AN IMPLICITLY RESTARTED SYMPLECTIC LANCZOS METHOD FOR THE SYMPLECTIC EIGENVALUE PROBLEM*

PETER BENNER† AND HEIKE FAßBENDER‡

Abstract. An implicitly restarted symplectic Lanczos method for the symplectic eigenvalue problem is presented. The Lanczos vectors are constructed to form a symplectic basis. The inherent numerical difficulties of the symplectic Lanczos method are addressed by inexpensive implicit restarts. The method is used to compute some eigenvalues and eigenvectors of large and sparse symplectic matrices.

Key words. eigenvalues, symplectic Lanczos method, implicit restarting, symplectic matrix.

AMS subject classifications. 65F15, 65F50, 15A18

1. Introduction. We consider the numerical solution of the real symplectic eigenvalue problem

$$Mx = \lambda x \tag{1.1}$$

where $M \in \mathbb{R}^{2n\times 2n}$ is large and possibly sparse. A matrix M is called *symplectic* iff

$$MJM^T = J, \tag{1.2}$$

or equivalently, $M^T JM = J$, where

$$J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \tag{1.3}$$

and I_n is the $n \times n$ identity matrix. The symplectic matrices form a group under multiplication. The eigenvalues of symplectic matrices occur in reciprocal pairs: If λ is an eigenvalue of M with right eigenvector x, then λ^{-1} is an eigenvalue of M with left eigenvector $(Jx)^T$. The computation of eigenvalues and eigenvectors of such matrices is an important task in applications like the discrete linear-quadratic regulator problem, discrete Kalman filtering, or the solution of discrete-time algebraic Riccati equations. See, e.g., [21, 22, 28] for applications and further references. Symplectic matrices also occur when solving linear Hamiltonian difference systems [6].

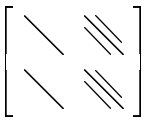
In order to develop fast, efficient, and reliable methods, the symplectic structure of the problem should be preserved and exploited. Then important properties of symplectic matrices (e.g., eigenvalues occurring in reciprocal pairs) will be preserved and not destroyed by rounding errors. Different structure-preserving methods for solving (1.1) have been proposed. In [25], Lin introduces the $S + S^{-1}$-transformation which can be used to compute the eigenvalues of a symplectic matrix by a structure-preserving method similar to Van Loan's square-reduced method for the Hamiltonian eigenvalue problem [38]. Flaschka, Mehrmann, and Zywietz show in [14] how to construct structure-preserving methods based on the SR method [10, 11, 26]. Patel [34, 33] and Mehrmann [27] developed structure-preserving algorithms for the symplectic generalized eigenproblem $L-\lambda N$ where $L, N \in \mathbb{R}^{2n\times 2n}$, and $LJL^T = NJN^T$.

*submitted in July 1998

†Universität Bremen, Fachbereich 3 – Mathematik und Informatik, Zentrum für Technomathematik, 28357 Bremen, FRG. E-mail: benner@math.uni-bremen.de

‡corresponding author, Universität Bremen, Fachbereich 3 – Mathematik und Informatik, Zentrum für Technomathematik, 28357 Bremen, FRG. E-mail: heike@math.uni-bremen.de

Recently, Banse and Bunse-Gerstner [2, 3] presented a new condensed form for symplectic matrices. The $2n \times 2n$ condensed matrix is symplectic, contains $8n-4$ nonzero entries, and is determined by $4n-1$ parameters. This condensed form, called *symplectic butterfly form*, can be depicted as a symplectic matrix of the following form:



Once the reduction of a symplectic matrix to butterfly form is achieved, the SR algorithm [10, 11, 26] is a suitable tool for computing the eigenvalues/eigenvectors of a symplectic matrix. The SR algorithm preserves the butterfly form in its iterations and can be rewritten in a parameterized form that works with the $4n-1$ parameters instead of the $(2n)^2$ matrix elements in each iteration. Hence, the symplectic structure, which will be destroyed in the numerical process due to roundoff errors, can be restored in each iteration for this condensed form. An analysis of the butterfly SR algorithm can be found in [2, 4, 5].

In [2, 3] an elimination process for computing the butterfly form of a symplectic matrix is given which uses elementary unitary symplectic transformations as well as non-unitary symplectic transformations. Unfortunately, this approach is not suitable when dealing with large and sparse symplectic matrices as an elimination process can not make full use of the sparsity. Hence, symplectic Lanczos methods which create the symplectic butterfly form if no breakdown occurs are derived in [2, 4]. Given $v_1 \in \mathbb{R}^{2n}$ and a symplectic matrix $M \in \mathbb{R}^{2n\times 2n}$, these Lanczos algorithms produce a matrix $S^{2n,2k} = [v_1, v_2, \ldots, v_k, w_1, w_2, \ldots, w_k] \in \mathbb{R}^{2n\times 2k}$ which satisfies a recursion of the form

$$MS^{2n,2k} = S^{2n,2k}B^{2k,2k} + r_{k+1}e_{2k}^T, \tag{1.4}$$

where $B^{2k,2k}$ is a butterfly matrix of order $2k \times 2k$, and the columns of $S^{2n,2k}$ are orthogonal with respect to the indefinite inner product defined by J (1.3). The latter property will be called *J–orthogonality* throughout this paper. The residual r_{k+1} depends on v_{k+1} and w_{k+1}; hence $(S^{2n,2k})^T J r_{k+1} = 0$. Such a symplectic Lanczos method will suffer from the well-known numerical difficulties inherent to any Lanczos method for unsymmetric matrices. In [2], a symplectic look-ahead Lanczos algorithm is presented which overcomes breakdown by giving up the strict butterfly form. Unfortunately, so far there do not exist eigenvalue methods that can make use of that special reduced form. Standard eigenvalue methods as QR or SR algorithms have to be employed resulting in a full symplectic matrix after only a few iteration steps.

A different approach to deal with the numerical difficulties of the Lanczos process is to modify the starting vectors by an implicitly restarted Lanczos process (see the fundamental work in [9, 35]); for the unsymmetric eigenproblem the implicitly restarted Arnoldi method has been implemented very successfully, see [24]). The problems are addressed by fixing the number of steps in the Lanczos process at a prescribed value k which depends upon the required number of approximate eigenvalues. J–orthogonality of the k Lanczos vectors is secured by re–J–orthogonalizing these vectors when necessary. The purpose of the implicit restart is to determine initial vectors such that the associated residual vectors are tiny. Given (1.4), an implicit Lanczos restart computes the Lanczos factorization

$$M\breve{S}^{2k} = \breve{S}^{2k}\breve{B}^{2k,2k} + \breve{r}_{k+1}e_{2k}^T$$

which corresponds to the starting vector

$$\breve{v}_1 = p(M)v_1$$

(where $p(M) \in \mathbb{R}^{2n\times 2n}$ is a polynomial) without having to explicitly restart the Lanczos process with the vector $\breve{v}_1$. Such an implicit restarting mechanism is derived here analogous to the technique introduced in [4, 18, 35].

Section 2 reviews the symplectic butterfly form and some of its properties that will be helpful for analyzing the symplectic Lanczos method which reduces a symplectic matrix to butterfly form. This symplectic Lanczos method is presented in Section 3. Further, that section is concerned with finding conditions for the symplectic Lanczos method terminating prematurely such that an invariant subspace associated with certain desired eigenvalues is obtained. We will also consider the important question of determining stopping criteria. The implicitly restarted symplectic Lanczos method itself is derived in Section 4. Numerical properties of the proposed algorithm are discussed in Section 5. In Section 6, we present some preliminary numerical examples.

2. The Symplectic Butterfly Form. A symplectic matrix

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} \diagdown & \diagdown\!\diagdown \\ \diagdown & \diagdown\!\diagdown \end{bmatrix}, \qquad \text{where } B_{ij} \in \mathbb{R}^{n\times n},$$

is called a *butterfly* matrix if B_{11} and B_{21} are diagonal, and B_{12} and B_{22} are tridiagonal. Banse and Bunse-Gerstner [2, 3] showed that for every symplectic matrix M, there exist numerous symplectic matrices S such that $B = S^{-1}MS$ is a symplectic butterfly matrix. In [2], an elimination process for computing the butterfly form of a symplectic matrix is presented (see also [4]).

In [4], an *unreduced butterfly matrix* is introduced in which the lower right tridiagonal matrix is unreduced, that is, the subdiagonal elements of B_{22} are nonzero. Using the definition of a symplectic matrix, one easily verifies that if B is an unreduced butterfly matrix, then B_{21} is nonsingular. This allows the decomposition of B into two simpler symplectic matrices:

$$B = \begin{bmatrix} B_{21}^{-1} & B_{11} \\ 0 & B_{21} \end{bmatrix} \begin{bmatrix} 0 & -I \\ I & T \end{bmatrix} = \begin{bmatrix} \diagdown & \diagdown \\ 0 & \diagdown \end{bmatrix} \begin{bmatrix} 0 & -I \\ I & \diagdown\!\diagdown \end{bmatrix}, \tag{2.1}$$

where $T = B_{21}^{-1}B_{22}$ is tridiagonal and symmetric. Hence $4n-1$ parameters that determine the symplectic matrix can be read off directly. The unreduced butterfly matrices play a role analogous to that of unreduced Hessenberg matrices in the standard QR theory [2, 4, 5].

We will frequently make use of the decomposition (2.1) and will denote it by

$$B_1 = \begin{bmatrix} B_{21}^{-1} & B_{11} \\ 0 & B21 \end{bmatrix} = \left[\begin{array}{ccc|ccc} a_1^{-1} & & & b_1 & & \\ & \ddots & & & \ddots & \\ & & a_n^{-1} & & & b_n \\ \hline & & & a_1 & & \\ & & & & \ddots & \\ & & & & & a_n \end{array}\right], \tag{2.2}$$

$$B_2^{-1} = \begin{bmatrix} 0 & -I \\ I & B_{21}^{-1}B_{22} \end{bmatrix} = \left[\begin{array}{ccc|ccc} & & & -1 & & \\ & & & & \ddots & \\ & & & & & -1 \\ \hline 1 & & & c_1 & d_2 & \\ & \ddots & & d_2 & \ddots & \ddots \\ & & \ddots & & \ddots & \ddots \\ & & 1 & & & d_n \end{array}\right. \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ c_n \end{array}\right] , \tag{2.3}$$

$$B = \left[\begin{array}{ccc|ccc} b_1 & & & b_1c_1 - a_1^{-1} & b_1d_2 & \\ & \ddots & & b_2d_2 & \ddots & \ddots \\ & & \ddots & & \ddots & \ddots \\ & & b_n & & & b_nd_n \\ \hline a_1 & & & a_1c_1 & a_1d_2 & \\ & \ddots & & a_2d_2 & \ddots & \ddots \\ & & \ddots & & \ddots & \ddots \\ & & a_n & & & a_nd_n \end{array}\right. \left.\begin{array}{c} \\ \\ b_{n-1}d_n \\ b_nc_n - a_n^{-1} \\ \\ \\ a_{n-1}d_n \\ a_nc_n \end{array}\right] . \tag{2.4}$$

REMARK 2.1. *(See [4].)*

a) Any unreduced butterfly matrix is similar to an unreduced butterfly matrix with $b_i = 1$, $|a_i| = 1$ *for* $i = 1, \ldots, n$ *and* $\operatorname{sign}(a_i) = \operatorname{sign}(d_i)$ *for* $i = 2, \ldots n$.

b) We will have deflation if $d_j = 0$ *for some* j. *Then the eigenproblem can be split into two smaller ones with unreduced symplectic butterfly matrices.*

Eigenvalues and eigenvectors of symplectic butterfly matrices can be computed efficiently by the SR algorithm [7], which is a QR like algorithm in which the QR decomposition is replaced by the SR decomposition. Almost every matrix $A \in \mathbb{R}^{2n\times 2n}$ can be decomposed into a product $A = SR$ where S is symplectic and R is J–triangular, that is

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix},$$

where all submatrices $R_{ij} \in \mathbb{R}^{n\times n}$ are upper triangular, and R_{21} is strictly upper triangular [12]. In the following a matrix $D \in \mathbb{R}^{2n\times 2n}$ will be called *trivial* if it is both symplectic and J–triangular. D is trivial if and only if it has the form

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

where C and F are diagonal matrices, C nonsingular.

If the SR *decomposition* $A = SR$ exists, then other SR decompositions of A can be built from it by passing trivial factors back and forth between S and R. That is, if D is a trivial matrix, $\tilde{S} = SD$ and $\tilde{R} = D^{-1}R$, then $A = \tilde{S}\tilde{R}$ is another SR decomposition of A. If A is nonsingular, then this is the only way to create other SR decompositions. In other words, the SR decomposition is unique up to trivial factors.

The SR algorithm is an iterative algorithm that performs an SR decomposition at each iteration. If B is the current iterate, then a *spectral transformation function*

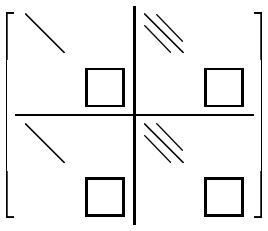
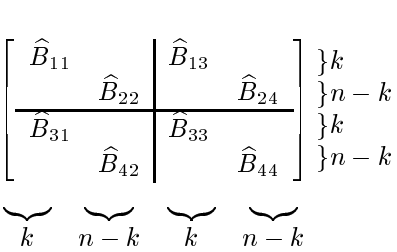
q is chosen (such that $q(B) \in \mathbb{R}^{2n\times 2n}$) and the SR decomposition of $q(B)$ is formed, if possible:

$$q(B) = SR.$$

Then the symplectic factor S is used to perform a similarity transformation on B to yield the next iterate, which we will call $\widehat{B}$:

$$\widehat{B} = S^{-1}BS. \tag{2.5}$$

If $\operatorname{rank}(q(B)) = 2n$ and B is an unreduced symplectic butterfly matrix, then so is $\widehat{B}$ in (2.5) [2, 3]. If rank $(p(B)) = 2n - \nu =: 2k$ and B is an unreduced symplectic butterfly matrix, then $\widehat{B}$ in (2.5) is of the form (see [4])

(2.6) $\widehat{B} =$  $=$ 

where

- $\begin{bmatrix} \widehat{B}_{11} & \widehat{B}_{13} \\ \widehat{B}_{31} & \widehat{B}_{33} \end{bmatrix}$ is a symplectic butterfly matrix and
- the eigenvalues of $\begin{bmatrix} \widehat{B}_{22} & \widehat{B}_{24} \\ \widehat{B}_{42} & \widehat{B}_{44} \end{bmatrix}$ are just the ν shifts that are eigenvalues of B.

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 $q(B)$ has to be computed explicitly. A preferred alternative is the implicit SR step, an analogue to the Francis QR step [15, 17, 20]. As the implicit SR step is analogous to the implicit QR step, this technique will not be discussed here (see [4, 5] for details).

A natural way to choose the spectral transformation function q is to choose a polynomial $p_2(\lambda) = (\lambda - \mu)(\lambda - \mu^{-1})$ for $\mu \in \mathbb{R}$ (or $\mu \in \mathbb{C}, |\mu| = 1$) or $p_4(\lambda) = (\lambda-\mu)(\lambda-\mu^{-1})(\lambda-\overline{\mu})(\lambda-\overline{\mu}^{-1})$ for $\mu \in \mathbb{C}$ as these choices make use of the symmetries of the spectrum of symplectic matrices. But, as explained in [5], a better choice is a Laurent polynomial to drive the SR step. For example, instead of $p_4(\lambda)$ we will use

$$q_4(\lambda) = p_4(\lambda)\lambda^{-2} = (\lambda+\lambda^{-1})^2 - (\mu+\mu^{-1}+\overline{\mu}+\overline{\mu}^{-1})(\lambda+\lambda^{-1}) + (\mu+\mu^{-1})(\overline{\mu}+\overline{\mu}^{-1}) - 2.$$

This reduces the size of the bulges that are introduced, thereby decreasing he number of computations required per iteration. Moreover, the use of Laurent polynomials improves the convergence and stability properties of the algorithm by effectively treating each reciprocal pair of eigenvalues as a unit. Using a generalized Rayleigh-quotient strategy, the butterfly SR algorithm is typically cubic convergent [5].

The right eigenvectors of unreduced butterfly matrices have the following property which will be helpful when analyzing the symplectic Lanczos method introduced in the next section.

LEMMA 2.2. *Suppose that $B \in \mathbb{R}^{2n\times 2n}$ is an unreduced butterfly matrix as in (2.4). If $Bx = \lambda x$ with $x \neq 0$ then $e_{2k}^T x \neq 0$.*

In order to proof this lemma we need the following definition. Let P_n be the permutation matrix

$$P_n := [e_1, e_3, \ldots, e_{2n-1}, e_2, e_4, \ldots, e_{2n}] \in \mathbb{R}^{2n\times 2n}. \tag{2.7}$$

If the dimension of P_n is clear from the context, we leave off the subscript.

Proof. The proof is by induction on the size of B. The entries of the eigenvector x will be denoted by x_i; $x = [x_1, x_2, \ldots, x_{2n}]^T$.

Suppose that $n = 2$. The second and fourth row of $Bx = \lambda x$ yield

$$b_2x_2 + b_2d_2x_3 + (b_2c_2 - a_2^{-1})x_4 = \lambda x_2, \tag{2.8}$$

$$a_2x_2 + a_2d_2x_3 + a_2c_2x_4 = \lambda x_4. \tag{2.9}$$

Since B is unreduced, we know that $a_2 \neq 0$ and $d_2 \neq 0$. If $x_4 = 0$ then from (2.9) we obtain

$$x_2 + d_2x_3 = 0, \tag{2.10}$$

while (2.8) gives $b_2(x_2 + d_2x_3) = \lambda x_2$. Using (2.10) we obtain $x_2 = 0$ and further $x_3 = 0$.

The third row of $Bx = \lambda x$ gives

$$a_1x_1 + a_1c_1x_3 + a_1d_2x_4 = \lambda x_3.$$

Using $x_2 = x_3 = x_4 = 0$ and $a_1 \neq 0$, since B is unreduced, we obtain $x_1 = 0$. Thus $x = 0$ which contradicts the assumption $x \neq 0$.

Assume that the lemma is true for matrices of order $2(n-1)$. Let $B^{2n,2n} \in \mathbb{R}^{2n\times 2n}$ be an unreduced butterfly matrix. For simplicity we will consider the permuted equation $B_P^{2n,2n}x_P = \lambda x_P$ where $B_P^{2n,2n} = PB^{2n,2n}P^T$ and $x_P = Px$. Partition $B_P^{2n,2n}, x_P$ as

$$B_P^{2n,2n} = \left[\begin{array}{c|cc} B_P^{2(n-1),2(n-1)} & 0 & d_n(b_{n-1}e_{2n-3} + a_{n-1}e_{2n-2}) \\ \hline b_nd_ne_{2n-2}^T & b_n & b_nc_n - a_n^{-1} \\ a_nd_ne_{2n-2}^T & a_n & a_nc_n \end{array}\right],$$

$$x_P = \begin{bmatrix} y \\ \tilde{x}_{2n-1} \\ \tilde{x}_{2n} \end{bmatrix},$$

where $B_P^{2(n-1),2(n-1)} \in \mathbb{R}^{(2n-2)\times(2n-2)}$ is an unreduced butterfly matrix and $y \in \mathbb{R}^{2n-2}$. Suppose $x_{2n} = \tilde{x}_{2n} = 0$. This implies

$$d_ny_{2n-2} + \tilde{x}_{2n-1} = 0 \tag{2.11}$$

since $a_n \neq 0$ as $B^{2n,2n}$ is unreduced. Further we have

$$b_n(d_ny_{2n-2} + \tilde{x}_{2n-1}) = \lambda\tilde{x}_{2n-1}.$$

Hence, using (2.11) we get $\tilde{x}_{2n-1} = 0$. This implies $B_P^{2n-1,2n-1}y = \lambda y$. Using $\tilde{x}_{2n-1} = \tilde{x}_{2n} = 0$ we further obtain from (2.11) $y_{2n-2} = 0$. This is a contradiction, because by induction hypothesis $e_{2n-2}^T y \neq 0$. ∎

REMARK 2.3. *If $Bx = \lambda x$, then $(Jx)^TB = \lambda^{-1}(Jx)^T$. Let y be the right eigenvector of B to λ^{-1} : $By = \lambda^{-1}y$, then $(Jy)^TB = \lambda(Jy)^T$. From Lemma 2.2 it follows that $e_{2n}^T y \neq 0$, hence the nth component of the left eigenvector of B corresponding to λ is $\neq 0$.*

3. A Symplectic Lanczos Method for Symplectic Matrices. In this section, we review the symplectic Lanczos method to compute the butterfly form (2.4) for a symplectic matrix M derived in [4]. The usual unsymmetric Lanczos algorithm generates two sequences of vectors. Due to the symplectic structure of M 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 [16].) Further, this section is concerned with finding conditions for the symplectic Lanczos method terminating prematurely such that an invariant subspace associated with certain desired eigenvalues is obtained. Finally we will consider the important question of determining stopping criteria.

In order to simplify the notation we use in the following permuted versions of M and B as in the previous section. Let

$$M_P = PMP^T, \quad B_P = PBP^T, \quad S_P = PSP^T, \quad J_P = PJP^T$$

with the permutation matrix P as in (2.7).

3.1. The symplectic Lanczos factorization. We want to compute a symplectic matrix S such that S transforms the symplectic matrix M to a symplectic butterfly matrix B; in the permuted version $MS = SB$ yields

$$M_P S_P = S_P B_P. \tag{3.1}$$

Equivalently, as $B = B_1 B_2^{-1}$, we can consider

$$M_P S_P (B_2)_P = S_P (B_1)_P \tag{3.2}$$

where

$$(B_1)_P = \begin{bmatrix} a_1^{-1} & -b_1 & & & & \\ 0 & a_1 & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & a_n^{-1} & -b_n \\ & & & & 0 & a_n \end{bmatrix}, \tag{3.3}$$

$$(B_2)_P = \begin{bmatrix} c_1 & 1 & d_2 & 0 & & & & \\ -1 & 0 & 0 & 0 & & & & \\ d_2 & 0 & c_2 & 1 & \ddots & & & \\ 0 & 0 & -1 & 0 & & \ddots & & \\ & & \ddots & & \ddots & & d_n & 0 \\ & & & \ddots & & \ddots & 0 & 0 \\ & & & & d_n & 0 & c_n & 1 \\ & & & & 0 & 0 & -1 & 0 \end{bmatrix}. \tag{3.4}$$

The structure preserving Lanczos method generates a sequence of permuted symplectic matrices

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

satisfying

$$M_P S_P^{2n,2k} = S_P^{2n,2k} B_P^{2k,2k} - d_{k+1}(b_{k+1} v_{k+1} + a_{k+1} w_{k+1}) e_{2k}^T \tag{3.5}$$

where $B_P^{2k,2k} = P_k B^{2k,2k} P_k{}^T$ is a permuted $2k \times 2k$ symplectic butterfly matrix.

The vector $r_{k+1} := d_{k+1}(b_{k+1}v_{k+1} + a_{k+1}w_{k+1})$ is the *residual vector* and is J_P–orthogonal to the columns of $S_P^{2n,2k}$, the *Lanczos vectors*. The matrix $B_P^{2k,2k}$ is the J_P–orthogonal projection of M_P onto the range of $S_P^{2n,2k}$,

$$B_P^{2k,2k} = J_P^{2k,2k}(S_P^{2n,2k})^T J_P M_P S_P^{2n,2k}.$$

Here $J_P^{2k,2k}$ denotes a permuted $2k \times 2k$ matrix J of the form (1.3). Equation (3.5) defines a length $2k$ Lanczos factorization of M_P. If the residual vector r_{k+1} is the zero vector, then equation (3.5) is called a *truncated Lanczos factorization* when $k < n$. Note that r_{n+1} must vanish since $(S_P^{2n,2n})^T J_P r_{n+1} = 0$ and the columns of $S_P^{2n,2n}$ form a J_P–orthogonal basis for $\mathbb{R}^{2n}$. In this case the symplectic Lanczos method computes a reduction to permuted butterfly form.

The symplectic Lanczos factorization is, up to multiplication by a trivial matrix, specified by the starting vector v_1 (see [4, Theorem 4.1]).

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

$$M_P S_P (B_2)_P e_j = S_P (B_1)_P e_j, \qquad j = 1, 2, \ldots .$$

From this we obtain the algorithm given in Table 3.1 (for a more detailed discussion see [4]).

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 $d_1 = ||\widetilde{v}_1||_2$ and $v_1 = \frac{1}{d_1}\widetilde{v}_1$.

for m = 1, 2, ... do

(update of w_m)

set

$\widetilde{w}_m = M_P v_m - b_m v_m$
$a_m = v_m^T J_P M_P v_m$
$w_m = \frac{1}{a_m}\widetilde{w}_m$

(computation of c_m)

$c_m = a_m^{-1} v_m^T J_P M_P^{-1} w_m$

(update of v_{m+1})

$\widetilde{v}_{m+1} = -d_m v_{m-1} - c_m v_m + w_m + a_m^{-1} M_P^{-1} v_m$
$d_{m+1} = ||\widetilde{v}_{m+1}||_2$
$v_{m+1} = \frac{1}{d_{m+1}}\widetilde{v}_{m+1}$

TABLE 3.1
Symplectic Lanczos Method

REMARK 3.1. *Using the derived formulae for w_{k+1}, the residual term $r_{k+1} = d_{k+1}(b_{k+1}v_{k+1} + a_{k+1}w_{k+1})$ can be expressed as*

$$r_{k+1} = d_{k+1} M_P v_{k+1}.$$

There is still some freedom in the choice of the parameters that occur in this algorithm. Essentially, the parameters b_m can be chosen freely. Here we set $b_m = 1$. Likewise a different choice of the parameters a_m, d_m is possible.

Note that $M_P^{-1} = -J_P M_P^T J_P$ since M is symplectic. Thus $M_P^{-1} v_m$ is just a matrix-vector-product with the transpose of M_P. Hence, 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 M_P and $2k$ is the number of Lanczos vectors computed (that is, the loop is executed k times). The algorithm as given in Table 3.1 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}).

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

$$v_j^T J_P \widetilde{v}_{m+1} \;=\; 0, \qquad\qquad w_j^T J_P \widetilde{v}_{m+1} \;=\; 0,$$

for $j = 1, \ldots, m$, the algorithm can be continued. The resulting butterfly 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 M_P with dimension $2m - 1$ is found (or a good approximation to such a subspace). In this case the parameter a_m will be zero (or close to zero). From Table 3.1 we further obtain that in this case $M_P v_m = b_m v_m$, i.e., b_m is a real eigenvalue of M_P (and hence of M) with corresponding eigenvector v_m ($P^T v_m$). Due to the symmetry of the spectrum of M, we also have that $1/b_m$ is an eigenvalue of M. Computing an eigenvector y of M_P corresponding to $1/b_m$, we can try to augment the $(2m-1)$–dimensional invariant subspace to an M_P–invariant subspace of even dimension. If this is possible, the space can be made J_P–orthogonal by J_P–orthogonalizing y against $\{ v_1, w_1, \ldots, v_{m-1}, w_{m-1} \}$ and normalizing such that $y^T J_P v_m = 1$.

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 $a_{m+1} = 0$, then the breakdown is serious. No reduction of the symplectic matrix to a symplectic butterfly matrix with v_1 as first column of the transformation matrix exists.

A convergence analysis for the symplectic Lanczos algorithm analogous to the one for the unsymmetric Lanczos algorithm presented by Ye [39] can be given. Moreover, an error analysis of the symplectic Lanczos algorithm in finite-precision arithmetic analogous to the analysis for the unsymmetric Lanczos algorithm presented by Bai [1] can also be derived. These results will be presented in [13]. As to be expected, the computed Lanczos vectors loose $J(J_P)$–orthogonality when some Ritz values begin to converge.

3.2. Truncated symplectic Lanczos factorizations. This section is concerned with finding conditions for the symplectic Lanczos method terminating prematurely. This is a welcome event since in this case we have found an invariant symplectic

[1] (Following [17], we define each floating point arithmetic operation together with the associated integer indexing as a flop.)

subspace $S^{2n,2k}$ and the eigenvalues of $B^{2k,2k}$ are a subset of those of M. We will first discuss the conditions under which the residual vector of the symplectic Lanczos factorization will vanish at some step k. Then we will show how the residual vector and the starting vector are related. Finally a result indicating when a particular starting vector generates an exact truncated factorization is given.

First the conditions under which the residual vector of the symplectic Lanczos factorization will vanish at some step k will be discussed. From the derivation of the algorithm it is immediately clear that if no breakdown occurs, then

$$\begin{aligned}
&\operatorname{span}\{v_1,\ldots,v_{k+1},w_1,\ldots,w_k\}\\
&\qquad=\operatorname{span}\{v_1,M_P^{-1}v_1,\ldots,M_P^{-k}v_1,M_Pv_1,\ldots,M_P^{k}v_1\}\\
&\qquad=\operatorname{span}\{\operatorname{span}(\mathcal{K}(M_P,v_1,k))\cup\{M_P^{-k}v_1\}\}\\
&\operatorname{span}\{v_1,\ldots,v_{k+1},w_1,\ldots,w_{k+1}\}\\
&\qquad=\operatorname{span}\{v_1,M_P^{-1}v_1,\ldots,M_P^{-k}v_1,M_Pv_1,\ldots,M_P^{k+1}v_1\}\\
&\qquad=\operatorname{span}(\mathcal{K}(M_P,v_1,k+1))
\end{aligned}$$

where $\mathcal{K}(X,v,k)=\{v,X^{-1}v,X^{-2}v\ldots,X^{-(k-1)}v,Xv,X^2v,\ldots,X^kv\}$. Further it is easy to see that

$$\text{(3.6)}\quad \dim\ \mathcal{K}(M_P,v_1,k)=d<2k \quad\Longrightarrow\quad \dim\ \mathcal{K}(M_P,v_1,j)=d \quad\forall\ j>k.$$

If $\dim\ \mathcal{K}(M_P,v_1,k+1)=2k+1$, then

$$w_{k+1}=M_Pv_{k+1}-b_{k+1}v_{k+1}\in\operatorname{span}\{v_1,\ldots,v_{k+1},w_1,\ldots,w_k\}.$$

Hence, there exist real scalars $\alpha_1,\ldots,\alpha_{k+1}$ and $\beta_1,\ldots,\beta_k$ such that

$$M_Pv_{k+1}=\alpha_1v_1+\ldots+\alpha_{k+1}v_{k+1}+\beta_1w_1+\ldots+\beta_kw_k.$$

Using the definition of a_{k+1} as given in Table 3.1 and the above expression we obtain because of J–orthogonality,

$$\begin{aligned}
a_{k+1}&=v_{k+1}^TJ_PM_Pv_{k+1}\\
&=\alpha_1v_{k+1}^TJ_Pv_1+\ldots+\alpha_{k+1}v_{k+1}^TJ_Pv_{k+1}+\beta_1v_{k+1}^TJ_Pw_1+\ldots+\beta_kv_{k+1}^TJ_Pw_k\\
&=0.
\end{aligned}$$

As $\widetilde{w}_{k+1}=a_{k+1}w_{k+1}=M_Pv_{k+1}-b_{k+1}v_{k+1}$ (see Table 3.1) it follows that $\widetilde{w}_{k+1}=0$. This implies that an invariant subspace of M_P with dimension $2k+1$ is found.

If $\dim\ \mathcal{K}(M_P,v_1,k+1)=2k$, then $M_P^{-1}v_k\in\operatorname{span}\{v_1,\ldots,v_k,w_1,\ldots,w_k\}$. Hence

$$a_k^{-1}M_P^{-1}v_k=\alpha_1v_1+\ldots+\alpha_kv_k+\beta_1w_1+\ldots+\beta_kw_k,$$

for properly chosen α_j,β_j and from the algorithm in Table 3.1

$$\begin{aligned}
\widetilde{v}_{k+1}=\alpha_1v_1+\ldots+\alpha_{k-2}v_{k-2}+(\alpha_{k-1}-d_k)v_{k-1}+(\alpha_k-c_k)v_k\\
+\beta_1w_1+\ldots+\beta_{k-1}w_{k-1}+(\beta_k+1)w_k.
\end{aligned}$$

Since $[v_1,w_1,\ldots,v_k,w_k]^TJ_P\widetilde{v}_{k+1}=[0,\ldots,0]$ we obtain

$$\begin{aligned}
v_j^TJ_P\widetilde{v}_{k+1}&=\beta_jv_j^TJ_Pw_j=\beta_j=0 && \text{for } j<k\\
v_k^TJ_P\widetilde{v}_{k+1}&=(\beta_k+1)v_k^TJ_Pw_k=\beta_k+1=0 &&\\
w_j^TJ_P\widetilde{v}_{k+1}&=\alpha_jw_j^TJ_Pv_j=-\alpha_j=0 && \text{for } j<k-2\\
w_{k-1}^TJ_P\widetilde{v}_{k+1}&=(\alpha_{k-1}-d_k)w_{k-1}^TJ_Pv_{k-1}=d_k-\alpha_{k-1}=0 &&\\
w_k^TJ_P\widetilde{v}_{k+1}&=(\alpha_k-c_k)w_k^TJ_Pv_k=c_k-\alpha_k=0. &&
\end{aligned}$$

Therefore $\widetilde{v}_{k+1} = 0$ and further $d_{k+1} = 0$. This implies that the residual vector of the symplectic Lanczos factorization will vanish at the first step k such that the dimension of $\mathcal{K}(M, v_1, k+1)$ is equal to $2k$ and hence is guaranteed to vanish for some $k \leq n$.

Next we will discuss the relation between the residual term and the starting vector. If dim $\mathcal{K}(M, v_1, n) = 2n$ then

$$MK(M, v_1, n) = K(M, v_1, n)C_n$$

where $K(M, v_1, n) = [v, M^{-1}v, M^{-2}v \ldots, M^{-(n-1)}v, Mv, M^2v, \ldots, M^n v] \in \mathbb{R}^{2n\times 2n}$, and C_n is a generalized companion matrix of the form

$$C_n = \left[\begin{array}{ccccc|ccc} 0 & 1 & & & & & & c_1 \\ & \ddots & \ddots & & & & & \vdots \\ & & \ddots & 1 & & & & \vdots \\ & & & 0 & & & & c_n \\ \hline 1 & & & & & 0 & & c_{n+1} \\ & & & & & 1 & \ddots & \vdots \\ & & & & & & \ddots & 0 \quad c_{2n-1} \\ & & & & & & & 1 \quad c_{2n} \end{array}\right]$$

(see [2, proof of Satz 3.6]). Thus,

$$MK(M, v_1, k) = K(M, v_1, k)C_k + (M^{k+1}v_1 - K(M, v_1, k)C_k e_{2k})e_{2k}^T. \tag{3.7}$$

Define the residual in (3.7) by

$$f_{k+1} := M^{k+1}v_1 - K(M, v_1, k)C_k e_{2k}. \tag{3.8}$$

Note that

$$f_{k+1} = p_k(M)v_1 \tag{3.9}$$

where

$$p_k(\lambda) := \lambda^{k+1} - \sum_{j=0}^{k-1}(c_{k+j+1}\lambda^{j+1} + c_{j+1}\lambda^{-j}).$$

We will now show that f_{k+1} is up to scaling the residual of the length $2k$ symplectic Lanczos iteration with starting vector v_1. Together with (3.9) this reveals the relation between residual and starting vectors. Since det $(C_k - \lambda I) = \lambda^{2k} - \sum_{j=0}^{k-1}(c_{k-j}\lambda^j + c_{k+j+1}\lambda^{k+j})$, we obtain

$$p_k(\lambda) = \lambda^{-(k-1)} \det\ (C_k - \lambda I).$$

Let $K(M, v_1, k) = S^{2n,2k}R$ where $S^{2n,2k} \in \mathbb{R}^{2n\times 2k}$ with J–orthogonal columns (that is, $(S^{2n,2k})^T J_n S^{2n,2k} = J_k$) and $R \in \mathbb{R}^{2k\times 2k}$ is a J–triangular matrix. Then $S^{2n,2k}e_1 = v_1$. The diagonal elements of R are nonzero if and only if the columns of $K(M, v_1, k)$ are linear independent. Choosing

$$c = \left[\begin{array}{c} c_1 \\ \vdots \\ c_{2k} \end{array}\right] = R^{-1}(-J^k(S^{2n,2k})^T J^n)M^{k+1}v_1$$

assures that $(-J^k(S^{2n,2k})^T J^n)f_{k+1} = 0$. Now multiplying (3.7) from the right by R^{-1} yields

$$\begin{aligned} & MK(M,v_1,k)R^{-1} - K(M,v_1,k)C_kR^{-1} = f_{k+1}e_{2k}^T R^{-1} \\ \iff \quad & MS^{2n,2k} - S^{2n,2k}B = f_{k+1}e_{2k}^T/r_{2k,2k} \end{aligned} \tag{3.10}$$

where $B = RC_kR^{-1}$ is an unreduced butterfly matrix (see [2, proof of Satz 3.6]) with the same characteristic polynomial as C_k. Equation (3.10) is a valid symplectic Lanczos recursion with starting vector $v_1 = S^{2n,2k}e_1$ and residual vector $f_{k+1}/r_{2k,2k}$. By (3.9) and due to the essential uniqueness of the symplectic Lanczos recursion any symplectic Lanczos recursion with starting vector v_1 yields a residual vector that can be expressed as a polynomial in M times the starting vector v_1.

REMARK 3.2. *From (3.8) it follows that if* $\dim \mathcal{K}(M,v_1,k+1) \leq 2k$*, then we can choose* $c_1,\ldots,c_{2k}$ *such that* $f_{k+1} = 0$*. This shows that if the Krylov subspace* $\mathcal{K}(M,v_1,k+1)$ *forms an* $2k$*–dimensional* M*–invariant subspace, the residual of the symplectic Lanczos recursion will be zero after* k *Lanczos steps such that the columns of* $S^{2n,2k}$ *span a symplectic basis for the subspace* $\mathcal{K}(M,v_1,k+1)$*.*

The final result of this section will give necessary and sufficient conditions for a particular starting vector to generate an exact truncated factorization in a similar way as stated for the Arnoldi method in [35]. This is desirable since then the columns of $S^{2n,2k}$ form a basis for an invariant symplectic subspace of M and the eigenvalues of $B^{2k,2k}$ are a subset of those of M. Here, $\hat{v}_k$, $\hat{w}_k$ will denote the Lanczos vectors after permuting them back, i.e., $\hat{v}_k = P^T v_k$, $\hat{w}_k = P^T w_k$.

THEOREM 3.3. *Let* $MS^{2n,2k} - S^{2n,2k}B^{2k,2k} = d_{k+1}(b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1})e_{2k}^T$ *be the symplectic Lanczos factorization after* k *steps, with* $B^{2k,2k}$ *unreduced. Then* $d_{k+1} = 0$ *if and only if* $v_1 = Xy$ *where* $MX = XJ$ *with rank* $(X) = 2k$ *and* J *a Jordan matrix of order* $2k$*.*

Proof. If $d_{k+1} = 0$, let $B^{2k,2k}\widetilde{X} = \widetilde{X}J$ be the Jordan canonical form of $B^{2k,2k}$ and put $X = S^{2n,2k}\widetilde{X}$. Then $MX = S^{2n,2k}B^{2k,2k}\widetilde{X} = S^{2n,2k}\widetilde{X}J = XJ$ and $v_1 = S^{2n,2k}e_1 = S^{2n,2k}\widetilde{X}\widetilde{X}^{-1}e_1 = Xy$ with $y = \widetilde{X}^{-1}e_1$.

Suppose now that $MX = XJ$, $\operatorname{rank}(X) = 2k$ and $v_1 = Xy$. Then $M^mX = XJ^m$ for $m \in \mathbb{N}$ and it follows that

$$M^m v_1 = M^m Xy = XJ^m y \in \operatorname{Range}(X)$$

for $m \in \mathbb{N}$. Hence by (3.6) $\dim \mathcal{K}(M,v_1,k+1) \leq \operatorname{rank}(X) = 2k$. Since $B^{2k,2k}$ is unreduced, $\dim \mathcal{K}(M,v_1,j) = 2j$ for $j = 1,\ldots,k$. Hence $\dim \mathcal{K}(M,v_1,k+1) = 2k$ and therefore, $d_{k+1} = 0$. □

A similar result may be formulated in terms of Schur vectors or symplectic Schur vectors (see, e.g., [28, 29] for the real symplectic Schur decomposition of a symplectic matrix). These theorems provide the motivation for the implicit restart developed in the next section. Theorem 3.3 suggests that one might find an invariant subspace by iteratively replacing the starting vector with a linear combination of approximate eigenvectors corresponding to eigenvalues of interest. Such approximations are readily available through the Lanczos factorization.

3.3. Stopping Criteria. Now assume that we have performed k steps of the symplectic Lanczos method and thus obtained the identity (after permuting back)

$$MS^{2n,2k} = S^{2n,2k}B^{2k,2k} + d_{k+1}(b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1})e_{2k}^T.$$

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

$$\begin{aligned} ||Mx - \lambda x|| &= ||(MS^{2n,2k} - S^{2n,2k}B^{2k,2k})y|| \\ &= |d_{k+1}|\,|e_{2k}^T y|\,||b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1}||. \end{aligned} \tag{3.11}$$

The vector x is referred to as *Ritz vector* and λ as *Ritz value* of M. If the last component of the eigenvector y is sufficiently small, the right-hand side of (3.11) is small and the pair $\{\lambda, x\}$ is a good approximation to an eigenvalue-eigenvector pair of M. Note that by Lemma 2.2 $|e_{2k}^T y| > 0$ if $B^{2k,2k}$ is unreduced. The pair (λ, x) is exact for the nearby problem

$$(M - E)x = \lambda x \qquad \text{where} \quad E = d_{k+1}(b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1})e_k^T (S^{2n,2k})^T J^n.$$

A small $||E||$ is not sufficient for the Ritz pair $\{\lambda, x\}$ being a good approximation to an eigenvalue-eigenvector pair of M. The advantage of using the *Ritz estimate* $|d_{k+1}|\,|e_{2k}^T y|\,||b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1}||$ is to avoid the explicit formation of the residual $(MS^{2n,2k} - S^{2n,2k}B^{2k,2k})y$ when deciding about the numerical accuracy of an approximate eigenpair.

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 $B^{2k,2k}$ is diagonalizable

$$Y^{-1}B^{2k,2k}Y = \left[\begin{array}{ccc|ccc} \lambda_1 & & & & & \\ & \ddots & & & & \\ & & \lambda_k & & & \\ \hline & & & \lambda_1^{-1} & & \\ & & & & \ddots & \\ & & & & & \lambda_k^{-1} \end{array}\right] = \Lambda.$$

Let $X = S^{2n,2k}Y = [x_1 \ldots, x_{2k}]$ and denote $d_{k+1}(b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1})$ by r_{k+1}. Since $MS^{2n,2k} = S^{2n,2k}B^{2k,2k} + r_{k+1}e_{2k}^T$, it follows that

$$MS^{2n,2k}Y = S^{2n,2k}YY^{-1}B^{2k,2k}Y + r_{k+1}e_{2k}^T Y$$

or $MX = X\Lambda + r_{k+1}e_{2k}^T Y$. Thus

$$Mx_i = \lambda_i x_i + y_{2k,i}r_{k+1} \quad \text{and} \quad Mx_{k+i} = \lambda_i^{-1}x_{k+i} + y_{2k,k+i}r_{k+1}$$

for $i = 1, \ldots, k$. The last equation can be re-written as

$$(Jx_{k+i})^T M = \lambda_i (Jx_{k+i})^T + y_{2k,k+i}\lambda_i r_{k+1}^T JM.$$

Using Theorem 2' of [19] we obtain that $(\lambda_i, x_i, (Jx_{k+i})^T)$ is an eigen-triplet of $M - E$ where

$$||E|| = \max\ \{\frac{||r_{k+1}||\ |y_{2k,i}|}{||x_i||}, \frac{||r_{k+1}^T JM||\ |y_{2k,k+i}\lambda_i|}{||Jx_{k+i}||}\}.$$

Furthermore, when $||E||$ is small enough, then

$$|\theta_i - \lambda_j| \leq \mathrm{cond}(\lambda_j)||E|| + \mathcal{O}(||E||^2),$$

where θ_i is an eigenvalue of M and

$$\mathrm{cond}(\lambda_j) = \frac{||x_i||_2||Jx_{k+i}||_2}{|x_{k+i}^T Jx_i|} = ||x_i||_2||x_{k+i}||_2.$$

Consequently, the symplectic Lanczos algorithm should be continued until both $||E||$ is small and $\mathrm{cond}(\lambda_j)||E||$ is below a given threshold for accuracy.

4. An Implicitly Restarted Symplectic Lanczos Method. In the previous sections we have briefly mentioned two algorithms for computing approximations to the eigenvalues of a symplectic matrix M. The symplectic Lanczos algorithm is appropriate when the matrix M is large and sparse. If only a small subset of the eigenvalues is desired, the length k symplectic Lanczos factorization may suffice. The analysis in the last chapter suggests that a strategy for finding $2k$ eigenvalues in a length k factorization is to find an appropriate starting vector that forces the residual r_{k+1} to vanish. The SR algorithm, on the other hand, computes approximations to all eigenvalues and eigenvectors of M. From Theorem 4.1 in [4] (an implicit Q-theorem for the SR case) we know that in exact arithmetic, when using the same starting vector, the SR algorithm and the length n Lanczos factorization generate the same symplectic butterfly matrices (up to multiplication by a trivial matrix). Forcing the residual for the symplectic Lanczos algorithm to zero has the effect of deflating a subdiagonal element during the SR algorithm: by Remark 3.1 $r_{k+1} = -d_{k+1}M_P v_{k+1}$ and from the symplectic Lanczos process we have $d_{k+1} = \|v_{k+1}\|_2$. Hence a zero residual implies a zero d_{k+1} such that deflation occurs for the corresponding butterfly matrix.

Our goal in this section will be to construct a starting vector that is a member of the invariant subspace of interest. Our approach is to implicitly restart the symplectic Lanczos factorization. This was first introduced by Sorensen [35] in the context of unsymmetric matrices and the Arnoldi process. The scheme is called implicit because the updating of the starting vector is accomplished with an implicit shifted SR mechanism on $B^{2j,2j}, j \leq n$. This allows to update the starting vector by working with a symplectic matrix in $\mathbb{R}^{2j\times 2j}$ rather than in $\mathbb{R}^{2n\times 2n}$ which is significantly cheaper.

The iteration starts by extending a length k symplectic Lanczos factorization by p steps. Next, $2p$ shifts are applied to $B^{2(k+p),2(k+p)}$ using double or quadruple SR steps. The last $2p$ columns of the factorization are discarded resulting in a length k factorization. The iteration is defined by repeating this process until convergence.

For simplicity let us first assume that $p = 1$ and that a $2n \times 2(k+1)$ matrix $S_P^{2n,2k+2}$ is known such that

$$M_P S_P^{2n,2k+2} = S_P^{2n,2k+2} B_P^{2k+2,2k+2} + r_{k+2} e_{2k+2}^T \tag{4.1}$$

as in (3.5). Let μ be a real shift and

$$q_2(B^{2k+2,2k+2}) = (B^{2k+2,2k+2} - \mu I)(B^{2k+2,2k+2} - \mu^{-1} I)(B^{2k+2,2k+2})^{-1} = SR.$$

Then (using (2.6) $S_P^{-1} B_P^{2k+2,2k+2} S_P$ will be a permuted butterfly matrix and S_P is an upper triangular matrix with two additional subdiagonals.

With this we can re-express (4.1) as

$$M_P(S_P^{2n,2k+2} S_P) = (S_P^{2n,2k+2} S_P)(S_P^{-1} B_P^{2k+2,2k+2} S_P) + r_{k+2} e_{2k+2}^T S_P.$$

Defining $\breve{S}_P^{2n,2k+2} = S_P^{2n,2k+2} S_P$, $\breve{B}_P^{2k+2,2k+2} = S_P^{-1} B_P^{2k+2,2k+2} S_P$ this yields

$$M_P \breve{S}_P^{2n,2k+2} = \breve{S}_P^{2n,2k+2} \breve{B}_P^{2k+2,2k+2} + r_{k+2} e_{2k+2}^T S_P. \tag{4.2}$$

The above equation fails to be a symplectic Lanczos factorization since the columns $2k, 2k+1, 2k+2$ of the matrix $d_{k+2}(b_{k+2}v_{k+2} + a_{k+2}w_{k+2})e_{2k+2}^T S_P$ are nonzero. Let s_{ij} be the (i,j)th entry of S_P. The residual term in (4.2) is

$$d_{k+2}(b_{k+2}v_{k+2} + a_{k+2}w_{k+2})(s_{2k+2,2k}e_{2k}^T + s_{2k+2,2k+1}e_{2k+1}^T + s_{2k+2,2k+2}e_{2k+2}^T).$$

Rewriting (4.2) as

$$M_P \breve{S}_P^{2n,2k+2} = [\breve{S}_P^{2n,2k},\ \breve{v}_{k+1},\ \breve{w}_{k+1},\ v_{k+2},\ w_{k+2}]Z$$

where Z is blocked as

$$\left[\begin{array}{c|c|c} \breve{B}_P^{2k,2k} & 0 & \breve{d}_{k+1}(\breve{b}_k e_{2k-1} + \breve{a}_k e_{2k}) \\ \hline \breve{b}_{k+1}\breve{d}_{k+1}e_{2k}^T & \breve{b}_{k+1} & \breve{b}_{k+1}\breve{c}_{k+1} - \breve{a}_{k+1}^{-1} \\ \breve{a}_{k+1}\breve{d}_{k+1}e_{2k}^T & \breve{a}_{k+1} & \breve{a}_{k+1}\breve{c}_{k+1} \\ \hline d_{k+2}b_{k+2}s_{2k+2,2k}e_{2k}^T & d_{k+2}b_{k+2}s_{2k+2,2k+1} & d_{k+2}b_{k+2}s_{2k+2,2k+2} \\ d_{k+2}a_{k+2}s_{2k+2,2k}e_{2k}^T & d_{k+2}a_{k+2}s_{2k+2,2k+1} & d_{k+2}a_{k+2}s_{2k+2,2k+2} \end{array}\right]$$

we obtain as a new Lanczos identity

$$M_P \breve{S}_P^{2n,2k} = \breve{S}_P^{2n,2k} \breve{B}_P^{2k,2k} + \breve{r}_{k+1} e_{2k}^T \tag{4.3}$$

where

$$\breve{r}_{k+1} = \breve{d}_{k+1}(\breve{b}_{k+1}\breve{v}_{k+1} + \breve{a}_{k+1}\breve{w}_{k+1}) + d_{k+2}s_{2k+2,2k}(b_{k+2}v_{k+2} + a_{k+2}w_{k+2}).$$

Here, $\breve{a}_{k+1}$, $\breve{b}_{k+1}$, $\breve{d}_{k+1}$ denote parameters of $\breve{B}_P^{2k+2,2k+2}$, while $a_{k+2}, b_{k+2}, d_{k+2}$ are parameters of $B_P^{2k+2,2k+2}$. In addition, $\breve{v}_{k+1}, \breve{w}_{k+1}$ are the last two column vectors from $\breve{S}_P^{2n,2k+2}$, while v_{k+2}, w_{k+2} are the two last column vectors of $S_P^{2n,2k+2}$.

As the space spanned by the columns of $S^{2n,2k+2} = (P_n)^T S_P^{2n,2k+2} P_{k+1}$ is J–orthogonal, and S_P is a permuted symplectic matrix, the space spanned by the columns of $\breve{S}^{2n,2k} = (P_n)^T \breve{S}_P^{2n,2k} P_k$ is J–orthogonal. Thus (4.3) is a valid symplectic Lanczos factorization. The new starting vector is $\breve{v}_1 = \rho q_2(M_P)v_1$ for some scalar $\rho \in \mathbb{R}$. This can be seen as follows: first note that for unreduced butterfly matrices $B^{2k+2,2k+2}$ we have $q_2(B_P^{2k+2,2k+2})e_1 \neq 0$. Hence, from $q_2(B_P^{2k+2,2k+2}) = S_P R_P$ we obtain $q_2(B_P^{2k+2,2k+2})e_1 = \rho S_P e_1$ for $\rho = e_1^T R_P e_1$ as R_P is an upper triangular matrix. As $q_2(B_P^{2k+2,2k+2})e_1 \neq 0$, we have $\rho \neq 0$. Using (4.3) it follows that

$$\begin{aligned} \breve{S}_P^{2n,2k} e_1 &= S_P^{2n,2k+2} S_P e_1 \\ &= \frac{1}{\rho} S_P^{2n,2k+2} q_2(B_P^{2k+2,2k+2}) e_1 \\ &= \frac{1}{\rho} S_P^{2n,2k+2} (B_P^{2k+2,2k+2} - \mu I)(B_P^{2k+2,2k+2} - \mu^{-1} I)(B_P^{2k+2,2k+2})^{-1} e_1 \\ &= \frac{1}{\rho} (M_P S_P^{2n,2k+2} - r_{k+2} e_{2k+2}^T - \mu S_P^{2n,2k+2})(I - \mu^{-1}(B_P^{2k+2,2k+2})^{-1}) e_1 \\ &= \frac{1}{\rho} (M_P S_P^{2n,2k+2} - \mu S_P^{2n,2k+2})(I - \mu^{-1}(B_P^{2k+2,2k+2})^{-1}) e_1 \end{aligned}$$

as $r_{k+2}e_{2k+2}^T(I-\mu^{-1}(B_P^{2k+2,2k+2})^{-1})e_1 = 0$. Thus using again (4.3) we get

$$\begin{aligned}
\breve{S}_P^{2n,2k}e_1 &= \frac{1}{\rho}(M_P-\mu I)(S_P^{2n,2k+2}-\mu^{-1}S_P^{2n,2k+2}(B_P^{2k+2,2k+2})^{-1})e_1 \\
&= \frac{1}{\rho}(M_P-\mu I)(S_P^{2n,2k+2}-\mu^{-1}M_P^{-1}S_P^{2n,2k+2})e_1 \\
&\qquad -\frac{1}{\rho}\mu^{-1}M_P^{-1}r_{k+2}e_{2k+2}^T(B_P^{2k+2,2k+2})^{-1}e_1 \\
&= \frac{1}{\rho}(M_P-\mu I)(I-\mu^{-1}M_P^{-1})S_P^{2n,2k+2}e_1 \\
&= q_2(M_P)v_1
\end{aligned}$$

as $e_{2k+2}^T(B_P^{2k+2,2k+2})^{-1})e_1 = 0$.

Note that in the symplectic Lanczos process the vectors v_j of $S_P^{2n,2k}$ satisfy the condition $||v_j||_2 = 1$ and the parameters b_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 $\breve{b}_j$ from $\breve{B}_P^{2k,2k}$ and thus for the new Lanczos factorization (4.3). Both properties could be forced using trivial factors. Numerical tests indicate that there is no obvious advantage in doing so.

Using standard polynomials as shift polynomials instead of Laurent polynomials as above results in the following situation: In $p_2(B_P^{2k+2,2k+2}) = (B_P^{2k+2,2k+2}-\mu I)(B_P^{2k+2,2k+2}-\mu^{-1}I) = S_PR_P\,S_P$ is an upper triangular matrix with four (!) additional subdiagonals. Therefore, the residual term in (4.2) has five nonzero entries. Hence not the last two, but the last four columns of (4.2) have to be discarded in order to obtain a new valid Lanczos factorization. That is, we would have to discard wanted information which is avoided by using Laurent polynomials.

This technique can be extended to the quadruple shift case using Laurent polynomials as the shift polynomials as discussed in Section 2. The implicit restart can be summarized as given in Table 4.1. In the course of the iteration we have to choose p shifts $\Delta = \{\mu_1,\ldots,\mu_p\}$ in order to apply $2p$ shifts: choosing a real shift μ_k implies that μ_k^{-1} is also a shift due to the symplectic structure of the problem. Hence, μ_k^{-1} is not added to Δ as the use of the Laurent polynomial q_2 guarantees that μ_k^{-1} is used as a shift once $\mu_k \in \Delta$. In case of a complex shift μ_k, $|\mu_k| = 1$, this implies that $\overline{\mu_k}$ is also a shift not added to Δ. For complex shifts μ_k, $|\mu_k| \neq 1$, we include μ_k, $\bar{\mu_k}$ in Δ.

Numerous choices are possible for the selection of the p shifts. One possibility is the case of choosing p "exact" shifts with respect to $B_P^{2(k+p),2(k+p)}$. That is, first the eigenvalues of $B_P^{2(k+p),2(k+p)}$ are computed (by the SR algorithm), then p unwanted eigenvalues are selected. One choice for this selection might be: sort the eigenvalues by decreasing magnitude. There will be $k+p$ eigenvalues with modulus greater than or equal to 1

$$\begin{aligned}
&|\lambda_1| \geq \ldots \geq |\lambda_k| \geq |\lambda_{k+1}| \geq \ldots |\lambda_{k+p}| \geq 1 \\
&\qquad \geq |\lambda_{k+p}^{-1}| \geq \ldots \geq |\lambda_{k+1}^{-1}| \geq |\lambda_k^{-1}| \geq \ldots \geq |\lambda_1^{-1}|.
\end{aligned}$$

Select the $2p$ eigenvalues with modulus closest to 1 as shifts. If λ_{k+1} is complex with $|\lambda_k| = |\lambda_{k+1}| \neq 1$, then we either have to choose $2p+2$ shifts or just $2p-2$ shifts, as λ_{k+1} belongs to a quadruple pair of eigenvalues of $B_P^{2(k+p),2(k+p)}$ and in order to preserve the symplectic structure either λ_k and λ_{k+1} have to be chosen or none.

Algorithm : k–step restarted symplectic Lanczos method

perform k steps of the symplectic Lanczos algorithm to compute $S_P^{2n,2k}$ and $B_P^{2k,2k}$
obtain the residual vector r_{k+1}
while $||r_{k+1}|| > tol$
    perform p additional steps of the symplectic Lanczos method
        to compute $S_P^{2n,2(k+p)}$ and $B_P^{2(k+p),2(k+p)}$
    select p shifts μ_i
    compute $\breve{B}_P^{2k,2k}$ and $\breve{S}_P^{2n,2k}$ via implicitly shifted SR steps
    set $S_P^{2n,2k} = \breve{S}_P^{2n,2k}$ and $B_P^{2k,2k} = \breve{B}_P^{2k,2k}$
    obtain the new residual vector r_{k+1}
end while

TABLE 4.1
k–step restarted symplectic Lanczos method

A different possibility of choosing the shifts is to keep those eigenvalues that are good approximations to eigenvalues of M. That is, eigenvalues for which (3.11) is small. Again we have to make sure that our set of shifts is complete in the sense described above.

Choosing eigenvalues of $B_P^{2(k+p),2(k+p)}$ as shifts has an important consequence for the next iterate. Assume for simplicity that $B_P^{2(k+p),2(k+p)}$ is diagonalizable. Let $\lambda(B_P^{2(k+p),2(k+p)}) = \{\theta_1, \ldots, \theta_{2k}\} \cup \{\mu_1, \ldots, \mu_{2p}\}$ be a disjoint partition of the spectrum of $B_P^{2(k+p),2(k+p)}$. Selecting the exact shifts $\mu_1, \ldots, \mu_{2p}$ in the implicit restart, following the rules mentioned above yields a matrix

$$\breve{B}_P^{2(k+p),2(k+p)} = \begin{bmatrix} \breve{B}_P^{2k,2k} & X \\ 0 & Y \end{bmatrix}$$

where $\lambda(\breve{B}_P^{2k,2k}) = \{\theta_1, \ldots, \theta_{2k}\}$ and $\lambda(Y) = \{\mu_1, \ldots, \mu_{2p}\}$. This follows from (2.6). Moreover, the new starting vector has been implicitly replaced by the sum of $2k$ approximate eigenvectors:

$$\breve{v}_1 = S_P^{2n,2(k+p)} S_P e_1 = \frac{1}{\rho} S_P^{2n,2(k+p)} q(B_P^{2(k+p),2(k+p)}) e_1 = \frac{1}{\rho} S_P^{2n,2(k+p)} \sum_{j=1}^{2k} \zeta_j y_j$$

where $\rho = e_1^T R_P e_1$, $B_P^{2(k+p),2(k+p)} y_j = \theta_j y_j$, and ζ_j is properly chosen. The last equation follows since $q(B_P^{2(k+p),2(k+p)}) e_1$ has no component along an eigenvector of $B_P^{2(k+p),2(k+p)}$ associated with $\mu_j, 1 \le j \le 2p$. Hence

$$\breve{v}_1 = \rho \sum_{j=1}^{2k} \zeta_j x_j \qquad \text{where } S_P^{2n,2(k+p)} y_j = x_j.$$

It should be mentioned that the k–step restarted symplectic Lanczos method as in Table 4.1 with exact shifts builds a J–orthogonal basis for a number of generalized Krylov subspaces simultaneously. The subspace of length $2(k+p)$ generated during a restart using exact shifts contains all the Krylov subspaces of dimension $2k$ generated from each of the desired Ritz vectors, for a detailed discussion see [13]. A similar

observation for Sorensen's restarted Arnoldi method with exact shifts was made by Morgan in [30]. For a discussion of this observation see [30] or [23]. Morgan infers *'the method works on approximations to all of the desired eigenpairs at the same time, without favoring one over the other'* [30, p. 1220,l. 7–8 from the bottom]. This remark can also be applied to the method presented here.

In the above discussion we have assumed that the permuted SR decomposition $q(B_P^{2(k+p),2(k+p)}) = S_P R_P$ exists. Unfortunately, this is not always true. During the bulge-chase in the implicit SR step, it may happen that a diagonal element a_j of B_1 (2.2) is zero (or almost zero). In that case no reduction to symplectic butterfly form with the corresponding first column $\breve{v}_1$ does exist. In the next section we will prove that a serious breakdown in the symplectic Lanczos algorithm is equivalent to such a breakdown of the SR decomposition. Moreover, it may happen that a subdiagonal element d_j of the $(2,2)$–block of B_2^{-1} (2.3) is zero (or almost zero) such that

$$\breve{B}_P^{2(k+p),2(k+p)} = \begin{bmatrix} \breve{B}_P^{2j,2j} & \\ & \hat{B}_P \end{bmatrix}.$$

The matrix $\breve{B}_P^{2(k+p),2(k+p)}$ is split, an invariant subspace of dimension j is found. If $j \geq k$ and all shifts have been applied, then the iteration is halted. Otherwise we continue similar to the procedure described by Sorensen in [35, Remark 3].

As the iteration progresses, some of the Ritz values may converge to eigenvalues of M 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. Lehoucq and Sorensen develop in [23, 36] locking and purging techniques to accomplish this in the context of unsymmetric matrices and the restarted Arnoldi method. These ideas can be carried over to the situation here.

5. Numerical Properties of the Implicitly Restarted Symplectic Lanczos Method.

5.1. 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 [32, 37]. Thus, Banse proposes in [2] 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

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

For the resulting algorithm and a discussion of it we refer to [2]. It is easy to see that $S_P^{-1}B_PS_P$ is no longer a permuted symplectic matrix, but it still has the desired form of a butterfly matrix. Unfortunately, an SR step does not preserve the structure of $S_P^{-1}B_PS_P$ 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_P–orthogonalize each Lanczos vector as soon as it is computed against the previous ones via

$$w_m \leftarrow w_m + \sum_{j=1}^{m-1} (< v_j, w_m >_{J_P} \; w_j - < w_j, w_m >_{J_P} \; v_j)\,,$$

$$v_{m+1} \leftarrow v_{m+1} + \sum_{j=1}^{m} (< v_j, v_{m+1} >_{J_P} \; w_j - < w_j, v_{m+1} >_{J_P} \; v_j)$$

where for $x, y \in \mathbb{R}^{2n}$, $< x, y >_{J_P^n} := x^T J_P^n y$ defines the indefinite inner product implied by J_P^n.

This re–J_P–orthogonalization is costly, it requires $16n(m-1)$ flops for the vector w_m and $16nm$ flops for v_{m+1}. Thus, if $2k$ Lanczos vectors $v_1, w_1, \ldots, v_k, w_k$ are computed, the re–J_P–orthogonalization adds a computational cost of the order of $16nk^2$ 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., [17]). Those ideas can be used to design analogous re–J_P–orthogonalizations for the symplectic Lanczos method. It should be noted that if k is small, the cost for re–J_P–orthogonalization is not too expensive.

Another important issue is the numerical stability of the SR step employed in the restart. During the SR step on the $2k \times 2k$ symplectic butterfly 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 (see [8]).

5.2. Breakdowns in the SR Factorization. If there is a starting vector $\breve{v}_1 = \rho q(M)v_1$ for which the explicitly restarted symplectic Lanczos method breaks down, then it is impossible to reduce the symplectic matrix M to symplectic butterfly form with a transformation matrix whose first column is $\breve{v}_1$. Thus, in this situation the SR decomposition of $q(B)$ 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 SR step, most of the transformations used are orthogonal symplectic transformations; their computation can not break down. The only source of breakdown can be one of the symplectic Gaussian eliminations L_j. For simplicity, we will discuss the double shift case. Only the following elementary elimination matrices are used in the implicit SR step: elementary symplectic Givens matrices [31]

$$G_k = \begin{bmatrix} C_k & -S_k \\ S_k & C_k \end{bmatrix},$$

where

$$C_k = I + (c_k - 1)e_k e_k^T, \qquad S_k = s_k e_k e_k^T, \qquad c_k^2 + s_k^2 = 1,$$

elementary symplectic Householder transformation

$$H(k,v) = \left[\begin{array}{cc|cc} I_{k-1} & & & \\ & P & & \\ \hline & & I_{k-1} & \\ & & & P \end{array}\right], \qquad \text{where } P = I_{n-k+1} - 2\frac{vv^T}{v^Tv},$$

and elementary symplectic Gaussian elimination matrices [8]

$$L_k = \begin{bmatrix} W_k & V_k \\ 0 & W_k^{-1} \end{bmatrix},$$

where

$$W_k = I + (w_k - 1)(e_{k-1}e_{k-1}^T + e_k e_k^T), \qquad V_k = v_k(e_{k-1}e_k^T + e_k e_{k-1}^T).$$

Assume that k steps of the symplectic Lanczos algorithm are performed, then from (3.5)

$$M_P S_P^{2n,2k} = S_P^{2n,2k} B_P^{2k,2k} + r_{k+1}e_{2k}^T. \tag{5.1}$$

Now an implicit restart is to be performed using an implicit double shift SR step. In the first step of the implicit SR step, a symplectic Householder matrix H_1 is computed such that

$$H_1^T q(B^{2k,2k})e_1 = \lambda e_1.$$

H_1 is applied to $B^{2k,2k}$

$$H_1^T B^{2k,2k} H_1$$

introducing a small bulge in the butterfly form: additional elements are found in the positions $(2,1)$, $(1,2)$, $(n+2,n+1)$, $(n+1,n+2)$, $(1,n+3)$, $(3,n+1)$, $(n+1,n+3)$ and $(n+3,n+1)$. The remaining implicit transformations perform a bulge-chasing sweep down the subdiagonal to restore the butterfly form. An algorithm for this is given in [2] or [4]; it can be summarized for the situation here as in Table 5.1, where $\tilde{G}_j$ and G_j both denote symplectic Givens transformation matrices acting in the same planes but with different rotation angles.

for $\ell = 1 : n-1$
compute $G_{\ell+1}$ such that $(G_{\ell+1}B^{2k,2k})_{n+\ell+1,\ell} = 0$
$B^{2k,2k} = G_{\ell+1}B^{2k,2k}G_{\ell+1}^T$
compute $L_{\ell+1}$ such that $(L_{\ell+1}B^{2k,2k})_{\ell+1,\ell} = 0$
$B^{2k,2k} = L_{\ell+1}B^{2k,2k}L_{\ell+1}^{-1}$
compute $\tilde{G}_{\ell+1}$ such that $(B^{2k,2k}\tilde{G}_{\ell+1})_{\ell,\ell+1} = 0$
$B^{2k,2k} = \tilde{G}_{\ell+1}^T B^{2k,2k}\tilde{G}_{\ell+1}$
compute $H_{\ell+1}$ such that $(B^{2k,2k}H_{\ell+1})_{\ell,n+\ell+2} = 0$
$B^{2k,2k} = H_{\ell+1}^T B^{2k,2k}H_{\ell+1}$
end

TABLE 5.1
Reduction to butterfly form – double shift case.

Suppose that the first $j-1$ Gaussian transformations, $j < k$, exist and that we have computed

$$\widehat{S} = H_1 G_2^T L_2^{-1} \tilde{G}_2 \dots H_{j-2} G_{j-1}^T L_{j-1}^{-1} \tilde{G}_{j-1} H_{j-1} G_j^T$$

In order to simplify the notation, we switch to the permuted version and rewrite the permuted symplectic matrix $\widehat{S}_P$ as

$$\widehat{S}_P = \begin{bmatrix} S_P & 0 \\ 0 & I_{2n-2j-2} \end{bmatrix}$$

where $S_P \in \mathbb{R}^{(2j+2)\times(2j+2)}$ making use of the fact that the accumulated transformations affect only the rows 1 to j and $n+1$ to $n+j$. The leading $(2j+2)\times(2j+2)$ principal submatrix of

$$\widehat{S}_P^{-1} B_P^{2k,2k} \widehat{S}_P$$

is given by

$$(5.2)\quad \widetilde{B}_P^{2j+2,2j+2} = \left[\begin{array}{c|cc|cc|cc} \breve{B}^{2j-4,2j-4} & 0 & \breve{b}_{j-2}\breve{d}_{j-1}e_{2j-5} & & & & \\ & 0 & \breve{a}_{j-2}\breve{d}_{j-1}e_{2j-4} & & & & \\ \hline \breve{b}_{j-1}\breve{d}_{j-1}e_{2j-4}^T & \breve{b}_{j-1} & \breve{b}_{j-1}\breve{c}_{j-1} - \breve{a}_{j-1}^{-1} & 0 & \hat{x} & & \\ \breve{a}_{j-1}\breve{d}_{j-1}e_{2j-4}^T & \breve{a}_{j-1} & \breve{a}_{j-1}\breve{c}_{j-1} & 0 & \hat{x} & & \\ \hline & 0 & \hat{y}_1 & \hat{b}_j & \hat{x} & \hat{x} & \hat{x} \\ & 0 & \hat{y}_2 & \hat{a}_j & \hat{x} & \hat{x} & \hat{x} \\ \hline & 0 & \hat{x}_1 & \hat{x}_2 & \hat{x} & \hat{x} & \hat{x} \\ & 0 & 0 & 0 & \hat{x} & \hat{x} & \hat{x} \end{array}\right]$$

where the hatted quantities denote unspecified entries that would change if the SR update could be continued. Next, the $(2j+1, 2j-1)$ entry should be annihilated by a permuted symplectic Gaussian elimination. This elimination will fail to exist if $\hat{a}_j = 0$; the SR decomposition of $q(B^{2k,2k})$ does not exist.

As will be needed later, $\hat{a}_j = 0$ implies that $\hat{y}_2 = 0$. This follows as $\widetilde{B}_P^{2j+2,2j+2}$ is J_P–orthogonal: From

$$e_{2j-2}^T \widetilde{B}_P^{2j+2,2j+2} J_P (\widetilde{B}_P^{2j+2,2j+2})^T e_{2j} = e_{2j-2}^T J_P e_{2j} = 0$$

we obtain

$$0 = [0 \cdots 0\ 0\ \breve{a}_{j-1}\breve{d}_{j-1}\ \breve{a}_{j-1}\ \breve{a}_{j-1}\breve{c}_{j-1}\ 0\ \hat{x}\ 0\ 0] J_P \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \hat{y}_2 \\ \hat{a}_j \\ \hat{x} \\ \hat{x} \\ \hat{x} \end{bmatrix} = -\breve{a}_{j-1}\hat{y}_2 - \hat{x}\hat{a}_j.$$

If $\hat{a}_j = 0$ we have $\hat{y}_2 = 0$ as $\breve{a}_{j-1} \neq 0$ (otherwise the last Gaussian transformation L_{j-1} did not exist).

Next we show that this breakdown in the SR decomposition implies a breakdown in the Lanczos process started with the starting vector $\breve{v}_1 = \rho q(M_P) v_1$.

For this we have to consider (5.1) multiplied from the right by $\widehat{S}_P$. From the derivations in the last section we know that the starting vector of that recursion is given by $\breve{v}_1 = \rho q(M_P)v_1$. As the trailing $(2n-2j-2)\times(2n-2j-2)$ principal submatrix of $\widehat{S}_P$ is the identity, we can just as well consider

$$M_P S_P^{2n,2j+2} = S_P^{2n,2j+2} B_P^{2j+2,2j+2} + r_{j+2} e_{2j+2}^T,$$

multiplied from the right by S_P

$$M_P \breve{S}_P^{2n,2j+2} = \breve{S}_P^{2n,2j+2} \breve{B}_P^{2j+2,2j+2} + r_{j+2} e_{2j+2}^T S_P, \tag{5.3}$$

where $\breve{B}_P^{2j+2,2j+2} = S_P^{-1} B_P^{2j+2,2j+2} S_P$ corresponds to the matrix in (5.2) (no butterfly form!) and $\breve{S}_P^{2n,2j+2} = S_P^{2n,2j+2} S_P = [\breve{v}_1, \breve{w}_1, \ldots, \breve{v}_{j-1}, \breve{w}_{j-1}, \hat{v}_j, \hat{w}_j, \hat{v}_{j+1}, \hat{w}_{j+1}]$. The columns of $\breve{S}_P^{2n,2j+2}$ are J_P–orthogonal

$$(\breve{S}_P^{2n,2j+2})^T (J_n)_P \breve{S}_P^{2n,2j+2} = (J_{j+1})_P. \tag{5.4}$$

The starting vector of the recursion (5.3) is given by $\breve{v}_1 = \rho q(M_P)v_1$. Deleting the last four columns of $\breve{S}_P^{2n,2j+2}$ in the same way as in the implicit restart we obtain a valid symplectic Lanczos factorization of length $2j-2$.

In order to show that a breakdown in the SR decomposition of $q(B)$ implies a breakdown in the above symplectic Lanczos recursion, we need to show

$$\hat{a}_j = 0 \quad \Longrightarrow \quad \breve{a}_j = \breve{v}_j^T (J_n)_P M_P \breve{v}_j = 0.$$

From (5.2) and (5.3) we obtain

$$\begin{aligned} M_P \breve{w}_{j-1} = {} & \breve{b}_{j-2}\breve{d}_{j-1}\breve{v}_{j-2} + \breve{a}_{j-2}\breve{d}_{j-1}\breve{w}_{j-2} + (\breve{b}_{j-1}\breve{c}_{j-1} - \breve{a}_{j-1}^{-1})\breve{v}_{j-1} \\ & + \breve{a}_{j-1}\breve{c}_{j-1}\breve{w}_{j-1} + \hat{y}_1\hat{v}_j + \hat{y}_2\hat{w}_j + \hat{x}_1\hat{v}_{j+1}, \end{aligned} \tag{5.5}$$

and

$$M_P \breve{v}_k = \breve{b}_k \breve{v}_k + \breve{a}_k \breve{w}_k, \qquad k \le j-1. \tag{5.6}$$

Further we do know from the symplectic Lanczos algorithm

$$\breve{v}_j = -\breve{d}_{j-1}\breve{v}_{j-2} - \breve{c}_{j-1}\breve{v}_{j-1} + \breve{w}_{j-1} + \breve{a}_{j-1}^{-1} M_P^{-1} \breve{v}_{j-1}, \tag{5.7}$$

all of these quantities are already known. Now consider

$$\begin{aligned} \breve{a}_j = \breve{v}_j^T J_P M_P \breve{v}_j = {} & -\underbrace{\breve{d}_{j-1}\breve{v}_j^T J_P M_P \breve{v}_{j-2}}_{x_1} - \underbrace{\breve{c}_{j-1}\breve{v}_j^T J_P M_P \breve{v}_{j-1}}_{x_2} \\ & + \underbrace{\breve{v}_j^T J_P M_P \breve{w}_{j-1}}_{x_3} + \underbrace{\breve{a}_{j-1}^{-1}\breve{v}_j^T J_P \breve{v}_{j-1}}_{x_4}. \end{aligned}$$

Obviously, $x_4 = 0$. Using (5.6) we obtain

$$\breve{v}_j^T J_P M_P \breve{v}_k = \breve{b}_k \breve{v}_j^T J_P \breve{v}_k + \breve{a}_k \breve{v}_j^T J_P \breve{w}_k = 0$$

for $k = j-1, j-2$. Hence $x_1 = x_2 = 0$. Using (5.5) end (5.4) will see that $x_3 = 0$:

$$\begin{aligned} \breve{v}_j^T J_P M_P \breve{w}_{j-1} = {} & \breve{b}_{j-2}\breve{d}_{j-1}\breve{v}_j^T J_P \breve{v}_{j-2} + \breve{a}_{j-2}\breve{d}_{j-1}\breve{v}_j^T J_P \breve{w}_{j-2} + \\ & (\breve{b}_{j-1}\breve{c}_{j-1} - \breve{a}_{j-1}^{-1})\breve{v}_j^T J_P \breve{v}_{j-1} + \breve{a}_{j-1}\breve{c}_{j-1}\breve{v}_j^T J_P \breve{w}_{j-1} + \\ & \hat{y}_1 \breve{v}_j^T J_P \hat{v}_j + \hat{y}_2 \breve{v}_j^T J_P \hat{w}_j + \hat{x}_1 \breve{v}_j^T J_P \hat{v}_{j+1} \\ = {} & \underbrace{\hat{y}_1 \breve{v}_j^T J_P \hat{v}_j}_{z_1} + \underbrace{\hat{y}_2 \breve{v}_j^T J_P \hat{w}_j}_{z_2} + \underbrace{\hat{x}_1 \breve{v}_j^T J_P \hat{v}_{j+1}}_{z_3}. \end{aligned}$$

As $\hat{a}_j = 0$, $\hat{y}_2 = 0$ and therefore $z_2 = 0$. With (5.7) we get

$$\begin{aligned} z_1 &= -\hat{y}_1 \hat{v}_j^T J_P \breve{v}_j \\ &= \hat{y}_1 (\breve{d}_{j-1} \hat{v}_j^T J_P \breve{v}_{j-2} - \breve{c}_{j-1} \hat{v}_j^T J_P \breve{v}_{j-1} - \hat{v}_j^T J_P \breve{w}_{j-1} - \breve{a}_{j-1}^{-1} \hat{v}_j^T J_P M_P^{-1} \breve{v}_{j-1}) \\ &= -\hat{y}_1 \breve{a}_{j-1}^{-1} \hat{v}_j^T M_P^T J_P \breve{v}_{j-1}. \end{aligned}$$

From (5.3) we obtain

$$M_P \hat{v}_j = \hat{b}_j \hat{v}_j + \hat{a}_j \hat{w}_j + \hat{x}_2 \hat{v}_{j+1}.$$

Hence using (5.4) yields

$$z_1 = -\hat{y}_1 \breve{a}_{j-1}^{-1} (\hat{b}_j \hat{v}_j^T J_P \breve{v}_{j-1} + \hat{a}_j \hat{w}_j^T J_P \breve{v}_{j-1} + \hat{x}_2 \hat{v}_{j+1}^T J_P \breve{v}_{j-1}) = 0.$$

Similar, it follows that $z_3 = 0$. Hence $x_3 = 0$, and therefore $\breve{v}_j^T J_P M_P \breve{v}_j = 0$.

This argumentation has shown that an SR breakdown implies a serious Lanczos breakdown. The opposite implication follows from the uniqueness of the Lanczos factorization. The result is summarized in the following theorem.

THEOREM 5.1. *Suppose the symplectic butterfly matrix $B^{2k,2k}$ corresponding to (3.5) is unreduced and let $\mu \in \mathbb{R}$. Let L_j be the jth symplectic Gauss transformation required in the SR step on $(B^{2k,2k} - \mu I)((B^{2k,2k} - \mu^{-1} I)(B^{2k,2k})^{-1}$. If the first $j-1$ symplectic Gauss transformations of this SR step exist, then L_j fails to exist if and only if $\breve{v}_j^T J_P M_P \breve{v}_j = 0$ with $\breve{v}_j$ as in (4.3).*

6. Numerical Experiments. Some examples to demonstrate the properties of the (implicitly restarted) symplectic Lanczos method are presented. The computational results are quite promising but certainly preliminary. All computations were done using MATLAB Version 5.1 on a Sun Ultra 1 with IEEE double-precision arithmetic and machine precision $\epsilon = 2.2204 \times 10^{-16}$.

Our code implements exactly the algorithm as given in Table 4.1. In order to detect convergence in the restart process, the rather crude criterion

$$||r_{k+1}|| \leq ||M|| * 10^{-6}$$

was used. This ad hoc stopping rule allowed the iteration to halt quite early. Usually, the eigenvalues largest in modulus (and their reciprocals) of the wanted part of the spectrum are much better approximated than the ones of smaller modulus. In a black-box implementation of the algorithm this stopping criterion has to be replaced with a more rigorous one to ensure that all eigenvalues are approximated to the desired accuracy (see the discussion in Section 3.3). Benign breakdown in the symplectic Lanczos process was detected by the criterion

$$||v_{m+1}|| \leq \epsilon * ||M|| \quad \text{or} \quad ||w_{m+1}|| \leq \epsilon * ||M||,$$

while a serious breakdown was detected by

$$v_{m+1} \neq 0, \quad w_{m+1} \neq 0, \quad |a_{m+1}| \leq \epsilon * ||M||.$$

Our implementation intends to compute the k eigenvalues of M largest in modulus and their reciprocals. In the implicit restart, we used exact shifts where we chose the shifts to be the $2p$ eigenvalues of $B^{2k+p,2k+p}$ closest to the unit circle.
Our observations have been the following.

- Re–J–orthogonalization is necessary; otherwise J–orthogonality of the computed Lanczos vectors is lost after a few steps, and *ghost eigenvalues* (see, e.g., [17]) appear. That is, multiple eigenvalues of $B^{2k,2k}$ correspond to simple eigenvalues of M.
- The implicit restart is more accurate than the explicit one.
- The leading end of the 'wanted' Ritz values (that is, the eigenvalues largest in modulus and their reciprocals) converge faster than the tail end (closest to cut off of the sort). The same behavior was observed in [35] for the implicitly restarted Arnoldi method. In order to obtain faster convergence, it seems advisable (similar to the implementation of Sorensen's implicitly restarted Arnoldi method in MATLAB's `eigs`) to increase the dimension of the computed Lanczos factorization. That is, instead of computing $S_P^{2n,2k}$ and $B_P^{2k,2k}$ as a basis for the restart, one should compute a slightly larger factorization, e.g. dimension $2(k+3)$ instead of dimension $2k$. When 2ℓ eigenvalues have converged, a subspace of dimension $2(k+3+\ell)$ is computed as a basis for the restart, followed by p additional Lanczos steps to obtain a factorization of length $k+3+\ell+p$. Using implicit SR steps this factorization is reduced to one of length $k+3+\ell$. If the symplectic Lanczos method would be implemented following this approach, the convergence check could be done using only the k Ritz values of largest modulus (and their reciprocals) or those that yield the smallest *Ritz residual*

$$|d_{k+1}|\,|e_{2k}^T y_j|\,||b_{k+1}\hat{v}_{k+1} + a_{k+1}\hat{w}_{k+1}||$$

  where the y_j are the eigenvectors of $B^{2k,2k}$.
- It is fairly difficult to find a good choice for k and p. Not for every possible choice of k, there exists an invariant subspace of dimension $2k$ associated to the k eigenvalues λ_i largest in modulus and their reciprocals. If λ_k is complex and $\overline{\lambda_{k+1}} = \lambda_k$ then we can not choose the $2p$ eigenvalues with modulus closest to the unit circle as shifts as this would tear a quadruple of eigenvalues apart resulting in a shift polynomial q such that $q(B_P^{2(k+p),2(k+p)}) \notin \mathbb{R}$. All we can do is to choose the $2p-2$ eigenvalues with modulus closest to 1 as shifts. In order to get a full set of $2p$ shifts we add as the last shift the real eigenvalue pair with largest Ritz residual. Depending on how good that real eigenvalue approximates an eigenvalue of M, this strategy worked, but the resulting subspace is no longer the subspace corresponding to the k eigenvalues largest in modulus and their reciprocals. If the real eigenvalue has converged to an eigenvalue of M, it is unlikely to remove that eigenvalue just by restarting, it will keep coming back. Only a purging technique like the one discussed by Lehoucq and Sorensen [23, 36] will be able to remove this eigenvalue. Moreover, there is no guarantee that there is a real eigenvalue of $B_P^{2(k+p),2(k+p)}$ that can be used here. Hence, in a black-box implementation one should either try to compute an invariant subspace of dimension $2(k-1)$ or of dimension $2(k+1)$. As this is not known a priori, the algorithm should adapt k during the iteration process appropriately. This is no problem, if as suggested above, one always computes a slightly larger Lanczos factorization than requested.

EXAMPLE 6.1. The first test performed concerned the loss of J–orthogonality of the computed Lanczos vectors during the symplectic Lanczos method and the ghost

eigenvalue problem (see, e.g. [17]). To demonstrate the effects of re–J–orthogonalization, a 100×100 symplectic matrix with eigenvalues

$$200, 100, 50, 47, \ldots, 4, 3, 2 \pm \mathtt{i}, 1/3, 1/4, \ldots, 1/47, 1/50, 1/100, 1/200$$

was used. A symplectic block-diagonal matrix with these eigenvalues on the block-diagonal was constructed and a similarity transformation with a randomly generated orthogonal symplectic matrix was performed to obtain a symplectic matrix M.

As expected, when using a random starting vector M's eigenvalues largest in modulus (and the corresponding reciprocals) tend to emerge right from the start, e.g., the eigenvalues of $B^{10,10}$ are

$$199.99997,\ 100.06771,\ 48.71752,\ 26.85083,\ 8.32399$$

and their reciprocals. Without any form of re–J–orthogonalization, the J–orthogonality of the Lanczos vectors is lost after a few iterations as indicated in Figure 6.1.

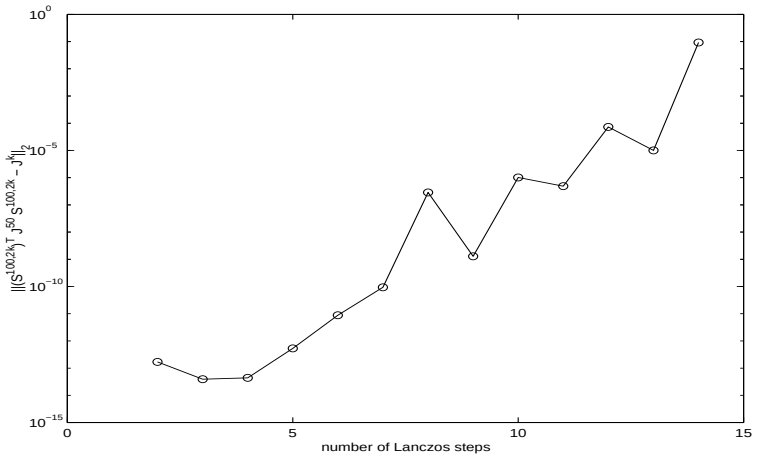


FIG. 6.1. *loss of J–orthogonality after k symplectic Lanczos steps*

The loss of J–orthogonality in the Lanczos vectors results, as in the standard Lanczos algorithm, in ghost eigenvalues. That is, multiple eigenvalues of $B^{2k,2k}$ correspond to simple eigenvalues of M. For example, using no re–J–orthogonalization, after 17 iterations the 6 eigenvalues largest in modulus of $B^{34,34}$ are

$$207.63389,\ 200,\ 100,\ 49.99982,\ 47.04542,\ 45.85367.$$

Using complete re–J–orthogonalization, this effect is avoided:

$$200,\ 100,\ 49.99992,\ 47.02461,\ 45.93018,\ 42.31199.$$

The second test performed concerned the question whether an implicit restart is more accurate than an explicit one. After nine steps of the symplectic Lanczos method (with a random starting vector) the resulting 18×18 symplectic butterfly matrix $B^{18,18}$ had the eigenvalues (using the MATLAB function `eig`)

200.000000000000	99.999999841718
50.070648930465	41.873264094053
35.891491504806	23.654512559868
13.344815062428	3.679215125563 ±5.750883779240i

and their reciprocals. Removing the 4 complex eigenvalues from $B^{18,18}$ using an implicit restart as described in Section 4, we obtain a symplectic butterfly matrix $B^{14,14}_{impl}$ whose eigenvalues are

200.000000000000	99.999999841719
50.070648930464	41.873264094053
35.891491504806	23.654512559868
13.344815062428	

and their reciprocals. From (2.6) it follows that these have to be the 14 real eigenvalues of $B^{18,18}$ which have not been removed. As can be seen, we lost one digit during the implicit restart (indicated by the 'underbar' under the 'lost' digits in the above table). Performing an explicit restart with the explicitly computed new starting vector $\breve{v}_1 = (M_P - \mu I)(M_P - \overline{\mu} I)(M_P - \mu^{-1} I)(M_P - \overline{\mu}^{-1} I) M_P^{-2} v_1$ yields a symplectic butterfly matrix $B^{14,14}_{expl}$ whose eigenvalues are

200.000000000000	99.999999841793
50.070648885030	41.873247045627
35.891922701991	23.654509163541
13.344810484061	

and their reciprocals. This time we lost up to nine digits.

The last set of tests performed on this matrix concerned the k–step restarted symplectic Lanczos method as given in Table 4.1. As M has only one quadruple of complex eigenvalues, and these eigenvalues are smallest in magnitude there is no problem in choosing $k \ll n$. For every such choice there exists an invariant symplectic subspace corresponding to the k eigenvalues largest in magnitude and their reciprocals. In the tests reported here, a random starting vector was used. Figure 6.2 shows a plot of $||r_{k+1}||$ versus the number of iterations performed. Iteration Step 1 refers to the norm of the residual after the first k Lanczos steps, no restart is performed. The three lines in Figure 6.2 present three different choice for k and p: $k = p = 8, k = 8, p = 16$ and $k = 5, p = 10$. Convergence was achieved for all three examples (and many more, not shown here). Obviously, the choice $k = 8, p = 2k$ results in faster convergence than the choice $k = p = 8$. Convergence is by no means monotonic, during the major part of the iteration the norm of the residual is changing quite dramatically. But once a certain stage is achieved, the norm of the residual converges. Although convergence for $k = 8, p = k$ or $p = 2k$ was quite fast, this does not imply that convergence is as fast for other choices of k and p. The third line in Figure 6.2 demonstrates that the convergence for $k = 5, p = 10$ does need twice as many iteration steps as for $k = 8, p = 16$.

EXAMPLE 6.2. Symplectic matrix pencils that appear in discrete-time linear-quadratic optimal control problems are typically of the form

$$L - \lambda N = \begin{bmatrix} F & 0 \\ C^T C & I \end{bmatrix} - \lambda \begin{bmatrix} I & -BB^T \\ 0 & F^T \end{bmatrix}, \qquad F \in \mathbb{R}^{n \times n},\ C \in \mathbb{R}^{p \times n},\ B \in \mathbb{R}^{n \times m}.$$

(Note: For $F \neq I$, L and N are not symplectic, but $L - \lambda N$ is a symplectic matrix pencil.) Assuming that L and N are nonsingular (that is, F is nonsingular), solving this generalized eigenproblem is equivalent to solving the eigenproblem for the symplectic matrix

$$N^{-1} L = \begin{bmatrix} I & -BB^T \\ 0 & F^T \end{bmatrix}^{-1} \begin{bmatrix} F & 0 \\ C^T C & I \end{bmatrix}.$$

If one is interested in computing a few of the eigenvalues of $L - \lambda N$, one can use the

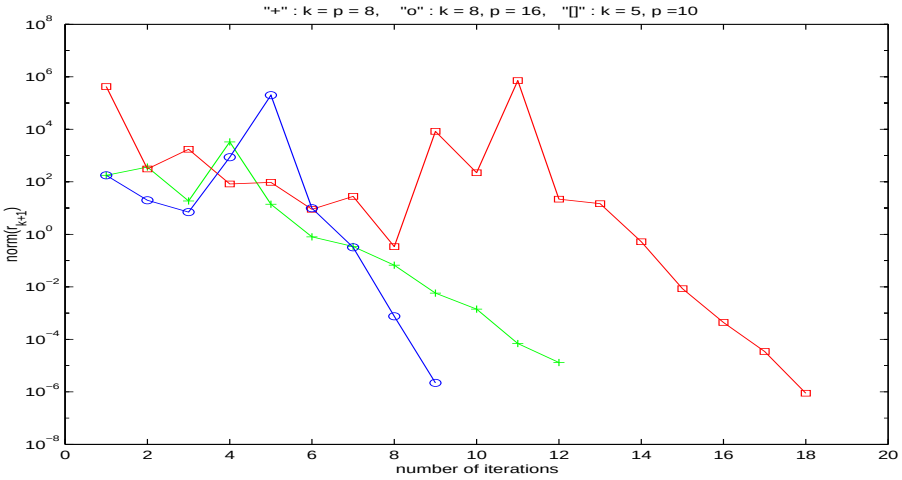


FIG. 6.2. *k–step restarted symplectic Lanczos method, different choices of k and p*

restarted symplectic Lanczos algorithm on $M = N^{-1}L$. In each step of the symplectic Lanczos algorithm, one has to compute matrix-vector products of the form Mx and M^Tx. Making use of the special form of L and N this can be done without explicitly inverting N: Let us consider the computation of $y = Mx$. First compute

$$Lx = \begin{bmatrix} F & 0 \\ C^TC & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} Fx_1 \\ C^TCx_1 + x_2 \end{bmatrix} =: \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = z$$

where $x \in \mathbb{R}^{2n}$ is written as $x = [x_1 \ x_2]^T, x_1, x_2 \in \mathbb{R}^n$. Next one has to solve the linear system $Ny = z$. Partition $y \in \mathbb{R}^{2n}$ analogous to x and z, then from $Ny = z$ we obtain

$$y_2 = F^{-T}z_2, \qquad y_1 = z_1 + BB^Ty_2.$$

In order to solve $y_2 = F^{-T}z_2$ we compute the LU decomposition of F and solve the linear system $F^Ty_2 = z_2$ using backward and forward substitution. Hence, the explicit inversion of N or F is avoided. In case F is a sparse matrix, sparse solvers can be employed. In particular, if the control system comes from some sort of discretization scheme, F is often banded which can be used here by computing an initial band LU factorization of F in order to minimize the cost for the computation of y_2. Note that in most applications, $p, m \ll n$ such that the computational cost for C^TCx_1 and BB^Ty_2 is significantly cheaper than a matrix-vector product with an $n \times n$ matrix. In case of single-input ($m = 1$) or single-output ($p = 1$) the corresponding operations come down to two dot products of length n each.

Using MATLAB's sparse matrix routine `sprandn` sparse normally distributed random matrices F, B, C (here, $p = m = n$) of different dimensions and with different densities of the nonzero entries were generated. Here an example of dimension $2n = 1000$ is presented, where the density of the different matrices was chosen to be

matrix	$\approx$ nonzero entries
F	$0.5n^2$
B	$0.2n^2$
C	$0.3n^2$

MATLAB computed the norm of the corresponding matrix $M = N^{-1}L$ to be $\approx 5.3 \times 10^5$.

In the first set of tests k was chosen to be 5, and we tested $p = k$ and $p = 2k$. As can be seen in Figure 6.3, for the first 3 iterations, the norm of the residual decreases for both choice of p, but then increases quite a bit. During the first step, the eigenvalues of $B^{10,10}$ are approximating the 5 eigenvalues of $L - \lambda N$ largest in modulus and their reciprocals. In step 4, a 'wrong' choice of the shifts is done in both cases. The extended matrices $B^{20,20}$ and $B^{30,30}$ both still approximate the 5 eigenvalues of $L - \lambda N$ largest in modulus, but there is a new real eigenvalue coming in, which is not a good approximation to an eigenvalue of $L - \lambda N$. But, due to the way the shifts are chosen here, this new eigenvalue is kept, while an already good approximated eigenvalue — a little smaller in magnitude — is shifted away, resulting in a dramatic increase of $||r_{k+1}||$. Modifying the choice of the shifts such that the good approximation is kept, while the new real eigenvalue is shifted away, the problem is resolved, the 'good' eigenvalues are kept and convergence occurs in a few steps (the 'o'-line in Figure 6.3).

Using a slightly larger Lanczos factorization as a basis for the restart, e.g., a factorization of length $k + 3$ instead of length k and using a locking technique to decouple converged approximate eigenvalues and associated invariant subspaces from the active part of the iteration, this problem is avoided.

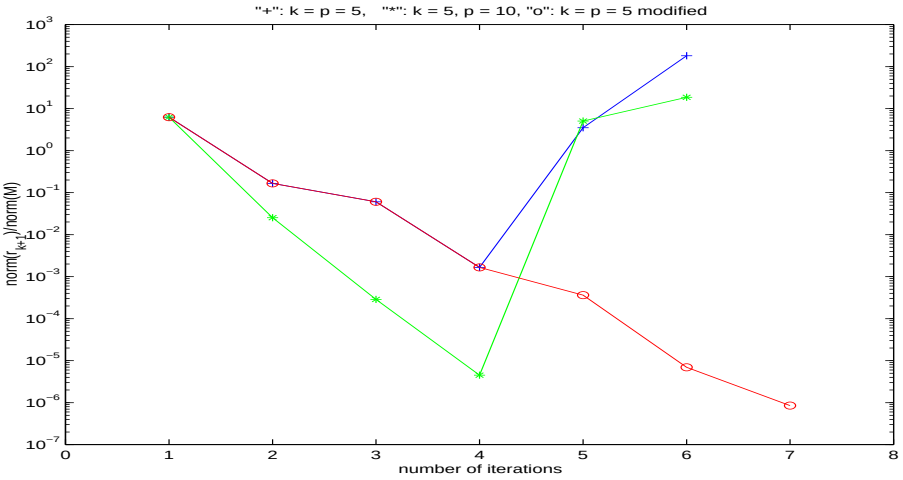


FIG. 6.3. *k–step restarted symplectic Lanczos method, different choices of the shifts*

Figure 6.4 displays the behavior of the k–step restarted symplectic Lanczos method for different choices of k and p, where k is quite small. Convergence is achieved in any case.

So far, in the tests presented, k was always chosen such that there exists a deflating subspace of $L - \lambda N$ corresponding to the k eigenvalues largest in modulus and their reciprocals. For $k = 20$, there is no such deflating subspace (there is one for $k = 19$ and one for $k = 21$). See Figure 6.5 for a convergence plot. The eigenvalues of $B^{2(k+p),2(k+p)}$ in the first iteration steps approximate the $k + j$ eigenvalues of largest modulus and their reciprocals (where $5 \leq j \leq p$) quite well. Our choice of shifts is to select the $2p$ eigenvalues with modulus closest to 1, but as λ_{k+1} is complex with $|\lambda_{k+1}| = |\lambda_k| \neq 1$, we can only choose $2(p-1)$ shifts that way. The last shift is chosen

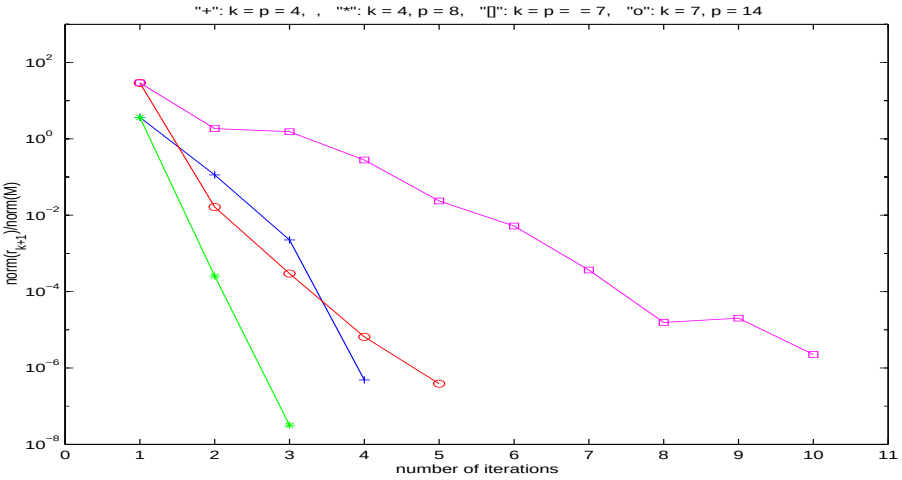


FIG. 6.4. *k–step restarted symplectic Lanczos method, different choices of k and p*

according to the strategy explained above. This eigenvalue keeps coming back before it is annihilated. A better idea to resolve the problem is to adapt k appropriately.

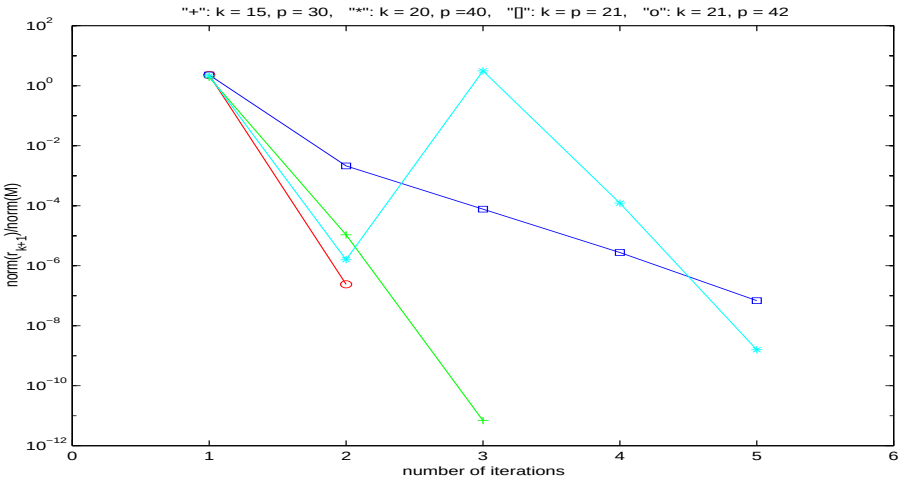


FIG. 6.5. *k–step restarted symplectic Lanczos method, different choices of k and p*

7. Concluding Remarks. We have investigated a symplectic Lanczos method for symplectic matrices. Employing the technique of implicitly restarting the method using double or quadruple shifts as zeros of the driving Laurent polynomials, this results in an efficient method to compute a few extremal eigenvalues of symplectic matrices and the associated eigenvectors or invariant subspaces. The residual of the Lanczos recursion can be made to zero by choosing proper shifts. It is an open problem how these shifts should be chosen in an optimal way. The preliminary numerical tests reported here show that for exact shifts, good performance is already achieved.

Before implementing the symplectic Lanczos process in a black-box algorithm, some more details need consideration: in particular, techniques for locking of con-

verged Ritz values as well as purging of converged, but unwanted Ritz values, needs to be derived in a similar way as it has been done for the implicitly restarted Arnoldi method.

REFERENCES

[1] Z. Bai, *Error analysis of the Lanczos algorithm for the nonsymmetric eigenvalue problem*, Math. Comp., 62 (1994), pp. 209–226.

[2] G. Banse, *Symplektische Eigenwertverfahren zur Lösung zeitdiskreter optimaler Steuerungsprobleme*, PhD thesis, Universität Bremen, Fachbereich 3 – Mathematik und Informatik, 28334 Bremen, Germany, 1995.

[3] G. Banse and A. Bunse-Gerstner, *A condensed form for the solution of the symplectic eigenvalue problem*, in Systems and Networks: Mathematical Theory and Applications, U. Helmke, R. Menniken, and J. Sauer, eds., Akademie Verlag, 1994, pp. 613–616.

[4] P. Benner and H. Fassbender, *The symplectic eigenvalue problem, the butterfly form, the SR algorithm and the Lanczos method*, Linear Algebra Appl., 275/276 (1998), pp. 19–47.

[5] P. Benner, H. Fassbender, and D. Watkins, *SR and SZ algorithms for the symplectic (butterfly) eigenproblem*, Linear Algebra Appl., (to appear).

[6] M. Bohner, *Linear Hamiltonian difference systems: Disconjugacy and Jacobi–type conditions*, J. Math. Anal. Appl., 199 (1996), pp. 804–826.

[7] A. Bunse-Gerstner, *Matrix factorization 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] 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.

[10] J. Della-Dora, *Sur quelques Algorithmes de recherche de valeurs propres*, Thése, L'Université Scientifique es Medicale de Grenoble, 1973.

[11] J. Della Dora, *Numerical linear algorithms and group theory*, Linear Algebra Appl., 10 (1975), pp. 267–283.

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

[13] H. Fassbender, *Symplectic Methods for Symplectic Eigenproblems*, Habilitationsschrift, Universität Bremen, Fachbereich 3 – Mathematik und Informatik, 28334 Bremen, Germany, 1998.

[14] U. Flaschka, V. Mehrmann, and D. Zywietz, *An analysis of structure preserving methods for symplectic eigenvalue problems*, RAIRO Automatique Productique Informatique Industrielle, 25 (1991), pp. 165–190.

[15] J. G. F. Francis, *The QR transformation Part I and Part II*, Comput. J., 4 (1961), pp. 265–271 and 332–345.

[16] R. Freund, *Transpose-free quasi-minimal residual methods for non-Hermitian linear systems*, numerical analysis manuscript 92-97, AT&T Bell Labs, 1992.

[17] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 2nd ed., 1989.

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

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

[20] 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.

[21] P. Lancaster and L. Rodman, *The Algebraic Riccati Equation*, Oxford University Press, Oxford, 1995.

[22] 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.

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

[24] R. Lehoucq, D. Sorensen, and C. Yang, *ARPACK user's guide. Solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, SIAM, Philadelphia, PA, 1998.

[25] W.-W. Lin, *A new method for computing the closed loop eigenvalues of a discrete-time algebraic Riccati equation*, Linear Algebra Appl., 6 (1987), pp. 157–180.

[26] V. Mehrmann, *Der SR-Algorithmus zur Berechnung der Eigenwerte einer Matrix*, Diplomarbeit, Universität Bielefeld, Bielefeld, FRG, 1979.

[27] ———, *A symplectic orthogonal method for single input or single output discrete time optimal linear quadratic control problems*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 221–248.

[28] ———, *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.

[29] V. Mehrmann and H. Xu, *Canonical forms for Hamiltonian and symplectic matrices and pencils*, Tech. Report SFB393/98-7, Fak. f. Mathematik, TU Chemnitz, 09107 Chemnitz, FRG, 1998. Available from `http://www.tu-chemnitz.de/sfb393/sfb98pr.html`.

[30] R. B. Morgan, *On restarting the Arnoldi method for large nonsymmetric eigenvalue problems*, Math. Comp., 65 (1996), pp. 1213–1230.

[31] C. Paige and C. Van Loan, *A Schur decomposition for Hamiltonian matrices*, Linear Algebra Appl., 14 (1981), pp. 11–32.

[32] 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.

[33] R. Patel, *Computation of the stable deflating subspace of a symplectic pencil using structure preserving orthogonal transformations*, in Proceedings of the 31st Annual Allerton Conference on Communication, Control and Computing, University of Illinois, 1993.

[34] ———, *On computing the eigenvalues of a symplectic pencil*, Linear Algebra Appl., 188/189 (1993), pp. 591–611.

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

[36] ———, *Deflation for implicitly restarted Arnoldi methods*, tech. report, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, 1998.

[37] D. R. Taylor, *Analysis of the look ahead Lanczos algorithm*, PhD thesis, Center for Pure and Applied Mathematics, University of California, Berkeley, CA, 1982.

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

[39] Q. Ye, *A convergence analysis for nonsymmetric Lanczos algorithms*, Math. Comp., 56 (1991), pp. 677–691.