Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

STRUCTURED QR ALGORITHMS FOR HAMILTONIAN SYMMETRIC MATRICES*

A. SALAM† AND D.S. WATKINS‡§

Abstract. Efficient, backward-stable, doubly structure-preserving algorithms for the Hamiltonian symmetric and skew-symmetric eigenvalue problems are developed. Numerical experiments confirm the theoretical properties of the algorithms. Also developed are doubly structure-preserving Lanczos processes for Hamiltonian symmetric and skew-symmetric matrices.

Key words. Hamiltonian matrix, Double structure, Eigenvalue, QR algorithm.

AMS subject classifications. 65F15, 15A18, 15A21.

1. Introduction. The problem of computing the complete eigensystem of a Hamiltonian matrix was posed some thirty years ago [16, 18] but still has not been resolved with complete satisfaction. The objective, to be precise, is to devise an efficient, backward stable algorithm that fully exploits the Hamiltonian structure and delivers the eigenvalues (and corresponding eigenvectors or invariant subspaces) in perfect pairs $\pm\lambda$ or quadruples $\pm\lambda$, $\pm\overline{\lambda}$. Progress toward this objective has been made in the works [1, 2, 7, 8, 9, 10, 16, 18, 21, 23], among others.

In this paper, we consider the special case in which the Hamiltonian matrix also possesses additional structure, either symmetric or skew-symmetric. These cases and others were studied by Bunse-Gerstner, Byers, and Mehrmann [6], who suggested the use of a quaternion QR algorithm [5] for the symmetric Hamiltonian case. Fassbender, Mackey, and Mackey [11] developed Jacobi algorithms for the symmetric and skew-symmetric cases. Stability questions for these algorithms have been studied by Tisseur [20]. In this paper we offer block QR algorithms for both the symmetric and skew-symmetric cases. These use exclusively orthogonal symplectic similarity transformations, so they preserve both structures and are backward stable. They can be expected to be more efficient than the Jacobi methods, and they are less exotic than the quaternion QR algorithm, as they use exclusively real arithmetic. We also develop

*Received by the editors on September 8, 2009. Accepted for publication on May 20, 2011. Handling Editor: Angelika Bunse-Gerstner.

†Univ Lille Nord de France, F-59650 Lille, ULCO LMPA, C.U. de la Mi-Voix, B.P. 699, F-62228 Calais, France (Ahmed.Salam@lmpa.univ-littoral.fr).

‡Department of Mathematics, Washington State University, Pullman, WA 99164-3113, USA (watkins@math.wsu.edu).

§This work was carried out while the second author was visiting the first author in Calais.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

structured Lanczos processes for the two cases.

2. Basic facts. We define a matrix $J \in \mathbb{R}^{2n\times 2n}$ by

$$J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}.$$

A matrix $H \in \mathbb{R}^{2n\times 2n}$ is *Hamiltonian* if JH is symmetric: $(JH)^T = JH$. This means exactly that H has the form

$$H = \begin{bmatrix} A & F \\ G & -A^T \end{bmatrix}; \quad F = F^T,\ G = G^T.$$

If v is an eigenvector of the Hamiltonian matrix H with eigenvalue λ, then Jv is an eigenvector of H^T with eigenvalue $-\lambda$. Therefore the eigenvalues of H occur in $\pm\lambda$ pairs. If λ is complex, then $\pm\lambda$, $\pm\overline{\lambda}$ form a quadruple of eigenvalues.

A major advantage of preserving structure is that the eigenvalues are always delivered in exact pairs or quadruples. If there are supposed to be n eigenvalues in the left half plane, then there always are exactly n. There is no danger of exactly one eigenvalue or a conjugate pair of complex eigenvalues slipping across the imaginary axis due to roundoff errors.

A matrix $S \in \mathbb{R}^{2n\times 2n}$ is *symplectic* if $S^T JS = J$. Symplectic matrices are clearly nonsingular, and the set of all such matrices is a group under matrix multiplication, the *symplectic group*. If H is Hamiltonian and S is symplectic, then $S^{-1}HS$ is also easily seen to be Hamiltonian. Thus the Hamiltonian structure is preserved under similarity transformations by symplectic matrices. Our method of preserving Hamiltonian structure will be to devise methods that use only symplectic similarity transformations.

In the interest of maintaining backward stability, it is desirable to use similarity transformations that are also orthogonal. The works [1, 2, 9, 10, 23] all use symplectic, orthogonal similarity transformations. However, there have also been some methods proposed in which orthogonality was sacrificed in the interest of efficiency [7, 8, 19].

In our present study we consider Hamiltonian matrices that also possess symmetry or skew-symmetry. Since these additional structures are preserved by orthogonal similarity transformations, we will use exclusively orthogonal, symplectic similarity transformations in this paper. In this way the two structures will be preserved, and we will achieve backward stability as well.

Let $S = \begin{bmatrix} U & V \end{bmatrix} \in \mathbb{R}^{2n\times 2n}$, where U, $V \in \mathbb{R}^{2n\times n}$. Then S is orthogonal and symplectic if and only if $U^T U = I$, $U^T JU = 0$, and $V = J^T U$. As a special case, if

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

$Q \in \mathbb{R}^{n\times n}$ is orthogonal, then

$$S = \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} \tag{2.1}$$

is orthogonal and symplectic. Our algorithms will use two types of orthogonal, symplectic similarity transforms. The first is the *double reflector*, a matrix of the form (2.1), where Q is an elementary reflector (Householder transformation) [25]. The second is the symplectic rotator. A *symplectic rotator* in the $(i, n+i)$ plane is a matrix of the form

$$S = \begin{bmatrix} \Gamma & -\Sigma \\ \Sigma & \Gamma \end{bmatrix},$$

where $\Gamma = \mathrm{diag}\{1, \ldots, 1, c, 1, \ldots, 1\}$, $\Sigma = \mathrm{diag}\{0, \ldots, 0, s, 0, \ldots, 0\}$, c and s lie in the ith position on the main diagonal, and $c^2 + s^2 = 1$. This is a special type of Givens rotation that also happens to be symplectic.

3. Reduction to condensed form. We consider first the symmetric case. If H is Hamiltonian and symmetric, then

$$H = \begin{bmatrix} A & G \\ G & -A \end{bmatrix}; \quad A = A^T,\ G = G^T.$$

By orthogonal symplectic similarity transformations we are able to transform H to a form

$$\tilde{H} = \begin{bmatrix} T & D \\ D & -T \end{bmatrix}, \tag{3.1}$$

where T is symmetric and tridiagonal, and D is diagonal. For example, in the case $n = 4$ we have

$$\tilde{H} = \left[\begin{array}{cccc|cccc} a_1 & b_1 & & & c_1 & & & \\ b_1 & a_2 & b_2 & & & c_2 & & \\ & b_2 & a_3 & b_3 & & & c_3 & \\ & & b_3 & a_4 & & & & c_4 \\ \hline c_1 & & & & -a_1 & -b_1 & & \\ & c_2 & & & -b_1 & -a_2 & -b_2 & \\ & & c_3 & & & -b_2 & -a_3 & -b_3 \\ & & & c_4 & & & -b_3 & -a_4 \end{array}\right]. \tag{3.2}$$

We describe the first step of the reduction, all other steps following the same pattern. First we use a symplectic double reflector to introduce zeros in the first column of G in positions $g_{31}, \ldots, g_{n1}$. This reflector acts on rows and columns 2

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

through n of both A and G. The next transformation is a symplectic rotator in the $(2, n+2)$ plane that transforms g_{21} to zero. Finally, a symplectic double reflector acting on rows and columns 2 through n of both A and G creates zeros in the first column of A in positions $a_{32}, \ldots, a_{n2}$. This completes the first step, which, taking preservation of symmetry of A and G into account, leaves the matrix in the form

$$\left[\begin{array}{cccc|cccc} * & * & & & * & & & \\ * & * & * & * & & * & * & * \\ & * & * & * & & * & * & * \\ & * & * & * & & * & * & * \\ \hline * & & & & * & * & & \\ & * & * & * & * & * & * & * \\ & * & * & * & & * & * & * \\ & * & * & * & & * & * & * \end{array}\right].$$

The second step acts similarly on the second column of A and G. After $n-1$ such steps, the matrix is reduced to the form (3.2).

This is actually just a special case of the Paige-Van Loan form [18]. For general Hamiltonian matrices this form is not particularly useful, but for the special case of symmetric Hamiltonian matrices it is. The flop count for the reduction is much less than for the general Paige-Van Loan reduction, as we just need to work with a single copy each of A and G, and these are both symmetric. Indeed the cost is approximately twice that of reducing a single $n \times n$ symmetric matrix to tridiagonal form. This is about $\frac{8}{3}n^3$ flops if the symplectic, orthogonal transforming matrix is not accumulated and $\frac{20}{3}n^3$ if it is accumulated. Notice that the flop count for reducing a full symmetric $2n \times 2n$ matrix to tridiagonal form, ignoring the Hamiltonian structure, is four times as great.

3.1. The skew-symmetric case. If H is Hamiltonian and skew-symmetric, it has the form

$$H = \begin{bmatrix} A & -G \\ G & A \end{bmatrix}; \quad A^T = -A,\ G^T = G.$$

As in the symmetric case, we are able to transform H to a form

$$\tilde{H} = \begin{bmatrix} T & -D \\ D & T \end{bmatrix},$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

where T is tridiagonal and D is diagonal. In this case, T is skew-symmetric. For example, in the case $n = 4$ we have

$$\tilde{H} = \left[\begin{array}{cccc|cccc} 0 & -b_1 & & & -c_1 & & & \\ b_1 & 0 & -b_2 & & & -c_2 & & \\ & b_2 & 0 & -b_3 & & & -c_3 & \\ & & b_3 & 0 & & & & -c_4 \\ \hline c_1 & & & & 0 & -b_1 & & \\ & c_2 & & & b_1 & 0 & -b_2 & \\ & & c_3 & & & b_2 & 0 & -b_3 \\ & & & c_4 & & & b_3 & 0 \end{array}\right]. \tag{3.3}$$

This is another special case of the reduction to Paige-Van Loan form. The flop count is slightly less than but asymptotically the same as that of the symmetric case.

This reduction is the precise analogue of the symmetric reduction given above. A different reduction (for the skew-symmetric case only), which might be preferable to this one, was given in [6]: Reduce H to the form

$$\left[\begin{array}{cc} 0 & -T \\ T & 0 \end{array}\right],$$

where T is symmetric and tridiagonal. Then the complete eigensystem of H can be found by applying the symmetric QR algorithm (or any of several other methods [25, § 7.2]) to T.

4. Structured Lanczos processes. Each of the reductions of the previous section has a structured Lanczos process associated with it. We briefly develop these processes, starting with the symmetric case. Let $S = \left[\begin{array}{cc} U & V \end{array}\right]$ denote the orthogonal, symplectic transforming matrix that maps the symmetric, Hamiltonian H to $\tilde{H}$ of the form (3.1). Then $HS = S\tilde{H}$, which we can write in block form as

$$H\left[\begin{array}{cc} U & V \end{array}\right] = \left[\begin{array}{cc} U & V \end{array}\right]\left[\begin{array}{cc} T & D \\ D & -T \end{array}\right]. \tag{4.1}$$

The first block equation from (4.1) is $HU = UT + VD$, and the second is $HV = UD + V(-T)$. The second is redundant; it can be obtained from the first by multiplying by J^T and using the properties $V = J^T U$ and $J^T H = HJ$. Thus we will focus on the first block equation, which we can write as

$$HU = UT + J^T UD.$$

Writing out the jth column of this equation, letting $u_1, \ldots, u_n$ denote the columns of U, we obtain

$$Hu_j = u_{j-1}b_{j-1} + u_j a_j + u_{j+1}b_j + J^T u_j c_j$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

or

$$u_{j+1}b_j = Hu_j - u_ja_j - u_{j-1}b_{j-1} - J^Tu_jc_j, \tag{4.2}$$

which gives us a structured Lanczos process, provided that we can come up with formulas for the coefficients. Multiplying (4.2) on the left by u_j^T and using the fact that the vectors are columns of an orthogonal, symplectic matrix, we find that $a_j = u_j^THu_j$. Similarly, multiplying by u_j^TJ, we find that $c_j = u_j^TJHu_j$. In practice we would compute

$$\tilde{u}_{j+1} = Hu_j - u_ja_j - u_{j-1}b_{j-1} - J^Tu_jc_j$$

and then compute $b_j = \|\tilde{u}_{j+1}\|_2$, which will get used as b_{j-1} on the following step, and $u_{j+1} = \tilde{u}_{j+1}/b_j$. We summarize these findings as an algorithm.

ALGORITHM 1. *Hamiltonian symmetric Lanczos process. Start with* $u_0 = 0$, $b_0 = 0$, *and* $u_1 =$ *a vector satisfying* $\|u_1\|_2 = 1$.

for $j = 1, \ldots, n-1$
- $w \leftarrow Hu_j$
- $a_j \leftarrow u_j^Tw$
- $c_j \leftarrow u_j^TJw$
- $u_{j+1} \leftarrow w - u_ja_j - u_{j-1}b_{j-1} - J^Tu_jc_j$
- $b_j = \|u_{j+1}\|_2$
- *if* $b_j = 0$
  - *stop (invariant subspace found)*
- *else*
  - $u_{j+1} = u_{j+1}/b_j$

If run to completion, this algorithm gives, in principle, the reduction to the condensed form (3.1). Of course we prefer the backward stable algorithm sketched in Section 3 for this purpose. The structured Lanczos algorithm can be used to detect eigenvalues and invariant subspaces of large, sparse matrices. In that context it is not run to completion.

4.1. The skew-symmetric case. Proceeding as in the symmetric case, we have

$$H\begin{bmatrix} U & V \end{bmatrix} = \begin{bmatrix} U & V \end{bmatrix}\begin{bmatrix} T & -D \\ D & T \end{bmatrix},$$

where the form of T and D is illustrated in (3.3). The first block equation is $HU = UT + VD$, and the second block equation is equivalent to the first. Rewriting the

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

first block equation as

$$HU = UT + J^T UD$$

and writing out the jth column of this equation, we obtain

$$Hu_j = -u_{j-1}b_{j-1} + u_{j+1}b_j + J^T u_j c_j$$

or

$$u_{j+1}b_j = Hu_j + u_{j-1}b_{j-1} - J^T u_j c_j. \tag{4.3}$$

Multiplying (4.3) on the left by $u_j^T J$, we find that $c_j = u_j^T JHu_j$. In practice we would compute

$$\tilde{u}_{j+1} = Hu_j + u_{j-1}b_{j-1} - J^T u_j c_j$$

and then compute $b_j = \| \tilde{u}_{j+1} \|_2$, which will get used as b_{j-1} on the following step, and $u_{j+1} = \tilde{u}_{j+1}/b_j$. These considerations yield the following algorithm.

ALGORITHM 2. *Hamiltonian skew-symmetric Lanczos process. Start with* $u_0 = 0$, $b_0 = 0$, *and* $u_1 =$ *a vector satisfying* $\| u_1 \|_2 = 1$.

for $j = 1, \ \ldots, n-1$
- $w \leftarrow Hu_j$
- $c_j \leftarrow u_j^T Jw$
- $u_{j+1} \leftarrow w + u_{j-1}b_{j-1} - J^T u_j c_j$
- $b_j = \| u_{j+1} \|_2$
- *if* $b_j = 0$
  - *stop (invariant subspace found)*
- *else*
  - $u_{j+1} = u_{j+1}/b_j$

5. Implicitly-shifted block QR iterations. Again we consider the symmetric case first. Suppose the Hamiltonian symmetric matrix H has been reduced to the condensed form exemplified by the matrix $\tilde{H}$ in (3.2). We now drop the tilde and refer to the condensed matrix as H. If a perfect shuffle permutation is applied to H,

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

the result is

$$H_s = \left[\begin{array}{cc|cc|cc|cc} a_1 & c_1 & b_1 & & & & & \\ c_1 & -a_1 & & -b_1 & & & & \\ \hline b_1 & & a_2 & c_2 & b_2 & & & \\ & -b_1 & c_2 & -a_2 & & -b_2 & & \\ \hline & & b_2 & & a_3 & c_3 & b_3 & \\ & & & -b_2 & c_3 & -a_3 & & -b_3 \\ \hline & & & & b_3 & & a_4 & c_4 \\ & & & & & -b_3 & c_4 & -a_4 \end{array}\right],$$

which is seen to be block tridiagonal with block size $k = 2$. It follows that we will be able to apply a block QR algorithm [17] to this structure. The block QR algorithm has not become popular because it is inefficient in general. If the block size is k, the cost in flops of applying one shift is about k times the cost of applying a single shift in the standard case. In our current situation we have $k = 2$, so the inefficiency is only a factor of 2. Moreover, because the matrix is Hamiltonian, it turns out that the operations are redundant by a factor of 2; only half of them have to be done. Thus there is no loss of efficiency here.

The eigenvalues of a symmetric Hamiltonian matrix are real and occur in $\pm\lambda$ pairs. The block QR algorithm that we will develop will be an implicitly-shifted bulge-chasing algorithm. In order to preserve Hamiltonian structure, we will need to extract the eigenvalues in $\pm\lambda$ pairs. To this end we will employ a double-shift algorithm with shifts taken in pairs $\pm\rho$. Each QR iteration will take the form

$$\widehat{H}_s = Q_s^{-1} H_s Q_s,$$

where Q_s is the orthogonal factor in the block QR decomposition [24, p. 157]

$$(H_s - \rho I)(H_s + \rho I) = Q_s R_s.$$

The factor R_s is block triangular with 2×2 blocks. Of course we do not perform this QR decomposition explicitly, nor do we compute the product $(H_s - \rho I)(H_s + \rho I) = H_s^2 - \rho^2 I$. We build the transformation Q_s bit by bit via a bulge-chasing process, for which we just need the first two columns of $H_s^2 - \rho^2 I$.

We prefer to describe the process in terms of the original unshuffled coordinate system (3.2), so we will work instead with $H^2 - \rho^2 I$, from which we need columns 1

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

and $n+1$. An easy computation shows that these are p and $J^T p$, respectively, where

$$p = \left[\begin{array}{c} a_1^2 + b_1^2 + c_1^2 - \rho^2 \\ b_1(a_1 + a_2) \\ b_2 b_1 \\ 0 \\ \hline 0 \\ (c_2 - c_1)b_1 \\ 0 \\ 0 \end{array}\right].$$

The iteration is set in motion by an orthogonal symplectic transformation Q_1 such that $Q_1^T p = \alpha e_1$. This can be built as a product of a symplectic rotator and a symplectic double reflector. First a symplectic rotator acting on entries 2 and $n+2$ transforms the $n+2$ entry $(c_2 - c_1)b_1$ to zero. Then a symplectic double reflector acting on entries 1, 2, and 3 (and $n+1$, $n+2$, and $n+3$) finishes the job. Notice that since the resulting Q_1 is orthogonal and symplectic, we have $JQ_1 = Q_1 J$. Thus $Q_1^T(J^T p) = J^T Q_1^T p = \alpha J^T e_1 = \alpha e_{n+1}$, so the elimination in $J^T p$ takes place automatically.

A similarity transformation by Q_1 yields a Hamiltonian symmetric matrix

$$H_1 = Q_1^{-1} H Q_1$$

that no longer has the form (3.2) but has bulges in both the T and D parts. In the case $n = 7$, H_1 has the form

$$\left[\begin{array}{ccccccc|ccccccc}
* & * & * & * & & & & * & * & * & & & & \\
* & * & * & * & & & & * & * & * & & & & \\
* & * & * & * & & & & * & * & * & & & & \\
* & * & * & * & * & & & & & & * & & & \\
 & & & * & * & * & & & & & & * & & \\
 & & & & * & * & * & & & & & & * & \\
 & & & & & * & * & & & & & & & * \\
\hline
* & * & * & & & & & * & * & * & * & & & \\
* & * & * & & & & & * & * & * & * & & & \\
* & * & * & & & & & * & * & * & * & & & \\
 & & & * & & & & * & * & * & * & * & & \\
 & & & & * & & & & & & * & * & * & \\
 & & & & & * & & & & & & * & * & * \\
 & & & & & & * & & & & & & * & *
\end{array}\right].$$

The rest of the block QR step consists of returning this matrix to the block tridiagonal form as in the reduction described in Section 3. This amounts to a bulge chase. Since

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

all of the transforming matrices are unitary and symplectic, the Hamiltonian and symmetric structures are preserved.

Once the matrix has been reduced to block tridiagonal form, it can be stored in the form of $3n-2$ parameters $a_1, \ldots, a_n$, $b_1, \ldots, b_{n-1}$, and $c_1, \ldots, c_n$. During the bulge chase an additional six storage spaces are needed for the bulge entries, and a few extra scratch spaces are needed, but the total memory requirement is clearly $O(n)$. Moreover, the cost in flops per step is also $O(n)$, as the the number of transforming matrices is $O(n)$, and each transforming matrix acts on $O(1)$ entries.

If, as in many applications, we need to update the transforming matrix as we go, the cost of this is $O(n)$ per transformation, hence $O(n^2)$ per iteration, as usual. Because the matrix that is being updated has the form $\begin{bmatrix} U & J^TU \end{bmatrix}$, a factor of 2 is saved in the flop count.

The iterations tend to drive the entries b_i to zero. Once all the b_i have become small enough to be considered to be zeros, the matrix has the form

$$H = \left[\begin{array}{cccc|cccc} a_1 & & & & c_1 & & & \\ & a_2 & & & & c_2 & & \\ & & a_3 & & & & c_3 & \\ & & & a_4 & & & & c_4 \\ \hline c_1 & & & & -a_1 & & & \\ & c_2 & & & & -a_2 & & \\ & & c_3 & & & & -a_3 & \\ & & & c_4 & & & & -a_4 \end{array}\right].$$

Each submatrix of the form

$$\begin{bmatrix} a_i & c_i \\ c_i & -a_i \end{bmatrix}$$

houses a pair of eigenvalues $\pm\sqrt{a_i^2+c_i^2}$. Letting $\hat{a}_i = \sqrt{a_i^2+c_i^2}$, we can diagonalize this matrix to the form

$$\begin{bmatrix} \hat{a}_i & 0 \\ 0 & -\hat{a}_i \end{bmatrix}$$

by a single rotator.

We coded the algorithm in MATLAB and tried it out with two different shifting strategies. The simplest is to take the eigenvalues of

$$\begin{bmatrix} a_n & c_n \\ c_n & -a_n \end{bmatrix},$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

which are $\pm\sqrt{a_n^2 + c_n^2}$, as the shifts. This *generalized Rayleigh quotient* shift strategy has local cubic convergence [26] and usually works well, but we found that occasionally it gets stuck and takes many iterations to find a pair of eigenvalues. We also tried a *generalized Wilkinson* shift strategy, in which we find the eigenvalues of

$$\left[\begin{array}{cc|cc} a_{n-1} & c_{n-1} & b_{n-1} & \\ c_{n-1} & -a_{n-1} & & -b_{n-1} \\ \hline b_{n-1} & & a_n & c_n \\ & -b_{n-1} & c_n & -a_n \end{array}\right],$$

and take the pair that is closer to $\pm\sqrt{a_n^2 + c_n^2}$ as shifts. This cured the problem of getting stuck. We observed cubic convergence, and we conjecture that this strategy is globally convergent.

For larger problems aggressive early deflation [4] and its associated shift strategy can be used to decrease the total number of iterations. In problems for which eigenvectors are needed, so that the transforming matrix needs to be accumulated, BLAS 3 speed [13] can be achieved by chasing bulges in bunches, as suggested in [3, 15]. Parallelism can be achieved as well [12, 14, 22]. In short, all of the methods that have been devised for speeding up the general (symmetric or unsymmetric) implicitly-shifted QR algorithm can also be employed here.

Since the algorithm preserves the structures, it delivers perfect $\pm\lambda$ pairs of real eigenvalues, and the corresponding eigenvectors are delivered in exact v, Jv pairs.

5.1. The skew-symmetric case. In this case the condensed form is (3.3). After a perfect shuffle of the rows and columns, we have the block tridiagonal form

$$H_s = \left[\begin{array}{cc|cc|cc} & -c_1 & -b_1 & & & \\ c_1 & & & -b_1 & & \\ \hline b_1 & & & -c_2 & -b_2 & \\ & b_1 & c_2 & & & -b_2 \\ \hline & & b_2 & & & -c_3 & -b_3 & \\ & & & b_2 & c_3 & & & -b_3 \\ \hline & & & & b_3 & & & -c_4 \\ & & & & & b_3 & c_4 & \end{array}\right].$$

The eigenvalues of a skew-symmetric Hamiltonian matrix are purely imaginary complex conjugate pairs $\pm i\mu$, so we use shifts $\pm i\rho$ with ρ real. To start the block double QR step, we need the first and $(n+1)$st columns of $(H - i\rho I)(H + i\rho I) =$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

$H^2 + \rho^2 I$. These are are p and $J^T p$, respectively, where

$$p = \left[\begin{array}{c} -b_1^2 - c_1^2 + \rho^2 \\ 0 \\ b_2 b_1 \\ 0 \\ \hline 0 \\ (c_2 - c_1) b_1 \\ 0 \\ 0 \end{array}\right].$$

A transformation Q_1 such that $Q_1^T p = \alpha e_1$ and $Q_1^T J^T p = \alpha e_{n+1}$ can be built just as in the symmetric case. A similarity transformation by Q_1 yields a matrix $Q_1^{-1} A Q_1$ with a bulge in the block triangular form. The bulge is then chased to complete the block QR iteration. The details are slightly simpler than in the symmetric case because of the extra zeros in the skew-symmetric form. We coded this algorithm, ran it, and obtained results similar to the symmetric case.

6. Conclusions. We have developed simple, robust, and backward stable algorithms for the real Hamiltonian symmetric and anti-symmetric eigenvalue problems. The methods preserve both structures and deliver eigenvalues in exact $\pm\lambda$ pairs.

REFERENCES

[1] P. Benner and D. Kressner. Fortran 77 subroutines for computing the eigenvalues of Hamiltonian matrices. *ACM Trans. Math. Software*, 32:352–373, 2006. Availiable at www.tu-chemnitz.de/mathematik/hapack/.

[2] P. Benner, V. Mehrmann, and H. Xu. A numerically stable, structure preserving method for computing the eigenvalues of real hamiltonian or symplectic pencils. *Numer. Math.*, 78:329–358, 1998.

[3] K. Braman, R. Byers, and R. Mathias. The multishift QR algorithm Part I: Maintaining well-focused shifts and level 3 performance. *SIAM J. Matrix Anal. Appl.*, 23:929–947, 2002.

[4] K. Braman, R. Byers, and R. Mathias. The multishift QR algorithm Part II: Aggressive early deflation. *SIAM J. Matrix Anal. Appl.*, 23:948–973, 2002.

[5] A. Bunse-Gerstner, R. Byers, and V. Mehrmann. A quaternion QR algorithm. *Numer. Math.*, 55:83–95, 1989.

[6] A. Bunse-Gerstner, R. Byers, and V. Mehrmann. A chart of numerical methods for structured eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 13:419–453, 1992.

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

[8] A. Bunse-Gerstner, V. Mehrmann, and D.S. Watkins. An SR algorithm for Hamiltonian matrices based on Gaussian elimination. *Methods Oper. Res.*, 58:339–357, 1989.

[9] R. Byers. A Hamiltonian QR algorithm. *SIAM J. Sci. Statist. Comput.*, 7:212–229, 1986.

[10] D. Chu, X. Liu, and V. Mehrmann. A numerical method for computing the Hamiltonian Schur form. *Numer. Math.*, 105:375–412, 2007.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 22, pp. 573-585, May 2011

ELA
http://math.technion.ac.il/iic/ela

[11] H. Fassbender, D.S. Mackey, and N. Mackey. Hamiltonian and Jacobi come full circle: Jacobi algorithms for structured Hamiltonian eigenproblems. *Linear Algebra Appl.*, 332/334:37–80, 2001.

[12] R. Granat, B. Kågström, and D. Kressner. A novel parallel QR algorithm for hybrid distributed memory HPC systems. *SIAM J. Sci. Comput.*, 32:2345–2378, 2010.

[13] J. Handy. *The Cache Memory Book*, 2nd edition. Academic Press, San Diego, 1998.

[14] G. Henry, D. Watkins, and J. Dongarra. A parallel implementation of the nonsymmetric QR algorithm for distributed memory architectures. *SIAM J. Sci. Comput.*, 24:284–311, 2002.

[15] B. Lang. Using level 3 BLAS in rotation-based algorithms. *SIAM J. Sci. Comput.*, 19:626–634, 1998.

[16] A.J. Laub. A Schur method for solving algebraic Riccati equations. *IEEE Trans. Automat. Control*, 24:913–921, 1979.

[17] G.S. Miminis and C.C. Paige. Implicit shifting in the QR algorithm and related algorithms. *SIAM J. Matrix Anal. Appl.*, 12:385–400, 1991.

[18] C.C. Paige and C.F. Van Loan. A Schur decomposition for Hamiltonian matrices. *Linear Algebra Appl.*, 41:11–32, 1981.

[19] A.C. Raines III and D.S. Watkins. A class of Hamiltonian-Symplectic methods for solving the algebraic Riccati equation. *Linear Algebra Appl.*, 205/206:1045–1060, 1994.

[20] F. Tisseur. Stability of structured Hamiltonian eigensolvers. *SIAM J. Matrix. Anal. Appl.*, 23:103–125, 2001.

[21] C.F. Van Loan. A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix. *Linear Algebra Appl.*, 61:233–251, 1984.

[22] D.S. Watkins. Shifting strategies for the parallel QR algorithm. *SIAM J. Sci. Comput.*, 15:953–958, 1994.

[23] D.S. Watkins. On the reduction of a Hamiltonian matrix to Hamiltonian Schur form. *Electron. Trans. Numer. Anal.*, 23:141–157, 2006.

[24] D.S. Watkins. *The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods.* SIAM, Philadelphia, 2007.

[25] D.S. Watkins. *Fundamentals of Matrix Computations*, 3rd edition. John Wiley and Sons, New York, 2010.

[26] D.S. Watkins and L. Elsner. Convergence of algorithms of decomposition type for the eigenvalue problem. *Linear Algebra Appl.*, 143:19–47, 1991.