Zentrum für Technomathematik
Fachbereich 3 – Mathematik und Informatik

Symplectic Balancing of Hamiltonian Matrices

Peter Benner

Report 00–05

Berichte aus der Technomathematik

Report 00–05 **Februar 2000**

SYMPLECTIC BALANCING OF HAMILTONIAN MATRICES

PETER BENNER*

Abstract. We discuss the balancing of Hamiltonian matrices by structure preserving similarity transformations. The method is closely related to balancing non-symmetric matrices for eigenvalue computations as proposed by Osborne [23] and Parlett/Reinsch [25] and implemented in most linear algebra software packages. It is shown that isolated eigenvalues can be deflated using similarity transformations with symplectic permutation matrices. Balancing is then based on equilibrating row and column norms of the Hamiltonian matrix using symplectic scaling matrices. Due to the given structure, it is sufficient to deal with the leading half rows and columns of the matrix. Numerical examples show that the method improves eigenvalue calculations of Hamiltonian matrices as well as numerical methods for solving continuous-time algebraic Riccati equations.

Key words. balancing, eigenvalues, Hamiltonian matrix, symplectic method.
AMS(MOS) subject classifications. 15A18, 65F15, 65F35.

1. Introduction. The eigenvalue problem for *Hamiltonian matrices*

$$H = \begin{bmatrix} A & G \\ Q & -A^T \end{bmatrix}, \tag{1.1}$$

where $A, G, Q \in \mathbb{R}^{n\times n}$ and G, Q are symmetric, plays a fundamental role in many algorithms of control theory and other areas of applied mathematics as well as computational physics and chemistry. Computing the eigenvalues of Hamiltonian matrices is required, e.g., when computing the $\mathcal{H}_\infty$–norm of transfer matrices (see, e.g., [9, 10]), calculating the stability radius of a matrix ([13, 29]), computing response functions [22], and many more. Hamiltonian matrices are also closely related to *continuous-time algebraic Riccati equations* (CARE) of the form

$$0 = Q + A^T X + XA - XGX \tag{1.2}$$

with A, G, Q as in (1.1) and $X \in \mathbb{R}^{n\times n}$ is a symmetric solution matrix. Many numerical methods for solving (1.2) are based on computing certain invariant subspaces of the related Hamiltonian matrices; see, e.g., [19, 21, 26, 28]. For a detailed discussion of the relations of Hamiltonian matrices and continuous-time algebraic Riccati equations (1.2) we refer to [18].

In eigenvalue computations, matrices and matrix pencils are often preprocessed using a *balancing* procedure as described in [23, 25] for a general matrix $A \in \mathbb{R}^{n\times n}$. First, A is permuted via similarity transformations in order to isolate eigenvalues, i.e., a permutation matrix $P \in \mathbb{R}^{n\times n}$ is computed such that

$$P^T A P = \begin{bmatrix} T_1 & X & Y \\ 0 & Z & W \\ 0 & 0 & T_2 \end{bmatrix}, \tag{1.3}$$

where $T_1 \in \mathbb{R}^{p\times p}$ and $T_2 \in \mathbb{R}^{q\times q}$ are upper triangular matrices. Then, a diagonal matrix

$$D := \begin{bmatrix} I_p & 0 & 0 \\ 0 & D_Z & 0 \\ 0 & 0 & I_q \end{bmatrix}, \qquad D_Z := \operatorname{diag}(d_{p+1}, \ldots, d_{n-q}),$$

*Universität Bremen, Fachbereich 3 – Mathematik und Informatik, Zentrum für Technomathematik, 28334 Bremen, Germany. E-mail: benner@math.uni-bremen.de

is computed such that rows and columns of $D_Z^{-1} Z D_Z$ are as close in norm as possible. That is, balancing consists of a permutation step and a scaling step. In the scaling step, the rows and columns of a matrix are scaled, which usually leads to a decrease of the matrix norm. This preprocessing step often improves the accuracy of computed eigenvalues significantly; *isolated eigenvalues* (i.e., those contained in T_1 and T_2) are even computed without roundoff error.

Unfortunately, applying this balancing strategy to a Hamiltonian matrix H as given in (1.1) will in general destroy the Hamiltonian structure. This is no problem if the subsequent eigenvalue algorithm does not preserve or use the Hamiltonian structure. But during the past fifteen years, several structure preserving methods for the Hamiltonian eigenproblem have been suggested. In particular, the square-reduced method [31], the Hamiltonian QR algorithm (if in (1.1), $\operatorname{rank} G = 1$ or $\operatorname{rank} Q = 1$) [12], the recently proposed algorithm based on a symplectic URV-like decomposition [7], or the implicitly restarted symplectic Lanczos method of [5] for large sparse Hamiltonian eigenproblems are appropriate choices for developing subroutines for library usage and raise the need for a symplectic balancing routine. Similarity transformations by symplectic matrices preserve the Hamiltonian structure. Thus, in order to balance a Hamiltonian matrix and to preserve its structure, the required permutation matrix and the diagonal scaling matrix should be symplectic.

In Section 2 we will give some necessary background. Isolating eigenvalues of Hamiltonian matrices without destroying the structure can be achieved using symplectic permutation matrices. This will be the topic of Section 3. How to equilibrate rows and norms of Hamiltonian matrices in a similar way as proposed in [25] using symplectic diagonal scaling matrices will be presented in Section 4. When invariant subspaces, eigenvectors, or solutions of algebraic Riccati equations are the target of the computations, some post-processing steps are required. These and some other applications of the proposed symplectic balancing method are discussed in Section 5. Some numerical examples on the use of the proposed balancing strategy for eigenvalue computation and numerical solution of algebraic Riccati equations are given in Section 6.

2. Preliminaries. The following classes of matrices will be employed in the sequel.

DEFINITION 2.1. *Let*

$$J := \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}, \tag{2.1}$$

where I_n is the $n \times n$ identity matrix.

a) A matrix $H \in \mathbb{R}^{2n\times 2n}$ is Hamiltonian *if $(HJ)^T = HJ$. The Lie Algebra of Hamiltonian matrices in $\mathbb{R}^{2n\times 2n}$ is denoted by $\mathcal{H}_{2n}$.*

b) A matrix $\mathcal{H} \in \mathbb{R}^{2n\times 2n}$ is skew-Hamiltonian *if $(HJ)^T = -HJ$. The Jordan algebra of skew-Hamiltonian matrices in $\mathbb{R}^{2n\times 2n}$ is denoted by $\mathcal{SH}_{2n}$.*

c) A matrix $S \in \mathbb{R}^{2n\times 2n}$ is symplectic *if $SJS^T = J$ or, equivalently, $S^T JS = J$. The Lie group of symplectic matrices in $\mathbb{R}^{2n\times 2n}$ is denoted by $\mathcal{S}_{2n}$.*

d) A matrix $U \in \mathbb{R}^{2n\times 2n}$ is unitary symplectic *if $U \in \mathcal{S}_{2n}$ and $UU^T = I_{2n}$. The compact Lie group of unitary symplectic matrices in $\mathbb{R}^{2n\times 2n}$ is denoted by $\mathcal{US}_{2n}$.*

Observe that every $H \in \mathcal{H}_{2n}$ must have the block representation given in (1.1).

In [11], an important relation between symplectic and Hamiltonian matrices is proved.

PROPOSITION 2.2. *Let $S \in \mathbb{R}^{2n\times 2n}$ be nonsingular. Then $S^{-1}HS$ is Hamiltonian for all $H \in \mathcal{H}_{2n}$ if and only if $S^TJS = \alpha J$ for some $\alpha \in \mathbb{R}\setminus\{0\}$.*

This result shows that in general, similarity transformations that preserve the Hamiltonian structure have to be symplectic up to scaling with a real scalar.

The following proposition shows that the structure of $2n \times 2n$ orthogonal symplectic matrices permits them to be represented as a pair of $n \times n$ matrices. Hence, the arithmetic cost and storage for accumulating orthogonal symplectic matrices can be halved.

PROPOSITION 2.3. *[24] An orthogonal matrix $U \in \mathbb{R}^{2n\times 2n}$ is symplectic if and only if it takes the form $U = \begin{bmatrix} U_1 & U_2 \\ -U_2 & U_1 \end{bmatrix}$ where $U_i \in \mathbb{R}^{n\times n}$, $i = 1, 2$.*

We have the following well-known property of the spectra of Hamiltonian matrices (see, e.g., [18, 21], and the references given therein).

PROPOSITION 2.4. *The spectrum of a real Hamiltonian matrix, denoted by $\sigma(H)$ is symmetric with respect to the imaginary axis, i.e., if $\lambda \in \sigma(H)$, then $-\lambda \in \sigma(H)$. The spectrum of Hamiltonian matrices can therefore be partitioned as*

$$\sigma(H) = \{\lambda_1, \ldots, \lambda_n\} \cup \{-\lambda_1, \ldots, -\lambda_n\} =: \Delta \cup (-\Delta),$$

where $Re(\lambda_j) \geq 0$, $j = 1, \ldots, n$.

When solving Hamiltonian eigenproblems one would like to compute a Schur form for Hamiltonian matrices analogous to the real Schur form for non-symmetric matrices. This should be done in a structure-preserving way.

DEFINITION 2.5. *a) Let $\hat{H} \in \mathcal{H}_{2n}$. If $\hat{H}$ has the form*

$$\hat{H} = \begin{bmatrix} \hat{A} & \hat{G} \\ 0 & -\hat{A}^T \end{bmatrix}, \tag{2.2}$$

where $\hat{A} \in \mathbb{R}^{n\times n}$ is in real Schur form (quasi-upper triangular) and $\hat{G} = \hat{G}^T$, then $\hat{H}$ is real Hamiltonian quasi-triangular.

b) If $H \in \mathcal{H}_{2n}$ and there exists $U \in \mathcal{US}_{2n}$ such that $\hat{H} = U^THU$ is real Hamiltonian quasi-triangular, then $\hat{H}$ is in real Hamiltonian Schur form *and U^THU is called a* Hamiltonian Schur decomposition.

If a Hamiltonian Schur decomposition exists such that $\hat{H}$ is as is (2.2), then U can be chosen such that $\sigma(\hat{A}) = -\Delta = \overline{-\Delta}$ [12].

Most of the structure-preserving methods for the Hamiltonian eigenproblem, i.e., those using symplectic (similarity) transformations, rely on the following result. For Hamiltonian matrices with no purely imaginary eigenvalues this result was first stated in [24] while in its full generality as given below it has been proved in [20].

THEOREM 2.6. *Let $H \in \mathcal{H}_{2n}$ and let $i\alpha_1, \ldots, i\alpha_\ell$ be its pairwise distinct nonzero purely imaginary eigenvalues. Furthermore, let the associated H-invariant subspaces be spanned by the columns of U_k, $k = 1, \ldots, \ell$. Then the following are equivalent.*

i) There exists $S \in \mathcal{S}_{2n}$ such that $S^{-1}HS$ is real Hamiltonian quasi-triangular.

ii) There exists $U \in \mathcal{US}_{2n}$ such that U^THU is in real Hamiltonian Schur form.

iii) $U_k^HJU_k$ is congruent to J for all $k = 1, \ldots, \ell$, where J is always of the appropriate dimension.

Note that from Theorem 2.6 it follows that purely imaginary eigenvalues of $H \in \mathcal{H}_{2n}$ (if any) must have even algebraic multiplicity in order for the Hamiltonian Schur form of H to exist.

3. Isolating Eigenvalues by Symplectic Permutations. Let P denote any $n \times n$ permutation matrix. It is easy to see that symplectic permutation matrices have the form

$$P_s = \begin{bmatrix} P & 0 \\ 0 & P \end{bmatrix}. \tag{3.1}$$

With matrices of type (3.1) it is possible to transform a Hamiltonian matrix to the form

$$\tilde{H} = P_s H P_s = \begin{array}{c} \left[\begin{array}{ccc|ccc} A_{11} & A_{12} & A_{13} & G_{11} & G_{12} & G_{13} \\ 0 & A_{22} & A_{23} & G_{12}^T & G_{22} & 0 \\ 0 & 0 & A_{33} & G_{13}^T & 0 & 0 \\ \hline 0 & 0 & Q_{13} & -A_{11}^T & 0 & 0 \\ 0 & Q_{22} & Q_{23} & -A_{12}^T & -A_{22}^T & 0 \\ Q_{13}^T & Q_{23}^T & Q_{33} & -A_{13}^T & -A_{23}^T & -A_{33}^T \end{array}\right] \begin{array}{l} \}p \\ \}n-p-q =: r \\ \}q \\ \}p \\ \}n-p-q = r \\ \}q \end{array} \\ \begin{array}{cccccc} \underbrace{\quad}_{p} & \underbrace{\quad}_{r} & \underbrace{\quad}_{q} & \underbrace{\quad}_{p} & \underbrace{\quad}_{r} & \underbrace{\quad}_{q} \end{array} \end{array} \tag{3.2}$$

where A_{11}, A_{33} are upper triangular and either $Q_{13} = 0$ or $G_{13} = 0$. The existence of such a P_s will be proved in a constructive way later by Algorithm 3.4 which transforms a Hamiltonian matrix to the form given in (3.2). From a Hamiltonian matrix having the form (3.2) a total of $2(p+q)$ eigenvalues of H can be read off directly as seen by the following result.

LEMMA 3.1. *Let $H \in \mathcal{H}_{2n}$ and $p, q, r \in \mathbb{N}_0$ with $r = n - p - q$, where H is of the form (3.2) and either $G_{13} = 0$ or $Q_{13} = 0$. Then there exists a permutation matrix $P \in \mathbb{R}^{2n \times 2n}$ such that*

$$T = P^T H P = \begin{bmatrix} T_1 & Y & Z \\ 0 & H_{22} & W \\ 0 & 0 & T_2 \end{bmatrix}, \tag{3.3}$$

where $T_1, T_2 \in \mathbb{R}^{(p+q)\times(p+q)}$ are upper triangular with

$$\sigma(T_1) = \sigma(-T_2) \tag{3.4}$$

and

$$H_{22} = \begin{bmatrix} A_{22} & G_{22} \\ Q_{22} & -A_{22}^T \end{bmatrix} \tag{3.5}$$

is a $2r \times 2r$ Hamiltonian submatrix of H.

Proof. Let H be as in (3.2) and

$$P_1 := \left[\begin{array}{ccc|ccc} 0 & I_p & 0 & 0 & 0 & 0 \\ 0 & 0 & I_r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_q \\ \hline 0 & 0 & 0 & 0 & I_p & 0 \\ 0 & 0 & 0 & I_r & 0 & 0 \\ I_q & 0 & 0 & 0 & 0 & 0 \end{array}\right]. \tag{3.6}$$

Then

$$P_1^T H P_1 = \left[\begin{array}{cc|cc|cc} -A_{33}^T & Q_{13}^T & Q_{23}^T & -A_{23}^T & -A_{13}^T & Q_{33} \\ G_{13}^T & A_{11} & A_{12} & G_{12} & G_{11} & A_{13} \\ \hline 0 & 0 & A_{22} & G_{22} & G_{12}^T & A_{23} \\ 0 & 0 & Q_{22} & -A_{22}^T & -A_{12}^T & Q_{23} \\ \hline 0 & 0 & 0 & 0 & -A_{11}^T & Q_{13} \\ 0 & 0 & 0 & 0 & G_{13} & A_{33} \end{array}\right]. \tag{3.7}$$

Define

$$\Pi_m := \begin{bmatrix} & & 1 \\ & \diagup & \\ 1 & & \end{bmatrix} \in \mathbb{R}^{m\times m}. \tag{3.8}$$

If $G_{13} = 0$ then set $P_2 := \mathrm{diag}\,(\Pi_q,\, I_p,\, I_r,\, I_r,\, \Pi_p,\, I_q)$. Otherwise ($Q_{13} = 0$), set

$$P_2 := \left[\begin{array}{cc|cc|cc} 0 & \Pi_q & & & & \\ I_p & 0 & & & & \\ \hline & & I_r & 0 & & \\ & & 0 & I_r & & \\ \hline & & & & 0 & \Pi_p \\ & & & & I_q & 0 \end{array}\right].$$

Thus, $T := P_2^T P_1^T H P_1 P_2$ has the desired form. The eigenvalue relation (3.4) follows from

$$\begin{bmatrix} 0 & I_p \\ -I_q & 0 \end{bmatrix}^{-1} \begin{bmatrix} -A_{11}^T & Q_{13} \\ G_{13} & A_{33} \end{bmatrix} \begin{bmatrix} 0 & I_p \\ -I_q & 0 \end{bmatrix} = - \begin{bmatrix} -A_{33}^T & Q_{13}^T \\ G_{13}^T & A_{11} \end{bmatrix}^T. \qquad \square$$

Lemma 3.1 is merely of theoretical interest and demonstrates that in order to solve the Hamiltonian eigenvalue problem, we can proceed by working only with H_{22}. But the transformations we have used in the proof are in general non-symplectic. If we want to compute invariant subspaces, eigenvectors, and/or the Hamiltonian Schur form given in Theorem 2.6, we can transform the Hamiltonian matrix in (3.2) such that it has Hamiltonian Schur form in rows and columns $1,\ldots,p+q$ and $n+1,\ldots,n+p+q$. But this can not be accomplished using only symplectic permutation matrices of the form (3.1). Therefore we need another class of transformation matrices.

DEFINITION 3.2. *A matrix $P_J \in \mathbb{R}^{2n\times 2n}$ is called a* J-permutation matrix *if*

a) it is symplectic, i.e., $P_J^T J P_J = J$,

b) $(P_J)_{j,k} \in \{-1,\, 0,\, 1\}$ for $j,k = 1,\ldots,2n$,

c) each row and column have exactly one nonzero entry.

As $P_J \in \mathcal{US}_{2n}$, it is clear that a similarity transformation by a J–permutation matrix preserves the Hamiltonian structure. In analogy to standard permutations, similarity transformations with P_J can be performed without floating point operations. Moreover, they can be represented by a signed integer vector `IP` of length n, where $\texttt{IP}(k) = \pm j$, $k = 1,\ldots,n$, $j = 1,\ldots,2n$, if rows and columns k, j are to be interchanged while the sign indicates if the corresponding entry in P_J is $+1$ or -1. The entries of P_J in rows $n+1$ to $2n$ can be deduced from `IP` and Proposition 2.3.

Furthermore, symplectic permutation matrices as given in (3.1) are J–permutation matrices.

LEMMA 3.3. *For any $H \in \mathcal{H}_{2n}$ having the form (3.2), there exists a J–permutation matrix P_J such that*

$$\hat{H} \;=\; P_J^T H P_J \;=\; \left[\begin{array}{cc|cc} \hat{A}_{11} & \hat{A}_{12} & \hat{G}_{11} & \hat{G}_{12} \\ 0 & \hat{A}_{22} & \hat{G}_{12}^T & \hat{G}_{22} \\ \hline 0 & 0 & -\hat{A}_{11}^T & 0 \\ 0 & \hat{Q}_{22} & -\hat{A}_{12}^T & -\hat{A}_{22}^T \end{array}\right] \begin{array}{l} \}p+q \\ \}n-p-q=r \\ \}p+q \\ \}n-p-q=r \end{array} \tag{3.9}$$

where $\hat{A}_{11}$ is upper triangular and with the notation in (3.2),

$$\hat{A}_{22} \;=\; A_{22}, \qquad \hat{G}_{22} \;=\; G_{22}, \qquad \hat{Q}_{22} \;=\; Q_{22}.$$

Proof. Let a Hamiltonian matrix H be given as in (3.2). We need a J–permutation matrix only in the first step. Let

$$P_1 \;:=\; \left[\begin{array}{ccc|ccc} I_p & 0 & 0 & 0 & 0 & 0 \\ 0 & I_r & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_q \\ \hline 0 & 0 & 0 & I_p & 0 & 0 \\ 0 & 0 & 0 & 0 & I_r & 0 \\ 0 & 0 & -I_q & 0 & 0 & 0 \end{array}\right].$$

Obviously, P_1 is a J–permutation matrix and

$$H_1 \;:=\; P_1^T H P_1 \;=\; \left[\begin{array}{ccc|ccc} A_{11} & A_{12} & -G_{13} & G_{11} & G_{12} & A_{13} \\ 0 & A_{22} & 0 & G_{12}^T & G_{22} & A_{23} \\ -Q_{13}^T & -Q_{23}^T & -A_{33}^T & A_{13}^T & A_{23}^T & -Q_{33} \\ \hline 0 & 0 & 0 & -A_{11}^T & 0 & Q_{13} \\ 0 & Q_{22} & 0 & -A_{12}^T & -A_{22}^T & Q_{23} \\ 0 & 0 & 0 & G_{13}^T & 0 & A_{33} \end{array}\right].$$

Now assume $G_{13} = 0$. Set $P_2 := \mathrm{diag}(\tilde{P}_2,\, \tilde{P}_2)$ where

$$\tilde{P}_2 \;:=\; \left[\begin{array}{ccc} 0 & I_p & 0 \\ 0 & 0 & I_r \\ I_q & 0 & 0 \end{array}\right].$$

Then,

$$H_2 \;:=\; P_2^T H_1 P_2 \;=\; \left[\begin{array}{ccc|ccc} -A_{33}^T & -Q_{13}^T & -Q_{23}^T & -Q_{33} & A_{13}^T & A_{23}^T \\ 0 & A_{11} & A_{12} & A_{13} & G_{11} & G_{12} \\ 0 & 0 & A_{22} & A_{23} & G_{12}^T & G_{22} \\ \hline 0 & 0 & 0 & A_{33} & 0 & 0 \\ 0 & 0 & 0 & Q_{13} & -A_{11}^T & 0 \\ 0 & 0 & Q_{22} & Q_{23} & -A_{12}^T & -A_{22}^T \end{array}\right].$$

We thus obtain the form (3.9) by another similarity transformation with

$$P_3 := \mathrm{diag}(\Pi_q,\, I_p,\, I_r,\, \Pi_q,\, I_p,\, I_r),$$

where Π_q is defined in (3.8). For the other case, i.e., $Q_{13} = 0$, set

$$\tilde{P}_2 \;=\; \begin{bmatrix} I_p & 0 & 0 \\ 0 & 0 & I_r \\ 0 & I_q & 0 \end{bmatrix},$$

and $P_2 := \mathrm{diag}(\tilde{P}_2,\, \tilde{P}_2)$, $P_3 := \mathrm{diag}(I_p,\, \Pi_q,\, I_r,\, I_p,\, \Pi_q,\, I_r)$.

In both cases, $\hat{P} := P_1 P_2 P_3$ is a J–permutation matrix and $\hat{H} := \hat{P}^T H \hat{P}$ is a Hamiltonian matrix having the desired form (3.9). □

In order to isolate eigenvalues, it is sufficient to restrict ourselves to symplectic permutations. But having computed the form (3.2), it is possible that there are still isolated eigenvalues in H_{22}. Applying the same procedure used to isolate eigenvalues in H to H_{22}, we can transform H_{22} to the form (3.2). This process can then be repeated until no more isolated eigenvalues are found. Accumulating all permutations in a symplectic permutation matrix P_s of the form (3.1), this results in a similarity transformation

(3.10) $\qquad \tilde{H} = P_s^T H P_s =$

$$= \left[\begin{array}{ccccc|ccccc}
A_{11} & \dots & A_{1,t} & \dots & A_{1,s} & G_{11} & \dots & G_{1,t} & \dots & G_{1,s} \\
0 & \ddots & \vdots & & \vdots & \vdots & & \vdots & \diagup & 0 \\
\vdots & \ddots & A_{t,t} & \dots & A_{t,s} & G_{1,t}^T & \dots & G_{t,t} & 0 & \vdots \\
\vdots & & \ddots & \ddots & \vdots & \vdots & \diagup & 0 & & \vdots \\
0 & \dots & \dots & 0 & A_{s,s} & G_{1,s}^T & 0 & \dots & \dots & 0 \\
\hline
0 & \dots & \dots & 0 & Q_{1,s} & -A_{11}^T & 0 & \dots & \dots & 0 \\
\vdots & & 0 & \diagup & \vdots & \vdots & \ddots & \ddots & & \vdots \\
\vdots & 0 & Q_{t,t} & \dots & Q_{t,s} & -A_{t,s}^T & \dots & -A_{t,t}^T & \ddots & \vdots \\
0 & \diagup & \vdots & & \vdots & \vdots & & \vdots & \ddots & 0 \\
Q_{1,s}^T & \dots & Q_{t,s}^T & \dots & Q_{s,s} & -A_{1,s}^T & \dots & -A_{t,s}^T & \dots & -A_{s,s}^T
\end{array}\right]
\begin{array}{l}
\}p_1 \\ \}p_2 + \ldots + p_{t-1} \\ \}p_t \\ \}p_{t+1} + \ldots + p_{s-1} \\ \}p_s \\ \\ \}p_1 \\ \}p_2 + \ldots + p_{t-1} \\ \}p_t \\ \}p_{t+1} + \ldots + p_{s-1} \\ \}p_s
\end{array} .$$

Here, $A_{j,j} \in \mathbb{R}^{p_j \times p_j}$, $j = 1, \ldots, s$, $j \neq t$, are upper triangular and for $j = 1, \ldots, t-1$, either $Q_{j,s-j+1} = 0$ or $G_{j,s-j+1} = 0$ and the Hamiltonian submatrix

$$H_{t,t} = \begin{bmatrix} A_{t,t} & G_{t,t} \\ Q_{t,t} & -A_{t,t}^T \end{bmatrix} \tag{3.11}$$

has no isolated eigenvalues. If we now define $p := \sum_{j=1}^{t-1} p_j$, $q := \sum_{j=t+1}^{s} p_j$, then we can partition $\tilde{H}$ in (3.10) as in (3.2). Then the first step in the proof of Lemma 3.1 can be performed to obtain the form (3.7). Just the block-structure of the upper left and lower right diagonal blocks in (3.7) are more complicated. But it is still possible to bring them to upper triangular form using repeatedly the same sequence of permutations used in the proof of Lemma 3.1. This shows that $2(p+q)$ eigenvalues of the Hamiltonian matrix can be read off directly from (3.10) and that $\tilde{H}$ is permutationally similar to

$$\begin{bmatrix} \tilde{T}_1 & \tilde{Y} & \tilde{Z} \\ 0 & H_{t,t} & \tilde{W} \\ 0 & 0 & \tilde{T}_2 \end{bmatrix} = \begin{bmatrix} \blacktriangle & \square & \square \\ & \square & \square \\ & & \blacktriangle \end{bmatrix}.$$

Further, we have $H_{t,t} \in \mathbb{R}^{2r \times 2r}$ where now, $r := p_t = n - p - q$, and

$$\sigma(H) = \sigma(H_{t,t}) \cup \bigcup_{\substack{j=1 \\ j \neq t}}^{s} \sigma(A_{j,j}) \cup \bigcup_{\substack{j=1 \\ j \neq t}}^{s} \sigma(-A_{j,j}^T). \tag{3.12}$$

If only eigenvalues are required, we can continue working only with $H_{t,t}$. If also eigenvectors and/or invariant subspaces are required, the similarity transformations used to solve the reduced-order eigenproblem for $H_{t,t}$ have to be applied to the whole matrix $\tilde{H}$. In that case, $\tilde{H}$ should be transformed to the form given in (3.9). Partitioning $\tilde{H}$ from (3.10) as in (3.2), we can perform the first step of the proof of Lemma 3.3 with the J-permutation matrix P_1. The subsequent steps to achieve upper triangular form in the first $p+q$ rows and columns have then to be performed for each of the first $t-1$ block columns, distinguishing the cases $Q_{j,s-j+1} = 0$ or $G_{j,s-j+1} = 0$ for $j = 1, \ldots, t-1$.

A procedure to transform a Hamiltonian matrix H to the form in (3.10) is given in the following algorithm. Note that in the given algorithm, $p_j = 1$ for all $j = 1, \ldots, s$, $j \neq t$.

ALGORITHM 3.4.

Input: *Matrices $A, G, Q \in \mathbb{R}^{n \times n}$, where $Q = Q^T$, $G = G^T$, defining a Hamiltonian matrix $H \in \mathbb{R}^{2n \times 2n}$.*

Output: *A symplectic permutation matrix P_s; matrices A, G, Q with $G = G^T$, $Q = Q^T$ defining a Hamiltonian matrix $H = P_s^T H P_s$ having the form (3.10); integers $i_l = p_1 + \ldots + p_{t-1} + 1 = p + 1$ and $i_h = i_l + p_t = n - q$.*

$P_s = I_{2n}$
$i_l = 1, \quad i_h = n$
$\tilde{i}_l = 0, \quad \tilde{i}_h = n$
`WHILE` $(i_l \neq \tilde{i}_l)$ `AND` $(i_h \neq \tilde{i}_h)$
  $\tilde{i}_l = i_l, \quad \tilde{i}_h = i_h$
  $i = i_h$
  `WHILE` $(i > i_l)$ `AND` $(i_h = \tilde{i}_h)$
    $\tilde{i}_h = i_h$
    $r = \sum_{\substack{j=i_l \\ j \neq i}}^{i_h} |a_{ij}| + \sum_{j=i_l}^{i_h} |g_{ij}|$
    `IF` $(r = 0)$ `THEN`
      $\tilde{P} = I_n + e_i e_{i_h}^T + e_{i_h} e_i^T - e_i e_i^T - e_{i_h} e_{i_h}^T$
      $A = \tilde{P}^T A \tilde{P}, \quad G = \tilde{P}^T G \tilde{P}, \quad Q = \tilde{P}^T Q \tilde{P}$
      $P = P \cdot \mathrm{diag}(\tilde{P}, \tilde{P})$
      $i_h = i_h - 1$
    `END IF`
    $i = i - 1$
  `END WHILE`
  $i = i_l$
  `WHILE` $(i < i_h)$ `AND` $(i_l = \tilde{i}_l)$
    $\tilde{i}_l = i_l$
    $c = \sum_{\substack{j=i_l \\ j \neq i}}^{i_h} |a_{ji}| + \sum_{j=i_l}^{i_h} |q_{ji}|$

```
            IF (c = 0) THEN
                P̃ = I_n + e_i e_{i_l}^T + e_{i_l} e_i^T − e_i e_i^T − e_{i_l} e_{i_l}^T
                A = P̃^T A P̃,      G = P̃^T G P̃,      Q = P̃^T Q P̃
                P = P · diag(P̃, P̃)
                i_l = i_l + 1
            END IF
            i = i + 1
         END WHILE
      END WHILE
END
```

In each execution of the outer `WHILE`-loop, we first search a row isolating an eigenvalue. If such a row is found, we look for a column isolating an eigenvalue. In this fashion it can be guaranteed that at the end, there are no more isolated eigenvalues although we always only touch the first n rows and columns of the Hamiltonian matrix.

In an actual implementation one would of course never form the permutation matrices explicitly but store the relevant information in an integer vector. Multiplications by permutation matrices are realized by swapping the data contained in the rows or columns to be permuted; for details, see, e.g., [3].

It is rather difficult to give a complete account of the cost of Algorithm 3.4. If there are no isolated eigenvalues, the algorithm requires $4n^2 - 2n$ floating point additions and $2n$ comparisons as opposed to $8n^2 - 4n$ additions and $4n$ comparisons for the unstructured permutation procedure from [25] as implemented in the LAPACK subroutine xGEBAL [3] when applied to $H \in \mathbb{R}^{2n\times 2n}$. The worst case for Algorithm 3.4 would be that in each execution of the outer `WHILE`-loop, an isolated eigenvalue is found in the last execution of the second inner `WHILE`-loop. In that case, the cost consists of $4n^3/3 + n^2 + O(n)$ floating point additions, $n^2 + n$ comparisons, and moving $2n^2 + 2n$ floating point numbers. But in this worst-case analysis, all eigenvalues are isolated such that after permuting, there is nothing left to do, and the Hamiltonian matrix is in Hamiltonian Schur form. A worst-case study for xGEBAL shows that the permutation part requires $8n^3/3 - 2n^2 + O(n)$ additions, $2n^2 + n$ comparisons, and moving $4n^2 + 2n$ floating point numbers. We can therefore conclude that Algorithm 3.4 is about half as expensive as the procedure proposed in [25] applied to a Hamiltonian matrix.

4. Symplectic Scaling. Suppose now that we have transformed the Hamiltonian matrix to the form (3.10). Since all subsequent transformations are determined from $H_{t,t}$, the scaling parameters to balance $H_{t,t}$ have now to be chosen such that the rows and columns of $H_{t,t}$ (instead of $\tilde{H}$) are as close in norm as possible.

In order to simplify notation we will in the sequel call the Hamiltonian matrix again H. Let H_{off} be the off-diagonal part of H, i.e.,

$$H_{\text{off}} = H - \text{diag}(H).$$

We may without loss of generality assume that none of the rows and columns of H_{off} vanishes identically. Otherwise, we could isolate another pair of eigenvalues.

Now we want to scale H such that the norms of its rows and columns are close in norm. As noted before, employing the technique of Parlett and Reinsch [25] destroys the Hamiltonian structure. Diagonal scaling has thus to be performed using a

symplectic diagonal matrix D_s. Such a matrix must have the form,

$$D_s = \begin{bmatrix} D & 0 \\ 0 & D^{-1} \end{bmatrix}, \tag{4.1}$$

where $D \in \mathbb{R}^{n\times n}$ is a nonsingular diagonal matrix.

Let us at first note an obvious result for Hamiltonian matrices. Here and in the sequel we will use the *colon notation* (see, e.g., [15]) $H(:,k)$, $H(j,:)$ to indicate the kth column and jth row, respectively, of a matrix H.

LEMMA 4.1. *Let $H \in \mathbb{R}^{2n\times 2n}$ be a Hamiltonian matrix. Then for all $p \geq 1$ and for all $i = 1,\ldots,n$,*

$$\|H(:,i)\|_p = \|H(n+i,:)^T\|_p, \tag{4.2}$$

$$\|H(i,:)^T\|_p = \|H(:,n+i)\|_p, \tag{4.3}$$

i.e., the p-norms of the ith column equals the norm of the $(n+i)$th row and the norm of the ith row equals the norm of the $(n+i)$th column.

Proof. The result is obvious by noting $\|x\|_p = \left(\sum_{k=1}^{2n} |x_k|^p\right)^{\frac{1}{p}}$ for $x \in \mathbb{R}^{2n}$ and observing that from the structure of Hamiltonian matrices, we have

$$\begin{aligned} H(:,i) &= [a_{1,i}, a_{2,i}, \ldots, a_{n,i}, q_{1,i}, q_{2,i}, \ldots, q_{n,i}]^T, \\ H(n+i,:) &= [q_{i,1}, q_{i,2}, \ldots, q_{i,n}, -a_{1,i}, -a_{2,i}, \ldots, -a_{n,i}], \end{aligned}$$

and furthermore, $q_{ij} = q_{ji}$ for all $1 \leq i,j \leq n$. Equation (4.3) follows analogously by noting $g_{ij} = g_{ji}$ for all $1 \leq i,j \leq n$. ∎

We can now conclude that it is sufficient to equilibrate the norms of the first n rows and columns of a $2n \times 2n$ Hamiltonian matrix by using a consequence of Lemma 4.1.

COROLLARY 4.2. *Let $H \in \mathbb{R}^{2n\times 2n}$ be a Hamiltonian matrix. Then for all $p \geq 1$ and for all $i = 1,\ldots,n$,*

$$\|H(:,i)\|_p = \|H(i,:)\|_p \iff \|H(:,n+i)\|_p = \|H(n+i,:)\|_p.$$

Since a similarity transformation with any diagonal matrix does not affect the diagonal elements of the transformed matrix, it is in the following sufficient to consider H_{off}. We will employ the notation

$$\begin{aligned} h_i &:= H_{\text{off}}(:,i) \;=\; i\text{th column of } H_{\text{off}}, \\ h^i &:= H_{\text{off}}(i,:)^T \;=\; \text{transpose of the } i\text{th row of } H_{\text{off}}. \end{aligned}$$

In the sequel, we will for convenience use $p = 1$. The results also hold for any other p-norm. From a computational point of view it is also reasonable to use the 1–norm, since its computation does not involve any floating point multiplications and furthermore, reducing the norm of H in one norm usually implies also a reduction in the other norms.

Equilibrating $\|h_i\|_1$ and $\|h^i\|_1$ can now be achieved in a similar way as in the Parlett/Reinsch method. If β denotes the base of the floating point arithmetic and σ_i is any signed integer, then they compute β^{σ_i} closest to the real scalar

$$\delta_i = \sqrt{\|h^i\|_1 / \|h_i\|_1}.$$

Thus, with $D^{(i)} = I_{2n} + (\delta_i - 1)e_i e_i^T$ and $\hat{H} = (D^{(i)})^{-1} H D^{(i)}$ it follows that $\|\hat{h}^i\|_1 = \|\hat{h}_i\|_1$. But if $\delta_i \neq 1$, $\hat{H}$ is in general no longer Hamiltonian. Unfortunately, using the symplectic diagonal matrix

$$D_s^{(i)} = \begin{bmatrix} D_i & 0 \\ 0 & D_i^{-1} \end{bmatrix},$$

where $D_i = I_n + (\delta_i - 1)e_i e_i^T$ and computing

$$\tilde{H} = (D_s^{(i)})^{-1} H D_s^{(i)} \tag{4.4}$$

we obtain

$$\|\tilde{h}_i\|_1 = \|\hat{h}_i\|_1 - \delta_i |q_{ii}| + \delta_i^2 |q_{ii}|, \qquad \|\tilde{h}^i\|_1 = \|\hat{h}^i\|_1 - \frac{1}{\delta_i}|g_{ii}| + \frac{1}{\delta_i^2}|g_{ii}| \tag{4.5}$$

and thus in general, $\|\tilde{h}_i\|_1 \neq \|\tilde{h}^i\|_1$.

Nevertheless, equilibrating the 1-norms of h_i and h^i can be achieved by requiring $\|\tilde{h}_i\|_1 = \|\tilde{h}^i\|_1$ and solving the resulting quartic equation

$$\delta_i(\|h_i\|_1 - |q_{ii}|) + \delta_i^2 |q_{ii}| = \frac{1}{\delta_i}(\|h^i\|_1 - |g_{ii}|) + \frac{1}{\delta_i^2}|g_{ii}|. \tag{4.6}$$

It remains to show that equation (4.6) has a positive solution.

THEOREM 4.3. *Let $H \in \mathbb{R}^{2n\times 2n}$ be a Hamiltonian matrix and denote its off-diagonal part by H_{off}. Assume that none of the rows and columns of H_{off} vanishes identically. Then there exists a unique real number $\delta_i > 0$ such that for $\tilde{H}$ as in (4.4) we have*

$$\|\tilde{h}_i\|_1 = \|\tilde{h}^i\|_1 = \|\tilde{h}_{n+i}\|_1 = \|\tilde{h}^{n+i}\|_1.$$

Proof. Solutions of Equation (4.6) are non-zero solutions of

$$0 = |q_{ii}|\delta_i^4 + (\|h_i\|_1 - |q_{ii}|)\delta_i^3 + (|g_{ii} - \|h^i\|_1)\delta_i - |g_{ii}| =: p(\delta_i), \tag{4.7}$$

where $p(t) = \sum_{k=0}^4 a_k t^k$. Recalling that $g_{ii} = (h^i)_{n+i}$, $q_{ii} = (h_i)_{n+i}$, the coefficients of the polynomial p satisfy

$$\begin{aligned} a_0 &= -|g_{ii}| \leq 0, \qquad a_1 = |g_{ii}| - \|h^i\|_1 \leq 0, \qquad a_2 = 0, \\ a_3 &= \|h_i\|_1 - |q_{ii}| \geq 0, \qquad a_4 = |q_{ii}| \geq 0. \end{aligned}$$

Since there is at most one change of sign in the coefficients of the polynomial p, Descartes' rule of signs shows that there is at most one positive zero of p. So if there exists a positive solution of (4.6), it is unique. By assumption, $\|h_i\|_1 \neq 0$ and $\|h^i\|_1 \neq 0$. Therefore, either $a_0 < 0$ or $a_1 < 0$ (as g_{ii} is part of h^i) and either $a_3 > 0$ or $a_4 > 0$ (as q_{ii} is part of h_i). Thus, we know that p is a polynomial of degree at least 3 with positive leading coefficient and hence, $\lim_{t\to\infty} p(t) = +\infty$.

On the other hand, if $g_{ii} \neq 0$, then $p(0) < 0$ and using the mean value theorem it follows that there exists a positive zero of p. If $g_{ii} = 0$, then $t = 0$ is a zero of p and $a_1 < 0$. The third order polynomial $q(t) = a_4 t^3 + a_3 t^2 + a_1$ has a positive zero because of the mean value theorem and $q(0) = a_1 < 0$ as well as $\lim_{t\to\infty} q(t) = +\infty$. Thus,

by $p(t) = tq(t)$, p has again at least one positive zero and hence equation (4.6) has at least one positive real solution regardless of the value of g_{ii}. On the other hand, it was already observed that there is at most one such solution and we can conclude that there exists a unique $\delta_i > 0$ solving equation (4.6) whence $\|\tilde{h}_i\|_1 = \|\tilde{h}^i\|_1$.

The other equalities follow immediately from Corollary 4.2. □

Computing the exact value δ_i equilibrating the ith and $(n+i)$th rows and columns would require the solution of the fourth-order equation (4.6). Since the diagonal similarity transformations are to be chosen from the set of machine numbers, it is sufficient to find the machine number β^{σ_i} closest to δ_i. This can be done similarly to the computation in the general case as proposed in [25] and implemented in the Fortran 77 subroutines BALANC from EISPACK [14] or its successor xGEBAL from LAPACK [3]. That is, starting with $\delta_i = 1$, the quantities in (4.5) are evaluated and compared. If $\|\tilde{h}_i\|_1 < \|\tilde{h}^i\|_1$, then this is repeated for $\delta_i = \beta^k$, $k = 1, 2, \ldots$ until $\|\tilde{h}_i\|_1 > \|\tilde{h}^i\|_1$. Otherwise, if $\|\tilde{h}_i\|_1 > \|\tilde{h}^i\|_1$, then we use $\delta_i = \beta^{-k}$, $k = 1, 2, \ldots$ until $\|\tilde{h}_i\|_1 < \|\tilde{h}^i\|_1$. This is achieved by the following algorithm.

ALGORITHM 4.4.

Input: *Hamiltonian matrix $H \in \mathcal{H}_{2n}$ having no isolated eigenvalues, β = base of floating point arithmetic.*

Output: *Diagonal matrix $D_s \in \mathcal{S}_{2n}$, H is overwritten by $D_s^{-1} H D_s \in \mathcal{H}_{2n}$ with row and column norms equilibrated as far as possible.*

$D_s = I_{2n}$
FOR $k = 1, 2, \ldots$ until convergence,
  FOR $i = 1, \ldots, n$,
    $\delta = 1$
    $c = \|h_i\|_1, \quad q_a = |h_{n+i,i}|, \quad c_a = c - q_a,$
    $r = \|h^i\|_1, \quad g_a = |h_{i,n+i}|, \quad r_a = r - g_a,$
    WHILE $c < r$
      $c_a = \beta c_a, \quad q_a = \beta^2 q_a, \quad c = c_a + q_a,$
      $r_a = r_a/\beta, \quad g_a = g_a/\beta^2, \quad r = r_a + g_a,$
      $\delta = \beta\delta,$
    END WHILE
    IF $\delta = 1$ THEN
      WHILE $r < c$
        $c_a = c_a/\beta, \quad q_a = q_a/\beta^2, \quad c = c_a + q_a,$
        $r_a = \beta(r_a - g_a), \quad g_a = \beta^2 g_a, \quad r = r_a + g_a,$
        $\delta = \delta/\beta,$
      END WHILE
    END IF
    $D_i = I + (\delta - 1) e_i e_i^T, \qquad D_s = D_s \mathrm{diag}\left(D_i, D_i^{-1}\right),$
    $H = \mathrm{diag}\left(D_i^{-1}, D_i\right) \cdot H \cdot \mathrm{diag}\left(D_i, D_i^{-1}\right),$
  END FOR i
END FOR k
END

One execution of the outer FOR-loop of Algorithm 4.4 can be considered as a sweep. The algorithm is terminated if for a whole sweep, all $D_i = I_n$. Usually, the row and column norms are approximately equal after very few sweeps. Afterwards, the iteration makes only very limited progress. Therefore, Parlett and Reinsch propose

in [25] a modification, which, translated to our problem, becomes:

> *Let δ_i be determined by the two inner* `WHILE`*-loops of Algorithm 4.4 and compute*
>
> $$c_i \;=\; \delta_i(\|h_i\|_1 - |q_{ii}|) + \delta_i^2|q_{ii}| \qquad r_i \;=\; \frac{1}{\delta_i}(\|h^i\|_1 - |g_{ii}|) + \frac{1}{\delta_i^2}|g_{ii}|. \tag{4.8}$$
>
> *If $(c_i + r_i) < \gamma(\|h_i\|_1 + \|h^i\|_1)$, (where γ is a given positive constant), then compute D_i as in Algorithm 4.4. Otherwise, set $D_i = I_n$.*

For $\gamma = 1$, the behavior is essentially the same as for Algorithm 4.4 (in a few cases, Algorithm 4.4 increases $\|h_i\|_1 + \|h^i\|_1$ which can not happen if $\gamma = 1$). For γ slightly smaller than one, a step is skipped if it would produce an insubstantial reduction of $\|h_i\|_1 + \|h^i\|_1$.

In an actual computation, the similarity transformations with the D_i's can be applied directly to the blocks A, G, and Q of the Hamiltonian matrix without forming the Hamiltonian matrix itself. Thus, each similarity transformation can be performed using only $4n - 4$ multiplications. When the standard (not structure preserving) scaling procedure from [25] is applied to H, each similarity transformation requires $4n - 2$ multiplications. (Recall that in Algorithm 4.4, two rows and columns are equilibrated at a time while only one row and column is treated in each step of the inner `FOR`-loop of the standard procedure.)

The number of sweeps required to converge is similar to those for the general case since the theory derived in [25] only requires the assumption of similarity transformations with diagonal matrices and that in step i of each sweep, the ith rows and columns are equilibrated as far as possible with $\delta_i = \beta^{\sigma_i}$. But this is accomplished by Algorithm 4.4. Moreover, if δ_i is taken as the exact solution of (4.6), the convergence of the sequence of similarity transformations to a stationary point can be proved as in [23, 16]. That is, if $\delta_i^{(k)}$ is the solution of (4.6) in sweep k, then $\lim_{k\to\infty} \delta_i^{(k)} = 1$ for all $i = 1, \ldots, n$ and hence in the limit, H is a balanced Hamiltonian matrix.

Note that here, each sweep has length n while in the standard balancing algorithm, one has to go through each row/column pair of the matrix and thus, each sweep has length $2n$. Thus, the computational cost for scaling a $2n \times 2n$ Hamiltonian matrix by Algorithm 4.4, assuming k_1 sweeps are required, is $4n^2k_1 + O(k_1 n)$ as opposed to $8n^2k_2 + O(k_2 n)$ for the standard scaling procedure as given in [25] with assumed k_2 sweeps required for convergence. In general, $k_1 \approx k_2$ such that the structure-preserving scaling strategy is about half as expensive as the standard procedure. These flop counts are based on the assumption that the cost for determining the δ_i can be considered as small ($O(1)$) compared to the similarity transformations.

REMARK 4.5. In [17] it is proposed to solve the matrix balancing problem using a convex programming approach. To compare the complexity of this approach to that of Algorithm 4.4, suppose that Algorithm 4.4 terminates after k_1 sweeps with $\big|\, \|h_i\|_1 - \|h^i\|_1 \,\big| < \mu_i$. For the matrix H to be balanced, let

$$\mathcal{E} := \{(i,j) \in \{1,\ldots,n\} \times \{1,\ldots,n\} \mid i \neq j,\ \ h_{ij} \neq 0\},$$

$\tau = \sum_{(i,j)\in\mathcal{E}} |h_{ij}|$, and $h_{\min} = \min\{|h_{ij}| \,|\, (i,j) \in \mathcal{E}\}$. Assume that $\sum_{i=1}^n \mu_i^2 =: \mu^2$ and $\mu < e\tau$. (Here, $e = \exp(1)$.) Then Theorem 5 in [17] states that the complexity of computing a diagonal matrix Y with positive diagonal entries such that the rows and columns of $Y^{-1}HY$ are balanced with the same accuracy as achieved by

Algorithm 4.4 is $O\left(n^4 \ln\left(\frac{ne\tau}{\mu}\ln\frac{\tau}{h_{\min}}\right)\right)$. From numerical experience, it can be assumed that $k_1 = O(1)$ with respect to n. Hence, Algorithm 4.4 can be considered to be of complexity $O(n^2)$. This complexity is clearly superior to that of the convex programming approach which is still the case if $k_1 = O(n)$.

Algorithm 4.4 requires a careful implementation to guard against over– and underflow due to a very large/small δ_i. Here, we can use the bounds discussed in [25] and implemented in LAPACK subroutine xGEBAL [3]; we just have to take into account that in each step we scale by $\beta^{\pm 2}$ rather than β as in xGEBAL.

5. Backtransformation, Ordering of Eigenvalues, and Applications. So far we have only considered the problem of computing the eigenvalues of a Hamiltonian matrix. In order to compute eigenvectors, invariant subspaces, and the solutions of algebraic Riccati equations, we have to transform the Hamiltonian matrix to real Schur form. As we are considering structure-preserving methods, the goal is to transform the Hamiltonian matrix to real Hamiltonian Schur form as given in Theorem 2.6 a) — if it exists.

Assume that we have applied Algorithm 3.4 to the Hamiltonian matrix and obtained a symplectic permutation matrix P_s such that $P_s^T H P_s$ has the form given in (3.10). Then, we have applied a J-permutation P_J to the permuted Hamiltonian matrix such that its rows and columns numbered $1, \ldots, p+q,\ n+1, \ldots, n+p+q$ are in Hamiltonian Schur form, i.e., $P_J^T P_s^T H P_s P_J$ has the form given in (3.9). (From Lemma 3.3 we know that such a P_J exists.) Next, we have applied Algorithm 4.4 to the Hamiltonian submatrix $H_{t,t} \in \mathcal{H}_{2r}$ from (3.11) and obtained a diagonal matrix $D = \operatorname{diag}\left(D_t, D_t^{-1}\right) \in \mathcal{S}_{2r}$. Let

$$D_s \;=\; \begin{bmatrix} I_{p+q} & 0 & 0 & 0 \\ 0 & D_r & 0 & 0 \\ 0 & 0 & I_{p+q} & 0 \\ 0 & 0 & 0 & D_r^{-1} \end{bmatrix}.$$

Then

$$\hat{H} \;:=\; D_s^{-1} P_J^T P_s^T H P_s P_j D_s \;=\; \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} & \hat{G}_{11} & \hat{G}_{12} \\ 0 & \hat{A}_{22} & \hat{G}_{12}^T & \hat{G}_{22} \\ 0 & 0 & -\hat{A}_{11}^T & 0 \\ 0 & \hat{Q}_{22} & -\hat{A}_{12}^T & -\hat{A}_{22}^T \end{bmatrix},$$

where $\hat{A}_{11} \in \mathbb{R}^{(p+q)\times(p+q)}$ is upper triangular and the Hamiltonian submatrix $\hat{H}_{22} := \begin{bmatrix} \hat{A}_{22} & \hat{G}_{22} \\ \hat{Q}_{22} & -\hat{A}_{22}^T \end{bmatrix}$ has no isolated eigenvalues and its rows and columns are equilibrated by Algorithm 4.4. Now assume the Hamiltonian Schur form of $\hat{H}_{22}$ exists and we have computed $U_{22} = \begin{bmatrix} \hat{U}_{22} & -\hat{V}_{22} \\ \hat{V}_{22} & \hat{U}_{22} \end{bmatrix} \in \mathcal{US}_{2r}$ that transforms $\hat{H}_{22}$ into real Hamiltonian Schur form. Set

$$U \;:=\; \begin{bmatrix} I_{p+q} & 0 & 0 & 0 \\ 0 & \hat{U}_{22} & 0 & -\hat{V}_{22} \\ 0 & 0 & I_{p+q} & 0 \\ 0 & \hat{V}_{22} & 0 & \hat{U}_{22} \end{bmatrix}.$$

and $S \;:=\; P_s P_J D_s U$. Then $H_1 := S^{-1} H S$ is real Hamiltonian quasi-triangular and $S \in \mathcal{S}_{2n}$. The first n columns of S span a Lagrangian H–invariant subspace. In most

applications, the c-stable H-invariant subspace is desired. Let us assume the method used to transform $\hat{H}_{22}$ to Hamiltonian Schur form chooses U_{22} such that the first r columns of U_{22}, i.e., the columns of $\begin{bmatrix} \hat{U}_{22} \\ \hat{V}_{22} \end{bmatrix}$, span the $\hat{H}_{22}$–invariant subspace of choice. But there is no guarantee that the isolated eigenvalues in $\hat{A}_{11}$ are the desired ones. In that case, we have to reorder the Hamiltonian Schur form in order to move the undesired eigenvalues to the lower right block of $\hat{H}$ and the desired ones to the upper left block. Assume that we want to compute the Lagrangian H–invariant subspace corresponding to a set $\Lambda = \{\lambda_1 \ldots, \lambda_n\} \subset \sigma(H)$ which is closed under complex conjugation. (Note that this is a necessary condition in order to obtain a Lagrangian invariant subspace [2]). Using the standard reordering algorithm for the real Schur form of an $n \times n$ unsymmetric matrix as given in [15, 30], we can find an orthogonal matrix $\tilde{U}_\Lambda$ such that with the orthogonal symplectic matrix $U_\Lambda = \mathrm{diag}\left(\tilde{U}_\Lambda, \tilde{U}_\Lambda\right)$, we have that

$$H_2 := U_\Lambda^T H_1 U_\Lambda = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} & \tilde{G}_{11} & \tilde{G}_{12} \\ 0 & \tilde{A}_{22} & \tilde{G}_{12}^T & \tilde{G}_{22} \\ 0 & 0 & -\tilde{A}_{11}^T & 0 \\ 0 & 0 & -\tilde{A}_{12}^T & -\tilde{A}_{22}^T \end{bmatrix} \begin{matrix} \}n-k \\ \}k \\ \}n-k \\ \}k \end{matrix} \tag{5.1}$$

where A_{11}, A_{22} are quasi-upper triangular, and

$$\Lambda = \sigma(\tilde{A}_{11}) \cup \sigma(-\tilde{A}_{22}^T), \qquad -\Lambda = \sigma(\tilde{A}_{22}) \cup \sigma(-\tilde{A}_{11}^T). \tag{5.2}$$

Therefore, we have to swap the eigenvalues in $\tilde{A}_{22}$ and $-\tilde{A}_{22}^T$. Note that the eigenvalues to be re-ordered are among the isolated eigenvalues and hence are real. This implies that $\tilde{A}_{22}$ is upper triangular. The re-ordering can be achieved analogously to the re-ordering of eigenvalues in the real Schur form as given in [15, 30]. The following procedure uses this standard re-ordering in order to swap eigenvalues within $\tilde{A}_{22}$ (and $-\tilde{A}_{22}^T$) and requires rotations working exclusively in rows and columns n and $2n$ in order to exchange eigenvalues from $\tilde{A}_{22}$ with eigenvalues from $-\tilde{A}_{22}^T$. Assume $(H_2)_{n,n} = (\tilde{A}_{22})_{kk} = -\lambda_n$. Then $(H_2)_{2n,2n} = (-\tilde{A}_{22}^T)_{kk} = \lambda_n$. Let $\begin{bmatrix} c_{\lambda_n} & -s_{\lambda_n} \\ s_{\lambda_n} & c_{\lambda_n} \end{bmatrix}$ be a 2×2 Givens rotation matrix that annihilates the second component of $\begin{bmatrix} \bar{g}_{nn} \\ -2\lambda_n \end{bmatrix}$ where $\tilde{g}_{nn} = (H_2)_{n,2n} = (\tilde{G}_{22})_{kk}$. Define

$$U_{\lambda_n} := I_{2n} + (c-1)(e_n e_n^T + e_{2n} e_{2n}^T) + s(e_{2n} e_n^T - e_n e_{2n}^T).$$

Then U_{λ_n} is a symplectic Givens rotation matrix acting in planes n and $2n$ and

$$U_{\lambda_n}^T H_2 U_{\lambda_n} = \left[\begin{array}{ccc|ccc} \tilde{A}_{11} & \tilde{A}_{12} & \bar{a}_{1,n} & \tilde{G}_{11} & \tilde{G}_{12} & \bar{g}_{1n} \\ 0 & \tilde{A}_{22} & \vdots & \tilde{G}_{12}^T & \tilde{G}_{22} & \vdots \\ 0 & 0 & \lambda_n & \bar{g}_{1n} & \ldots & \bar{g}_{nn} \\ \hline 0 & 0 & 0 & -\tilde{A}_{11}^T & 0 & 0 \\ 0 & 0 & 0 & -\tilde{A}_{12}^T & -\tilde{A}_{22}^T & 0 \\ 0 & 0 & 0 & -\bar{a}_{1,n} & \ldots & -\lambda_n \end{array}\right] \begin{matrix} \}n-k \\ \}k-1 \\ \}1 \\ \}n-k \\ \}k-1 \\ \}1 \end{matrix}.$$

Here, the bar indicates elements changed by the similarity transformation.

The next step is now to move λ_n up in the upper diagonal block using again the standard ordering subroutine such that we obtain again the form given in (5.1), just

$\tilde{A}_{11} \in \mathbb{R}^{(n-k+1)\times(n-k+1)}$ and again, the relations (5.2) hold. This procedure has now to be repeated until $k = 0$ and $\Lambda = \sigma(\hat{A}_{11})$.

REMARK 5.1. If the Hamiltonian matrix has the form $H = \begin{bmatrix} A & BB^T \\ C^TC & -A^T \end{bmatrix}$ which corresponds to a linear system $\dot{x} = Ax + Bu$, $y = Cx$, with (A, B) stabilizable and (C, A) detectable, then each isolated eigenvalue in (5.1) given by the diagonal elements of $\tilde{A}_{11}$ has negative real part. Otherwise, these eigenvalues are unstable or undetectable and can not be stabilized/detected. Therefore, if we have not mixed up blocks by the J-permutation matrix P_J (i.e., in Algorithm 3.4, $i_h = n$) and the c-stable H-invariant subspace is required, no re-ordering is necessary.

REMARK 5.2. When solving algebraic Riccati equations using any approach based on the Hamiltonian eigenproblem, the symplectic balancing strategy proposed here is often not enough to minimize errors caused by ill-scaling. This is due to the effect that for a balanced Hamiltonian matrix $H = \begin{bmatrix} A & G \\ Q & -A^T \end{bmatrix}$ we still may have $\|Q\| \gg \|G\|$ which may cause large errors when computing invariant subspaces [27]. Therefore, another symplectic scaling using a similarity transformation with diag $\left(\sqrt{\rho}I_n, \frac{1}{\sqrt{\rho}}I_n\right) \in \mathcal{S}_{2n}$, $\rho \in \mathbb{R}$, should be applied to H in order to achieve $\|A\| \approx \|G\| \approx \|Q\|$ as far as possible; see [4] for details and a discussion of several heuristic strategies to achieve this.

REMARK 5.3. Everything derived so far for Hamiltonian matrices can be applied in the same way to skew-Hamiltonian matrices. If $N \in \mathcal{SH}_{2n}$, then $N = \begin{bmatrix} A & G \\ Q & A^T \end{bmatrix}$ with $G = -G^T$, $Q = -Q^T$. The skew-Hamiltonian structure is again preserved under symplectic similarity transformations. Hence, isolating eigenvalues, re-ordering, etc., can be achieved in the same way as for Hamiltonian matrices as all considered transformations do not depend on the signs in the matrix blocks A, G, Q, but only on the distinction zero/non-zero when isolating eigenvalues and on the absolute values of the entries when equilibrating rows and norms. Note that Algorithm 4.4 even simplifies quite a lot for *real* skew-Hamiltonian matrices. As $q_{ii} = g_{ii} =$ for all $i = 1, \ldots, n$, the scaling factor δ_i can be computed as in the general balancing algorithm for non-symmetric matrices because in (4.5) we obtain $\|\tilde{h}_i\|_1 = \|\tilde{h}^i\|_1$.

Eigenvalues of skew-Hamiltonian matrices as well as a skew-Hamiltonian Schur form can be computed in a numerically strong backward stable way by Van Loan's method [31]. It is advisable to balance skew-Hamiltonian matrices using the proposed strategies prior to applying this algorithm.

REMARK 5.4. We have considered so far only real Hamiltonian and skew-Hamiltonian matrices. Isolating eigenvalues and equilibrating rows and columns for complex (skew-)Hamiltonian matrices can be achieved in exactly the same way. A structure-preserving, numerically backward stable (and hence numerically strong backward stable) method for solving the complex (skew-)Hamiltonian eigenproblem has recently been proposed [8]. The proposed symplectic balancing method can (and should) also be used prior to applying this algorithm.

6. Numerical Examples. We have tested the symplectic balancing strategy for eigenvalue computations. The computations were done in MATLAB[1] Version 5.2 with machine precision $\varepsilon \approx 2.2204 \times 10^{-16}$. Algorithms 3.4 and 4.4 were implemented as MATLAB functions. We used the modified algorithm as suggested by (4.8) where we set $\gamma = 0.95$ as suggested in [25] and implemented in the LAPACK subroutine xGEBAL [3]. The eigenvalues of the balanced and the unbalanced Hamiltonian matrix

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

were computed by the square-reduced method using a MATLAB function `sqred` which implements the explicit version of the square-reduced method (see [31]).

We also tested the effects of symplectic balancing for the numerically backward stable, structure-preserving method for the Hamiltonian eigenvalue problem presented in [7]. Like the square-reduced method, this algorithm uses the square of the Hamiltonian matrix. But it avoids forming the square explicitly using a symplectic URV-type decomposition of the Hamiltonian matrix.

As reference values we used the eigenvalues computed by the unsymmetric QR algorithm with Parlett/Reinsch balancing as implemented in the LAPACK expert driver routine DGEEVX [3], applied to the Hamiltonian matrix and using quadruple precision.

Moreover, we tested the effects of balancing when solving algebraic Riccati equations with the structure-preserving multishift method presented in [1] for the examples from the benchmark collection [6]. We only present some of the most intriguing results.

EXAMPLE 6.1. [6, Example 6] The system data come from an optimal control problem for a J–100 jet engine as a special case of a multivariable servomechanism problem. The resulting Hamiltonian matrix $H \in \mathbb{R}^{60\times 60}$ has 8 isolated eigenvalues: triple eigenvalues at ± 20.0 and simple eigenvalues at ± 33.3.

Algorithm 3.4 returns $i_l = 5$ and $i_h = 30$ and for the permuted Hamiltonian matrix we have

$$\begin{aligned} H_{1:i_l-1,1:i_l-1} &= \mathrm{diag}(\,-33.3,\,-20.0,\,-20.0,\,-20.0\,) \\ H_{n+1:n+i_l-1,n+1:n+i_l-1} &= \mathrm{diag}(\,33.3,\,20.0,\,20.0,\,20.0\,). \end{aligned}$$

Next, the Hamiltonian submatrix

$$H_{22} \;=\; \begin{bmatrix} H_{5:30,5:30} & H_{5:30,35:60} \\ H_{35:60,5:30} & -H_{5:30,5:30}^T \end{bmatrix}$$

is scaled using Algorithm 4.4. After six sweeps, we obtain the balanced Hamiltonian matrix H_b. We have $\|H\|_2 = 1.44 \times 10^8$ and $\|H_b\|_2 = 6.54 \times 10^2$, that is, we have decreased the 2-norm of the matrix used in the subsequent eigenvalue computation by more than five orders of magnitude. If the eigenvalues are computed by the square-reduced method applied to the unbalanced Hamiltonian matrix, the triplet of isolated eigenvalues is returned as a pair of conjugate complex eigenvalues with relative errors $\approx 4.06 \times 10^{-11}$ and a simple eigenvalue with relative error $\approx 3.96 \times 10^{-11}$. For the simple eigenvalue at 33.3, the relative error is $\approx 7.7 \times 10^{-15}$. For the balanced version, these eigenvalues are returned with full accuracy since they are not affected by roundoff errors. The relative errors for the other (not isolated) eigenvalues are given in Figure 6.1 where we use the relative distance of the computed eigenvalues to those computed by DGEEVX as an estimate of the real relative error.

Figure 6.1 only contains the relative errors for the eigenvalues with positive real parts as `sqred` returns the eigenvalues as exact plus-minus pairs. The '+' for the 26th eigenvalue is missing as the computed relative error for the balanced version is zero with respect to machine precision. The eigenvalues are ordered by increasing absolute values. From Figure 6.1, the increasing accuracy for decreasing ratio $\|H\|_2/|\lambda|$ is obvious — with or without balancing. All computed eigenvalues of the balanced matrix are more accurate than for the unbalanced one. The increase in accuracy is more significant for the eigenvalues of smaller magnitude. This reflects the decrease

of the ratios $\|H\|_2/|\lambda|$ which more or less determines the accuracy of the computed eigenvalues; see [31]. The decrease factor for $\|H\|_2$ is about 5×10^{-6}. The accuracy for the eigenvalues of smaller magnitude increases by almost the same factor.

From Figure 6.2 we see that symplectic balancing also improves the eigenvalues computed by the method proposed in [7]. As the method does not suffer from the $\|H\|_2/|\lambda|$ perturbation, the accuracy for all computed eigenvalues is similar. Also note that in the unbalanced version, the isolated eigenvalues are computed with a relative accuracy ranging from 7.0×10^{-14} to 1.2×10^{-15}.

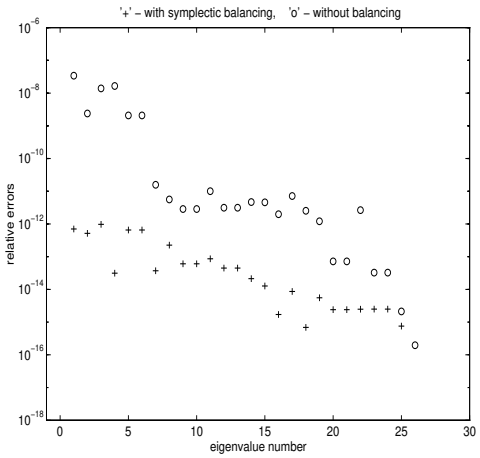

FIG. 6.1. *square-reduced method.*

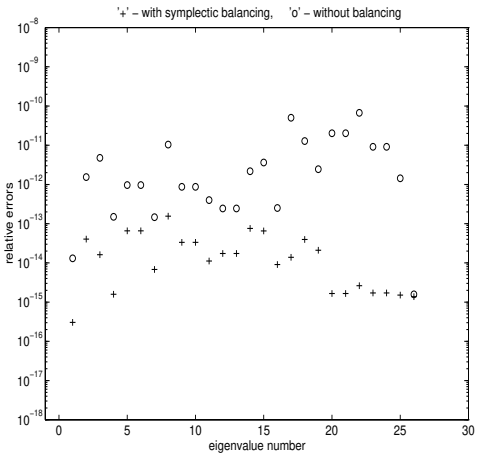

FIG. 6.2. *symplectic URV + periodic QR.*

Using the balanced matrix in order to solve algebraic Riccati equations by the multishift method as described in [1], we obtain the following results: if the multishift method is applied to the unbalanced data, the computed solution yields a residual

$$r_F := \|Q + A^T X + XA - XGX\|_F \tag{6.1}$$

of size 1.5×10^{-6} while using the balanced Hamiltonian matrix we get $r_F = 8.1 \times 10^{-10}$. This shows that numerical methods for solving algebraic Riccati equations can substantially be improved employing balancing.

EXAMPLE 6.2. [6, Example 13] The Hamiltonian matrix is defined as in (1.1) with

$$A = \begin{bmatrix} 0 & 0.4 & 0 & 0 \\ 0 & 0 & 0.345 & 0 \\ 0 & -0.524 \cdot \tau & -0.465 \cdot \tau & 0.262 \cdot \tau \\ 0 & 0 & 0 & -\tau \end{bmatrix}, \qquad \begin{aligned} G &= \tau^2 e_4 e_4^T, \\ Q &= \operatorname{diag}(1,\ 0,\ 1,\ 0), \end{aligned}$$

where $\tau = 10^6$ and e_4 denotes the fourth unit vector. After four sweeps of Algorithm 4.4, $\|H\|_2$ is reduced from 10^{12} to 1.5×10^6. The accuracy of the computed eigenvalues did not improve significantly, but for the stabilizing solution of the algebraic Riccati equation, the Frobenius norm of the residual as defined in (6.1) dropped from $r_F = 5.4 \times 10^{-11}$ to $r_F = 1.8 \times 10^{-15}$.

7. Concluding Remarks. We have seen that isolated eigenvalues of a real Hamiltonian matrix can be deflated using similarity transformations with symplectic

permutation matrices, the deflated problem can be scaled in order to reduce the norm of the deflated Hamiltonian matrix and to equilibrate its row and column norms, and the remaining (not isolated) eigenvalues can then be determined by computing the eigenvalues of the deflated, balanced Hamiltonian submatrix. If invariant subspaces are required, then we can use J-permutation matrices and a symplectic re-ordering strategy in order to obtain the desired invariant subspaces. The same method can be applied in order to balance skew-Hamiltonian and complex (skew-)Hamiltonian matrices.

Numerical examples demonstrate that symplectic balancing can significantly improve the accuracy of eigenvalues of Hamiltonian matrices as well as the accuracy of solutions of the associated algebraic Riccati equations computed by structure-preserving methods.

Final Remark and Acknowledgments. The work presented in this article continues preliminary results derived in [4]. The author would like to thank Ralph Byers, Heike Faßbender, and Volker Mehrmann for helpful suggestions.

REFERENCES

[1] G. Ammar, P. Benner, and V. Mehrmann, *A multishift algorithm for the numerical solution of algebraic Riccati equations*, Electr. Trans. Num. Anal., 1 (1993), pp. 33–48.

[2] G. Ammar and V. Mehrmann, *On Hamiltonian and symplectic Hessenberg forms*, Linear Algebra Appl., 149 (1991), pp. 55–72.

[3] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide*, SIAM, Philadelphia, PA, second ed., 1995.

[4] P. Benner, *Contributions to the Numerical Solution of Algebraic Riccati Equations and Related Eigenvalue Problems*, Dissertation, Fakultät für Mathematik, TU Chemnitz–Zwickau, 09107 Chemnitz (Germany), Feb. 1997.

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

[6] P. Benner, A. Laub, and V. Mehrmann, *A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case*, Tech. Rep. SPC 95_22, Fakultät für Mathematik, TU Chemnitz–Zwickau, 09107 Chemnitz, FRG, 1995. Available from `http://www.tu-chemnitz.de/sfb393/spc95pr.html`.

[7] 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 (1998), pp. 329–358.

[8] ———, *A note on the numerical solution of complex Hamiltonian and skew-Hamiltonian eigenvalue problems*, Electr. Trans. Num. Anal., 8 (1999), pp. 115–126.

[9] S. Boyd, V. Balakrishnan, and P. Kabamba, *A bisection method for computing the $\mathcal{H}_\infty$ norm of a transfer matrix and related problems*, Math. Control, Signals, Sys., 2 (1989), pp. 207–219.

[10] N. Bruinsma and M. Steinbuch, *A fast algorithm to compute the H_∞-norm of a transfer function matrix*, Sys. Control Lett., 14 (1990), pp. 287–293.

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

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

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

[14] B. Garbow, J. Boyle, J. Dongarra, and C. Moler, *Matrix Eigensystem Routines—EISPACK Guide Extension*, vol. 51 of Lecture Notes in Computer Science, Springer-Verlag, New York, 1977.

[15] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, third ed., 1996.

[16] J. Grad, *Matrix balancing*, Comput. J., 14 (1971), pp. 280–284.

[17] B. Kalantari, L. Khachiyan, and A. Shokoufandeh, *On the complexity of matrix balancing*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 450–463.

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

[19] A. J. 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.

[20] W.-W. Lin, V. Mehrmann, and H. Xu, *Canonical forms for Hamiltonian and symplectic matrices and pencils*, Linear Algebra Appl., 301–303 (1999), pp. 469–533.

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

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

[23] E. E. Osborne, *On pre-conditioning of matrices*, J. ACM, 7 (1960), pp. 338–345.

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

[25] B. Parlett and C. Reinsch, *Balancing a matrix for calculation of eigenvalues and eigenvectors*, Numer. Math., 13 (1969), pp. 296–304.

[26] P. Petkov, N. Christov, and M. Konstantinov, *Computational Methods for Linear Control Systems*, Prentice-Hall, Hertfordshire, UK, 1991.

[27] P. Petkov, M. Konstantinov, D. Gu, and I. Postlethwaite, *Solving continuous-time matrix algebraic Riccati equations with condition and accuracy estimates*, Tech. Rep. 94–64, Control Systems Research, Department of Engineering, Leicester University, Leicester LE1 7RH, UK, Dec. 1994.

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

[29] J. Sreedhar, P. Van Dooren, and A. Tits, *A fast algorithm to compute the real structured stability radius*, in Proc. Conf. Centennial Hurwitz on Stability Theory, Ticino (CH), May 21-26, 1995.

[30] G. W. Stewart, *Algorithm 506—HQR3 and EXCHNG: Fortran subroutines for calculating and ordering the eigenvalues of a real upper Hessenberg matrix*, ACM Trans. Math. Software, 2 (1976), pp. 275–280.

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

Berichte aus der Technomathematik ISSN 1435-7968

`http://www.math.uni-bremen.de/zetem/berichte.html`

— Vertrieb durch den Autor —

Reports **Stand: 23. Februar 2000**

98–01. Peter Benner, Heike Faßbender:
An Implicitly Restarted Symplectic Lanczos Method for the Symplectic Eigenvalue Problem, Juli 1998.

98–02. Heike Faßbender:
Sliding Window Schemes for Discrete Least-Squares Approximation by Trigonometric Polynomials, Juli 1998.

98–03. Peter Benner, Maribel Castillo, Enrique S. Quintana-Ortí:
Parallel Partial Stabilizing Algorithms for Large Linear Control Systems, Juli 1998.

98–04. Peter Benner:
Computational Methods for Linear–Quadratic Optimization, August 1998.

98–05. Peter Benner, Ralph Byers, Enrique S. Quintana-Ortí, Gregorio Quintana-Ortí:
Solving Algebraic Riccati Equations on Parallel Computers Using Newton's Method with Exact Line Search, August 1998.

98–06. Lars Grüne, Fabian Wirth:
On the rate of convergence of infinite horizon discounted optimal value functions, November 1998.

98–07. Peter Benner, Volker Mehrmann, Hongguo Xu:
A Note on the Numerical Solution of Complex Hamiltonian and Skew-Hamiltonian Eigenvalue Problems, November 1998.

98–08. Eberhard Bänsch, Burkhard Höhn:
Numerical simulation of a silicon floating zone with a free capillary surface, Dezember 1998.

99–01. Heike Faßbender:
The Parameterized SR Algorithm for Symplectic (Butterfly) Matrices, Februar 1999.

99–02. Heike Faßbender:
Error Analysis of the symplectic Lanczos Method for the symplectic Eigenvalue Problem, März 1999.

99–03. Eberhard Bänsch, Alfred Schmidt:
Simulation of dendritic crystal growth with thermal convection, März 1999.

99–04. Eberhard Bänsch:
Finite element discretization of the Navier-Stokes equations with a free capillary surface, März 1999.

99–05. Peter Benner:
Mathematik in der Berufspraxis, Juli 1999.

99–06. Andrew D.B. Paice, Fabian R. Wirth:
Robustness of nonlinear systems and their domains of attraction, August 1999.

99–07. Peter Benner, Enrique S. Quintana-Ortí, Gregorio Quintana-Ortí:
Balanced Truncation Model Reduction of Large-Scale Dense Systems on Parallel Computers, September 1999.

99–08. Ronald Stöver:
Collocation methods for solving linear differential-algebraic boundary value problems, September 1999.

99–09. Huseyin Akcay:
Modelling with Orthonormal Basis Functions, September 1999.

99–10. Heike Faßbender, D. Steven Mackey, Niloufer Mackey:
Hamilton and Jacobi come full circle: Jacobi algorithms for structured Hamiltonian eigenproblems, Oktober 1999.

99–11. Peter Benner, Vincente Hernández, Antonio Pastor:
On the Kleinman Iteration for Nonstabilizable System, Oktober 1999.

99–12. Peter Benner, Heike Faßbender:
A Hybrid Method for the Numerical Solution of Discrete-Time Algebraic Riccati Equations, November 1999.

99–13. Peter Benner, Enrique S. Quintana-Ortí, Gregorio Quintana-Ortí:
Numerical Solution of Schur Stable Linear Matrix Equations on Multicomputers, November 1999.

99–14. Eberhard Bänsch, Karol Mikula:
Adaptivity in 3D Image Processing, Dezember 1999.

00–01. Peter Benner, Volker Mehrmann, Hongguo Xu:
Perturbation Analysis for the Eigenvalue Problem of a Formal Product of Matrices, Januar 2000.

00–02. Ziping Huang:
Finite Element Method for Mixed Problems with Penalty, Januar 2000.

00–03. Gianfrancesco Martinico:
Recursive mesh refinement in 3D, Februar 2000.

00–04. Eberhard Bänsch, Christoph Egbers, Oliver Meincke, Nicoleta Scurtu:
Taylor-Couette System with Asymmetric Boundary Conditions, Februar 2000.

00–05. Peter Benner:
Symplectic Balancing of Hamiltonian Matrices, Februar 2000.