], and model reduction [9, 10, 11, 12, 13]. Another source of eigenproblems exhibiting Hamiltonian structure is the linearization of certain quadratic eigen-

*Corresponding author

value problems [14, 15, 16, 17]. Further applications can be found in computational physics and chemistry, e.g., symplectic integrators for molecular dynamics [18, 19], methods for random phase approximation (RPA) [20, 21, 22] and many more.

Many of the abovementioned applications involve large and sparse Hamiltonian matrices and mostly, a few extremal or interior eigenvalues are required. An appropriate tool to solve these kind of problems is the (shift-and-invert) symplectic Lanczos method [23, 24]. It projects the large, sparse $2n \times 2n$ Hamiltonian matrix H onto a small $2k \times 2k$ Hamiltonian J–Hessenberg matrix $\widetilde{H}$, $k \ll n$. Hamiltonian J–Hessenberg (also called Hamiltonian J–triangular) matrices can be depicted as



That is, due to the Hamiltonian structure, it can be represented by $4k-1$ parameters instead of the usual k^2 matrix entries. As observed in [25], the SR algorithm preserves the Hamiltonian J–Hessenberg form and can be implemented working only with the $4k-1$ parameters [26].

An ubiquitous matrix when dealing with Hamiltonian eigenvalue problems is the skew-symmetric matrix

$$J = J_n = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}, \tag{2}$$

where I_n denotes the $n \times n$ identity matrix. By straightforward algebraic manipulation one can show that a Hamiltonian matrix H is equivalently defined by the property

$$HJ = (HJ)^T. \tag{3}$$

In other words, Hamiltonian matrices are skew-adjoint with respect to the bilinear form induced by J, i.e., $\langle x, y \rangle_J := y^T J x$ for $x, y \in \mathbb{R}^{2n}$. Any matrix $S \in \mathbb{R}^{2n \times 2n}$ satisfying

$$S^T J S = S J S^T = J \tag{4}$$

is called symplectic, i.e., symplectic matrices are orthogonal with respect to $\langle \, . \, , \, . \, \rangle_J$ and are therefore also called J–orthogonal. Symplectic similarity transformations preserve the Hamiltonian structure:

$$(S^{-1}HS)J = S^{-1}HJS^{-T} = S^{-1}J^T H^T S^{-T} = [(S^{-1}HS)J]^T.$$

One of the most remarkable properties of a Hamiltonian matrix is that its eigenvalues always occur in pairs $\{\lambda, -\lambda\}$ if λ is real or purely imaginary, or in quadruples $\{\lambda, -\lambda, \overline{\lambda}, -\overline{\lambda}\}$ otherwise. Hence, the spectrum of any Hamiltonian matrix is symmetric with respect to both the real and imaginary axis. We call this property *Hamiltonian symmetry*. Numerical methods that take this structure into account are capable of preserving the eigenvalue pairings despite the presence of roundoff errors and thus return physically meaningful results.

Moreover, employing the structure usually leads to more efficient and sometimes more accurate algorithms.

In [23], the ideas of implicitly restarted Lanczos methods [27, 28, 29] together with ideas to reflect the Hamiltonian structure are used to derive the implicitly restarted symplectic Lanczos algorithm for Hamiltonian eigenproblems. There are several variants of symplectic Lanczos processes for Hamiltonian matrices available which create a Hamiltonian J–Hessenberg matrix, [23, 24, 30] as well as other attempts to create structure–preserving methods using a symplectic Lanczos method ([31] which works with the squared Hamiltonian matrix and suffers from stability problems as well as from breakdown or a symplectic look–ahead Lanczos algorithm [32] which overcomes breakdown by giving up the strict Hamiltonian J–Hessenberg form).

A different approach for solving large scale Hamiltonian eigenproblems makes use of the following observation: for any Hamiltonian matrix H the matrices H^2, $(H-\sigma I)^{-1}(H+\sigma I)^{-1}$ with $\sigma \in \mathbb{R}, \imath\mathbb{R}$ and $(H-\sigma I)^{-1}(H+\sigma I)^{-1}(H-\overline{\sigma} I)^{-1}(H+\overline{\sigma} I)^{-1}$ with $\sigma \in \mathbb{C}$ are skew-Hamiltonian matrices. The standard (implicitly restarted) Arnoldi method [29] automatically preserves this structure. This led to the development of the SHIRA method [15, 33] as a structure-preserving (shift-and-invert) Arnoldi method for Hamiltonian matrices.

Here we consider the structure–preserving Lanczos method which generates a sequence of matrices

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

satisfying the symplectic Lanczos recursion

$$HS^{2n,2k} = S^{2n,2k}\widetilde{H}^{2k,2k} + \zeta_{k+1} v_{k+1} e_{2k}^T, \tag{5}$$

where $\widetilde{H}^{2k,2k}$ is a $2k \times 2k$ Hamiltonian J–Hessenberg matrix and the columns of $S^{2n,2k}$ are J–orthogonal. In the following, we call (5) a *symplectic Lanczos decomposition*. An implicit Lanczos restart computes the Lanczos decomposition

$$H\breve{S}^{2n,2k} = \breve{S}^{2n,2k}\breve{H}^{2k,2k} + \breve{r}_{k+1} e_{2k}^T \tag{6}$$

which corresponds to the starting vector

$$\breve{s}_1 = p(H)s_1$$

(where $p(H) \in \mathbb{R}^{2n\times 2n}$ is a polynomial) without having to explicitly restart the Lanczos process with the vector $\breve{s}_1$. This process is iterated until the residual vector r_{k+1} is tiny. J–orthogonality of the k Lanczos vectors is secured by re–J–orthogonalizing these vectors when necessary. This idea was investigated in [34]. As the iteration progresses, some of the Ritz values may converge to eigenvalues of H long before the entire set of wanted eigenvalues have. These converged Ritz values may be part of the wanted or unwanted portion of the spectrum. In either case it is desirable to deflate the converged Ritz values and corresponding Ritz vectors from the unconverged portion of the factorization. If the converged

Ritz value is wanted then it is necessary to keep it in the subsequent factorizations; if it is unwanted then it must be removed from the current and the subsequent factorizations. Locking and purging techniques to accomplish this in the context of implicitly restarted Arnoldi/Lanczos methods were first introduced by Sorensen [29, 35, 36, 37]. These techniques are fairly involved and do not easily carry over to the symplectic Lanczos method. Most of the complications in the purging and deflating algorithms come from the need to preserve the structure of the decomposition, in particular, to preserve the J–Hessenberg form and the zero structure of the vector e_{2k}^T. In [38], Stewart shows how to relax the definition of an Arnoldi decomposition such that the purging/locking and deflation problems can be solved in a natural and efficient way. Since the method is centered about the Schur decomposition of the Hessenberg matrix, the method is called the Krylov-Schur method.

In this paper we will discuss how to adapt the Krylov-Schur restarting to the symplectic Lanczos method for Hamiltonian matrices. First, the symplectic Lanczos method and the Hamiltonian SR method are briefly reviewed in Sections 2 and 3. Next the Krylov-Schur-like restarted symplectic Lanczos method is developed in Section 4 while locking and purging techniques are considered in Section 5. Before numerical experiments are reported in Section 8, stopping criteria and shift-and-invert techniques are briefly discussed in Sections 6 and 7. At the end, some conclusions are given in Section 9.

Throughout this paper, $\|\,.\,\|$ will denote the Euclidian norm of a vector and the corresponding spectral norm for matrices. I_m stands for the identity matrix of size $m \times m$ with columns e_k $(k = 1, \ldots m)$, H will in general denote Hamiltonian matrices and S matrices with J–orthogonal columns.

2. The symplectic Lanczos method

The usual nonsymmetric Lanczos algorithm generates two sequences of vectors, see, e.g., [39]. 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 [40].)

The structure-preserving symplectic Lanczos method [23, 24] generates a sequence of matrices that satisfy the Lanczos recursion

$$HS^{2n,2k} = S^{2n,2k}\widetilde{H}^{2k,2k} + \zeta_{k+1}v_{k+1}e_{2k}^T. \tag{7}$$

Here, $\widetilde{H}^{2k,2k}$ is a Hamiltonian J–Hessenberg matrix

$$\widetilde{H}^{2k,2k} = \left[\begin{array}{cccc|cccc} \delta_1 & & & & \beta_1 & \zeta_2 & & \\ & \delta_2 & & & \zeta_2 & \beta_2 & \ddots & \\ & & \ddots & & & \ddots & \ddots & \zeta_k \\ & & & \delta_k & & & \zeta_k & \beta_k \\ \hline \nu_1 & & & & -\delta_1 & & & \\ & \nu_2 & & & & -\delta_2 & & \\ & & \ddots & & & & \ddots & \\ & & & \nu_k & & & & -\delta_k \end{array}\right]. \tag{8}$$

The space spanned by the columns of $S^{2n,2k}$ is symplectic since

$$S^{2n,2k^T} J^n S^{2n,2k} = J^k$$

where J^j is a $2j \times 2j$ matrix of the form (2). The vector $r_{k+1} := \zeta_{k+1} v_{k+1}$ is the residual vector and is J–orthogonal to the columns of $S^{2n,2k}$, the Lanczos vectors. The matrix $\widetilde{H}^{2k,2k}$ is the J–orthogonal projection of H onto the range of $S^{2n,2k}$,

$$\widetilde{H}^{2k,2k} = (J^k)^T (S^{2n,2k})^T J^n H S^{2n,2k}.$$

Equation (7) defines a length $2k$ Lanczos factorization of H. If the residual vector r_{k+1} is the zero vector, then equation (7) is called a truncated Lanczos factorization when $k < n$. Note that r_{n+1} must vanish since $(S^{2n,2n})^T J r_{n+1} = 0$ and the columns of $S^{2n,2n}$ form a J–orthogonal basis for $\mathbb{R}^{2n}$. In this case the symplectic Lanczos method computes a reduction to permuted J–Hessenberg form.

A symplectic Lanczos factorization exists for almost all starting vectors $Se_1 = v_1$. Moreover, the symplectic Lanczos factorization is, up to multiplication by a trivial matrix, specified by the starting vector v_1, see, e.g., [23, 41]. Hence, 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. Assume that no such breakdowns occur, and let $S^{2n,2n} = [v_1, v_2, \ldots, v_n, w_1, w_2, \ldots, w_n]$. For a given v_1, a Lanczos method constructs the matrix S columnwise from the equations

$$HSe_j = S\widetilde{H}^{2n,2n} e_j, \qquad j = 1, n+1, 2, n+2, 3, \ldots 2n-1.$$

This yields Algorithm 1, where the freedom in the choice of the parameters δ_m (which are set to 1 in [23] and to 0 in [30]) is used to retain a local orthogonality condition, i.e., $w_m \perp v_m$, in addition to the global J–orthogonality of the basis vectors. This choice of δ_m is first suggested in [24] and is proved in [42] to minimize the condition number of the symplectic Lanczos basis when the other parameters are chosen as in Algorithm 1. Eigenvalues and eigenvectors of Hamiltonian J–Hessenberg matrices can be computed efficiently by the SR

Algorithm 1 Symplectic Lanczos method

INPUT: $H \in \mathbb{R}^{2n,2n}$ and $m \in \mathbb{N}$

OUTPUT: $S \in \mathbb{R}^{2n,2m}$, $\delta_1, \ldots, \delta_m$, $\beta_1, \ldots, \beta_m$, $\nu_1, \ldots, \nu_m$, $\zeta_2, \ldots, \zeta_{m+1}$ and v_{m+1} defining a Hamiltonian J–Hessenberg matrix as in (8).

Choose start vector $\tilde{v}_1 \neq 0 \in \mathbb{R}^{2n}$.
$v_0 = 0 \in \mathbb{R}^{2n}$
$\zeta_1 = \|\tilde{v}_1\|_2$
$v_1 = \frac{1}{\zeta_1}\tilde{v}_1$
for $m = 1, 2, \ldots$ **do**
  `% Computation of matrix-vector products`
  $v = Hv_m$
  $w = Hw_m$
  `% Computation of` δ_m
  $\delta_m = v_m^T v$
  `% Computation of` w_m
  $\tilde{w}_m = v - \delta_m v_m$
  $\nu_m = \langle v_m, v \rangle_J$
  $w_m = \frac{1}{\nu_m}\tilde{w}_m$
  `% Computation of` β_m
  $\beta_m = -\langle w_m, w \rangle_J$
  `% Computation of` v_{m+1}
  $\tilde{v}_{m+1} = w - \zeta_m v_{m-1} - \beta_m v_m + \delta_m w_m$
  $\zeta_{m+1} = \|\tilde{v}_{m+1}\|_2$
  $v_{m+1} = \frac{1}{\zeta_{m+1}}\tilde{v}_{m+1}$
end for

algorithm. This has been discussed to some extent in [13, 23, 25, 26, 41], see Section 3.

The symplectic Lanczos method described above inherits all numerical difficulties of Lanczos-like methods for nonsymmetric matrices, in particular serious breakdown is possible. One approach to deal with the numerical difficulties of Lanczos-like algorithms is to implicitly restart the Lanczos factorization. This approach was introduced by Sorensen [29] in the context of nonsymmetric matrices and the Arnoldi process and adapted to the symplectic Lanczos process in [23]. The latter paper lacks a discussion of locking and purging converged and unwanted eigenvalues from the restarted iteration as such techniques are quite difficult to accomplish for the symplectic Lanczos method. Note that purging is in principle achieved by using the unwanted eigenvalues as exact shifts in the polynomial filter used for restarting, but due to numerical roundoff, they will often reappear. Thus, it is necessary to ensure that the next Krylov subspace built will remain J–orthogonal to the unwanted eigenspace.

Before we discuss the new restarting technique for the symplectic Lanczos method based on Krylov-Schur-like decompositions, we briefly recall some facts about the Hamiltonian SR algorithm.

3. The Hamiltonian *SR* algorithm

Eigenvalues and eigenvectors of Hamiltonian J–Hessenberg matrices (8) can be computed efficiently by the SR algorithm. This has already been discussed to some extent in [13, 23, 25, 26, 41]. If H is the current iterate, then a spectral transformation function q is chosen (such that $q(H) \in \mathbb{R}^{2n\times 2n}$) and the SR decomposition of $q(H)$ is formed, if possible:

$$q(H) = SR.$$

Then the symplectic factor S is used to perform a similarity transformation on H to yield the next iterate:

$$\widehat{H} = S^{-1}HS. \tag{9}$$

An algorithm for computing S and R explicitly is presented in [25]. As with explicit QR steps, the expense of explicit SR steps comes from the fact that $q(H)$ has to be computed explicitly. A preferred alternative is the implicit SR step, an analogue to the Francis QR step [43], yielding the same iterate as the explicit SR step due to the implicit S-theorem [25, 41]. The first implicit transformation S_1 is selected in order to introduce a bulge into the J–Hessenberg matrix H. That is, a symplectic matrix S_1 is determined such that

$$S_1^{-1}q(H)e_1 = \alpha e_1, \qquad \alpha \in \mathbb{R},$$

where $q(H)$ is an appropriately chosen spectral transformation function. Applying this first transformation to the J–Hessenberg matrix yields a Hamiltonian matrix $S_1^{-1}HS_1$ with almost J–Hessenberg form having a small bulge. The remaining implicit transformations perform a bulge-chasing sweep down the subdiagonals to restore the J–Hessenberg form. That is, a symplectic matrix S_2 is determined such that $S_2^{-1}S_1^{-1}HS_1S_2$ is of J–Hessenberg form again. If H is an unreduced J–Hessenberg matrix and $\operatorname{rank}(q(H)) = 2n$, then $\widetilde{H} = S_2^{-1}S_1^{-1}HS_1S_2$ is also an unreduced J–Hessenberg matrix. Hence, by the implicit S-theorem [25, 41], there will be parameters $\widetilde{\delta}_1, \ldots, \widetilde{\delta}_n, \widetilde{\beta}_1, \ldots, \widetilde{\beta}_n, \widetilde{\zeta}_1, \ldots, \widetilde{\zeta}_n, \widetilde{\nu}_2, \ldots, \widetilde{\nu}_n$ which determine $\widetilde{H}$. An efficient implementation of the SR step for Hamiltonian J–Hessenberg matrices involves $\mathcal{O}(n)$ arithmetic operations ($\mathcal{O}(n^2)$ if the symplectic similarity transformation is to be accumulated), see [25, 26, 41].

Due to the special Hamiltonian eigenstructure, the spectral transformation function will be chosen either as

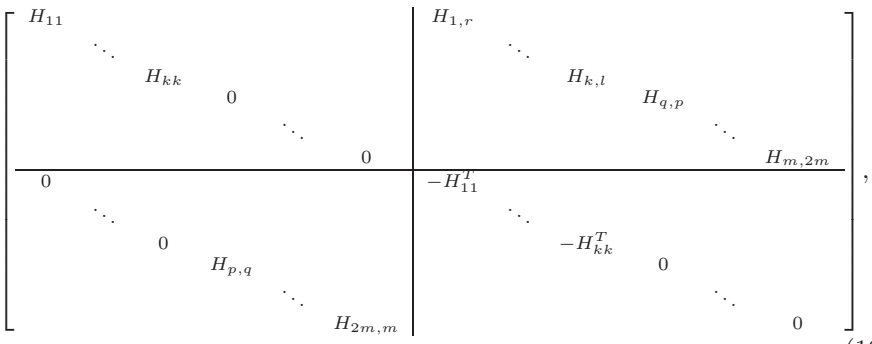
$$q_2(H) = (H - \mu I)(H + \mu I), \qquad \mu \in \mathbb{R} \ \text{ or } \ \mu = i\omega, \omega \in \mathbb{R},$$

or

$$q_4(H) = (H - \mu I)(H + \mu I)(H - \overline{\mu} I)(H + \overline{\mu} I), \qquad \mu \in \mathbb{C}, \operatorname{Re}(\mu) \neq 0.$$

If the chosen shifts are good approximate eigenvalues, we expect deflation. As proposed in [25], a shift strategy similar to that used in the standard QR algorithm should be used. By applying a sequence of quadruple shift SR steps to

a Hamiltonian J–Hessenberg matrix H it is possible to reduce the tridiagonal block in H to quasi-diagonal form with 1×1 and 2×2 blocks on the diagonal. The eigenproblem decouples into a number of simple Hamiltonian 2×2 or 4×4 eigenproblems



(10)

where $l = m + k$, $p = l + 1$, $q = k + 1$, $r = m + 1$ and the blocks H_{11} to H_{kk} represent the real (size of the block 1×1) and complex (size of the block 2×2) eigenvalues with negative real part. The other blocks represent purely imaginary eigenvalues and are of size 1×1. Any ordering of the small blocks on the diagonal of the $(1, 1)$ block are possible (the other diagonals have to be reordered accordingly). An efficient implementation of the SR algorithm for Hamiltonian J–Hessenberg matrices involves $\mathcal{O}(n)$ arithmetic operations ($\mathcal{O}(n^2)$ if the symplectic similarity transformation is to be accumulated) [25, 26, 41].

4. Krylov-Schur-like restarted symplectic Lanczos method

To implement an efficient implicitly restarted Lanczos process it is necessary to introduce deflation techniques. The basic ideas were developed by Sorensen and Lehoucq in [29, 35, 36, 37]. The focus is on purging and locking eigenvalues during the iteration process, where locking means to fix the converged and wanted eigenvalues and purging means to purge the unwanted but converged eigenvalues. Unfortunately, these techniques are hard to implement, especially when the eigenblock is of size (2×2). Moreover, this strategy appears to be difficult to adopt to the symplectic Lanczos process and so far has defied its realization. In [38], Stewart shows how to relax the definition of an Arnoldi decomposition such that the purging and deflating problems can be solved in a natural and efficient way. Since the method is centered about the Schur decomposition of the Hessenberg matrix, the method is called the Krylov-Schur method. In this section we develop a Krylov-Schur like variant of the symplectic Lanczos method for Hamiltonian matrices. An initial version of this method was developed in [44]. This and the following sections make use of the results derived there without further notice.

So far, we have considered symplectic Lanczos factorizations of order $2k$ of the form (7):

$$HS^{2n,2k} = S^{2n,2k}\widetilde{H}^{2k,2k} + \zeta_{k+1}v_{k+1}e_{2k}^T.$$

More generally, we will speak of a Hamiltonian Krylov-Schur-type decomposition of order $2k$ if $2k+1$ linearly independent vectors $u_1, u_2, \ldots, u_{2k+1} \in \mathbb{R}^{2n}$ are given such that

$$HU^{2n,2k} = U^{2n,2k}B^{2k,2k} + u_{2k+1}b_{2k+1}^T, \tag{11}$$

where $U^{2n,2k} = [u_1, u_2, \ldots, u_{2k}]$. Equivalently, we can write

$$HU^{2n,2k} = U^{2n,2k+1}\widehat{B}^{2k,2k},$$

where $U^{2n,2k+1} = [U^{2n,2k} \;\; u_{2k+1}]$ and

$$\widehat{B}^{2k+1,2k} = \begin{bmatrix} B^{2k,2k} \\ b_{2k+1}^T \end{bmatrix}.$$

This definition removes practically all the restrictions imposed on a symplectic Lanczos decomposition. The vectors of the decomposition are not required to be J–orthogonal and the vector b_{2k+1} and the matrix $B^{2k,2k}$ are allowed to be arbitrary.

If the columns of $U^{2n,2k+1}$ are J–orthogonal, we say that the Hamiltonian Krylov-Schur-type decomposition is J–orthogonal. Please note, that no particular form of $\widehat{B}^{2k+1,2k}$ is assumed here. It is uniquely determined by the basis $U^{2n,2k+1}$. For if $[V^{2n,2k} \; v]^T$ is any left inverse for $U^{2n,2k+1}$, then it follows from (11) that

$$B^{2k,2k} = (V^{2n,2k})^T HU^{2n,2k}$$

and

$$b_{2k+1}^T = v^T HU^{2n,2k}.$$

In particular, $B^{2k,2k}$ is a Rayleigh quotient of H with respect to the J–orthogonal Lanczos basis $\operatorname{span} U^{2n,2k}$ and is thus Hamiltonian.

We say that the Hamiltonian Krylov-Schur-type decomposition spans the space spanned by the columns of $U^{2n,2k+1}$. Two Hamiltonian Krylov-Schur-type decompositions spanning the same space are said to be equivalent.

For any nonsingular matrix $Q \in \mathbb{R}^{2k,2k}$ we obtain from (11) an equivalent Hamiltonian Krylov-Schur-type decomposition

$$H(U^{2n,2k}Q) = (U^{2n,2k}Q)(Q^{-1}B_P^{2k,2k}Q) + u_{2k+1}(b_{2k+1}^T Q).$$

In this case, the two Hamiltonian Krylov-Schur-type decompositions are said to be similar to each other. Note that similar Hamiltonian Krylov-Schur-type decompositions are also equivalent.

If, in (11) the vector u_{2k+1} can be written as $u_{2k+1} = \gamma\widehat{u}_{2k+1} + U^{2n,2k}a, \gamma \neq 0$, then we have that the Hamiltonian Krylov-Schur-type decomposition

$$HU^{2n,2k} = U^{2n,2k}(B + ab_{2k+1}^T) + \gamma\widehat{u}_{2k+1}b_{2k+1}^T$$

is equivalent to the original one, as the space spanned by the columns of $[U^{2n,2k} \ u_{2k+1}]$ is the same as the space spanned by the columns of $[U^{2n,2k} \ \widehat{u}_{2k+1}]$.

Any symplectic Lanczos factorization (7) is at the same time a Hamiltonian Krylov-Schur type decomposition; multiplying (7) from the right by a symplectic matrix S yields an equivalent and similar Hamiltonian Krylov-Schur type decomposition

$$H(S^{2n,2k}S) = (S^{2n,2k}S)(S^{-1}\widetilde{H}^{2k,2k}S) + \zeta_{k+1}v_{k+1}e_{2k}^T S.$$

Moreover, we have the following result.

Theorem 4.1. *Almost every Hamiltonian Krylov-Schur-type decomposition is equivalent to a symplectic Lanczos factorization.*

Note: the symplectic Lanczos factorization may be reduced in the sense that the corresponding Hamiltonian J–Hessenberg matrix is reduced. Moreover, for our purposes, we can drop "almost any" in the theorem above as the Hamiltonian Krylov-Schur-type decompositions that we will use are always J–orthogonal. As will be seen in the proof below, the "almost any" comes from the need of an SR decomposition in the general case which is not necessary in the J–orthogonal case.

For the proof Theorem 4.1, we need the following two observations.

Theorem 4.2. *Suppose $H \in \mathbb{R}^{2n\times 2n}$ is an unreduced J–Hessenberg matrix. If $Hs = \lambda s$ with $s \in \mathbb{K}^{2n}\backslash\{0\}$ and $H^T u = \lambda u$ with $u \in \mathbb{K}^{2n}\backslash\{0\}$, then $e_n^T s \neq 0$ and $e_1^T u \neq 0$.*

Proof. Performing a perfect shuffle of the rows and columns of H yields an unreduced upper Hessenberg matrix. Hence, the theorem follows immediately from the corresponding theorem for Hessenberg matrices [45].

The algorithm for reducing a (general) matrix to J–Hessenberg form as given in [25] reduces the matrix columnwise. In the proof of Theorem 4.1, we will need to reduce a matrix rowwise to J–Hessenberg form. This can be done as given in Algorithm 2. This algorithm makes use of the following elementary symplectic transformations.

- Symplectic Givens transformations $G_k = G(k, c, s)$

$$G_k = \left[\begin{array}{cc|cc} I_{k-1} & & & \\ & c & & s \\ & & I_{n-k} & \\ \hline & & I_{k-1} & \\ & -s & & c \\ & & & I_{n-k} \end{array}\right], \quad c^2 + s^2 = 1, \ c, s \in \mathbb{R}.$$

- Symplectic Householder transformations $H_k = H(k, v)$

$$H_k = \left[\begin{array}{cc|cc} I_{k-1} & & & \\ & P & & \\ \hline & & I_{k-1} & \\ & & & P \end{array}\right], \quad P = I_{n-k+1} - \frac{2}{v^T v}vv^T, \ v \in \mathbb{R}^{n-k+1}.$$

- Symplectic Gauss transformations $L_k = L(k, c, d)$

$$L_k = \left[\begin{array}{ccc|ccc} I_{k-2} & & & & & \\ & c & & & d & \\ & & c & & & \\ & & & I_{n-k} & & \\ \hline & & & I_{k-2} & & \\ & & & & c^{-1} & \\ & & & & & c^{-1} \\ & & & & & & I_{n-k} \end{array}\right], \quad c, d \in \mathbb{R}.$$

The symplectic Givens and Householder transformations are orthogonal, while the symplectic Gauss transformations are nonorthogonal. See, e.g, [25, 46, 47] on how to compute these matrices.

Now we can provide the proof for Theorem 4.1.

Proof of Theorem 4.1. We begin with the Hamiltonian Krylov-Schur-type decomposition

$$HU = UB + ub^T,$$

where for convenience we have dropped all sub- and superscripts. Let $U = SR$ be the SR decomposition of U. Note that almost any U has such a decomposition as the set of matrices having an SR decomposition is dense in $\mathbb{R}^{2n\times 2k}$ [48, Theorem 3.8]. Then

$$HS = H(UR^{-1}) = (UR^{-1})(RBR^{-1}) + u(b^TR^{-1}) =: S\tilde{B} + u\tilde{b}^T$$

is an equivalent decomposition, in which the matrix S is J–orthogonal. Next let

$$\tilde{u} := \gamma^{-1}(u - Sa)$$

be a vector of norm one such that $\tilde{u}$ is J–orthogonal to the span of U, that is, $U^TJ\tilde{u} = 0$. (The vector $\tilde{u}$ is obtained by J–orthogonalizing u w.r.t. the range of S, i.e., a is the vector containing the J–orthogonalization coefficients.) Then the decomposition

$$HS = S(\tilde{B} + a\tilde{b}^T) + \tilde{u}(\gamma\tilde{b}^T) =: S\tilde{\tilde{B}} + \tilde{u}\tilde{\tilde{b}}^T$$

is an equivalent J–orthogonal Hamiltonian Krylov-Schur-type decomposition. Finally, let $\tilde{S}$ be a J–orthogonal matrix such that $\tilde{\tilde{b}}^T\tilde{S} = \|b\|_2 e_{2k}^T$ and $\tilde{S}^{-1}\tilde{\tilde{B}}\tilde{S} = \widetilde{H}$ is in Hamiltonian J–Hessenberg form (this reduction has to be performed rowwise from bottom to top in order to achieve $\tilde{\tilde{b}}^T\tilde{S} = \|b\|_2 e_{2k}^T$, see Algorithm 2 for an algorithm which constructs such an $\tilde{S}$). Then the equivalent decomposition

$$H\tilde{\tilde{S}} = H(S\tilde{S}) = (S\tilde{S})(\tilde{S}^{-1}\tilde{\tilde{B}}\tilde{S}) + \tilde{u}(\tilde{\tilde{b}}^T\tilde{S}) = \tilde{\tilde{S}}\widetilde{H} + \tilde{\tilde{u}}e_{2k}^T$$

is a possibly reduced symplectic Lanczos factorization.

Algorithm 2 Rowwise JHESS algorithm

INPUT : $A \in \mathbb{R}^{2n,2n}$
OUTPUT : $S \in \mathbb{R}^{2n,2n}$ and $A \in \mathbb{R}^{2n,2n}$ such that $A = S^{-1}AS$ is in J–Hessenberg form.

for $j = 1$ to $n-1$ **do**
  for $k = 1$ to $n-j$ **do**
    Compute G_k such that $(AG_k)_{2n-j+1,k} = 0$;
    $A = G_k^T A G_k$;
    $S = SG_k$;
  end for
  if $j < n-1$ **then**
    Compute H_j such that $(AH_j)_{2n-j+1,n+1:2n-j+1} = 0$;
    $A = H_j^T A H_j$;
    $S = SH_j$;
  end if
  if $A_{2n-j+1,2n-j} \neq 0$ and $A_{2n-j+1,n-j+1} = 0$ **then**
    Decomposition does not exist; STOP.
  end if
  Compute L_{j+1} such that $(AL_{j+1})_{2n-j+1,2n-j} = 0$;
  $A = L_{j+1}^{-1} A L_{j+1}$;
  $S = SL_{j+1}$;
  for $k = 1$ to $n-j$ **do**
    Compute G_k such that $(AG_k)_{n-j+1,k} = 0$;
    $A = G_k^T A G_k$;
    $S = SG_k$;
  end for
  if $j < n-1$ **then**
    Compute H_j such that $(AH_j)_{n-j+1,n+1:2n-j-1} = 0$;
    $A = H_j^T A H_j$;
    $S = SH_j$;
  end if
end for

Employing Algorithm 2, the proof of Theorem 4.1 describes in a constructive way how to pass from a Hamiltonian Krylov-Schur-type recursion to a symplectic Lanczos recursion.

Next, we will describe how to construct a Hamiltonian Krylov-Schur-type factorization which enables us to efficiently perform locking and purging. For this, let us assume we have constructed a symplectic Lanczos factorization of order $2(k+p) = 2m$ of the form (7)

$$HS^{2n,2m} = S^{2n,2m}\widetilde{H}^{2m,2m} + \zeta_{m+1} v_{m+1} e_{2m}^T. \tag{12}$$

Applying the SR algorithm to $\widetilde{H}^{2m,2m}$ yields a symplectic matrix $\breve{S}$ such that

$$\breve{S}^{-1}\widetilde{H}^{2m,2m}\breve{S} = \begin{bmatrix} \widetilde{A} & \widetilde{G} \\ \widetilde{Q} & -\widetilde{A}^T \end{bmatrix} = \breve{H}^{2m,2m}$$

decouples into 1×1 or 2×2 blocks on the diagonals of each of the four subblocks $\widetilde{A}, \widetilde{G}$ and $\widetilde{Q}$:

$$\left[\begin{array}{cccc|cccc} \widetilde{A}_{11} & & & & \widetilde{G}_{11} & & & \\ & \widetilde{A}_{22} & & & & \widetilde{G}_{22} & & \\ & & \ddots & & & & \ddots & \\ & & & \widetilde{A}_{mm} & & & & \widetilde{G}_{mm} \\ \hline \widetilde{Q}_{11} & & & & -\widetilde{A}_{11}^T & & & \\ & \widetilde{Q}_{22} & & & & -\widetilde{A}_{22}^T & & \\ & & \ddots & & & & \ddots & \\ & & & \widetilde{Q}_{mm} & & & & -\widetilde{A}_{mm}^T \end{array}\right]. \tag{13}$$

Assume furthermore, that $\breve{S}$ has been constructed such that the desired eigenvalues of $\widetilde{H}^{2m,2m}$ have been moved to the leading parts of the four submatrices such that

$$\breve{H}^{2m,2m} = \left[\begin{array}{cc|cc} \widetilde{A}_1 & & \widetilde{G}_1 & \\ & \widetilde{A}_2 & & \widetilde{G}_2 \\ \hline \widetilde{Q}_1 & & -\widetilde{A}_1^T & \\ & \widetilde{Q}_2 & & -\widetilde{A}_2^T \end{array}\right]$$

and

$$\breve{H} = \begin{bmatrix} \widetilde{A}_1 & \widetilde{G}_1 \\ \widetilde{Q}_1 & -\widetilde{A}_1^T \end{bmatrix}$$

contains the desired eigenvalues. This can easily be achieved by J–orthogonal permutation matrices $\widetilde{P}$ of the form

$$\widetilde{P} = \operatorname{diag}(P, P), \qquad P = \begin{bmatrix} I_j & & & & \\ & & & I_k & \\ & & I_i & & \\ & I_p & & & \\ & & & & I_q \end{bmatrix}$$

for appropriate i, j, k, p, q as

$$\begin{bmatrix} P^T & \\ & P^T \end{bmatrix} \begin{bmatrix} A & G \\ Q & -A^T \end{bmatrix} \begin{bmatrix} P & \\ & P \end{bmatrix} = \begin{bmatrix} P^TAP & P^TGP \\ P^TQP & -P^TA^TP \end{bmatrix}$$

and

$$\begin{bmatrix} 0 & I_2 \\ I_1 & 0 \end{bmatrix} \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \begin{bmatrix} 0 & I_1 \\ I_2 & 0 \end{bmatrix} = \begin{bmatrix} B_2 & 0 \\ 0 & B_1 \end{bmatrix}$$

interchanges the diagonal blocks B_1 and B_2. Here, the size of the identity matrices I_1, I_2 is the same as that of B_1 and B_2.

Then postmultiplying (12) by $\breve{S}$,

$$HS^{2n,2m}\breve{S} = S^{2n,2m}\breve{S}\breve{S}^{-1}\widetilde{H}^{2m,2m}\breve{S} + \zeta_{m+1}v_{m+1}e_{2m}^T\breve{S},$$

yields a Hamiltonian Krylov-Schur-type decomposition

$$H\breve{S}^{2n,2m} = \breve{S}^{2n,2m}\breve{H}^{2m,2m} + \zeta_{m+1}v_{m+1}\breve{s}_{2m}^T$$

similar to the symplectic Lanczos factorization (12). Due to the special form of $\breve{H}^{2m,2m}$, the Hamiltonian Krylov-Schur-type decomposition can be partitioned in the form

$$H[\breve{S}^1\ \breve{S}^2\ \breve{S}^3\ \breve{S}^4] = [\breve{S}^1\ \breve{S}^2\ \breve{S}^3\ \breve{S}^4] \left[\begin{array}{cc|cc} \widetilde{A}_1 & & \widetilde{G}_1 & \\ & \widetilde{A}_2 & & \widetilde{G}_2 \\ \hline \widetilde{Q}_1 & & -\widetilde{A}_1^T & \\ & \widetilde{Q}_2 & & -\widetilde{A}_2^T \end{array}\right] + \zeta_{m+1}v_{m+1}\breve{s}_{2m}^T, \tag{14}$$

where

$$\breve{S}^1 = [v_1, \ldots, v_\ell],\ \breve{S}^2 = [v_{\ell+1}, \ldots, v_m],\ \breve{S}^3 = [w_1, \ldots, w_\ell],\ \breve{S}^4 = [w_{\ell+1}, \ldots, w_m]$$

if $\widetilde{A}_1, \widetilde{G}_1, \widetilde{Q}_1 \in \mathbb{R}^{\ell,\ell}$. Then with $\breve{s}_{2m}^T = [\breve{s}_{2m,1}, \ldots, \breve{s}_{2m,2m}]^T$ and

$$\breve{s}_{2l}^T = [\breve{s}_{2m,1}, \ldots, \breve{s}_{2m,\ell}, \breve{s}_{2m,m+1}, \ldots, \breve{s}_{2m,m+\ell}]^T$$

we see that

$$H\breve{S}^{2n,2\ell} = \breve{S}^{2n,2\ell}\breve{H} + \zeta_{m+1}v_{m+1}\breve{s}_{2\ell}^T \tag{15}$$

is also a Hamiltonian Krylov-Schur-type decomposition, where

$$\breve{S}^{2n,2\ell} = [\breve{S}^1\ \breve{S}^3] \qquad \text{and} \qquad \breve{H} = \left[\begin{array}{c|c} \widetilde{A}_1 & \widetilde{G}_1 \\ \hline \widetilde{Q}_1 & -\widetilde{A}_1^T \end{array}\right].$$

In other words, a Hamiltonian Krylov-Schur-type decomposition splits at any point where its Rayleigh quotient is block diagonal. By Theorem 4.1 there is an equivalent symplectic Lanczos factorization

$$H\breve{S}^{2n,2\ell} = \breve{S}^{2n,2\ell}\breve{H}^{2\ell,2\ell} + \breve{v}_{2\ell+1}e_{2\ell}^T$$

where $\check{H}^{2\ell,2\ell}$ is in Hamiltonian J–Hessenberg form and the columns of $\check{S}^{2n,2\ell}$ are J–orthogonal. Thus, the purging problem can be solved by applying the SR algorithm to $\widetilde{H}^{2k,2k}$, moving the unwanted Ritz values to the Hamiltonian submatrix

$$\left[\begin{array}{c|c} \widetilde{A}_2 & \widetilde{G}_2 \\ \hline \widetilde{Q}_2 & -\widetilde{A}_2^T \end{array}\right],$$

truncating the decomposition and returning to a symplectic Lanczos factorization. Restarting then becomes

1. expanding this symplectic Lanczos factorization,
2. computing the Hamiltonian Krylov-Schur-type decomposition,
3. moving the desired Ritz values to the top,
4. purging the rest of the decomposition, and
5. transforming the Hamiltonian Krylov-Schur-type decomposition back to a symplectic Lanczos one.

The symplectic Lanczos factorization achieved in this way is equivalent to the one the implicitly restarted symplectic Lanczos algorithm will achieve if the same Ritz values are discarded in both (and those Ritz values are distinct from the other Ritz values). The proof follows the lines of the proof of Theorem 3.1 in [38] or Theorem 2.4 of Chapter 5 in [49].

Remark 4.1. *From standard techniques of rounding error analysis it can be shown that as the Krylov-Schur-like restarted symplectic Lanczos algorithm proceeds, the computed J–orthogonal Hamiltonian Krylov-Schur type decomposition satisfies*

$$HS = SB + sb^T + F,$$

where the columns of S are J–orthogonal. Due to the Gauss transformations needed for the SR algorithm as well as for the row-wise reduction to J–Hessenberg form the norm of F is theoretically not of the order of the rounding error. In practice, of course, the condition number of the Gaussian transformations needed can be controlled. With $F = -ES$ and $E = FJ^nS^TJ^k$ this can be rewritten as

$$(H + E)S = SB + sb^T,$$

where

$$\frac{\|F\|}{\|S\|} \leq \|E\| \leq \|F\| \, \|S\|,$$

and $\|\,.\,\|$ denotes the spectral norm of a matrix. The upper bound follows from taking norms in the definition of $E = FJ^nS^TJ^K$ and the lower bound from taking norms in $F = -ES$. Due to the use of Gaussian transformations there is no a priori guarantee that the norms of F or S are small, but their growth can be monitored with significant computational overhead (see, e.g., [23, 25]) such that in practice, if small norms cannot be achieved (which is rarely the case), a warning can be issued.

5. Locking and Purging

As a Krylov-type iteration progresses, the Ritz estimates will converge at different rates. When a Ritz estimate is small enough, the corresponding Ritz value is said to have converged. The converged Ritz value may be wanted or unwanted. Unwanted ones can be deflated from the current factorization using the above procedure (in order to make sure that these eigenvalues do not creep back into our computations the corresponding columns of S must be kept and used in the J–orthogonalization step). Wanted ones should be deflated in the following sense in order to speed up convergence.

Assume that we have achieved a Hamiltonian Krylov-Schur-type decomposition as in (14),

$$\begin{aligned} & H[\breve{S}^1 \ \breve{S}^2 \ \breve{S}^3 \ \breve{S}^4] \qquad (16) \\ &= [\breve{S}^1 \ \breve{S}^2 \ \breve{S}^3 \ \breve{S}^4] \left[\begin{array}{cc|cc} \widetilde{A}_1 & & \widetilde{G}_1 & \\ & \widetilde{A}_2 & & \widetilde{G}_2 \\ \hline \widetilde{Q}_1 & & -\widetilde{A}_1^T & \\ & \widetilde{Q}_2 & & -\widetilde{A}_2^T \end{array}\right] + \zeta_{m+1} v_{m+1} [0 \ s_2^T \ 0 \ s_4^T], \end{aligned}$$

where we assume that the Ritz values contained in the Hamiltonian submatrix defined by $\widetilde{A}_1$, $\widetilde{G}_1$, $\widetilde{Q}_1$ (with the same partitioning as in (14)) have converged. Hence, zeroes appear in the vector $\breve{s}_{2m}$ at the corresponding positions so that

$$\begin{aligned} \breve{s}_{2m}^T &= [\underbrace{0, \ldots, 0}_{\ell}, \breve{s}_{2m,\ell+1}, \ldots, \breve{s}_{2m,m}, \underbrace{0, \ldots, 0}_{\ell}, \breve{s}_{2m,m+\ell+1}, \ldots, \breve{s}_{2m,2m}] \\ &=: [\, 0, s_2, 0, s_4 \,] \qquad \text{with} \quad s_2, s_4 \in \mathbb{R}^{m-\ell}. \end{aligned}$$

That is, with

$$\breve{S}^{2n,2\ell} = [\breve{S}^1 \ \breve{S}^3] \qquad \text{and} \qquad \breve{H}_1 = \left[\begin{array}{c|c} \widetilde{A}_1 & \widetilde{G}_1 \\ \hline \widetilde{Q}_1 & -\widetilde{A}_1 \end{array}\right]$$

we have $H\breve{S}^{2n,2\ell} = \breve{S}^{2n,2\ell}\breve{H}$, so that the columns of $\breve{S}^{2n,2\ell}$ span an eigenspace of H. We say a Hamiltonian Krylov-Schur-type decomposition has been deflated if it can be partitioned in this form. After deflation, equating the last $(m-\ell)$ columns of each of the two column blocks in (16) results in

$$H\breve{S} = \breve{S}\breve{H}_2 + \breve{v}_{m+1}[s_2^T \ s_4^T],$$

where

$$\breve{S} = [\breve{S}^2 \ \breve{S}^4] \qquad \text{and} \qquad \breve{H}_2 = \left[\begin{array}{c|c} \widetilde{A}_2 & \widetilde{G}_2 \\ \hline \widetilde{Q}_2 & -\widetilde{A}_2 \end{array}\right].$$

As Stewart [38, 49] points out, there are two advantages to deflating converged eigenspaces. First, by freezing it at the beginning of the Hamiltonian

Krylov-Schur-type decomposition, we insure that the remaining space of the decomposition remains J–orthogonal to it. In particular, this gives algorithms the opportunity to compute more than one independent eigenvector corresponding to a multiple eigenvalue.

The second advantage of the deflated decomposition is that we can save operations in the contraction phase of the Krylov-Schur-type cycle. Only the rightmost part of the Hamiltonian Krylov-Schur-type decomposition will be transformed back to a symplectic Lanczos factorization, yielding

$$H[\breve{S}^1\ \tilde{\tilde{S}}^2\ \breve{S}^3\ \tilde{\tilde{S}}^4] = [\breve{S}^1\ \tilde{\tilde{S}}^2\ \breve{S}^3\ \tilde{\tilde{S}}^4] \left[\begin{array}{cc|cc} \widetilde{A}_1 & & \widetilde{G}_1 & \\ & \widetilde{\widetilde{A}}_2 & & \widetilde{\widetilde{G}}_2 \\ \hline \widetilde{Q}_1 & & -\widetilde{A}_1^T & \\ & \widetilde{\widetilde{Q}}_2 & & -\widetilde{\widetilde{A}}_2^T \end{array}\right] + \tilde{\tilde{v}}_{m+1} e_{2m}^T.$$

The expansion phase does not change, and we end up with a decomposition of the form

$$\begin{aligned} H[\breve{S}^1\ \tilde{\tilde{S}}^2\ \tilde{\tilde{S}}^{2n}\ \breve{S}^3\ \tilde{\tilde{S}}^4\ \tilde{\tilde{S}}^{4n}] = \\ [\breve{S}^1\ \tilde{\tilde{S}}^2\ \tilde{\tilde{S}}^{2n}\ \breve{S}^3\ \tilde{\tilde{S}}^4\ \tilde{\tilde{S}}^{4n}] \left[\begin{array}{ccc|ccc} \widetilde{A}_1 & & & \widetilde{G}_1 & & \\ & \tilde{\tilde{A}}_2 & & & \tilde{\tilde{G}}_2 & \\ & & \tilde{\tilde{A}}_3 & & & \tilde{\tilde{G}}_3 \\ \hline \widetilde{Q}_1 & & & -\widetilde{A}_1^T & & \\ & \tilde{\tilde{Q}}_2 & & & -\tilde{\tilde{A}}_2^T & \\ & & \tilde{\tilde{Q}}_3 & & & -\tilde{\tilde{A}}_3^T \end{array}\right] \\ + \breve{v}_{m+p+1} e_{2(m+p)}^T. \end{aligned}$$

Since $\breve{H}_1$ is uncoupled from the rest of the Rayleigh quotient, we can apply all subsequent transformations exclusively to the eastern part of the Rayleigh quotient and to $[\tilde{\tilde{S}}^2\ \tilde{\tilde{S}}^4]$. If the order of $\breve{H}_1$ is small, the savings will be marginal; but as its size increases during the course of the algorithm, the savings become significant.

Figures 1 and 2 illustrate the process of locking an eigenvalue block. The red blocks are the ones to be locked. Once they are moved to the left upper corner we can work with the rest of the matrix (blue parts).

While the implicitly restarted symplectic Lanczos factorization (6) can restart with an arbitrary filter polynomial, the Krylov-Schur-type method discussed here cannot do that. When it comes to exact shifts the Krylov-Schur-type method is to be preferred because exchanging eigenvalues in a Schur-type form is a more reliable process than using implicit SR steps to deflate as swapping blocks can be obtained by permutations only.

6. Stopping Criteria

Now assume that we have performed k steps of the symplectic Lanczos method and thus obtained the identity

$$HS^{2n,2k} = S^{2n,2k} H^{2k,2k} + \zeta_{k+1} v_{k+1} e_{2k}^T.$$

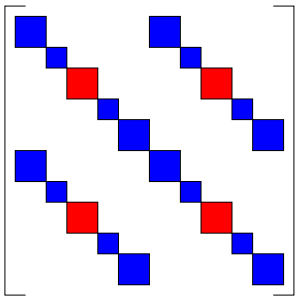


Figure 1: Eigenvalue corresponding to the red block is converged. Structure before swapping the red block

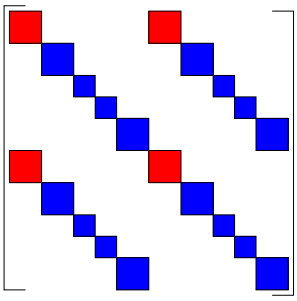


Figure 2: Structure after the red block is moved into the left upper corner

If the norm of the residual vector is small, the $2k$ eigenvalues of $H^{2k,2k}$ are approximations to the eigenvalues of H. Numerical experiments indicate that the norm of the residual rarely becomes small by itself. Nevertheless, some eigenvalues of $H^{2k,2k}$ may be good approximations to eigenvalues of H. Let θ be an eigenvalue of $H^{2k,2k}$ with the corresponding eigenvector y. Then the vector $x = S^{2n,2k}y$ satisfies

$$\|Hx - \theta x\| = \|(HS^{2n,2k} - S^{2n,2k}H^{2k,2k})y\| = |\zeta_{k+1}|\,|e_{2k}^T y|\,\|v_{k+1}\|. \qquad (17)$$

The vector x is referred to as Ritz vector and θ as Ritz value of H. If the last component of the eigenvector y is sufficiently small, the right-hand side of (17) is small and the pair $\{\theta, x\}$ is a good approximation to an eigenvalue-eigenvector pair of H. Thus, a small y_{2k} indidcates a possible deflation. Note that by Lemma 4.2 $|e_{2k}^T y| > 0$ if $H^{2k,2k}$ is unreduced. The pair (θ, x) is exact for the nearby problem

$$(H + E)x = \theta x \qquad \text{where} \quad E = -\zeta_{k+1} v_{k+1} e_k^T (S^{2n,2k})^T J^{2,2n},$$

as

$$\begin{aligned}(H+E)x &= (H+E)S^{2n,2k}y \\ &= S^{2n,2k}H^{2k,2k}y + \zeta_{k+1}v_{k+1}e_{2k}^T y + ES^{2n,2k}y \\ &= \theta x + \zeta_{k+1}v_{k+1}e_{2k}^T y + ES^{2n,2k}y.\end{aligned}$$

Note that a small $\|E\|$ is not sufficient for the Ritz pair $\{\theta, x\}$ being a good approximation to an eigenvalue-eigenvector pair of H. Thus we develop a more appropriate stopping criterion below. Also observe that the explicit formation of the residual $(HS^{2n,2k} - S^{2n,2k}H^{2k,2k})y$ can be avoided when deciding about the numerical accuracy of an approximate eigenpair, one can use the Ritz estimate $|\zeta_{k+1}|\,|e_{2k}^T y|\,\|v_{k+1}\|$ instead due to (17).

It is well-known that for non-normal matrices the norm of the residual of an approximate eigenvector is not by itself sufficient information to bound the error in the approximate eigenvalue. It is sufficient however to give a bound on

the distance to the nearest matrix to which the given approximation is exact. In the following, we will give a computable expression for the error. Assume that $H^{2k,2k}$ is diagonalizable

$$Y^{-1}H^{2k,2k}Y = \left[\begin{array}{ccc|ccc} -\theta_1 & & & & & \\ & \ddots & & & & \\ & & -\theta_k & & & \\ \hline & & & \theta_1 & & \\ & & & & \ddots & \\ & & & & & \theta_k \end{array}\right] =: \Theta;$$

Y can be chosen symplectic [50, 51]. Let $X = S^{2n,2k}Y = [x_1 \ldots, x_{2k}]$ and denote the residual term $\zeta_{k+1}v_{k+1}$ by r_{k+1}. Since $HS^{2n,2k} = S^{2n,2k}H^{2k,2k} + r_{k+1}e_{2k}^T$, it follows that

$$HS^{2n,2k}Y = S^{2n,2k}YY^{-1}H^{2k,2k}Y + r_{k+1}e_{2k}^TY$$

or $HX = X\Theta + r_{k+1}e_{2k}^TY$. Thus

$$Hx_i = -\theta_i x_i + y_{2k,i}r_{k+1} \quad \text{and} \quad Hx_{k+i} = \theta_i x_{k+i} + y_{2k,k+i}r_{k+1}$$

for $i = 1, \ldots, k$. From this, we can conclude a relation for the left eigenvectors corresponding to $\pm\theta$. Premultiplying

$$Hx_i = -\theta_i x_i + y_{2k,i}r_{k+1}$$

by J yields

$$JHx_i = -\theta_i Jx_i + y_{2k,i}Jr_{k+1}.$$

As H is Hamiltonian

$$(HJ)^Tx_i = -\theta_i Jx_i + y_{2k,i}Jr_{k+1},$$

and

$$H^T(Jx_i) = \theta_i(Jx_i) - y_{2k,i}Jr_{k+1}.$$

From this, we conclude

$$(Jx_i)^TH = \theta_i(Jx_i)^T - y_{2k,i}r_{k+1}^TJ.$$

Similarly, we obtain

$$(Jx_{k+i})^TH = -\theta_i(Jx_{k+i})^T + y_{2k,k+i}r_{k+1}^TJ.$$

Using Theorem 2' of [52] we obtain that $(-\theta_i, x_i, (Jx_{k+i})^T)$ is an eigen-triplet of $H - F_{-\theta_i}$ where

$$\begin{aligned} \|F_{-\theta_i}\|_2 &= \max\ \{\frac{\|r_{k+1}\|\ |y_{2k,i}|}{\|x_i\|_2}, \frac{\|r_{k+1}^TJ\|\ |y_{2k,k+i}|}{\|Jx_{k+i}\|_2}\} \\ &= \|r_{k+1}\| \max\ \{\frac{|y_{2k,i}|}{\|x_i\|_2}, \frac{|y_{2k,k+i}|}{\|x_{k+i}\|_2}\}. \end{aligned} \tag{18}$$

Furthermore, when $\|F_{-\theta_i}\|$ is small enough, then

$$|\lambda_i + \theta_j| \leq \mathrm{cond}(-\theta_j)\|F_{-\theta_i}\| + \mathcal{O}(\|F_{-\theta_i}\|^2),$$

where λ_i is an eigenvalue of H and $\mathrm{cond}(-\theta_j)$ is the condition number of the Ritz value $-\theta_j$

$$\mathrm{cond}(-\theta_j) = \frac{\|x_i\|\|Jx_{k+i}\|}{|x_{k+i}^T Jx_i|} = \frac{\|x_i\|\|x_{k+i}\|}{|x_{k+i}^T Jx_i|}.$$

Similarly, we obtain that $\{\theta_i, x_{k+i}, (Jx_i)^T\}$ is an eigen-triplet of $H - F_{\theta_i}$ where

$$\begin{aligned} \|F_{\theta_i}\|_2 &= \max_i \left\{ \frac{\|r_{k+1}\|\ |y_{2k,k+i}|}{\|x_{k+i}\|_2}, \frac{\|r_{k+1}^T J\|\ |y_{2k,i}|}{\|Jx_i\|_2} \right\} \\ &= \|r_{k+1}\| \max_i \left\{ \frac{|y_{2k,k+i}|}{\|x_{k+i}\|_2}, \frac{|y_{2k,i}|}{\|x_i\|_2} \right\}. \end{aligned} \tag{19}$$

Consequently, as θ_i and $-\theta_i$ are treated alike,

$$\|F_{-\theta_i}\|_2 = \|F_{\theta_i}\|_2.$$

The symplectic Lanczos algorithm should be continued until $\|F_{-\theta_i}\|_2$ is small, and until $\mathrm{cond}(-\theta_j)\|F_{-\theta_i}\|_2$ is below a given threshold for accuracy. Note that as in the Ritz estimate, in the criteria derived here the essential quantities are $|\zeta_{k+1}|$ and the last component of the desired eigenvectors $|y_{2k,i}|$ and $|y_{2k,k+i}|$.

7. Shift-and-invert techniques for the symplectic Lanczos method

As noted before, eigenvalues of real Hamiltonian matrices occur in pairs $\{\lambda, -\lambda\}$ or in quadruples $\{\lambda, -\lambda, \overline{\lambda}, -\overline{\lambda}\}$. A structure-preserving algorithm will extract entire pairs and quadruples intact. The symplectic Lanczos algorithm described above will, in general, compute approximations to a few of the largest eigenvalues of a Hamiltonian matrix H. Sometimes only a few of its smallest eigenvalues are needed. Since these are also the largest eigenvalues of H^{-1}, a Krylov subspace method can be applied to H^{-1} to find them. Since H^{-1} inherits the Hamiltonian structure of H, the symplectic Lanczos method is an appropriate method in the interest of efficiency, stability and accuracy. In situations where some prior information is given, one might prefer to use a shift before inverting. Specifically, if we know that the eigenvalues of interest lie near τ, we might prefer to work with $(H - \tau I)^{-1}$. Unfortunately, the shift destroys the Hamiltonian structure. In light of the symmetry of the spectrum, one might think of working with $(H-\tau I)^{-1}(H+\tau I)^{-1}$, in case τ is real or purely imaginary. All eigenvalues near to $\pm\tau$ are mapped simultaneously to values of large modulus. But this matrix is not Hamiltonian as well, it is skew-Hamiltonian. (This approach led to the development of SHIRA, a structure-preserving Arnoldi algorithm for Hamiltonian matrices [15]). The Cayley transform $(H-\tau I)^{-1}(H+\tau I)$

might come to mind next, but this matrix is symplectic (and would require the appropriate symplectic Lanczos process described in [34]). In order to apply the Hamiltonian Krylov-Schur-like method developed in the previous sections, we need to stay within the Hamiltonian structure. This is accomplished when working with the Hamiltonian matrices

$$H_1 = H^{-1}(H - \tau I)^{-1}(H + \tau I)^{-1} = (H^3 - \tau^2 H)^{-1},$$

or

$$H_2 = H(H - \tau I)^{-1}(H + \tau I)^{-1} = (H - \tau^2 H^{-1})^{-1}, \tag{20}$$

for example (see also [15, 30]). Although these expressions look fairly complicated, the action of the encoded operators on a vector can often be implemented in a very efficient way in most interesting applications: frequently, only one sparse LU decomposition of size $n \times n$ is necessary to apply the operators, see [14, 15] and Section 8 for some examples.

In order to obtain the eigenvalues λ of H from the eigenvalues ϖ of these shifted Hamiltonian matrices, a cubic polynomial equation

$$\lambda^3 - \tau^2 \lambda - \frac{1}{\varpi} = 0$$

has to be solved in case H_1 is used, while a quadratic polynomial equation

$$\lambda^2 - \frac{1}{\varpi}\lambda - \tau^2 = 0 \tag{21}$$

has to be solved in case H_2 is used. In case a complex shift σ is used, we can work with the Hamiltonian matrix

$$\begin{aligned} H_3 &= H^{-1}(H - \sigma I)^{-1}(H + \sigma I)^{-1}(H - \overline{\sigma} I)^{-1}(H + \overline{\sigma} I)^{-1} \\ &= (H^5 - (\overline{\sigma}^2 + \sigma^2)H^3 + |\sigma|^4 H)^{-1} \end{aligned}$$

or

$$\begin{aligned} H_4 &= H(H - \sigma I)^{-1}(H + \sigma I)^{-1}(H - \overline{\sigma} I)^{-1}(H + \overline{\sigma} I)^{-1} \\ &= (H^3 - (\overline{\sigma}^2 + \sigma^2)H + |\sigma|^4 H^{-1})^{-1}. \end{aligned} \tag{22}$$

Similar as before, in order to obtain the eigenvalues λ of H from the eigenvalues of the shifted matrices, polynomial equations of order five or four have to be solved: in case H_3 is used, this is

$$\lambda^5 - (\overline{\sigma}^2 + \sigma^2)\lambda^3 + |\sigma|^4 \lambda - \frac{1}{\varpi} = 0$$

while for H_4, we need to solve

$$\lambda^4 - (\overline{\sigma}^2 + \sigma^2)\lambda^2 - \frac{1}{\varpi}\lambda + |\sigma|^4 = 0.$$

Let us consider the case H_2 more closely. The eigenvalues λ of H are mapped to

$$\varpi = \frac{\lambda}{\lambda^2 - \tau^2}.$$

No matter whether $\tau \in \mathbb{R}$ or $\tau \in i\mathbb{R}$, τ^2 is always real. Hence, a real λ is mapped onto a real ϖ, a purely imaginary λ onto a purely imaginary ϖ and a complex λ onto a complex ϖ. Eigenvectors stay invariant, $Hx = \lambda x$ implies $H^{-1}x = \frac{1}{\lambda}x$ and therefore $H_2 x = \varpi x$ as

$$H_2^{-1}x = (H - \tau^2 H^{-1})x = (\lambda - \frac{\tau^2}{\lambda})x = \frac{1}{\varpi}x.$$

Unfortunately, two distinct eigenvalues λ_1 and λ_2 of H can be mapped to the same eigenvalue ϖ of H_2 by an unlucky choice of τ. Whenever $\tau^2 = -\lambda_1\lambda_2$ is chosen, this is the case (Please note, that τ can be real or purely imaginary, hence τ^2 can be a negative real. Moreover, λ_1 and λ_2 might both be real or purely imaginary, hence the above equation can be fulfilled.)

Applying the symplectic Lanczos process to H_2 yields eigenvalues of the matrix H_2, but we actually want to compute eigenvalues of H. A straightforward approach to compute the eigenvalues λ of H from the eigenvalues ϖ of H_2 is to solve the quadratic equation (21). It has the solution

$$\lambda_{1,2} = \frac{1}{2\varpi} \pm \sqrt{\frac{1}{4\varpi^2} + \tau^2}. \tag{23}$$

Unfortunately, only one of these solutions corresponds to an eigenvalue of H. In order to decide which one is correct, let us assume that the symplectic Lanczos process is run to achieve a negligible ζ_{k+1},

$$H_2 S^{2n,2k} \approx S^{2n,2k}\widetilde{H}^{2k,2k} \tag{24}$$

(and, that H is nonderogatory). The space spanned by the vectors

$$\{v_1, w_1, \ldots, v_k, w_k\}$$

is, up to rounding errors, an invariant subspace of H_2. Normally it is also invariant under H as H_2 is a rational function of H. The space spanned by $\{v_1, w_1, \ldots, v_k, w_k\}$ can fail to be invariant under H only if two distinct eigenvalues of H are mapped to the same eigenvalue of H_2. Let us assume for the moment that the shift τ is chosen such that this does not happen; that is $\tau^2 \neq -\lambda_1\lambda_2$ for all eigenvalues λ_1, λ_2 of H. If the SR algorithm is used to compute the eigenvalues and eigenvectors of $\widetilde{H}^{2k,2k}$ so that

$$\widetilde{H}^{2k,2k}\widetilde{S} = \widetilde{S}\widehat{H},$$

then the eigenvalues λ_j of H can be obtained via (23). In order to decide, which of the two possible solutions to choose, we can now check the residual

$$\|H\widehat{s}_j - \widehat{s}_j\lambda_j\|_F$$

where $\widehat{s}_j$ denotes the jth column of $\widehat{S} = S^{2n,2k}\widetilde{S}$ and λ_j has been obtained using the eigenvalue ϖ of $\widehat{H}$ corresponding to the jth column of $\widetilde{S}$. In case the residual is small, λ_j can be accepted as an eigenvalue of H.

A different approach circumventing the difficulties with the above approach in order to determine the eigenvalues of H from the Lanczos recursion (24) is to calculate the Ritz values of H with respect to the space spanned by the vectors $\{v_1, w_1, \ldots, v_k, w_k\}$ [15]; that is, we calculate the eigenvalues λ_i of

$$X = J_k (S^{2n,2k})^T J_n H S^{2n,2k}.$$

As X is Hamiltonian, but not of J–Hessenberg form, it is suggested to compute its eigenvalues by the numerically backward stable structure-preserving HAPACK routine `haeig` [53]. Moreover, the residual

$$\| H S^{2n,2k} - S^{2n,2k} X \|_F$$

can be used to check whether or not the space spanned by the vectors $\{v_1, w_1, \ldots, v_k, w_k\}$ really is invariant under H. Hence, this approach can be used in order to detect if an unlucky shift τ has been chosen.

8. Numerical Examples

In this section, we report the results of numerical experiments obtained with the Krylov-Schur-type method for Hamiltonian eigenproblems. All experiments are performed in MATLAB R2006a using double precision on a Pentium M notebook with 512 MB main memory.

The accuracy of computed eigenvalues and eigenvectors is compared using relative residuals

$$\frac{\| Q(\tilde{\theta}) \tilde{x} \|}{\| Q(\tilde{\theta}) \|}, \tag{25}$$

where $(\tilde{\theta}, \tilde{x})$ is a computed Ritz pair and $Q(\theta) = H - \theta I$ (unless stated otherwise).

It should be noted that the Hamiltonian Krylov-Schur-type method exhibits nice properties for certain quadratic eigenproblems with Hamiltonian symmetry, particularly for stable gyroscopic systems. As this case is extensively studied already in [14] (where the theory from a preliminary version of this paper has been employed), we refrain here from showing examples for quadratic eigenproblems and concentrate on eigenproblems from other application areas.

8.1. Heat transfer equation (HEAT)

The data of this example come from the autonomous linear-quadratic optimal control problem of one dimensional heat flow and is taken from [54, Example 18] (see also [55]). The problem is described in terms of infinite-dimensional operators on a Hilbert space. Using a standard finite element approach based on linear B-splines, a finite dimensional approximation may be obtained by the solution of Hamiltonian eigenproblems. The Hamiltonian matrix H is of the following form:

$$H = \begin{bmatrix} M_N^{-1} K_N & -B B^T \\ -C^T C & -\left(M_N^{-1} K_N \right)^T \end{bmatrix}$$

with

$$M_N = \frac{1}{6N}\begin{bmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{bmatrix},\ K_N = -\alpha N \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix},$$

where $B = M_N^{-1} b_N$ and $C = c_N^T$, $M_N, K_N \in \mathbb{R}^{N\times N}$. The vectors b_N and c_N can be obtained by integrating the B-Spline basis functions $\{\phi_i^N\}_{i=1}^{N-1}$ for the chosen finite dimensional subspace of the underlying Hilbert space

$$\begin{aligned} (b_N)_j &= \int_0^1 \beta(s)\phi_i^N(s)ds \qquad j = 1,\ldots,N-1, \\ (c_N)_j &= \int_0^1 \gamma(s)\phi_i^N(s)ds \qquad j = 1,\ldots,N-1, \end{aligned}$$

where the functions $\beta, \gamma \in L_2(0,1)$ are given by

$$\begin{aligned} \beta(s) &= \begin{cases} 1, & s \in [0.1, 0.5], \\ 0, & \text{otherwise}, \end{cases} \\ \gamma(s) &= \begin{cases} 1, & s \in [0.1, 0.5], \\ 0, & \text{otherwise}. \end{cases} \end{aligned}$$

In the computational experiments reported below, we set $\alpha = 0.05$ (default in [55]). Making use of the special form of H we can set up efficient operators for computing Hx, $H^{-1}x$, H_2x (20) or H_4x (22) for an arbitrary vector x, so that we are able to apply the Hamiltonian Krylov-Schur method to one of these operators in order to find a few of the largest or smallest eigenvalues or eigenvalues close to a shift target. Here we will report on numerical experiments in which we asked for a few of the smallest eigenvalues. For this purpose, we use H^{-1} which can be efficiently implemented using as the only "inverse operation" the banded Cholesky factorization of K_N based on the following considerations: note that H can be written as (using the symmetry of M_N, K_N)

$$H = \begin{bmatrix} M_N^{-1} & \\ & I_n \end{bmatrix} \left(\begin{bmatrix} K_N & \\ & -K_N \end{bmatrix} + UV^T \right) \begin{bmatrix} I_N & \\ & M_N^{-1} \end{bmatrix}$$

with the rank-2 update

$$UV^T = \begin{bmatrix} -b_N & \\ & c_N \end{bmatrix} \begin{bmatrix} & -c_N \\ b_N & \end{bmatrix}^T.$$

Hence, using the Sherman-Morrison-Woodbury formula (see, e.g., [39, Section 2.1.3])

$$H^{-1} = \begin{bmatrix} I_N & \\ & M_N \end{bmatrix} \left(\begin{bmatrix} K_N^{-1} & \\ & -K_N^{-1} \end{bmatrix} + \hat{U} W^{-1} \hat{V}^T \right) \begin{bmatrix} M_N & \\ & I_n \end{bmatrix},$$

where

$$\hat{U} = \begin{bmatrix} & K_N^{-1}b_N \\ K_N^{-1}c_N & \end{bmatrix}, \quad \hat{V} = \begin{bmatrix} & K_N^{-1}c_N \\ K_N^{-1}b_N & \end{bmatrix},$$
$$W = \begin{bmatrix} 1 & -b_N^T K_N^{-1}c_N \\ -c_N^T K_N^{-1}b_N & 1 \end{bmatrix}.$$

Hence, application to a vector x requires besides the precomputed tridiagonal Cholesky factorization of K_N two sparse matrix-vector products with M_N, two forward-backward solves with the Cholesky factor of K_N plus a few dot products and solving a 2-by-2 linear system with W as coefficient matrix. (Note that $K_N^{-1}b_N, K_N^{-1}c_N$ can be precomputed.)

For the reported example we choose $N = 2000$, that is $H \in \mathbb{R}^{4000\times 4000}$. We compare the Hamiltonian Krylov-Schur-like symplectic Lanczos method (called KRSCHUR) and MATLAB's `eigs`. Both algorithms use the same starting vector and a tolerance of 10^{-10}. We are looking for 6 eigenpairs (that is 12 eigenvalues) in a search space of size 24. In the Tables 1 and 2 the computed eigenvalues and the associated residual (25) is given for the computed eigenvalues with negative real part and the computed eigenvectors. The KRSCHUR algorithm was slightly faster than `eigs` in terms of numbers of iterations needed to converge. The residuals for KRSCHUR are slightly larger than those for `eigs`, in both cases one step of inverse iteration reduces the residual to machine precision (last column in the Tables). For other choices of the number of eigenvalues to be computed and size of the search space similar results are obtained.

Eigenvalue	Residual	Residual with refined eigenvector
-0.53742837879709	$2.2 \cdot 10^{-12}$	$2.9 \cdot 10^{-16}$
-1.99375748667056	$6.3 \cdot 10^{-12}$	$5.0 \cdot 10^{-17}$
-4.44183939202748	$1.5 \cdot 10^{-11}$	$2.7 \cdot 10^{-16}$
-7.89595335914986	$2.3 \cdot 10^{-11}$	$5.4 \cdot 10^{-17}$
-12.33706885545842	$1.2 \cdot 10^{-11}$	$1.2 \cdot 10^{-16}$
-17.76547171343604	$2.1 \cdot 10^{-11}$	$4.1 \cdot 10^{-17}$

Table 1: KRSCHUR results for HEAT after 2 iterations with a maximal condition number of 212.22

8.2. A Semi-discretized Heat Transfer Problem for Optimal Cooling of Steel Profiles (STEEL)

The problem of optimal cooling of steel profiles arises in a rolling mill when different steps in the production process require different temperatures of the raw material. To achieve a high production rate, economical interests suggest to reduce the temperature as fast as possible to the required level before entering the next production phase. At the same time, the cooling process, which is realized by spraying cooling fluids on the surface, has to be controlled so that

Eigenvalue	Residual	Residual with refined eigenvector
-0.53742837811615	$1.3 \cdot 10^{-14}$	$9.1 \cdot 10^{-17}$
-1.99375748661981	$4.8 \cdot 10^{-14}$	$1.4 \cdot 10^{-16}$
-4.44183939138648	$1.0 \cdot 10^{-13}$	$2.3 \cdot 10^{-16}$
-7.89595335914068	$1.8 \cdot 10^{-13}$	$5.3 \cdot 10^{-17}$
-12.33706885551394	$1.7 \cdot 10^{-13}$	$5.4 \cdot 10^{-17}$
-17.76547171346281	$3.2 \cdot 10^{-13}$	$4.4 \cdot 10^{-17}$

Table 2: *eigs* results for HEAT after 3 iterations.

material properties, such as durability or porosity, achieve given quality standards. Large gradients in the temperature distributions of the steel profile may lead to unwanted deformations, brittleness, loss of rigidity, and other undesirable material properties, see [56] for more details and further references on the mathematical model and its discretization.

For the reported example we choose $n = 20209$, that is $H \in \mathbb{R}^{40418 \times 40418}$. Again we compare KRSCHUR and MATLAB's `eigs`. Both algorithms use the same starting vector and a tolerance of 10^{-10}. We ask again for 6 eigenpairs (that is 12 eigenvalues) in a search space of size 24. In the Tables 3 and 4 the computed eigenvalues and the associated residual (25) is given for the computed eigenvalues with negative real part and the computed eigenvectors. As we do not have H at hand in order to compute its 1-norm, we use a 1-norm-condition estimator as explained in [57, Algorithm 14.3]. Both algorithms need the same number of iterations in order to converge. The residuals for KRSCHUR are slightly larger than those for `eigs`, in both cases one step of inverse iteration reduces the residual to machine precision. For other choices of the number of eigenvalues to be computed and size of the search space similar results are obtained.

Eigenvalue $\cdot 10^{-3}$	Residual
-0.01807591600155	$1.1 \cdot 10^{-13}$
-0.03087837032047	$4.3 \cdot 10^{-13}$
-0.08814494716421	$5.5 \cdot 10^{-14}$
-0.19258460926318	$9.5 \cdot 10^{-13}$
-0.26388595299809	$8.1 \cdot 10^{-13}$
-0.33668742939977	$1.2 \cdot 10^{-11}$

Table 3: KRSCHUR results for STEEL after 3 iterations with a maximal condition number of 572.65

Eigenvalue $\cdot 10^{-3}$	Residual
-0.01807591600154	$8.1 \cdot 10^{-17}$
-0.03087837032049	$1.5 \cdot 10^{-16}$
-0.08814494716419	$1.4 \cdot 10^{-16}$
-0.19258460926304	$2.5 \cdot 10^{-16}$
-0.26388595299811	$3.8 \cdot 10^{-16}$
-0.33668742939988	$2.1 \cdot 10^{-15}$

Table 4: *eigs* results for STEEL after 3 iterations.

8.3. Random Phase Approximation (RPA)

Random phase approximation (RPA) is a popular technique in computational (quantum) chemistry. It can be considered as part of time-dependent Hartree-Fock (or density functional) theory and is used for calculating excitation energies. For determining the excitation spectra, RPA requires the solution of the Hamiltonian eigenproblem

$$Hx := \begin{bmatrix} A & B \\ -B & -A \end{bmatrix} x = \lambda x, \quad A = A^T,\ B = B^T,$$

where the eigenvalues of smallest magnitude of H are required [20, 21, 22].

For the results reported in this section, we use matrices A, B related to direct RPA for the fluorine dimer F_2 provided by Tom Henderson (Department of Chemistry, Rice University, Houston). In direct RPA, B is positive definite and the eigenvalues of H are all real. For this data set, we have $n = 2484$, that is $H \in \mathbb{R}^{4968\times 4968}$. As before, we compare KRSCHUR and MATLAB's `eigs`. Both algorithms use the same starting vector. For applying the operator H^{-1}, we use a sparse LU decomposition of H. We are looking for 6 eigenpairs (that is 12 eigenvalues) in a search space of size 24. As before the computed eigenvalues and the associated residual (25) is given for the computed eigenvalues with negative real part and the computed eigenvectors. Note that the residuals reported in Tables 5 and 7 are sometimes slightly larger than the prescribed convergence tolerance. The reason for this lies in the fact that in KRSCHUR, residuals are evaluated on the basis of the Ritz estimates while the reported residuals are computed using H and the computed Ritz pair.

In the first set of experiments, we set the convergence tolerance for the residuals of the Ritz pairs to 10^{-10} . With this convergence tolerance, KRSCHUR needs 3 iterations less than `eigs` in order to converge. The KRSCHUR residuals shown in Table 5 are slightly larger than those for `eigs` reported in Table 6, in both cases one step of inverse iteration reduces the residual to machine precision. Note that `eigs` computes a complex conjugate pair of Ritz values. Though the imaginary parts are small in magnitude, this contradicts the theoretically known fact that all eigenvalues of H are real.

Eigenvalue	Residual
-0.781645350746793	$4.9 \cdot 10^{-12}$
-0.781645350746791	$6.1 \cdot 10^{-12}$
-0.811136520635671	$1.6 \cdot 10^{-10}$
-0.811136520635672	$2.5 \cdot 10^{-11}$
-0.874712875414053	$9.7 \cdot 10^{-13}$
-0.881337321551920	$6.6 \cdot 10^{-12}$

Table 5: KRSCHUR results for RPA with tol = 10^{-10} after 10 iterations with a maximal condition number of 9.28.

Eigenvalue	Residual
-0.781645350746785	$3.1 \cdot 10^{-14}$
-0.781645350746800	$2.0 \cdot 10^{-14}$
$-0.811136520635689 - 0.000000000000015i$	$1.1 \cdot 10^{-10}$
$-0.811136520635689 + 0.000000000000015i$	$1.1 \cdot 10^{-10}$
-0.874712875414049	$5.1 \cdot 10^{-14}$
-0.881337321551929	$7.2 \cdot 10^{-13}$

Table 6: *eigs* results for RPA with tol = 10^{-10} after 13 iterations.

If we decrease the tolerance to 10^{-12}, no complex eigenvalues show up any longer in the `eigs` computations, but now it needs about twice as many iterations as KRSCHUR to converge! The results are shown in Tables 7 and 8, respectively.

Eigenvalue	Residual
-0.781645350746794	$1.1 \cdot 10^{-12}$
-0.781645350746797	$1.6 \cdot 10^{-14}$
-0.781645350746803	$6.7 \cdot 10^{-14}$
-0.811136520635681	$8.2 \cdot 10^{-14}$
-0.811136520635665	$9.7 \cdot 10^{-14}$
-0.874712875414078	$9.4 \cdot 10^{-14}$

Table 7: KRSCHUR results for RPA with tol = 10^{-12} after 16 iterations with a maximal condition number of 15.83.

9. Conclusions

We have derived a Krylov-Schur-type method for the Hamiltonian eigenproblem. The main ingredients are the symplectic Lanczos process for extending the Krylov space and the SR algorithm for obtaining a Schur-like form for the Hamiltonian Rayleigh quotient. The Krylov-Schur technique allows a simpler restart-

Eigenvalue	Residual
-0.781645350746783	$6.0 \cdot 10^{-14}$
-0.781645350746784	$1.4 \cdot 10^{-12}$
-0.781645350746796	$4.1 \cdot 10^{-13}$
-0.781645350746797	$2.7 \cdot 10^{-14}$
-0.811136520635683	$3.2 \cdot 10^{-16}$
-0.811136520635672	$6.8 \cdot 10^{-16}$

Table 8: *eigs* results for RPA with tol = 10^{-12} after 30 iterations.

ing mechanism for exact shifts as compared to the previously employed implicit restarting using polynomial filters. In particular, locking, purging, and deflation can easily be incorporated in the Krylov-Schur variant while these techniques turn out to be prohibitively difficult to realize in the implicit restarting variant of the symplectic Lanczos process. Though nonorthogonal transformations are employed in the method, the resulting method turns out to be fairly robust in practice. A significant advantage over general purpose eigensolvers for nonsymmetric eigenproblems is obtained from the fact the Hamiltonian symmetry of the eigenvalues is exploited and preserved. Thus, physical meaningfully paired Ritz values are computed in contrast to, e.g., the standard (shift-and-invert) Arnoldi method. Numerical examples demonstrate that the accuracy of the eigenvalue approximations obtained by the Krylov-Schur-like method is comparable to the Arnoldi process. Besides the advantage of a correct pairing of the eigenvalue approximations, also the number of restarts needed to converge is often lower (sometimes significantly) than for implicitly restarted Arnoldi method.

Acknowledgments

We would like to express our thanks to Thomas M. Henderson (Department of Chemistry, Rice University, Houston) for providing the data used in Section 8.3.

References

[1] P. Benner, Computational methods for linear-quadratic optimization, Supplemento ai Rendiconti del Circolo Matematico di Palermo, Serie II No. 58 (1999) 21–56.

[2] V. Mehrmann, The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution, no. 163 in Lecture Notes in Control and Information Sciences, Springer-Verlag, Heidelberg, 1991.

[3] V. Sima, Algorithms for Linear-Quadratic Optimization, Vol. 200 of Pure and Applied Mathematics, Marcel Dekker, Inc., New York, NY, 1996.

[4] E. Sontag, Mathematical Control Theory, 2nd Edition, Springer-Verlag, New York, NY, 1998.

[5] K. Zhou, J. Doyle, K. Glover, Robust and Optimal Control, Prentice-Hall, Upper Saddle River, NJ, 1996.

[6] S. Boyd, V. Balakrishnan, P. Kabamba, A bisection method for computing the H_∞ norm of a transfer matrix and related problems, Math. Control, Signals, Sys. 2 (1989) 207–219.

[7] J. Burke, A. Lewis, M. Overton, Robust stability and a criss-cross algorithm for pseudospectra, IMA J. Numer. Anal. 23 (2003) 359–375.

[8] R. Byers, A bisection method for measuring the distance of a stable to unstable matrices, SIAM J. Sci. Statist. Comput. 9 (1988) 875–881.

[9] A. Antoulas, A new result on positive real interpolation and model reduction, Systems & Control Letters 54 (2005) 361–374.

[10] P. Benner, H. Faßbender, Computing passive reduced-order LTI models using structured Krylov subspace methods, Preprint, TU Braunschweig, Germany (2006).

[11] K. Ito, K. Kunisch, Reduced order control based on approximate inertial manifolds, Linear Algebra Appl. 415 (2–3) (2006) 531–541.

[12] D. Sorensen, Passivity preserving model reduction via interpolation of spectral zeros, Systems & Control Letters 54 (2005) 347–360.

[13] N. Wong, V. Balakrishnan, C.-K. Koh, Passivity-preserving model reduction via a computationally efficient project-and-balance scheme, in: Proc. Design Automation Conference, San Diego, CA, 2004, pp. 369–374.

[14] P. Benner, H. Faßbender, M. Stoll, Solving large-scale quadratic eigenvalue problems with Hamiltonian eigenstructure using a structure-preserving Krylov subspace method, Electron. Trans. Numer. Anal. 29 (2007/08) 212–229.

[15] V. Mehrmann, D. Watkins, Structure-preserving methods for computing pairs of large sparse skew-Hamiltonian/Hamiltonian pencils, SIAM J. Sci. Statist. Comput. 22 (2001) 1905–1925.

[16] C. Pester, Hamiltonian eigenvalue symmetry for quadratic operator eigenvalue problems, J. Integral Equations Appl. 17 (1) (2005) 71–89.

[17] F. Tisseur, K. Meerbergen, The quadratic eigenvalue problem, SIAM Review 43 (2001) 235–286.

[18] T. Eirola, Krylov integrators for Hamiltonian systems, Workshop on Exponential Integrators, October 20-23, 2004 Innsbruck, Austria, available at `http://techmath.uibk.ac.at/numbau/alex/events/files04/slides/timo.pdf` (2004).

[19] L. Lopez, V. Simoncini, Preserving geometric properties of the exponential matrix by block Krylov subspace methods, BIT 46 (2006) 813–830.

[20] M. Lucero, A. Niklasson, S. Tretiak, M. Challacombe, Molecular-orbital-free algorithm for excited states in time-dependent perturbation theory, J. Chem. Phy. 129 (6) (2008) 064114–1–064114–8.

[21] G. Scuseria, T. Henderson, D. Sorensen, The ground state correlation energy of the random phase approximation from a ring coupled cluster doubles approach, J. Chem. Phy. 129 (23) (2008) 231101–1–231101–4.

[22] S. Tretiak, C. Isborn, A. Niklasson, M. Challacombe, Representation independent algorithms for molecular response calculations in time-dependent self-consistent field theories, J. Chem. Phy. 130 (5) (2009) 054111–1–054111–16.

[23] P. Benner, H. Faßbender, An implicitly restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem, Linear Algebra Appl. 263 (1997) 75–111.

[24] W. Ferng, W.-W. Lin, C.-S. Wang, The shift-inverted J-Lanczos algorithm for the numerical solutions of large sparse algebraic Riccati equations, Comput. Math. Appl. 33 (1997) 23–40.

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

[26] H. Faßbender, The parametrized SR algorithm for Hamiltonian matrices, Electron. Trans. Numer. Anal. 26 (2007) 121–145.

[27] D. Calvetti, L. Reichel, D. Sorensen, An implicitly restarted Lanczos method for large symmetric eigenvalue problems, Electr. Trans. Num. Anal. 2 (1994) 1–21.

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

[29] D. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, SIAM J. Matrix Anal. Appl. 13 (1) (1992) 357–385.

[30] D. Watkins, On Hamiltonian and symplectic Lanczos processes, Linear Algebra Appl. 385 (2004) 23–45.

[31] 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).

[32] R. Freund, V. Mehrmann, A symplectic look-ahead Lanczos algorithm for the Hamiltonian eigenvalue problem, manuscript (1994).

[33] T. Apel, V. Mehrmann, D. Watkins, Structured eigenvalue methods for the computation of corner singularities in 3d anisotropic elastic structures, Comput. Methods Appl. Mech. Engrg. 191 (2002) 4459–4473.

[34] P. Benner, H. Faßbender, An implicitly restarted Lanczos method for the symplectic eigenvalue problem, SIAM J. Matrix Anal. Appl. 22 (2000) 682–713.

[35] R. Lehoucq, D. Sorensen, Deflation techniques for an implicitly restarted Arnoldi iteration, SIAM J. Matrix Anal. Appl. 17 (1996) 789–821.

[36] D. Sorensen, Deflation for implicitly restarted Arnoldi methods, Tech. rep., Department of Computational and Applied Mathematics, Rice University, Houston, Texas (1998).

[37] D. Sorensen, Numerical methods for large eigenvalue problems, Acta Numerica (2002) 519–584.

[38] G. Stewart, A Krylov-Schur algorithm for large eigenproblems, SIAM J. Matrix Anal. Appl. 23 (3) (2001) 601–614.

[39] G. Golub, C. Van Loan, Matrix Computations, 3rd Edition, Johns Hopkins University Press, Baltimore, 1996.

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

[41] H. Faßbender, A detailed derivation of the parameterized SR algorithm and the symplectic Lanczos method for Hamiltonian matrices, internal report, TU Braunschweig, Institut *Computational Mathematics*, available from `http://www.icm.tu-bs.de/~hfassben/papers/ham_evp.pdf`. (2006).

[42] P. Benner, Structured Krylov subspace methods for eigenproblems with spectral symmetries, Workshop on Theoretical and Computational Aspects of Matrix Algorithms, Schloß Dagstuhl, October 12-17, 2003, available from `http://www.tu-chemnitz.de/~benner/talks/dagstuhl03.pdf`. (Oct. 2003).

[43] J. Francis, The QR transformation, Part I and Part II, Comput. J. 4 (1961) 265–271 and 332–345.

[44] M. Stoll, Locking und Purging für den Hamiltonischen Lanczos-Prozess, Diplomarbeit, TU Chemnitz, Fakultät für Mathematik, Germany (2005).

[45] G. Golub, C. Van Loan, Matrix Computations, 3rd Edition, Johns Hopkins University Press, Baltimore, 1996.

[46] C. Paige, C. Van Loan, A Schur decomposition for Hamiltonian matrices, Linear Algebra Appl. 41 (1981) 11–32.

[47] A. Bunse-Gerstner, V. Mehrmann, D. Watkins, An SR algorithm for Hamiltonian matrices based on Gaussian elimination, Methods of Operations Research 58 (1989) 339–356.

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

[49] G. Stewart, Matrix Algorithms, Volume II: Eigensystems, SIAM , Philadelphia, USA, 2001.

[50] A. Laub, K. Meyer, Canonical forms for symplectic and Hamiltonian matrices, Celestial Mechanics 9 (1974) 213–238.

[51] W.-W. Lin, V. Mehrmann, H. Xu, Canonical forms for Hamiltonian and symplectic matrices and pencils, Linear Algebra Appl. 302-303 (1999) 469–533.

[52] W. Kahan, B. Parlett, E. Jiang, Residual bounds on approximate eigensystems of nonnormal matrices, SIAM J. Numer. Anal. 19 (1982) 470–484.

[53] P. Benner, D. Kressner, Algorithm 854: Fortran 77 subroutines for computing the eigenvalues of Hamiltonian matrices, ACM Transactions on Mathematical Software 32 (2) (2006) 352–373.

[54] P. Benner, A. Laub, V. Mehrmann, A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case, Tech. Rep. SPC 95_22, Fakultät für Mathematik, TU Chemnitz–Zwickau (1995).

[55] J. Abels, P. Benner, CAREX – a collection of benchmark examples for continuous-time algebraic Riccati equations (version 2.0), SLICOT Working Note 1999-14, available from `www.slicot.de` (Nov. 1999).

[56] P. Benner, J. Saak, A semi-discretized heat transfer model for optimal cooling of steel profiles, in: P. Benner, V. Mehrmann, D. Sorensen (Eds.), Dimension Reduction of Large-Scale Systems, Vol. 45 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, Berlin/Heidelberg, Germany, 2005, pp. 353–356.

[57] N. Higham, Accuracy and Stability of Numerical Algorithms, SIAM Publications, Philadelphia, PA, 1996.

RECENT REPORTS

04/09	Stability analysis of reaction-diffusion systems with timem-dependent coefficients on growing domains	Madzvamuse Gaffney Maini
05/09	Onsager reciprocity in premelting solids	Peppin Spannuth Wettlaufer
06/09	Inherent noise can facilitate coherence in collective swarm motion	Yates *et al.*
07/09	Solving the Coupled System Improves Computational Efficiency of the Bidomain Equations	Southern Plank Vigmond Whiteley
08/09	Model reduction using a posteriori analysis	Whiteley
09/09	Equilibrium Order Parameters of Liquid Crystals in the Laudau-De Gennes Theory	Majumdar
10/09	Landau-De Gennes theory of nematic liquid crystals: the Oseen-Frank limit and beyond	Majumdar Zarnescu
11/09	A Comparison of Numerical Methods used for Finite Element Modelling of Soft Tissue Deformation	Pathmanathan Gavaghan Whiteley
12/09	From Individual to Collective Behaviour of Unicellular Organisms: Recent Results and Open Problems	Xue Othmer Erban
13/09	Stochastic modelling of reaction-diffusion processes: algorithms for bimolecular reactions	Erban Chapman
14/09	Chaste: a test-driven approach to software development for physiological modelling	Pitt-Francis *et al.*
15/09	Block triangular preconditioners for PDE constrained optimization	Rees Stoll
16/09	From microscopic to macroscopic descriptions of cell migration on growing domains	Baker Yates Erban
17/09	The Influence of Gene Expression Time Delays on Gierer-Meinhardt Pattern Formation Systems	Seirin Lee Gaffney Monk

18/09	Analysis of a stochastic chemical system close to a sniper bifurcation of its mean field model	Erban *et al.*
19/09	On the existence and the applications of modified equations for stochastic differential equations	Zygalakis
20/09	Pebble bed: reflector treatment and pressure velocity coupling	Charpin *et al.*
21/09	A finite difference method for free boundary problems	Fornberg
22/09	Tangent unit-vector fields: nonabelian homotopy invariants and the Dirichlet energy	Majumdar Robbins Zyskin
23/09	Morphological instability of a nonequilibrium icecolloid interface	Peppin Majumdar Wettlaufer
24/09	The effect of polar lipids on tear film dynamics	Aydemir Breward Witelski
25/09	Preconditioning for active set and projected gradient methods as semi-smooth Newton methods for PDE-constrained optimization with control constraints	Stoll Wathen
26/09	Functional differential equations arising in cell-growth	Wake Begg
27/09	A Cell Growth Model Revisited	Derfel van Brunt Wake

Copies of these, and any other OCCAM reports can be obtained from:

Oxford Centre for Collaborative Applied Mathematics
Mathematical Institute
24 - 29 St Giles'
Oxford
OX1 3LB
England
www.maths.ox.ac.uk/occam