A numerically strongly stable method for computing the Hamiltonian Schur form

Delin Chu * Xinmin Liu † Volker Mehrmann ‡

October 4, 2004

[1]Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543 matchudl@math.nus.edu.sg

[2]Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576.

[3]Institut für Mathematik MA 4-5, TU Berlin, Straße des 17. Juni 136, D-10623 Berlin, Germany mehrmann@math.TU-Berlin.DE. This author was supported by *Deutsche Forschungsgemeinschaft*, Research Grant Me 790/11-3

Technische Universität Berlin

Institut für Mathematik

A numerically strongly stable method for computing the Hamiltonian Schur form

Delin Chu, Xinmin Liu, and Volker Mehrmann

Preprint 24/2004

Preprint-Reihe des Instituts für Mathematik
Technische Universität Berlin

Report 24/2004 **October 2004**

Abstract

In this paper we solve a long-standing open problem in numerical analysis called *Van Loan's Curse*. We derive a new numerical method for computing the Hamiltonian Schur form of a Hamiltonian matrix that has no purely imaginary eigenvalues. The proposed method is numerically strongly backward stable, i.e., it computes the exact Hamiltonian Schur form of a nearby Hamiltonian matrix, and it is of complexity $\mathbf{O}(n^3)$ and thus Van Loan's curse is lifted. We demonstrate the quality of the new method by showing its performance for the benchmark collection of continuous-time algebraic Riccati equations.

Keywords. Hamiltonian matrix, skew-Hamiltonian matrix, real Hamiltonian Schur form, real skew-Hamiltonian Schur form, symplectic URV-decomposition, stable invariant subspace.

AMS subject classification. 65F15, 93B36, 93B40, 93C60.

1 Introduction

It has been a long-standing problem, see [30], to compute the Hamiltonian Schur form and the invariant subspace associated with the eigenvalues in the left half plane for Hamiltonian matrices. Recall the following definitions.

Definition 1 *Let*

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

where I_n is the identity matrix in $\mathbb{R}^{n\times n}$.

(i) A matrix $\mathcal{M} \in \mathbb{R}^{2n\times 2n}$ is called Hamiltonian *if $\mathcal{M}J_n = (\mathcal{M}J_n)^T$. Every Hamiltonian matrix $\mathcal{M} \in \mathbb{R}^{2n\times 2n}$ is of the form*

$$\mathcal{M} = \begin{bmatrix} F & G \\ H & -F^T \end{bmatrix},$$

where $F, G, H \in \mathbb{R}^{n\times n}$, $G = G^T$, and $H = H^T$.

(ii) A matrix $\mathcal{N} \in \mathbb{R}^{2n\times 2n}$ is called skew-Hamiltonian *if $\mathcal{N}J_n = -(\mathcal{N}J_n)^T$. Every skew-Hamiltonian matrix $\mathcal{N} \in \mathbb{R}^{2n\times 2n}$ is of the form*

$$\mathcal{N} = \begin{bmatrix} F & G \\ H & F^T \end{bmatrix},$$

where $F, G, H \in \mathbb{R}^{n\times n}$, $G = -G^T$, and $H = -H^T$.

(iii) A matrix $\mathcal{S} \in \mathbb{R}^{2n\times 2n}$ *is called* symplectic *if* $\mathcal{S}J_n\mathcal{S}^T = J_n$. *A matrix* $\mathcal{U} \in \mathbb{R}^{2n\times 2n}$ *is called* orthogonal-symplectic *if* $\mathcal{U}$ *is symplectic and orthogonal. Every orthogonal-symplectic matrix* $\mathcal{U} \in \mathbb{R}^{2n\times 2n}$ *is of the form*

$$\mathcal{U} = \begin{bmatrix} U_1 & -U_2 \\ U_2 & U_1 \end{bmatrix},$$

where $U_1, U_2 \in \mathbb{R}^{n\times n}$.

(iv) A subspace L *of* $\mathbb{R}^{2n}$ *is called* isotropic *if* $x^T J_n y = 0$ *for all* $x, y \in L$. *It is called* Lagrangian *if it is maximal isotropic, i.e. it has dimension* n.

Definition 2 *(i) A real, block upper-triangular matrix of the form*

$$T = \begin{array}{c} \begin{array}{ccc} n_1 & \cdots & n_l \end{array} \\ \begin{bmatrix} T_{1,1} & \cdots & T_{1l} \\ & \ddots & \vdots \\ & & T_{ll} \end{bmatrix} \end{array} \begin{array}{c} \\ n_1 \\ \vdots \\ n_l \end{array},$$

with diagonal blocks $T_{i,i},\ i = 1, \cdots, l$ *that are either* 1×1 *or* 2×2 *with a pair of (non-real) complex conjugate eigenvalues is called a* real Schur form, *see e.g. [18].*

(ii) Let $\mathcal{M} \in \mathbb{R}^{2n\times 2n}$ *be Hamiltonian. If there exists an orthogonal-symplectic matrix* $\mathcal{U} \in \mathbb{R}^{2n\times 2n}$ *such that*

$$\mathcal{U}^T\mathcal{M}\mathcal{U} = \begin{bmatrix} M_{1,1} & M_{1,2} \\ 0 & -M_{1,1}^T \end{bmatrix}, \tag{1}$$

where $M_{1,1}$ *is in real Schur form, then (1) is called a* real Hamiltonian Schur form of $\mathcal{M}$.

(iii) Let $\mathcal{N} \in \mathbb{R}^{2n\times 2n}$ *be skew-Hamiltonian. If there exists an orthogonal-symplectic matrix* $\mathcal{U} \in \mathbb{R}^{2n\times 2n}$ *such that*

$$\mathcal{U}^T\mathcal{N}\mathcal{U} = \begin{bmatrix} N_{1,1} & N_{1,2} \\ 0 & N_{1,1}^T \end{bmatrix}, \tag{2}$$

where $N_{1,1}$ *is in real Schur form, then (2) is called a* real skew-Hamiltonian Schur form of $\mathcal{N}$.

The *Hamiltonian eigenvalue problem*, i.e., to compute eigenvalues and invariant subspaces of Hamiltonian matrices, is of great importance in many applications in control theory and signal processing, see [20, 24, 29, 39], since it is at the heart of almost any solution method for determining optimal and robust controllers. Computational methods for this problem are well established and have been constantly improved, since the landmark papers of Laub [25] and Paige/Van Loan [30]. In the latter paper the authors posed the open question to derive an $\mathbf{O}(n^3)$ method that is numerically strongly backwards stable in the sense of [13], i.e., the method computes the exact eigenvalues and invariant subspaces of a nearby Hamiltonian matrix. This problem is known as *Van Loan's curse*. Many attempts have been made to derive such a method, see [14, 26, 29] and the references therein, but it has been shown in [1] that a modification of the standard QR-method to solve this problem is in general hopeless, due to the missing reduction to a Hessenberg-like form. Only in special cases such a method has been found [15, 16].

A major step in the direction of deriving a structure preserving method for the Hamiltonian eigenvalue problem was made in [6, 7, 8], where new methods are described that improve the so-called *square-reduced method* of Van Loan [35]. The basis for these methods is the observation that one can compute the real skew-Hamiltonian Schur form of $\mathcal{N} = \mathcal{M}^2$, without forming the square, by exploiting the relationship between the invariant subspaces of the Hamiltonian matrix $\mathcal{M}$, the extended matrix $\begin{bmatrix} 0 & \mathcal{M} \\ \mathcal{M} & 0 \end{bmatrix}$ and the symplectic URV-decomposition introduced in [7]. Our new approach is strongly based on this symplectic URV-decomposition which we will summarize and slightly modify in Section 2. In Section 3 we will then derive our new strongly backward stable method of complexity $\mathbf{O}(n^3)$ to determine the Hamiltonian Schur form. Some algorithmic details are discussed in Section 4. In Section 5 we will demonstrate the properties of the new method by its performance on the benchmark collection [5]. Conclusions are given in Section 6.

Although some of the results may be extended to the general case, in this paper we only consider the Hamiltonian eigenvalue problem for matrices that have no purely imaginary eigenvalues. For Hamiltonian matrices with purely imaginary eigenvalues, the situation is much more complicated, since in this case not always a Hamiltonian Schur form exists, see [27, 31, 32]. A complete parameterization of all possible Hamiltonian Schur forms and corresponding Lagrange invariant subspaces has recently been given in [17].

2 Previous results

The basis of our new method to compute the Hamiltonian Schur form for a real Hamiltonian matrix $\mathcal{M}$ is the computation of the real skew-Hamiltonian Schur form of $\mathcal{N} = \mathcal{M}^2$. The real skew-Hamiltonian Schur form was originally suggested in [35], where also a numerical method was presented that computes this form and that is backward stable for $\mathcal{N}$. But there are several drawbacks of this method. First, it has to compute $\mathcal{N} = \mathcal{M}^2$, which in a worst case situation creates an error of order $\sqrt{\text{eps}}$ for the eigenvalues that are small in modulus, where eps is the machine precision. A backward stable method to compute the real skew-Hamiltonian Schur form of $\mathcal{N} = \mathcal{M}^2$ without forming $\mathcal{M}^2$ explicitly was developed in [7]. The method uses the following decomposition.

Lemma 3 *[7] (*Symplectic URV-decomposition.*) Let $\mathcal{M} \in \mathbb{R}^{2n\times 2n}$ be a Hamiltonian matrix. Then there exist orthogonal-symplectic matrices $\mathcal{U}, \mathcal{V} \in \mathbb{R}^{2n\times 2n}$ and integers $n_1, \cdots, n_l$ such that*

$$\mathcal{U}^T\mathcal{M}\mathcal{V} = \begin{bmatrix} \Xi & \Gamma \\ 0 & -\Theta^T \end{bmatrix}, \quad \mathcal{V}^T\mathcal{M}\mathcal{U} = \begin{bmatrix} \Theta & \Gamma^T \\ 0 & -\Xi^T \end{bmatrix}, \tag{3}$$

with block upper-triangular matrices

$$\Xi = \begin{array}{c} \begin{array}{ccc} n_1 & \cdots & n_l \end{array} \\ \left[\begin{array}{ccc} \Xi_{1,1} & \cdots & \Xi_{1,l} \\ & \ddots & \vdots \\ & & \Xi_{l,l} \end{array}\right] \end{array} \begin{array}{c} \\ n_1 \\ \vdots \\ n_l \end{array}, \quad \Theta = \begin{array}{c} \begin{array}{ccc} n_1 & \cdots & n_l \end{array} \\ \left[\begin{array}{ccc} \Theta_{1,1} & \cdots & \Theta_{1,l} \\ & \ddots & \vdots \\ & & \Theta_{l,l} \end{array}\right] \end{array} \begin{array}{c} \\ n_1 \\ \vdots \\ n_l \end{array}, \tag{4}$$

where the block-sizes satisfy $1 \le n_i \le 2$ and the diagonal blocks $\Xi_{i,i}$ are upper-triangular, $i = 1, \cdots, l$.

The method of [7] to compute this symplectic URV-decomposition consists of two parts. The first part is a Hessenberg-like reduction using orthogonal-symplectic transformation matrices $\mathcal{U}_0, \mathcal{V}_0 \in \mathbb{R}^{2n\times 2n}$ such that

$$\mathcal{U}_0^T\mathcal{M}\mathcal{V}_0 = \begin{bmatrix} M_{1,1} & M_{1,2} \\ 0 & -M_{2,2}^T \end{bmatrix}, \tag{5}$$

with $M_{1,1}$ upper-triangular and $M_{2,2}$ upper-Hessenberg. The second part is the computation of the periodic Schur form of $M_{1,1}M_{2,2}$, which determines (without forming the product) real orthogonal matrices U_1 and U_2 such that $U_1^T M_{1,1}(U_2U_2^T)M_{2,2}U_1 = \Xi\Theta$ as well as $U_2^T M_{2,2}(U_1U_1^T)M_{1,1}U_2 = \Theta\Xi$ are in real Schur form. Different versions of periodic Schur forms and corresponding computational methods were developed in [11, 19, 33, 36]. For new computational approaches to compute periodic Schur decompositions and its perturbation analysis, see [9, 23, 22].

The second drawback of the approach to solve the Hamiltonian eigenvalue problem via the skew-Hamiltonian eigenvalue problem for $\mathcal{N} = \mathcal{M}^2$ is that squaring the matrix maps the eigenvalues λ and $-\lambda$ both to λ^2. This makes it difficult to compute the invariant subspace associated with the eigenvalues in the left half plane, which is one of the major tasks in applications, see [29] and the references therein. A method to overcome the second drawback was suggested in [37, 38], but the method is not backward stable. An improvement here is the extension of [7] in [6] which also allows the computation of the invariant subspace of the Hamiltonian matrix $\mathcal{M}$ associated with the eigenvalues in the left half plane. The new method that we present below is strongly based on all these ideas.

In the symplectic URV-decomposition (3)-(4) the square roots of eigenvalues of $\Xi_{i,i}\Theta_{i,i}$ $(i = 1, \cdots, l)$ are eigenvalues of $\mathcal{M}$ and, because we may choose any order of diagonal blocks, we may order $\Xi_{i,i}$ and $\Theta_{i,i}$ in (3)-(4) such that the magnitude of the real parts of the square roots of the eigenvalues of $\Xi_{i,i}\Theta_{i,i}$ $(i = 1, \cdots, l)$ is decreasing.

Also, in the symplectic URV-decomposition (3)-(4) as it was derived in [7] it is not required that a 2×2-block $\Xi_{i,i}\Theta_{i,i}$ has only non-real eigenvalues. If 2×2 blocks with real eigenvalues occur, then we modify the URV-decomposition slightly by carefully computing an orthogonal matrix $\mathcal{U}_i \in \mathbb{R}^{2\times 2}$ such that $\mathcal{U}_i^T \Xi_{i,i}\Theta_{i,i}\mathcal{U}_i$ is upper-triangular and then updating sub-blocks $\Xi_{i,j}$, $\Theta_{k,i}$ $(j = i, \cdots, l,\ k = 1, \cdots, i)$ and the orthogonal-symplectic matrix $\mathcal{U}$ in (3)-(4).

In the following we assume that we have performed these two modifications, i.e., we assume that an orthogonal-symplectic matrix $\mathcal{U}$ as in (3)-(4) has been determined without forming $\mathcal{M}^2$ explicitly such that $\mathcal{U}^T\mathcal{M}^2\mathcal{U}$ is in real skew-Hamiltonian Schur form

$$\mathcal{U}^T\mathcal{M}^2\mathcal{U} = \begin{bmatrix} \Phi & \Pi \\ 0 & \Phi^T \end{bmatrix}, \tag{6}$$

with

$$\Phi = \begin{array}{c} \begin{array}{ccc} n_1 & \cdots & n_l \end{array} \\ \begin{bmatrix} \Phi_{1,1} & \cdots & \Phi_{1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix} \end{array} \begin{array}{c} \\ n_1 \\ \vdots \\ n_l \end{array}, \tag{7}$$

where $\Phi_{i,i} = \Xi_{i,i}\Theta_{i,i}$ is in real Schur form, i.e., either 2×2 with a pair of non-real complex conjugate eigenvalues, or 1×1 real, and the magnitude of the real parts of the square roots of the eigenvalues of $\Phi_{i,i}$ $(i = 1, \cdots, l)$ is decreasing.

Throughout the paper we will frequently make use of the following simple but important observations.

Proposition 4 *Suppose that $D \in \mathbb{R}^{2\times 2}$ has a pair of non-real conjugate eigenvalues. If $DX - XD = 0$ for some nonzero matrix $X \in \mathbb{R}^{2\times 2}$, then X is nonsingular.*

Proof. By our assumption, we may assume w.l.o.g. that $D = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ with $b \neq 0$. Then all solutions take the form $X = \begin{bmatrix} x & y \\ -y & x \end{bmatrix}$ and hence the result follows trivially. □

Proposition 5 *Consider a Hamiltonian matrix $\mathcal{M}$, let $\mathcal{U}$ be an orthogonal-symplectic matrix such that $\mathcal{U}^T\mathcal{M}^2\mathcal{U}$ is in real skew-Hamiltonian Schur form (6)–(7) and let $\mathcal{H} = \mathcal{U}^T\mathcal{M}\mathcal{U}$. Then the columns of $\mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}$ form an invariant subspace of $\mathcal{H}^2$ associated with the eigenvalues of $\Phi_{1,1}$.*

Proof. We have that $\mathcal{H}^2 = \mathcal{U}^T\mathcal{M}^2\mathcal{U} = (\mathcal{U}^T\mathcal{M}\mathcal{U})^2$ is in real skew-Hamiltonian Schur form (6)–(7) and hence

$$\mathcal{H}^2(\mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}) = \mathcal{H}(\mathcal{H}^2\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}) = \mathcal{H}(\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}\Phi_{1,1}) = (\mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix})\Phi_{1,1}. \tag{8}$$

□

By recalling some previous results on the skew-Hamiltonian Schur form we have now set the stage for the new method that we will describe in the next section.

3 Computation of the Hamiltonian Schur form

In this section we describe our new method to compute the Hamiltonian Schur form. Suppose that we have determined the symplectic URV-decomposition (3)–(4) of the Hamiltonian matrix $\mathcal{M}$, i.e., by construction $\mathcal{U}^T\mathcal{M}^2\mathcal{U}$

is in skew-Hamiltonian Schur form (6)-(7). We form the real Hamiltonian matrix

$$\begin{aligned}\mathcal{H} &= \mathcal{U}^T \mathcal{M} \mathcal{U} = \left[\begin{array}{c|c} F & G \\ \hline H & -F^T \end{array}\right] \\ &= \left[\begin{array}{ccc|ccc} F_{1,1} & \dots & F_{1,l} & G_{1,1} & \dots & G_{1,l} \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ F_{l,1} & \dots & F_{l,l} & G_{l,1} & \dots & G_{l,l} \\ \hline H_{1,1} & \dots & H_{1,l} & -F_{1,1}^T & \dots & -F_{l,1}^T \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ H_{l,1} & \dots & H_{l,l} & -F_{1,l}^T & \dots & -F_{l,l}^T \end{array}\right]\end{aligned} \tag{9}$$

with

$$F_{i,j}, G_{i,j}, H_{i,j} \in \mathbb{R}^{n_i \times n_j}, \quad i, j = 1, \cdots, l, \tag{10}$$

partitioned conformably with (6)–(7).

Our new method determines an orthogonal-symplectic matrix $\mathcal{Q}$ such that the Hamiltonian matrix $\mathcal{Q}^T \mathcal{H} \mathcal{Q}$ can be partitioned as

$$\mathcal{Q}^T \mathcal{H} \mathcal{Q} = \left[\begin{array}{cc|cc} \hat{F}_1 & * & * & * \\ 0 & \hat{F} & * & \hat{G} \\ \hline 0 & 0 & -\hat{F}_1^T & 0 \\ 0 & \hat{H} & * & -\hat{F}^T \end{array}\right], \tag{11}$$

where $\hat{F}_1$ is an $n_1 \times n_1$ or $2n_1 \times 2n_1$ matrix in real Hamiltonian Schur form, $\begin{bmatrix} \hat{F} & \hat{G} \\ \hat{H} & -\hat{F}^T \end{bmatrix}$ is Hamiltonian, and $\begin{bmatrix} \hat{F} & \hat{G} \\ \hat{H} & -\hat{F}^T \end{bmatrix}^2$ is again in real skew-Hamiltonian Schur form. Once this form has been computed, we can apply the same procedure inductively to the matrix $\begin{bmatrix} \hat{F} & \hat{G} \\ \hat{H} & -\hat{F}^T \end{bmatrix}$.

In this section we discuss only the theoretical basis for the new method, for the algorithm see Section 4. We study two cases depending on blocks of the first (block) column of (9) being zero or not. Observe that if all the blocks $F_{2,1}, \dots, F_{l,1}$ and $H_{1,1}, \dots, H_{l,1}$ in (9) are zero, then $\mathcal{H}$ is already in the form (11), since $\mathcal{H}$ is Hamiltonian, with $\hat{F}_1 \in \mathbb{R}^{n_1 \times n_1}$ and we have $\mathcal{Q} = I_{2n}$.

In the first case we assume that the blocks $H_{1,1}, \dots, H_{l,1}$ in (9) are all zero but at least one of the blocks $F_{2,1}, \dots, F_{l,1}$ does not vanish. We then have the following Lemma.

Lemma 6 *Consider a Hamiltonian matrix $\mathcal{H}$ of the form (9) such that $\mathcal{H}^2$ is in real skew-Hamiltonian Schur form and suppose that the blocks $H_{1,1}, \ldots, H_{l,1}$ are all zero but at least one of the blocks $F_{2,1}, \ldots, F_{l,1}$ does not vanish. If k is the largest index of a non-zero block in*

$$\begin{bmatrix} F_{2,1} \\ \vdots \\ F_{l,1} \end{bmatrix},$$

then $n_k = n_1$, the two matrices $\Phi_{1,1}$ and $\Phi_{k,k}$ in (6)–(7) have the same eigenvalues, and $F_{k,1} \in \mathbb{R}^{n_1 \times n_1}$ is invertible.

Proof. Since $\mathcal{H}^2 = \mathcal{U}^T \mathcal{M}^2 \mathcal{U} = (\mathcal{U}^T \mathcal{M} \mathcal{U})^2$ is in real skew-Hamiltonian Schur form (6)–(7), by Proposition 5 we obtain

$$\begin{bmatrix} \Phi_{1,1} & \cdots & \Phi_{1,k} \\ & \ddots & \vdots \\ & & \Phi_{k,k} \end{bmatrix} \begin{bmatrix} F_{1,1} \\ \vdots \\ F_{k,1} \end{bmatrix} = \begin{bmatrix} F_{1,1} \\ \vdots \\ F_{k,1} \end{bmatrix} \Phi_{1,1}.$$

So, the Sylvester equation

$$\Phi_{k,k} F_{k,1} - F_{k,1} \Phi_{1,1} = 0 \tag{12}$$

has to hold and we have the following possibilities.

If $n_1 = 2$ and $n_k = 1$ then, since the eigenvalues of $\Phi_{1,1}$ are non-real and $\Phi_{k,k}$ is real, we have a contradiction to $F_{k,1} \neq 0$, since (12) would then only have the solution $F_{k,1} = 0$. The same argument holds if $n_1 = 1$ and $n_k = 2$. So it follows that $n_1 = n_k$.

If $n_1 = 1$ then clearly by (12), $\Phi_{k,k} = \Phi_{1,1}$, since $F_{k,1} \neq 0$.

Analogously, if $n_1 = n_k = 2$ and the matrices $\Phi_{1,1}$ and $\Phi_{k,k}$ have no common eigenvalue, then (12) leads to a contradiction to $F_{k,1} \neq 0$. But since $\Phi_{1,1}$ and $\Phi_{k,k}$ have in this case only a pair of complex conjugate eigenvalues, we have that $\Phi_{1,1}$ and $\Phi_{k,k}$ are similar.

The invertibility of $F_{k,1}$ is clear if $n_1 = 1$ and follows from Proposition 4 if $n_1 = 2$. □

In the situation of Lemma 6 we introduce the $n \times n$ matrix

$$Y_1 = \left[\begin{array}{cccccccc} I_{n_1} & F_{1,1} & & & & & & \\ & F_{2,1} & I_{n_2} & & & & & \\ & \vdots & & \ddots & & & & \\ & F_{k-1,1} & & & I_{n_{k-1}} & & & \\ & F_{k,1} & & & 0 & 0 & & \\ & 0 & & & 0 & I_{n_{k+1}} & & \\ & \vdots & & & & & \ddots & \\ & 0 & & & & & & I_{n_l} \end{array}\right], \tag{13}$$

and determine the QR-factorization

$$Y_1 = Q_1 R_1, \tag{14}$$

where $Q_1 \in \mathbb{R}^{n\times n}$ and $R_1 \in \mathbb{R}^{n\times n}$ is upper-triangular and nonsingular, since $F_{k,1}$ is nonsingular.

Remark 1 The orthogonal matrix Q_1 in (14) can be obtained by

$$[(\sum_{i=2}^{k} n_i - 1) + (n_1 - 1)(\sum_{i=2}^{k} n_i - 2)] < 2n$$

Givens rotations, see Algorithm 2 in Appendix A.

Since $\mathcal{H}^2 \begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} = \begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} \Phi_{1,1}$, we have the identity

$$\mathcal{H}\left[\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}, \ \mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} \right] = \left[\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}, \ \mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} \right] \begin{bmatrix} 0 & \Phi_{1,1} \\ I_{n_1} & 0 \end{bmatrix}, \tag{15}$$

and it follows that

$$\mathcal{H}\left[\begin{array}{c} Q_1 \begin{bmatrix} I_{2n_1} \\ 0 \end{bmatrix} \\ \hline 0 \end{array}\right] = \left[\begin{array}{c} Q_1 \begin{bmatrix} I_{2n_1} \\ 0 \end{bmatrix} \\ \hline 0 \end{array}\right] \Delta, \tag{16}$$

where

$$\Delta = \left[\begin{array}{c} Q_1 \begin{bmatrix} I_{2n_1} \\ 0 \end{bmatrix} \\ \hline 0 \end{array}\right]^T \mathcal{M} \left[\begin{array}{c} Q_1 \begin{bmatrix} I_{2n_1} \\ 0 \end{bmatrix} \\ \hline 0 \end{array}\right]. \tag{17}$$

Finally we compute an orthogonal matrix $Q_2 \in \mathbb{R}^{2n_1 \times 2n_1}$ such that $Q_2^T \Delta Q_2$ is in real Schur form.

Remark 2 Note that if $n_1 = 1$, then $\Delta \in \mathbb{R}^{2\times 2}$ has a double real eigenvalue, since we have assumed that the problem has no purely imaginary eigenvalues. If $n_1 = 2$, then $\Delta \in \mathbb{R}^{4\times 4}$ has 2 double pairs of complex conjugate eigenvalues. In general Δ is not in real Schur form. So we may compute an orthogonal matrix $Q_2 \in \mathbb{R}^{2n_1\times 2n_1}$ such that $Q_2^T \Delta Q_2$ is in real Schur form, but due to the double eigenvalues, in general this computation is sensitive to perturbations. In those applications, where a block-Schur form is sufficient, we may just skip the transformation to real Schur from for this block and take Q_2 to be the identity matrix.

Theorem 7 *In the situation of Lemma 6, with the orthogonal-symplectic matrix*

$$\mathcal{Q} = \left[\begin{array}{c|c} Q_1 \begin{bmatrix} Q_2 & \\ & I \end{bmatrix} & 0 \\ \hline 0 & Q_1 \begin{bmatrix} Q_2 & \\ & I \end{bmatrix} \end{array}\right], \tag{18}$$

we have that $\mathcal{Q}^T \mathcal{H} \mathcal{Q}$ has the form (11) with $\hat{F}_1 \in \mathbb{R}^{2n_1\times 2n_1}$.

Proof. Since (16) holds, $\mathcal{H}$ is Hamiltonian and $\mathcal{Q}$ is orthogonal-symplectic, it follows that (11) holds with $\hat{F}_1 = Q_2^T \Delta Q_2$ in real Schur form.

By Lemma 6 we have that $F_{k,1}$ is nonsingular and $n_1 = n_k$, and hence with $t = n_1 + \ldots + n_{k-1}$ there exists a matrix $\mathcal{D} \in \mathbb{R}^{t\times n_1}$ such that

$$Y_1 \begin{bmatrix} I_{t+n_k} \\ 0 \end{bmatrix} \mathcal{D} = \begin{array}{cl} n_k & \\ \begin{bmatrix} 0 \\ I_{n_k} \\ 0 \end{bmatrix} & \begin{array}{l} t \\ n_k \\ n-t-n_k \end{array} \end{array},$$

where Y_1 is as in (13). Defining

$$\begin{array}{cl} \begin{array}{ccc} n_{k+1} & \cdots & n_l \end{array} & \\ \begin{bmatrix} \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ \tilde{\Phi}_{1,k+1} & \cdots & \tilde{\Phi}_{1,l} \\ \hat{\Phi}_{2,k+1} & \cdots & \hat{\Phi}_{2,l} \\ \vdots & \cdots & \vdots \\ \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \end{bmatrix} & \begin{array}{l} n_1 \\ n_1 \\ n_2 \\ \vdots \\ n_{k-1} \end{array} \end{array} := \begin{bmatrix} \Phi_{1,k+1} & \cdots & \Phi_{1,l} \\ 0 & \cdots & 0 \\ \Phi_{2,k+1} & \cdots & \Phi_{2,l} \\ \vdots & \cdots & \vdots \\ \Phi_{k-1,k+1} & \cdots & \Phi_{k-1,l} \end{bmatrix} + \mathcal{D} \begin{bmatrix} \Phi_{k,k+1} & \cdots & \Phi_{k,l} \end{bmatrix},$$

then, using (6)–(7) and introducing the matrices

$$Y_2 := \begin{bmatrix} \Phi_{1,1} & 0 & \Phi_{1,2} & \cdots & \Phi_{1,k-1} & \Phi_{1,k+1} & \cdots & \Phi_{1,l} \\ & \Phi_{1,1} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ & & \Phi_{2,2} & \cdots & \Phi_{2,k-1} & \Phi_{2,k+1} & \cdots & \Phi_{2,l} \\ & & & \ddots & \vdots & \vdots & \cdots & \vdots \\ & & & & \Phi_{k-1,k-1} & \Phi_{k-1,k+1} & \cdots & \Phi_{k-1,l} \\ & & & & & \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & & & & & & \ddots & \vdots \\ & & & & & & & \Phi_{l,l} \end{bmatrix},$$

and

$$Y_3 := \begin{bmatrix} \Phi_{1,1} & 0 & \Phi_{1,2} & \cdots & \Phi_{1,k-1} & \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ & \Phi_{1,1} & 0 & \cdots & 0 & \tilde{\Phi}_{1,k+1} & \cdots & \tilde{\Phi}_{1,l} \\ & & \Phi_{2,2} & \cdots & \Phi_{2,k-1} & \hat{\Phi}_{2,k+1} & \cdots & \hat{\Phi}_{2,l} \\ & & & \ddots & \vdots & \vdots & \cdots & \vdots \\ & & & & \Phi_{k-1,k-1} & \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \\ & & & & & \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & & & & & & \ddots & \vdots \\ & & & & & & & \Phi_{l,l} \end{bmatrix},$$

a direct calculation yields that

$$\begin{aligned} \mathcal{H}^2 \left[\begin{array}{c} Y_1 \\ \hline 0 \end{array}\right] &= \left[\begin{array}{c} Y_1 \\ \hline 0 \end{array}\right] Y_2 + \begin{bmatrix} 0 \\ I_{n_k} \\ 0 \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 & \Phi_{k,k+1} & \cdots & \Phi_{k,l} \end{bmatrix} \\ &= \left[\begin{array}{c} Y_1 \\ \hline 0 \end{array}\right] Y_2 + \left[\begin{array}{c} Y_1 \\ \hline 0 \end{array}\right] \begin{bmatrix} I_{t+n_k} \\ 0 \end{bmatrix} \mathcal{D} \begin{bmatrix} 0 & \cdots & 0 & \Phi_{k,k+1} & \cdots & \Phi_{k,l} \end{bmatrix} \\ &= \left[\begin{array}{c} Y_1 \\ \hline 0 \end{array}\right] Y_3. \end{aligned}$$

Hence,

$$\mathcal{H}^2 \left[\begin{array}{c} Q_1 \\ \hline 0 \end{array}\right] = \left[\begin{array}{c} Q_1 \\ \hline 0 \end{array}\right] R_1 Y_3 R_1^{-1}$$

Because R_1 is upper triangular and $Q^T \mathcal{H}^2 Q$ is skew-Hamiltonian, we have that

$$
\begin{array}{c} \\ \mathcal{Q}^T\mathcal{H}^2\mathcal{Q} = \end{array}
\begin{array}{c}
\begin{array}{cccc} 2n_1 & n-2n_1 & 2n_1 & n-2n_1 \end{array} \\
\left[\begin{array}{cccc}
\Psi_{1,1} & \Psi_{1,2} & \Psi_{13} & \Psi_{14} \\
0 & \Psi_{2,2} & \Psi_{23} & \Psi_{24} \\
0 & 0 & \Psi_{1,1}^T & 0 \\
0 & 0 & \Psi_{1,2}^T & \Psi_{2,2}^T
\end{array}\right]
\end{array},
$$

where

$$
\begin{bmatrix} \Psi_{1,1} & \Psi_{1,2} \\ 0 & \Psi_{2,2} \end{bmatrix} = \begin{bmatrix} Q_2^T & \\ & I \end{bmatrix} R_1 Y_3 R_1^{-1} \begin{bmatrix} Q_2 & \\ & I \end{bmatrix}.
$$

So, $\Psi_{2,2}$ is in real Schur form with diagonal blocks of sizes $n_2, \cdots, n_{k-1}$, $n_{k+1}, \cdots, n_l$, and the proof is complete. □

Our second case is that at least one of the blocks $H_{1,1}, \ldots, H_{l,1}$ in (9) does not vanish. In this case we have the following lemma.

Lemma 8 *Consider a Hamiltonian matrix $\mathcal{H}$ of the form (9) such that $\mathcal{H}^2$ is in real skew-Hamiltonian Schur form and suppose that at least one of the blocks $H_{1,1}, \ldots, H_{l,1}$ in (9) does not vanish. Then*

$$
\operatorname{rank} \begin{bmatrix} H_{1,1} \\ \vdots \\ H_{l,1} \end{bmatrix} = n_1. \tag{19}
$$

Proof. Since $\mathcal{H}^2 = \mathcal{U}^T\mathcal{M}^2\mathcal{U} = (\mathcal{U}^T\mathcal{M}\mathcal{U})^2$ is in real skew-Hamiltonian Schur form (6)–(7), by Proposition 5 we obtain

$$
\begin{bmatrix} \Phi_{1,1} & \cdots & \Phi_{1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix}^T \begin{bmatrix} H_{1,1} \\ \vdots \\ H_{l,1} \end{bmatrix} = \begin{bmatrix} H_{1,1} \\ \vdots \\ H_{k,1} \end{bmatrix} \Phi_{1,1}. \tag{20}
$$

Let k be the smallest index for which $H_{k,1} \neq 0$. Then (20) is reduced to

$$
\begin{bmatrix} \Phi_{k,k} & \cdots & \Phi_{k,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix}^T \begin{bmatrix} H_{k,1} \\ \vdots \\ H_{l,1} \end{bmatrix} = \begin{bmatrix} H_{k,1} \\ \vdots \\ H_{k,1} \end{bmatrix} \Phi_{1,1}.
$$

Thus, the Sylvester equation

$$
\Phi_{k,k}^T H_{k,1} - H_{k,1}\Phi_{1,1} = 0
$$

holds. The proof follows then analogous to the proof of the invertibility of $F_{k,1}$ in Lemma 6. □

Next, we perform a skinny QR-factorization

$$\left[\begin{array}{c} F_{2,1} \\ \vdots \\ F_{l,1} \\ \hline H_{1,1} \\ \vdots \\ H_{l,1} \end{array}\right] = P_1 S_1, \tag{21}$$

where P_1 is column-orthogonal and S_1 is upper triangular and nonsingular (because of (19)). Since (15) holds, an elementary calculation shows that

$$\mathcal{H} \left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right] = \left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right] \Sigma, \tag{22}$$

where

$$\Sigma = \left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right]^T \mathcal{H} \left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right] \tag{23}$$

has the same spectrum as $\left[\begin{array}{cc} 0 & \Phi_{1,1} \\ I_{n_1} & 0 \end{array}\right]$.

Since (22) means that the columns of $\left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right]$ form an invariant subspace of $\mathcal{H}$, and since $\mathcal{H}$ has no purely imaginary eigenvalues, it follows that Σ has no purely imaginary eigenvalues. By computing the real Schur decomposition of Σ we can determine an orthogonal matrix $P_2 \in \mathbb{R}^{2n_1 \times 2n_1}$ such that

$$P_2^T \Sigma P_2 = \left[\begin{array}{cc} \Sigma_{1,1} & \Sigma_{1,2} \\ 0 & \Sigma_{2,2} \end{array}\right] \tag{24}$$

is in real Schur form with the spectrum of $\Sigma_{1,1}$ being in the open left half plane.

Lemma 9 *Let*

$$\mathcal{W} := \left[\begin{array}{cc} I_{n_1} & 0 \\ 0 & P_1 \end{array}\right] P_2 \left[\begin{array}{c} I_{n_1} \\ 0 \end{array}\right] =: \left[\begin{array}{c} V_1 \\ \vdots \\ V_l \\ \hline W_1 \\ \vdots \\ W_l \end{array}\right] \tag{25}$$

be partitioned as in (6)–(7). Then

$$\mathcal{H}\mathcal{W} = \mathcal{W}\Sigma_{1,1}, \quad \mathcal{W}^T J_n \mathcal{W} = 0, \quad \text{and} \begin{bmatrix} 0 & I_n \end{bmatrix} \mathcal{W} \neq 0. \tag{26}$$

Furthermore, if $k > 0$ is the smallest index for which $W_k \neq 0$ in (25), then $n_k = n_1$ and $W_k \in \mathbb{R}^{n_1 \times n_1}$ is invertible.

Proof. The identity $\mathcal{H}\mathcal{W} = \mathcal{W}\Sigma_{1,1}$ follows directly from (22) and (24).

Because $\Sigma_{1,1}$ has only eigenvalues in the open left half plane and $\mathcal{H}$ is Hamiltonian, it follows that the columns of $\mathcal{W}$ span an isotropic subspace of $\mathcal{H}$, i.e., $\mathcal{W}^T J_n \mathcal{W} = 0$. In addition, since (19) holds, $\mathcal{H}\mathcal{W} = \mathcal{W}\Sigma_{1,1}$, $\mathcal{H}\begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} \neq \begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix} Z$ for any $Z \in \mathbb{R}^{n_1 \times n_1}$ by assumption, and

$$\mathcal{W} = \begin{bmatrix} I_{n_1} & 0 \\ 0 & F_{2,1} \\ \vdots & \vdots \\ 0 & F_{l,1} \\ 0 & H_{1,1} \\ \vdots & \vdots \\ 0 & H_{l,1} \end{bmatrix} \begin{bmatrix} I_{n_1} & 0 \\ 0 & S_1^{-1} \end{bmatrix} P_2 \begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix},$$

we must have that $\begin{bmatrix} 0 & I_n \end{bmatrix} \mathcal{W} \neq 0$.

Finally, we know that $\mathcal{H}\mathcal{W} = \mathcal{W}\Sigma_{1,1}$ and $\mathcal{H}^2 = \mathcal{U}^T \mathcal{M}^2 \mathcal{U}$ is of the form (6)–(7), thus $\mathcal{H}^2\mathcal{W} = \mathcal{W}\Sigma_{1,1}^2$ which gives that

$$\begin{bmatrix} \Phi_{k,k} & \cdots & \Phi_{k,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix}^T \begin{bmatrix} W_k \\ \vdots \\ W_l \end{bmatrix} = \begin{bmatrix} W_k \\ \vdots \\ W_l \end{bmatrix} \Sigma_{1,1}^2,$$

hence the second part follows analogous to the proof of Lemma 6. □

Lemma 10 *In the situation of Lemma 9, with k being the smallest index of a non-zero block W_i, let*

$$t = n_1 + \cdots + n_{k-1}$$

and

$$\hat{V} := \begin{bmatrix} V_{k+1} \\ \vdots \\ V_l \end{bmatrix}, \quad \hat{W} := \begin{bmatrix} W_{k+1} \\ \vdots \\ W_l \end{bmatrix}.$$

Then there exists an orthogonal matrix U_1 with partitioning

$$
U_1 =: \begin{array}{c} \\ \\ \end{array}\begin{array}{cc} n_k & n-t-n_k \\ \multicolumn{2}{c}{\begin{bmatrix} U_{1,1} & U_{1,2} \\ U_{2,1} & U_{2,2} \end{bmatrix}} \end{array} \begin{array}{l} \\ n_k \\ n-t-n_k \end{array}, \tag{27}
$$

such that $U_{1,1}$ and $U_{2,2}$ are nonsingular, $U_{2,2}$ is upper triangular, and furthermore,

$$
U_1^T \begin{bmatrix} W_k \\ \hat{W} \end{bmatrix} = \begin{bmatrix} \hat{W}_k \\ 0 \end{bmatrix}, \tag{28}
$$

where $\hat{W}_k \in \mathbb{R}^{n_k \times n_k}$ is nonsingular.

Proof. The proof is given constructively by Algorithm 3 in Appendix B. □

With $\hat{V}$, $\hat{W}$, t, $U_{1,2}$ and $U_{2,2}$ as in Lemma 10, introduce

$$
Y = \left[\begin{array}{ccccc} V_1 & I_{n_1} & & & \\ \vdots & & \ddots & & \\ V_{k-1} & & & I_{n_{k-1}} & \\ V_k & 0 & \cdots & 0 & U_{1,2} \\ \hat{V} & 0 & \cdots & 0 & U_{2,2} \\ \hline 0 & 0 & \cdots & 0 & 0 \\ W_k & 0 & \cdots & 0 & 0 \\ \hat{W} & 0 & \cdots & 0 & 0 \end{array}\right] = \left[\mathcal{W}, \begin{bmatrix} I_t \\ 0 \end{bmatrix}, \begin{bmatrix} 0_{t \times n_k} \\ U_{1,2} \\ U_{2,2} \\ 0 \end{bmatrix} \right]. \tag{29}
$$

It is obvious that Y has full column rank.

We perform the skinny QR-factorization

$$
Y = Q_1 R_1, \tag{30}
$$

where $Q_1 \in \mathbb{R}^{2n \times n}$ is column orthogonal and R_1 is upper triangular and nonsingular and then introduce

$$
\mathcal{Q} = \begin{bmatrix} Q_1 & -J_n Q_1 \end{bmatrix}. \tag{31}
$$

Remark 3 With t and U_1 as in Lemma 10 we have

$$
\begin{bmatrix} I_t & 0 & 0 & 0 \\ 0 & U_1^T & 0 & 0 \\ 0 & 0 & I_t & 0 \\ 0 & 0 & 0 & U_1^T \end{bmatrix} Y = \begin{bmatrix} Z_1 & I & 0 \\ Z_2 & 0 & 0 \\ Z_3 & 0 & I \\ 0 & 0 & 0 \\ \hat{W}_k & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{array}{l} t \\ n_k \\ n-n_k-t \\ t \\ n_k \\ n-n_k-t \end{array},
$$

where

$$\begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \end{bmatrix} = \begin{bmatrix} I_t & 0 & 0 \\ 0 & U_1^T & 0 \end{bmatrix} \mathcal{W} = \begin{bmatrix} I_t & 0 \\ 0 & U_1^T \end{bmatrix} \begin{bmatrix} V_1 \\ \vdots \\ V_l \end{bmatrix}.$$

Thus, it follows that the column orthogonal matrix Q_1 in (30) can be obtained by applying $[n_1(n+1+n_{k+1}+\cdots+n_l)-1] < 4n$ Givens rotations to $\mathcal{W}$ without computing the factorization (28), i.e., U_1 or the left hand side of (30) explicitly, see Algorithm 4 in Appendix C.

Theorem 11 *In the situation of Lemma 9, the matrix $\mathcal{Q}$ defined by (31) is orthogonal-symplectic and satisfies (11) with $\hat{F}_1 \in \mathbb{R}^{n_1 \times n_1}$.*

Proof. A simple calculation yields that the matrix Y defined in (29) satisfies $Y^T J_n Y = 0$ and thus $Q_1^T J_n Q_1 = 0$ and $\mathcal{Q} = \begin{bmatrix} Q_1 & -J_n Q_1 \end{bmatrix}$ is orthogonal-symplectic. The form (11) with $\hat{F}_1 \in \mathbb{R}^{n_1 \times n_1}$ follows directly from (26) and the fact that $\Sigma_{1,1}$ in (24) is in real Schur form.

Since $U_{2,2}$ is invertible, we may form the matrix

$$\begin{aligned} K &:= \Phi_{k,k} U_{1,2} + \begin{bmatrix} \Phi_{k,k+1} & \cdots & \Phi_{k,l} \end{bmatrix} U_{2,2} \\ &- U_{1,2} U_{2,2}^{-1} \begin{bmatrix} \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix} U_{2,2}. \end{aligned}$$

We have that W_k is invertible and (26) holds with $\Sigma_{1,1}$ having only eigenvalues in the open left half plane. Furthermore, $\mathcal{H}^2 \mathcal{W} = \mathcal{W} \Sigma_{1,1}^2$ and $\mathcal{H}^2$ is of the form (6)–(7). All these properties, together with

$$\begin{bmatrix} U_{1,2}^T & U_{2,2}^T \end{bmatrix} \begin{bmatrix} W_k \\ \hat{W} \end{bmatrix} = 0,$$

imply that $K = 0$. Note that

$$\begin{aligned} &\mathcal{H}^2 \begin{bmatrix} 0_{t \times n_k} \\ U_{1,2} \\ U_{2,2} \\ 0 \end{bmatrix} \\ = &\begin{bmatrix} I_t \\ 0 \end{bmatrix} \left(\begin{bmatrix} \Phi_{1,k} \\ \vdots \\ \Phi_{k-1,k} \end{bmatrix} U_{1,2} + \begin{bmatrix} \Phi_{1,k+1} & \cdots & \Phi_{1,l} \\ \vdots & \cdots & \vdots \\ \Phi_{k-1,k+1} & \cdots & \Phi_{k-1,l} \end{bmatrix} U_{2,2} \right) \end{aligned}$$

$$
+ \begin{bmatrix} 0_{t\times n_k} \\ U_{1,2} \\ U_{2,2} \\ 0 \end{bmatrix} U_{2,2}^{-1} \begin{bmatrix} \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix} U_{2,2} + \begin{bmatrix} 0_{t\times n_k} \\ I_{n_k} \\ 0 \end{bmatrix} K
$$

$$
= \begin{bmatrix} I_t \\ 0 \end{bmatrix} \left(\begin{bmatrix} \Phi_{1,k} \\ \vdots \\ \Phi_{k-1,k} \end{bmatrix} U_{1,2} + \begin{bmatrix} \Phi_{1,k+1} & \cdots & \Phi_{1,l} \\ \vdots & \cdots & \vdots \\ \Phi_{k-1,k+1)} & \cdots & \Phi_{k-1,l} \end{bmatrix} U_{2,2} \right)
$$

$$
+ \begin{bmatrix} 0_{t\times n_k} \\ U_{1,2} \\ U_{2,2} \\ 0 \end{bmatrix} U_{2,2}^{-1} \begin{bmatrix} \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix} U_{2,2}
$$

$$
= \begin{bmatrix} I_t & 0 \\ 0 & U_{1,2} \\ 0 & U_{2,2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ \vdots & \cdots & \vdots \\ \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \\ \hat{\Phi}_{k+1,k+1} & \cdots & \hat{\Phi}_{k+1,l} \\ & \ddots & \vdots \\ & & \hat{\Phi}_{l,l} \end{bmatrix},
$$

where

$$
\begin{bmatrix} \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ \vdots & \cdots & \vdots \\ \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \end{bmatrix} = \begin{bmatrix} \Phi_{1,k} \\ \vdots \\ \Phi_{k-1,k} \end{bmatrix} U_{1,2} + \begin{bmatrix} \Phi_{1,k+1} & \cdots & \Phi_{1,l} \\ \vdots & \cdots & \vdots \\ \Phi_{k-1,k+1} & \cdots & \Phi_{k-1,l} \end{bmatrix} U_{2,2},
$$

and

$$
\begin{bmatrix} \hat{\Phi}_{k+1,k+1} & \cdots & \hat{\Phi}_{k+1,l} \\ & \ddots & \vdots \\ & & \hat{\Phi}_{l,l} \end{bmatrix} = U_{2,2}^{-1} \begin{bmatrix} \Phi_{k+1,k+1} & \cdots & \Phi_{k+1,l} \\ & \ddots & \vdots \\ & & \Phi_{l,l} \end{bmatrix} U_{2,2},
$$

which is in real Schur form with diagonal blocks of sizes $n_i \times n_i$, $i = k+1, \cdots, l$ (because $U_{2,2}$ is upper triangular). Hence,

$$
\mathcal{H}^2 Y = Y \begin{bmatrix} \Sigma_{1,1} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ & \Phi_{1,1} & \cdots & \Phi_{1,k-1} & \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ & & \ddots & \vdots & \vdots & \cdots & \vdots \\ & & & \Phi_{k-1,k-1} & \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \\ & & & & \hat{\Phi}_{k+1,k+1} & \cdots & \hat{\Phi}_{k+1,l} \\ & & & & & \ddots & \vdots \\ & & & & & & \hat{\Phi}_{l,l} \end{bmatrix}.
$$

Thus, we have

$$\mathcal{H}^2 Q_1 = Q_1 \begin{bmatrix} \Psi_{1,1} & \Psi_{1,2} \\ 0 & \Psi_{2,2} \end{bmatrix},$$

and

$$\mathcal{Q}^T \mathcal{H}^2 \mathcal{Q} = \begin{bmatrix} \Psi_{1,1} & \Psi_{1,2} & \Psi_{1,3} & \Psi_{1,4} \\ 0 & \Psi_{2,2} & \Psi_{2,3} & \Psi_{2,4} \\ 0 & 0 & \Psi_{1,1}^T & 0 \\ 0 & 0 & \Psi_{1,2}^T & \Psi_{2,2}^T \end{bmatrix},$$

where

$$\begin{array}{cc} n_1 & n - n_1 \end{array}$$
$$\begin{bmatrix} \Psi_{1,1} & \Psi_{1,2} \\ 0 & \Psi_{2,2} \end{bmatrix} \begin{array}{l} n_1 \\ n - n_1 \end{array}$$

$$= R_1 \begin{bmatrix} \Sigma_{1,1} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ & \Phi_{1,1} & \cdots & \Phi_{1,k-1} & \hat{\Phi}_{1,k+1} & \cdots & \hat{\Phi}_{1,l} \\ & & \ddots & \vdots & \vdots & \cdots & \vdots \\ & & & \Phi_{k-1,k-1} & \hat{\Phi}_{k-1,k+1} & \cdots & \hat{\Phi}_{k-1,l} \\ & & & & \hat{\Phi}_{k+1,k+1} & \cdots & \hat{\Phi}_{k+1,l} \\ & & & & & \ddots & \vdots \\ & & & & & & \hat{\Phi}_{l,l} \end{bmatrix} R_1^{-1},$$

and $\Psi_{2,2}$ is in real Schur form with diagonal blocks of sizes $n_i \times n_i$, $i = 1, \cdots, k-1, k+1, \cdots, l$ (because R_1 is upper triangular). This finishes the proof. $\square$

After having deflated the first block column and having shown that the remaining matrix $\begin{bmatrix} \hat{F} & \hat{G} \\ \hat{H} & -\hat{F}^T \end{bmatrix}$ is again of the form that its square is in real skew-Hamiltonian Schur form, we can apply this approach inductively and obtain an algorithm for computing the Hamiltonian Schur form of a real Hamiltonian matrix $\mathcal{H}$ that has no purely imaginary eigenvalues.

The construction in this section was of a theoretical nature, and many details have to be considered when turning this into a numerical algorithm. Some of these are discussed in the following section.

4 Algorithms

In this section we describe some algorithmic details that are needed to implement the new algorithm for computing the Hamiltonian Schur form of a real Hamiltonian matrix $\mathcal{H}$ that has no purely imaginary eigenvalues.

For the following algorithm describing the inductive procedure of deflating blocks in the Hamiltonian Schur as in (11), at every step of the loop we have to decide whether the block column given by the blocks $H_{1,1}, \ldots, H_{l,1}$ in (9) are all zero and if this is case then we have to decide which is the largest index of a non-zero block among $F_{2,1}, \ldots, F_{l,1}$. To do this we first compute the norms of the blocks separately, i.e. we form $h_i = \|H_{i,1}\|_2$ $(i = 1, \ldots, l)$ and $f_i = \|F_{i,1}\|_2$ $(i = 2, \ldots, l)$ and then for a given tolerance Tol identify all blocks for which $h_i <$ Tol or $f_i <$ Tol as zero blocks.

With this decision procedure the framework of our new algorithm is the following.

Algorithm 1
Input: *Real Hamiltonian matrix* $\mathcal{M} \in \mathbb{R}^{2n\times 2n}$ *without purely imaginary eigenvalues and a deflation tolerance* Tol.
Output: *Orthogonal symplectic matrix* $\mathcal{U}$ *such that* $\mathcal{U}^T\mathcal{M}\mathcal{U}$ *is in real Hamiltonian Schur form.*
Step 1: *Use a slight modification of Algorithm 4.3 in [7] to determine orthogonal-symplectic matrices* $\mathcal{U}, \mathcal{V} \in \mathbb{R}^{2n\times 2n}$*, integers* n_i *with* $1 \le n_i \le 2$ $(i = 1, \cdots, l)$ *and a symplectic* URV*-decomposition (3)–(4) with the property that* $\mathcal{U}^T\mathcal{M}^2\mathcal{U}$ *is in real skew-Hamiltonian Schur form (6)–(7) with diagonal blocks that are* 1×1 *real or* 2×2 *with non-real complex-conjugate eigenvalues (and that the magnitude of the real parts of the square roots of the eigenvalues of* $\Phi_{i,i}$ $(i = 1, \cdots, l)$ *is decreasing if necessary).*

Set $m := n$, $\mathcal{S} := \left[\begin{array}{ccc} n_1 & \cdots & n_l \end{array}\right]$*, and compute (by directly updating* $\mathcal{H}$ *during the* URV*-decomposition) the Hamiltonian matrix* $\mathcal{H} := \mathcal{U}^T\mathcal{M}\mathcal{U}$.
Step 2:
WHILE $(m > 0)$ *DO .* form of $\mathcal{U}$

Partition $\mathcal{H}$ *as (9)–(10) and compute* $h_i = \|H_{i,1}\|_2$ $(i = 1, \ldots, l)$ *and* $f_i = \|F_{i,1}\|_2$ $(i = 2, \ldots, l)$*;*

IF (i) $\max\{\max_{i=1,\ldots,l} h_i, \max_{i=2,\ldots,l} f_i\} <$ *Tol THEN*

Set

$$\mathcal{H} := \left[\begin{array}{ccc|ccc} F_{2,2} & \cdots & F_{2,l} & G_{2,2} & \cdots & G_{2,l} \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ F_{l,2} & \cdots & F_{l,l} & G_{l,2} & \cdots & G_{l,l} \\ \hline H_{2,2} & \cdots & H_{2,l} & -F_{2,2}^T & \cdots & -F_{l,2}^T \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ H_{l,2} & \cdots & H_{l,l} & -F_{2,l}^T & \cdots & -F_{l,l}^T \end{array}\right],$$

$$m := m - \mathcal{S}(1), \quad \mathcal{S} := \left[\begin{array}{ccc} \mathcal{S}(2) & \cdots & \mathcal{S}(l) \end{array} \right], \quad l := l - 1.$$

ELSE IF (ii)($\max_{i=1,\ldots,l} h_i <$ *Tol AND* $\max_{i=2,\ldots,l} f_i \geq$ *Tol) THEN*

Determine the largest index k for $f_k \geq$ Tol;

Perform Algorithm 2 in Appendix A with $n_i = \mathcal{S}(i)$, $i = 1, \cdots, l$ and n being replaced by m to determine an orthogonal matrix Q_1 that generates the QR-factorization (14) and update

$$\mathcal{U} := \mathcal{U} \left[\begin{array}{cc} Q_1 & \\ & Q_1 \end{array} \right], \quad \mathcal{H} := \left[\begin{array}{cc} Q_1 & \\ & Q_1 \end{array} \right]^T \mathcal{H} \left[\begin{array}{cc} Q_1 & \\ & Q_1 \end{array} \right];$$

If a block-Hamiltonian Schur form is satisfactory then set Q_2 to be the identity of size $2\mathcal{S}(1)$ otherwise compute an orthogonal Q_2 such that $Q_2^T \mathcal{H}(1 : 2\mathcal{S}(1), 1 : 2\mathcal{S}(1))Q_2$ is in real Schur form, and form

$$\begin{aligned} \mathcal{U} &:= \mathcal{U} \left[\begin{array}{cccc} Q_2 & & & \\ & I & & \\ & & Q_2 & \\ & & & I \end{array} \right], \\ \mathcal{H} &= \left[\begin{array}{cccc} Q_2 & & & \\ & I & & \\ & & Q_2 & \\ & & & I \end{array} \right]^T \mathcal{H} \left[\begin{array}{cccc} Q_2 & & & \\ & I & & \\ & & Q_2 & \\ & & & I \end{array} \right]. \end{aligned}$$

To deflate the first block we then set $\nu = 2\mathcal{S}(1)$ and redefine

$$\mathcal{H} := \left[\begin{array}{cc} \mathcal{H}(\nu + 1 : m, \nu + 1 : m) & \mathcal{H}(\nu + 1 : m, m + \nu + 1 : 2m) \\ \mathcal{H}(m + \nu + 1 : 2m, \nu + 1 : m) & \mathcal{H}(m + \nu + 1 : 2m, m + \nu + 1 : 2m) \end{array} \right],$$

$$m := m - \nu, \ \mathcal{S} := \left[\begin{array}{cccccc} \mathcal{S}(2) & \cdots & \mathcal{S}(k-1) & \mathcal{S}(k+1) & \cdots & \mathcal{S}(l) \end{array} \right], \ l := l - 2.$$

ELSE IF (iii) ($\max_{i=1,\ldots,l} h_i \geq$ *Tol) THEN*

Compute the skinny QR-factorization (21);

Compute Σ by (23), compute an orthogonal matrix P_2 as in (24), and then compute $\mathcal{W}$ as in (25);

Determine the smallest index $k > 0$ for $\|W_k\|_2 \geq$ Tol and then perform Algorithm 4 in Appendix C with $n_i = \mathcal{S}(i)$, $i = 1, \cdots, l$, and n being replaced by m to determine a column orthogonal matrix Q_1;

Set $\nu = \mathcal{S}(1)$, $\mathcal{U} := \mathcal{U}\left[\begin{array}{cc} Q_1 & -J_m Q_1 \end{array}\right]$, *and*
$\mathcal{H} := \left[\begin{array}{cc} Q_1 & -J_m Q_1 \end{array}\right]^T \mathcal{H} \left[\begin{array}{cc} Q_1 & -J_m Q_1 \end{array}\right]$, *and redefine*

$$\mathcal{H} := \left[\begin{array}{cc} \mathcal{H}(\nu+1:m, \nu+1:m) & \mathcal{H}(\nu+1:m, m+\nu+1:2m) \\ \mathcal{H}(m+\nu+1:2m, \nu+1:m) & \mathcal{H}(m+\nu+1:2m, m+\nu+1:2m) \end{array}\right],$$

$m := m-\nu$, $\mathcal{S} = \left[\begin{array}{cccccc} \mathcal{S}(1) & \cdots & \mathcal{S}(k-1) & \mathcal{S}(k+1) & \cdots & \mathcal{S}(l) \end{array}\right]$, $l := l-1$.

END

END

To compute the stable invariant subspace of $\mathcal{M}$, i.e., the invariant subspace of $\mathcal{M}$ associated with the eigenvalues in the open left half plane, we only need to reorder the real Hamiltonian Schur form of $\mathcal{M}$ determined in Algorithm 1 using the algorithm of Byers [15, 16].

In order to identify this variation of Algorithm 1 we denote it in Tables 4, 5 and 6 of Section 5 by Algorithm 1^r.

Remark 4 The complexity analysis of Algorithm 1 yields the following results.

- The computation of the symplectic URV-decomposition including the reordering and the computation of $\mathcal{H} := \mathcal{U}^T \mathcal{M} \mathcal{U}$ in Step 1 need $\mathbf{O}(n^3)$ flops [7];
- In the inductive procedure of Step 2:
  - The computations in Case (i) need $\mathbf{O}(1)$ flops;
  - In Case (ii), since $m \leq n$, the orthogonal matrix Q_1 is computed by at most $2n$ Givens rotations (see Algorithm 2 in Appendix A), thus, all computations in Case (ii) can be carried out in $\mathbf{O}(n^2)$ flops;
  - In Case (iii), the skinny QR-factorization (21) can be computed with $\mathbf{O}(n)$ flops, and the matrices Σ in (23), P_2 in (24) and $\mathcal{W}$ in (25) can be computed with $\mathbf{O}(m^2)$ flops. The column orthogonal matrix Q_1 is computed by at most $4n$ Givens rotations (see Algorithm 4 in Appendix C). Thus , all computations in Case (iii) can be carried out with $\mathbf{O}(n^2)$ flops.

Note that $m \leq n$ and Step 2 consists of at most n inductive steps (because l in the symplectic URV-decomposition (3)–(4)) is at most n), hence the overall Step 2 needs $\mathbf{O}(n^3)$ flops.

Thus we conclude that the whole algorithm needs $\mathbf{O}(n^3)$ flops.

Remark 5 Since the positive and negative square roots of the eigenvalues of $\Xi_{i,i}\Theta_{i,i}$ $(i = 1, \cdots, l)$ in (3)–(4) are the eigenvalues of $\mathcal{M}$, so if the moduli of the real parts of the square roots of the eigenvalues of $\Xi_{i,i}\Theta_{i,i}$ $(i = 1, \cdots, l)$ are ordered decreasingly, in Algorithm 1, then the eigenvalues of $\mathcal{M}$ with larger moduli of the real parts are deflated first and those nearer to or on the imaginary axis are deflated in later steps. Then after deflating all eigenvalues of $\mathcal{M}$ with the modulus of the real parts larger than a given value, a reduced Hamiltonian matrix which usually is of small dimension is obtained. All eigenvalues of this reduced Hamiltonian matrix are near or on the imaginary axis. Hence, if it is necessary, we can apply some expensive method to this reduced Hamiltonian matrix to compute its Hamiltonian Schur form with desired accuracy.

Remark 6 The perturbation and error analysis of the new method is a combination of the analysis for the URV-decomposition, see [7], and the analysis for the deflation procedure in Step 2 of Algorithm 1. Since we only update the Hamiltonian matrix $\mathcal{H}$ with orthogonal-symplectic similarity transformations and since we can force the Hamiltonian structure easily, we obtain that the resulting matrix is the exact Hamiltonian Schur form of a nearby Hamiltonian matrix $\mathcal{H} + \Delta\mathcal{H}$ and hence the method is a strongly backward stable for computing the Hamiltonian Schur form.

A difficulty in the analysis, however, lies in the fact that a complete structured perturbation analysis for the Hamiltonian Schur form is not known if eigenvalues are on or very close to the imaginary axis, see [21] for the non-critical case and [12] for an extension to matrix pencils. The consequence of this difficulty is that is very hard to decide what the stable invariant subspace is. Thus despite the fact that we may have computed a very good and structured approximation to the Hamiltonian Schur form, the stable invariant subspace that we compute based on this Schur form may be highly corrupted or even wrong, see [17, 31, 32].

Due to this missing analysis, it is not clear, how much effect the small perturbations $\Delta\mathcal{H}$ that we have committed in the computation of the Schur form will have. But this is not a feature of our method but of the mathematical problem of computing the stable invariant subspace is.

A second difficulty lies in the decision process which leads to two different QR-factorizations in Step 2 of Algorithm 1. A detailed perturbation analysis of this decision process and what the best choice for the tolerance Tol is, has not been carried out so far. In our implementation we have tried different tolerance criteria and have used an absolute tolerance Tol $= 10^{-14}$ in Examples $1-17$, 19. We had to decrease this tolerance to Tol $= 10^{-11}$, however, in Example 18. The numerical results of Section 5 indicate that these tolerances form a good choice, but more analysis is needed here.

5 Numerical Results

In order to demonstrate the numerical properties of the new method we have implemented the new method in MATLAB version 6.5 [28] and compared this implementation with the MATLAB codes `haeig.m` and `hastab.m` which are implementations of the algorithms in [7] and [6], respectively and which are freely available from the HAPACK package [10, 3]. For a comparison of `haeig.m` and `hastab.m` with other Hamiltonian eigenvalue methods such as the Schur method in [25] and the deflating subspace method in [2, 34], see [6, 7]. In our implementation the reordering of the diagonal blocks in the symplectic URV-decomposition (3)–(4) was not considered and the eigenvalue reordering in the Hamiltonian Schur form was carried out using the MATLAB code `haschord.m` [10].

For comparison we have applied the new method to the problems 1-19 of the benchmark collection for continuous-time algebraic Riccati equations [4, 5]. Example 20 from the benchmark collection is missing, since it required more memory than available. All computations were performed on an IBM PC-Pentium-4 with IEEE standard double precision arithmetic and machine precision $\epsilon \approx 2.22 \times 10^{-16}$.

- Table 1 shows the (relative) residuals of the Hamiltonian Schur form $\mathcal{M} = \mathcal{U}^T \mathcal{M} \mathcal{U}$ computed by Algorithm 1. As measure we use

$$\text{Residual}_{\text{Schur}} := \frac{\|\mathcal{U}^T \mathcal{M} \mathcal{U} - \tilde{\mathcal{M}}\|_2}{\|\mathcal{M}\|_2},$$

  where $\tilde{\mathcal{M}}$ is the Hamiltonian Schur form of $\mathcal{M}$ computed by Algorithm 1 in finite precision arithmetic.

- Table 2 shows the maximal errors of the eigenvalues read from the Hamiltonian Schur form computed by Algorithm 1 compared with those computed by the MATLAB code `haeig.m` [10] with the exact eigenvalues of $\mathcal{M}$, respectively.

  The related measure that we use is defined by

$$\text{Residual}_{\text{eigenvalue}} := \frac{\max|\lambda - \lambda_{\text{exact}}|}{\|\mathcal{M}\|_2},$$

where $\lambda_{\rm exact}$ is a corresponding exact eigenvalue of $\mathcal{M}$. We list only those examples for which exact eigenvalues of $\mathcal{M}$ are available.

- Because the method in [7] computes the eigenvalues of $\mathcal{M}$ to full possible accuracy when no eigenvalues of $\mathcal{M}$ are close to the imaginary axis, Table 3 shows the maximal errors of the eigenvalues read from the Hamiltonian Schur form computed by Algorithm 1 with those given by `haeig.m`. As measure for the error we use

$$\text{Residual}_{\rm eigenvalue} := \frac{\max|\lambda - \lambda_{\rm haeig}|}{\|\mathcal{M}\|_2},$$

where $\lambda_{\rm haeig}$ denotes the eigenvalues of $\mathcal{M}$ computed by `haeig.m`. We list only those examples for which exact eigenvalues of $\mathcal{M}$ are not available.

- Table 4 shows the relative errors of the computed stabilizing solution of the algebraic Riccati equations associated with the Hamiltonian matrices. For

$$\mathcal{M} = \begin{bmatrix} F & -G \\ -H & -F^T \end{bmatrix}$$

this is the algebraic Riccati equation $0 = H + F^T X + XF - XGX$.

The solution of the algebraic Riccati equation is computed from the computed transformation matrix $\tilde{\mathcal{U}}_1 \in \mathbb{R}^{2n\times n}$ via $\tilde{X} = \tilde{\mathcal{U}}_1(n+1 : 2n, 1 : n)\tilde{\mathcal{U}}_1(1 : n, 1 : n)^{-1}$, where $\tilde{\mathcal{U}}_1$ was computed by Algorithm 1 with reordering [15, 16] and the columns of the computed $\tilde{\mathcal{U}}_1$ span the stable invariant subspace of the Hamiltonian matrix $\mathcal{M}$. We depict the relative error

$$\text{Rel} - \text{error}_{\rm solution} := \frac{\|\tilde{X} - X_{\rm exact}\|_2}{\|X_{\rm exact}\|_2},$$

where $X_{\rm exact}$ is the exact stabilizing solution of the corresponding algebraic Riccati equation. Note that we list only those examples for which the exact stabilizing solution is available.

- Table 5 shows the residuals for the algebraic Riccati equations, i.e.,

$$\text{Residual}_{\rm ARE} := \|Q + A^T\tilde{X} + \tilde{X}A\tilde{-}XG\tilde{X}\|_2,$$

- Since $\tilde{\mathcal{U}}_1$ is column orthogonal and its columns span an invariant subspace of $\mathcal{M}$, so in exact arithmetic we have

$$\mathcal{M}\tilde{\mathcal{U}}_1 = \tilde{\mathcal{U}}_1(\tilde{\mathcal{U}}_1^T\mathcal{M}\tilde{\mathcal{U}}_1).$$

In Table 6 we give the errors of the computed stable invariant subspaces, measured as

$$\mathrm{Residual}_{\mathrm{Subspace}} := \frac{\|\mathcal{M}\tilde{\mathcal{U}}_1 - \tilde{\mathcal{U}}_1(\tilde{\mathcal{U}}_1^T\mathcal{M}\tilde{\mathcal{U}}_1)\|_2}{\|\mathcal{M}\|_2}.$$

Table 1: Errors for the computed Hamiltonian Schur forms

Examples in [4]	Algorithm 1
1	7.4408×10^{-17}
2	5.6290×10^{-16}
3	5.1027×10^{-16}
4	8.3861×10^{-16}
5	1.3177×10^{-15}
6	1.5907×10^{-13}
$7(\epsilon = 1)$	1.6558×10^{-16}
$7(\epsilon = 10^{-6})$	4.1850×10^{-16}
$8(\epsilon = 1)$	1.9667×10^{-17}
$8(\epsilon = 10^{-8})$	1.1839×10^{-16}
$9(\epsilon = 1)$	3.3978×10^{-16}
$9(\epsilon = 10^{6})$	2.1403×10^{-16}
$9(\epsilon = 10^{-6})$	1.1102×10^{-16}
$10(\epsilon = 1)$	3.7063×10^{-16}
$10(\epsilon = 10^{-5})$	1.7520×10^{-16}
$10(\epsilon = 10^{-7})$	2.2993×10^{-16}
$11(\epsilon = 1)$	1.5535×10^{-16}
$11(\epsilon = 0)$	3.7743×10^{-9}
$12(\epsilon = 1)$	5.7923×10^{-16}
$12(\epsilon = 10^{6})$	3.5352×10^{-16}
$13(\epsilon = 1)$	6.4849×10^{-16}
$13(\epsilon = 10^{-6})$	1.2583×10^{-16}
$14(\epsilon = 1)$	4.8108×10^{-16}
$14(\epsilon = 10^{-6})$	1.1955×10^{-4}
$15(n = 39)$	2.3978×10^{-15}
$15(n = 119)$	7.2804×10^{-15}
$15(n = 199)$	8.4665×10^{-15}
$16(n = 8)$	1.2084×10^{-15}
$16(n = 64)$	4.2936×10^{-15}
$17(q = 1,\ r = 1,\ n = 21)$	4.1814×10^{-15}
$17(q = 100,\ r = 100,\ n = 21)$	6.7294×10^{-16}
$18(n = 100)$	5.1086×10^{-15}
19	4.6501×10^{-15}

Table 2: Maximal errors for the computed eigenvalues

Examples in [4]	Algorithm 1	`haeig`(M)
1	1.9511×10^{-16}	0
2	1.6620×10^{-16}	1.4166×10^{-16}
$7(\epsilon = 1)$	4.3928×10^{-16}	1.5067×10^{-16}
$7(\epsilon = 10^{-6})$	7.4910×10^{-16}	1.6750×10^{-16}
$9(\epsilon = 1)$	1.3723×10^{-16}	9.7037×10^{-17}
$9(\epsilon = 10^{6})$	7.5443×10^{-14}	1.6078×10^{-19}
$9(\epsilon = 10^{-6})$	5.6125×10^{-16}	8.2767×10^{-17}
$10(\epsilon = 1)$	4.4961×10^{-16}	5.0634×10^{-16}
$10(\epsilon = 10^{-5})$	1.7418×10^{-16}	2.8188×10^{-17}
$10(\epsilon = 10^{-7})$	1.8202×10^{-17}	1.7578×10^{-16}
$11(\epsilon = 1)$	1.2568×10^{-16}	5.3404×10^{-17}
$11(\epsilon = 0)$	3.0590×10^{-9}	2.6246×10^{-9}
$12(\epsilon = 1)$	9.4206×10^{-16}	2.8067×10^{-16}
$12(\epsilon = 10^{6})$	8.8696×10^{-16}	3.8336×10^{-16}
$16(n = 8)$	3.4323×10^{-15}	1.0376×10^{-15}
$16(n = 64)$	8.8078×10^{-15}	6.2549×10^{-15}

Table 3: Maximal errors for the computed eigenvalues

Examples in [4]	Algorithm 1
3	5.6325×10^{-16}
4	1.7224×10^{-15}
5	3.0522×10^{-15}
6	4.2463×10^{-15}
$8(\epsilon = 1)$	2.3678×10^{-19}
$8(\epsilon = 10^{-8})$	6.5125×10^{-15}
$13(\epsilon = 1)$	6.3992×10^{-16}
$13(\epsilon = 10^{-6})$	4.2133×10^{-17}
$14(\epsilon = 1)$	3.8169×10^{-16}
$14(\epsilon = 10^{-6})$	1.0813×10^{-15}
$15(n = 39)$	1.1462×10^{-15}
$15(n = 119)$	4.0021×10^{-15}
$15(n = 199)$	4.0995×10^{-15}
$17(q = 1,\ r = 1,\ n = 21)$	5.9603×10^{-15}
$17(q = 100,\ r = 100,\ n = 21)$	2.7613×10^{-15}
$18(n = 100)$	1.3842×10^{-14}
19	9.9952×10^{-15}

Table 4: Relative errors for the computed solutions of the algebraic Riccati equations, for Example 17: $|X(1,n) - \sqrt{qr}|/\sqrt{qr}$

Examples in [4]	Algorithm 1^r	`hastab`(M)
1	0	3.3165×10^{-16}
2	1.9252×10^{-15}	1.6434×10^{-15}
$7(\epsilon = 1)$	4.9151×10^{-16}	$*$
$7(\epsilon = 10^{-6})$	7.2106×10^{-5}	$*$
$9(\epsilon = 1)$	4.6602×10^{-16}	1.9585×10^{-15}
$9(\epsilon = 10^{6})$	8.4608×10^{-11}	7.3535×10^{-11}
$9(\epsilon = 10^{-6})$	8.2399×10^{-16}	7.2842×10^{-11}
$10(\epsilon = 1)$	5.9204×10^{-16}	2.6715×10^{-15}
$10(\epsilon = 10^{-5})$	5.2781×10^{-16}	3.2400×10^{-6}
$10(\epsilon = 10^{-7})$	1.9771×10^{-16}	1.3392×10^{-1}
$11(\epsilon = 1)$	3.8688×10^{-16}	3.6493×10^{-16}
$11(\epsilon = 0)$	1.2757×10^{-8}	2.8584×10^{-9}
$12(\epsilon = 1)$	4.4747×10^{-16}	1.1958×10^{-14}
$12(\epsilon = 10^{6})$	5.7378×10^{-4}	4.0628×10^{0}
$16(n = 8)$	5.6889×10^{-16}	8.4747×10^{-16}
$16(n = 64)$	3.1619×10^{-15}	3.8677×10^{-14}
$17(q = 1,\ r = 1,\ n = 21)$	3.3776×10^{-7}	1.1969×10^{-7}
$17(q = 100,\ r = 100,\ n = 21)$	1.6418×10^{-4}	1.4917×10^{-5}

Table 5: Residuals for the compared algebraic Riccati equations

Examples in [4]	Algorithm 1^r	`hastab`(M)
1	0	4.3179×10^{-15}
2	1.3496×10^{-13}	2.3185×10^{-13}
3	4.8014×10^{-14}	1.3303×10^{-13}
4	9.0788×10^{-15}	2.8576×10^{-14}
5	1.0291×10^{-12}	9.2309×10^{-14}
6	4.2666×10^{-4}	2.0973×10^{-5}
$7(\epsilon = 1)$	3.8500×10^{-15}	$*$
$7(\epsilon = 10^{-6})$	2.8840×10^{8}	$*$
$8(\epsilon = 1)$	2.5793×10^{-11}	9.3096×10^{-10}
$8(\epsilon = 10^{-8})$	4.6067×10^{-6}	5.2482×10^{-4}
$9(\epsilon = 1)$	2.8335×10^{-15}	4.2428×10^{-15}
$9(\epsilon = 10^{6})$	5.9306×10^{-5}	1.5117×10^{-4}
$9(\epsilon = 10^{-6})$	1.7258×10^{-10}	3.5607×10^{-10}
$10(\epsilon = 1)$	2.1893×10^{-14}	1.0742×10^{-13}
$10(\epsilon = 10^{-5})$	8.1029×10^{-15}	5.1841×10^{-5}
$10(\epsilon = 10^{-7})$	1.7764×10^{-15}	1.8558×10^{0}
$11(\epsilon = 1)$	3.3845×10^{-15}	2.0324×10^{-15}
$11(\epsilon = 0)$	1.7764×10^{-15}	2.5121×10^{-15}
$12(\epsilon = 1)$	1.4572×10^{-14}	4.4648×10^{-13}
$12(\epsilon = 10^{6})$	1.8770×10^{16}	2.4218×10^{19}
$13(\epsilon = 1)$	1.7186×10^{-14}	3.3088×10^{-14}
$13(\epsilon = 10^{-6})$	5.7851×10^{-9}	1.5912×10^{-4}
$14(\epsilon = 1)$	3.2692×10^{-14}	2.2859×10^{-14}
$14(\epsilon = 10^{-6})$	9.8739×10^{-15}	4.2958×10^{-15}
$15(n = 39)$	2.0548×10^{-13}	1.7791×10^{-13}
$15(n = 119)$	1.1381×10^{-12}	5.5183×10^{-13}
$15(n = 199)$	4.3443×10^{-12}	1.0243×10^{-12}
$16(n = 8)$	2.0003×10^{-15}	2.2298×10^{-15}
$16(n = 64)$	1.5552×10^{-14}	1.0248×10^{-13}
$17(q = 1,\ r = 1,\ n = 21)$	9.4512×10^{2}	2.5232×10^{2}
$17(q = 100,\ r = 100,\ n = 21)$	3.3820×10^{7}	5.6661×10^{6}
$18(n = 100)$	6.2213×10^{-12}	2.8868×10^{-15}
19	2.6634×10^{-12}	1.0219×10^{-12}

Table 6: Errors for the computed stable invariant subspaces

Examples in [4]	Algorithm 1^r	`hastab`(M)
1	3.5381×10^{-16}	1.8680×10^{-16}
2	2.0848×10^{-16}	7.8597×10^{-17}
3	1.5629×10^{-15}	5.5160×10^{-15}
4	1.7530×10^{-15}	3.6437×10^{-15}
5	1.8743×10^{-15}	3.6329×10^{-16}
6	1.4660×10^{-14}	2.4606×10^{-16}
$7(\epsilon = 1)$	9.0249×10^{-16}	1.6846×10^{-16}
$7(\epsilon = 10^{-6})$	1.3748×10^{-15}	1.7981×10^{-16}
$8(\epsilon = 1)$	5.3455×10^{-18}	1.0213×10^{-17}
$8(\epsilon = 10^{-8})$	6.4915×10^{-17}	1.0937×10^{-16}
$9(\epsilon = 1)$	2.2295×10^{-16}	8.8174×10^{-16}
$9(\epsilon = 10^{6})$	1.8807×10^{-16}	8.5549×10^{-17}
$9(\epsilon = 10^{-6})$	2.4825×10^{-16}	2.6388×10^{-16}
$10(\epsilon = 1)$	1.8744×10^{-16}	1.0298×10^{-15}
$10(\epsilon = 10^{-5})$	2.6096×10^{-16}	1.1905×10^{-6}
$10(\epsilon = 10^{-7})$	1.5632×10^{-16}	5.5723×10^{-2}
$11(\epsilon = 1)$	1.2556×10^{-16}	1.2079×10^{-16}
$11(\epsilon = 0)$	9.8072×10^{-17}	5.2525×10^{-17}
$12(\epsilon = 1)$	8.4036×10^{-16}	4.3272×10^{-15}
$12(\epsilon = 10^{6})$	8.7429×10^{-16}	2.4336×10^{-4}
$13(\epsilon = 1)$	9.6134×10^{-16}	3.1353×10^{-15}
$13(\epsilon = 10^{-6})$	1.3206×10^{-21}	1.8338×10^{-17}
$14(\epsilon = 1)$	7.3677×10^{-16}	4.0682×10^{-16}
$14(\epsilon = 10^{-6})$	1.5438×10^{-15}	5.8296×10^{-16}
$15(n = 39)$	1.6189×10^{-15}	9.4700×10^{-16}
$15(n = 119)$	4.6274×10^{-15}	1.2624×10^{-15}
$15(n = 199)$	3.3113×10^{-15}	1.5971×10^{-15}
$16(n = 8)$	2.3503×10^{-15}	6.5223×10^{-16}
$16(n = 64)$	5.4938×10^{-15}	1.8782×10^{-14}
$17(q = 1,\ r = 1,\ n = 21)$	5.3092×10^{-15}	4.8028×10^{-15}
$17(q = 100,\ r = 100,\ n = 21)$	6.1542×10^{-16}	2.5168×10^{-16}
$18(n = 100)$	7.5253×10^{-15}	1.6386×10^{-15}
19	3.7382×10^{-15}	1.9190×10^{-14}

Since the benchmark collection [5] contains some really ill-conditioned and difficult examples we comment on some of the results in these tables:

- Example 7 with $\epsilon = 1$ or $\epsilon = 10^{-6}$: In Tables 4 and 5, $\tilde{\mathcal{U}}_1(1:n, 1:n)$ computed by `hastab.m` is singular and so the solution X cannot be obtained. However, one sees from Tables 6 and 2 that the computed subspace by `hastab.m` is still a good approximation to the stable invariant subspace of $\mathcal{M}$.

- Example 11 with $\epsilon = 0$: In this example, the eigenvalues of $\mathcal{M}$ are $\pm i$, i.e., the theoretical basis for our method is not given. However, Table 6 indicates that the columns of $\tilde{\mathcal{U}}_1$ computed by Algorithm 1^r, span a very good approximation to the stable invariant subspace of $\mathcal{M}$. We see from

  $$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 2.2260 \times 10^{-8}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 3.7743 \times 10^{-9},$$

  that the symplecticity of the matrix $\left[\begin{array}{cc} \tilde{\mathcal{U}}_1 & -J_n\tilde{\mathcal{U}}_1 \end{array}\right]$ is perturbed. Perturbations of the same order arise when using the method in [6]. The code `hastab.m` yields errors

  $$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 4.9889 \times 10^{-9}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 8.4591 \times 10^{-10}.$$

  In this example, with

  $$\left[\begin{array}{cc} \tilde{\mathcal{U}}_1 & -J_n\tilde{\mathcal{U}}_1 \end{array}\right]^{-1} \mathcal{M} \left[\begin{array}{cc} \tilde{\mathcal{U}}_1 & -J_n\tilde{\mathcal{U}}_1 \end{array}\right] =: \begin{array}{c} \begin{array}{cc} n & n \end{array} \\ \left[\begin{array}{cc} \Psi_{1,1} & \Psi_{1,2} \\ \Psi_{21} & \Psi_{2,2} \end{array}\right] \end{array} \begin{array}{c} \\ n \\ n \end{array}, \tag{32}$$

  we obtain

  $$\frac{\|\Psi_{21}\|_2}{\|M\|_2} = \begin{cases} 7.9385 \times 10^{-17}, & \text{if } \mathcal{U} \text{ is computed by Algorithm } 1^{\mathrm{r}}, \\ 4.2683 \times 10^{-17}, & \text{if } \mathcal{U} \text{ is computed by } \texttt{hastab.m}. \end{cases}$$

- Example 14 with $\epsilon = 10^{-6}$:

  In this example, $\mathcal{M}$ has 4 eigenvalues very close to the imaginary axis, i.e., the problem is very ill-conditioned. The transformation matrix looses its symplecticity, we get

  $$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 5.0642 \times 10^{-4}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 1.1955 \times 10^{-4}.$$

Nevertheless, Table 6 indicates that the columns of $\tilde{\mathcal{U}}_1$ computed by Algorithm 1^r span a very good approximation to the stable invariant subspace of $\mathcal{M}$. The same problem arises with the method in [6], where we get

$$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 1.9486 \times 10^{-4}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 4.5999 \times 10^{-5}.$$

If we again use (32), then we have

$$\frac{\|\Psi_{21}\|_2}{\|\mathcal{M}\|_2} = \begin{cases} 1.1600 \times 10^{-15}, & \text{if } \mathcal{U} \text{ is computed by Algorithm } 1^{\mathrm{r}}, \\ 4.9252 \times 10^{-16}, & \text{if } \mathcal{U} \text{ is computed by } \texttt{hastab.m}. \end{cases}$$

We also applied the MATLAB code `are.m` to these two examples, which is a MATLAB implementation of the Schur vector method in [25]. Denote by $\tilde{X}$ the computed stabilizing solution of the related algebraic Riccati equation, and determine $\tilde{\mathcal{U}}_1$ as an orthogonal basis of the subspace spanned by $\begin{bmatrix} I \\ \tilde{X} \end{bmatrix}$, which is supposed to be associated with the eigenvalues of $\mathcal{M}$ in the left half plane, then in Example 11 with $\epsilon = 0$ `are.m` yields

$$\frac{|\tilde{X}(2,1) - \tilde{X}(1,2)|}{\|\tilde{X}\|_2} = 1.0924 \times 10^{-8},$$

$$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 2.4958 \times 10^{-8}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 4.2318 \times 10^{-9}.$$

For the computed solution of the algebraic Riccati equation in Example 14 with $\epsilon = 10^{-6}$ `are.m` yields

$$\frac{\|\tilde{X}(1:2,3:4) - \tilde{X}(3:4,1:2)\|_2}{\|\tilde{X}\|_2} = 1.9992 \times 10^{-6},$$

$$\|\tilde{\mathcal{U}}_1^T J_n \tilde{\mathcal{U}}_1\|_2 = 7.9163 \times 10^{-4}, \quad \frac{\|\tilde{\mathcal{U}}_1^T J_n \mathcal{M} \tilde{\mathcal{U}}_1\|_2}{\|\mathcal{M}\|_2} = 1.8688 \times 10^{-4}.$$

So, the computed solutions $\tilde{X}$ are not symmetric and the associated orthogonal basis for the stable invariant subspaces in Example 11 with $\epsilon = 0$ and Example 14 with $\epsilon = 10^{-6}$ is not any longer isotropic.

6 Conclusions

In this paper we have developed a new method for computing the Hamiltonian Schur form and the stable invariant subspace of real Hamiltonian matrices without purely imaginary eigenvalues. The new method is numerically strongly backward stable, it preserves the Hamiltonian structure, and has complexity $\mathbf{O}(n^3)$. The numerical performance of the new method has been demonstrated using the problems of the benchmark collection for the continuous-time algebraic Riccati equations [5].

Even though much numerical detail has to be taken care of until a production code implementation of this method is available we may say at this stage: **Van Loan's curse is lifted!**

References

[1] G. S. Ammar and V. Mehrmann. On Hamiltonian and symplectic Hessenberg forms. *Linear Algebra Appl.*, 149:55–72, 1991.

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

[3] P. Benner and D. Kressner. Fortran 77 subroutines for computing the eigenvalues of Hamiltonian matrices II. Submitted. See also `http://www.math.tu-berlin.de/~kressner/hapack/`, 2004.

[4] P. Benner, A. Laub, and V. Mehrmann. A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case. Technical Report SPC 95_22, Fak. f. Mathematik, TU Chemnitz–Zwickau, 09107 Chemnitz, FRG, 1995. Available as `SPC95_22.ps` via anonymous ftp from `ftp.tu-chemnitz.de`, directory `/pub/Local/mathematik/Benner`.

[5] P. Benner, A.J. Laub, and V. Mehrmann. Benchmarks for the numerical solution of algebraic Riccati equations. *IEEE Contr. Syst. Mag.*, 7:18–28, 1997.

[6] P. Benner, V. Mehrmann, and H. Xu. A new method for computing the stable invariant subspace of a real Hamiltonian matrix. *J. Comput. Appl. Math.*, 86:17–43, 1997.

[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(3):329–358, 1998.

[8] P. Benner, V. Mehrmann, and H. Xu. A note on the numerical solution of complex Hamiltonian and skew-Hamiltonian eigenvalue problems. *Electr. Trans. Num. Anal.*, 8:115–126, 1999.

[9] P. Benner, V. Mehrmann, and H. Xu. Perturbation analysis for the eigenvalue problem of a formal product of matrices. *BIT*, 42(1):1–43, 2002.

[10] P. Benner and D. Kreßner. HAPACK - software for (skew-)Hamiltonian eigenvalue problems, 2003.

[11] A. Bojanczyk, G.H. Golub, and P. Van Dooren. The periodic Schur decomposition; algorithms and applications. In *Proc. SPIE Conference, vol. 1770*, pages 31–42, 1992.

[12] S. Bora and V. Mehrmann. Perturbation theory for structured matrix pencils arising in control theory. Preprint 16–2004, Institut für Mathematik, TU Berlin, D-10623 Berlin, Germany, 2004. Submitted for publications.

[13] J.R. Bunch. The weak and strong stability of algorithms in numerical algebra. *Linear Algebra Appl.*, 88:49–66, 1987.

[14] A. Bunse-Gerstner, R. Byers, and V. Mehrmann. Numerical methods for algebraic Riccati equations. In S. Bittanti, editor, *Proc. Workshop on the Riccati Equation in Control, Systems, and Signals*, pages 107–116, Como, Italy, 1989.

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

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

[17] G. Freiling, V. Mehrmann, and H. Xu. Existence, uniqueness and parametrization of Lagrangian invariant subspaces. *SIAM J. Matrix Anal. Appl.*, 23:1045–1069, 2002.

[18] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, third edition, 1996.

[19] J.J. Hench and A.J. Laub. Numerical solution of the discrete-time periodic Riccati equation. *IEEE Trans. Automat. Control*, 39:1197–1210, 1994.

[20] H. Abou Kandil, G. Freiling, V. Ionescu, and G. Jank. *Matrix Riccati Equations in Control and Systems Theory.* Birkhäuser, Verlag Basel, 2003.

[21] M.M. Konstantinov, V. Mehrmann, and P.Hr. Petkov. Perturbation analysis for the Hamiltonian Schur form. *SIAM J. Matrix Anal. Appl.*, 23:387–424, 2002.

[22] D. Kreßner. *Numerical Methods and Software for General and Structured Eigenvalue Problems.* PhD thesis, TU Berlin, Institut für Mathematik, Berlin, Germany, 2004.

[23] D. Kreßner. The periodic QR algorithm is a disguised QR algorithm, 2004. To appear in *Linear Algebra Appl.*

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

[25] A.J. Laub. A Schur method for solving algebraic Riccati equations. *IEEE Trans. Automat. Control*, AC-24:913–921, 1979. (See also *Proc. 1978 CDC (Jan. 1979)*, pp. 60-65).

[26] A.J. Laub. Invariant subspace methods for the numerical solution of Riccati equations. In S. Bittanti, A.J. Laub, and J.C. Willems, editors, *The Riccati Equation*, pages 163–196. Springer-Verlag, Berlin, 1991.

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

[28] The MathWorks, Inc., Cochituate Place, 24 Prime Park Way, Natick, Mass, 01760. *MATLAB Version 6.5*, 2002.

[29] V. Mehrmann. *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution.* Number 163 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Heidelberg, July 1991.

[30] C.C. Paige and C.F. Van Loan. A Schur decomposition for Hamiltonian matrices. *Linear Algebra Appl.*, 14:11–32, 1981.

[31] A.C.M. Ran and L. Rodman. Stability of invariant Lagrangian subspaces I. In I. Gohberg, editor, *Operator Theory: Advances and Applications*, volume 32, pages 181–218. Birkhäuser-Verlag, Basel, Switzerland, 1988.

[32] A.C.M. Ran and L. Rodman. Stability of invariant Lagrangian subspaces II. In H.Dym, S. Goldberg, M.A. Kaashoek, and P. Lancaster, editors, *Operator Theory: Advances and Applications*, volume 40, pages 391–425. Birkhäuser-Verlag, Basel, Switzerland, 1989.

[33] J. Sreedhar and P. Van Dooren. Periodic Schur forms and some matrix equations. *Systems and Networks: Mathematical Theory and Applications*, 77:339–362, 1994.

[34] P. Van Dooren. A generalized eigenvalue approach for solving Riccati equations. *SIAM J. Sci. Statist. Comput.*, 2:121–135, 1981.

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

[36] C.F. Van Loan. Computing the CS and the generalized singular value decomposition. *Numer. Math.*, 46:479–491, 1985.

[37] H. Xu. *Solving Algebraic Riccati Equations via Skew-Hamiltonian Matrices*. PhD thesis, Inst. of Math., Fudan University, Shanghai, P.R. China, April 1993.

[38] H. Xu and L. Lu. Properties of a quadratic matrix equation and the solution of the continuous-time algebraic Riccati equation. *Linear Algebra Appl.*, 222:127–146, 1995.

[39] K. Zhou, J.C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice-Hall, Upper Saddle River, NJ, 1995.

Appendix A

The orthogonal matrix Q_1 in (14) can be computed by the following Algorithm 2, which consists of $[(\sum_{i=2}^{k} n_i - 1) + (n_1 - 1)(\sum_{i=2}^{k} n_i - 2)] < 2n$ Givens rotations:

Algorithm 2

Input: $X := \begin{bmatrix} F_{2,1} \\ \vdots \\ F_{k,1} \end{bmatrix}.$

Output: *Orthogonal matrix* Q_1 *in (14).*

Step 1: *Set* $\mu := n_2 + \cdots + n_k$ *and* $Q_1 = I_\mu$.

Step 2:

FOR $j = 1 : n_1$,

FOR $i = 1 + j : \mu$,

Compute a Givens rotation such that

$$G^T X(\mu + j - i : \mu + 1 + j - i, j) = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}.$$

IF $(j = 1 \ \& \ n_1 = 2)$ *THEN set*

$$X(\mu + j - i : \mu + 1 + j - i, 2) \\ := G^T X(\mu + j - i : \mu + 1 + j - i, 2).$$

END

$$X(\mu + j - i : \mu + 1 + j - i, j) := \begin{bmatrix} \gamma \\ 0 \end{bmatrix},$$

$$Q_1(:, \mu + j - i : \mu + 1 + j - i) \\ := Q_1(:, \mu + j - i : \mu + 1 + j - i)G.$$

END

END

Step 3: *Set*

$$Q_1 := \begin{bmatrix} I_{n_1} & & \\ & Q_1 & \\ & & I \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

Appendix B

We prove Lemma 10 constructively by the following algorithm:

Algorithm 3

Input: $X := \begin{bmatrix} W_{k,1} \\ \vdots \\ W_{l,1} \end{bmatrix}.$

Output: *Orthogonal matrix* U_1 *as in Lemma 10.*
Step 1: *Set* $\mu := n_k + \cdots + n_l$ *and* $U_1 = I_\mu$.
Step 2:

FOR $j = 1 : n_1$,
FOR $i = j+1 : \mu$,
Compute a Givens rotation such that

$$G^T \begin{bmatrix} X(j,j) \\ X(i,j) \end{bmatrix} = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}.$$

IF ($j = 1$ & $n_1 = 2$) *THEN set*

$$\begin{aligned} X(j,2) &:= \begin{bmatrix} 1 & 0 \end{bmatrix} G^T \begin{bmatrix} X(j,2) \\ X(i,2) \end{bmatrix}, \\ X(i,2) &:= \begin{bmatrix} 0 & 1 \end{bmatrix} G^T \begin{bmatrix} X(j,2) \\ X(i,2) \end{bmatrix}, \end{aligned}$$

END
Set

$$\begin{aligned} X(j,j) &:= \gamma, \quad X(i,j) := 0, \\ U_1(:,j) &:= \begin{bmatrix} U_1(:,j) & U_1(:,i) \end{bmatrix} G \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ U_1(:,i) &:= \begin{bmatrix} U_1(:,j) & U_1(:,i) \end{bmatrix} G \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{aligned}$$

END
END

If U_1 computed by Algorithm 3 is partitioned as in (27), then a simple calculation gives that U_1 yields the QR-factorization (28) and $U_{2,2}$ is upper triangular. Furthermore, since W_k and $\hat{W}_k$ are nonsingular, and

$$W_k = U_{1,1}\hat{W}_k,$$

so $U_{1,1}$ is nonsingular. Consequently, we have by the generalized CS-decomposition [18] of the orthogonal matrix U_1 that $U_{2,2}$ is also nonsingular. □

Appendix C

The column orthogonal matrix Q_1 in (30) can be computed by the following algorithm which consists of $[n_1(n+1+n_{k+1}+\cdots n_l)-1] < 4n$ Givens rotations:

Algorithm 4

Input: $X := \mathcal{W} = \begin{bmatrix} V_1 \\ \vdots \\ V_l \\ 0 \\ W_k \\ \vdots \\ W_l \end{bmatrix}.$

Output: *Column orthogonal matrix Q_1 in the QR-factorization (30).*

Step 1: *Set $\mu := n_k + \cdots + n_l$ and $Q_1 = I_{2n}$.*

Step 2:

FOR $j = 1 : n_1$,

FOR $i = j + 1 : \mu$,

Compute a Givens rotation such that

$$G^T \begin{bmatrix} X(2n + j - \mu, j) \\ X(2n + i - \mu, j) \end{bmatrix} = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}.$$

IF ($j = 1$ & $n_1 = 2$) THEN set

$$\begin{aligned} X(2n + j - \mu, 2) &:= \begin{bmatrix} 1 & 0 \end{bmatrix} G^T \begin{bmatrix} X(2n + j - \mu, 2) \\ X(2n + i - \mu, 2) \end{bmatrix}, \\ X(2n + i - \mu, 2) &:= \begin{bmatrix} 0 & 1 \end{bmatrix} G^T \begin{bmatrix} X(2n + j - \mu, 2) \\ X(2n + i - \mu, 2) \end{bmatrix}. \end{aligned}$$

END

$$\begin{aligned} X(2n + j - \mu, j) &:= \gamma, \quad X(2n + i - \mu, j) := 0, \\ X(n + j - \mu, :) &:= \begin{bmatrix} 1 & 0 \end{bmatrix} G^T \begin{bmatrix} X(n + j - \mu, :) \\ X(n + i - \mu, :) \end{bmatrix}, \\ X(n + i - \mu, :) &:= \begin{bmatrix} 0 & 1 \end{bmatrix} G^T \begin{bmatrix} X(n + j - \mu, :) \\ X(n + i - \mu, :) \end{bmatrix}, \\ Q_1(:, n + j - \mu) &:= \begin{bmatrix} Q_1(: .n + j - \mu) & Q_1(: n + i - \mu) \end{bmatrix} G \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ Q_1(:, n + i - \mu) &:= \begin{bmatrix} Q_1(: .n + j - \mu) & Q_1(: n + i - \mu) \end{bmatrix} G \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \\ Q_1(:, 2n + j - \mu) &:= \begin{bmatrix} Q_1(: .2n + j - \mu) & Q_1(: 2n + i - \mu) \end{bmatrix} G \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ Q_1(:, 2n + i - \mu) &:= \begin{bmatrix} Q_1(: .2n + j - \mu) & Q_1(: 2n + i - \mu) \end{bmatrix} G \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{aligned}$$

END
END

Step 3*: Compute an orthogonal matrix G using $[(n_1 + (n_1 - 1)2]$ Givens rotations such that*

$$G^T \begin{bmatrix} X(n-\mu+1 : n-\mu+n_1, 1 : n_1) \\ X(2n-\mu+1 : 2n-\mu+n_1, 1 : n_1) \end{bmatrix} = \begin{bmatrix} \Gamma \\ 0 \end{bmatrix},$$

where $\Gamma \in \mathbb{R}^{n_1 \times n_1}$ is upper-triangular. Set

$$\begin{aligned} &X(n-\mu+1 : n-\mu+n_1, 1 : n_1) := \Gamma, \\ &X(2n-\mu+1 : 2n-\mu+n_1, 1 : n_1) := 0, \\ &Q_1(:, n-\mu+1 : n-\mu+n_1) := \\ &\begin{bmatrix} Q_1(:, n-\mu+1 : n-\mu+n_1) & Q_1(:, 2n-\mu+1 : 2n-\mu+n_1) \end{bmatrix} G \begin{bmatrix} I_{n_1} \\ 0 \end{bmatrix}. \end{aligned}$$

Step 4*:*

FOR $j = 1 : n_1$,
FOR $i = 1 : \mu - n_k$,
Compute a Givens rotation such that

$$G^T \begin{bmatrix} X(n-\mu+j, j) \\ X(n+1-i, j) \end{bmatrix} = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}.$$

IF ($j = 1$ & $n_1 = 2$) THEN set

$$\begin{aligned} X(n-\mu+j, 2) &:= \begin{bmatrix} 1 & 0 \end{bmatrix} G^T \begin{bmatrix} X(n-\mu+j, 2) \\ X(n+1-i, 2) \end{bmatrix}, \\ X(n+1-i, 2) &:= \begin{bmatrix} 0 & 1 \end{bmatrix} G^T \begin{bmatrix} X(n-\mu+j, 2) \\ X(n+1-i, 2) \end{bmatrix}, \end{aligned}$$

$$\begin{aligned} X(n-\mu+j, j) &:= \gamma, \quad X(n+1-i, j) := 0, \\ Q_1(:, n-\mu+j) &:= \begin{bmatrix} Q_1(:, n-\mu+j) & Q_1(:, n+1-i) \end{bmatrix} G \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ Q_1(:, n+1-i) &:= \begin{bmatrix} Q_1(:, n-\mu+j) & Q_1(:, n+1-i) \end{bmatrix} G \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{aligned}$$

END

END

END

Step 5*:*

FOR $j = 1 : n_1$,

FOR $i = 1 : n - \mu$,

Compute a Givens rotation such that

$$G^T X(n - \mu + j - i : n - \mu + j + 1 - i, j) = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}.$$

IF $(j = 1 \;\&\; n_1 = 2)$ *THEN set*

$$X(n - \mu + j - i : n - \mu + j + 1 - i, 2) := G^T X(n - \mu + j - i : n - \mu + j + 1 - i, 2)$$

END

$$X(n - \mu + j - i : n - \mu + j + 1 - i, j) := \begin{bmatrix} \gamma \\ 0 \end{bmatrix},$$

$$Q_1(:, n - \mu + j - i : n - \mu + j + 1 - i) = Q_1(:, n - \mu + j - i : n - \mu + j + 1 - i)G.$$

END

END

Step 6*: Set* $Q_1 := Q_1 \begin{bmatrix} I_n \\ 0 \end{bmatrix}$.