COMPUTATIONAL EXPERIENCE WITH STRUCTURE-PRESERVING HAMILTONIAN SOLVERS IN COMPLEX SPACES

Vasile Sima
Department of Advanced Research
National Institute for Research & Development in Informatics
Romania
vsima@ici.ro

Abstract
Structure-preserving numerical techniques for computation of eigenvalues and stable deflating subspaces of complex skew-Hamiltonian/Hamiltonian matrix pencils, with applications in control systems analysis and design, are presented. The techniques use specialized algorithms to exploit the structure of such matrix pencils: the skew-Hamiltonian/Hamiltonian Schur form decomposition and the periodic QZ algorithm. The structure-preserving approach has the potential to avoid the numerical difficulties which are encountered for an unstructured solution, implemented by the currently available software tools.

Key words
Computational methods, Control systems design, Eigenvalue problems, Linear multivariable systems, Numerical algorithms.

1 Introduction

Several basic computational problems in optimal and robust systems analysis and design involve structured, e.g., Hamiltonian and symplectic, matrix pencils. Two important problems, with many applications, are discussed below.

Some definitions are first recalled. A matrix pencil $\lambda M - N$ is *Hamiltonian* if $N\mathcal{J}M^H = -M\mathcal{J}N^H$, and it is *symplectic* if $N\mathcal{J}N^H = M\mathcal{J}M^H$, where

$$\mathcal{J} := \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}, \qquad \mathcal{J}^T = -\mathcal{J} = \mathcal{J}^{-1},$$

the superscripts H and T denote the conjugate-transpose and transpose, respectively, and I_n denotes the identity matrix of order n. If $M = I_{2n}$, definitions for Hamiltonian and symplectic matrices are obtained; for instance, N is *Hamiltonian* if $(N\mathcal{J})^H = N\mathcal{J}$, and it is *skew-Hamiltonian* if $(N\mathcal{J})^H = -N\mathcal{J}$. Note that $M \in \mathbb{C}^{2n\times 2n}$ is skew-Hamiltonian if and only if $\imath M$ is Hamiltonian, where $\imath$ denotes the purely imaginary unit. A matrix pencil $\lambda M - N$ is *skew-Hamiltonian/Hamiltonian* if M is skew-Hamiltonian, and N is Hamiltonian. These pencils have spectra which are symmetric with respect to the imaginary axis. Such pencils arrise in various domains of applied mathematics, computational physics, chemistry, optimal control, etc.

The *skew-Hamiltonian/Hamiltonian Schur form* of a skew-Hamiltonian/Hamiltonian pencil $\lambda M - N$ is

$$\mathcal{J}\mathcal{Q}\mathcal{J}^T \left(\lambda \begin{bmatrix} M_{11} & M_{12} \\ 0 & M_{11}^H \end{bmatrix} - \begin{bmatrix} N_{11} & N_{12} \\ 0 & -N_{11}^H \end{bmatrix} \right) \mathcal{Q}^H \tag{1}$$

where $\mathcal{Q}$ is unitary, $M_{11}, N_{11} \in \mathbb{C}^{n\times n}$ are upper triangular, M_{12} is *skew-Hermitian*, and N_{12} is *Hermitian*, i.e., $M_{12} = -M_{12}^H$, and $N_{12} = N_{12}^H$. This form displays the pencil eigenvalues. Some pencils which lack this form can be embedded in pencils which always have a skew-Hamiltonian/Hamiltonian Schur form. For a matrix, the definition above can be specialized to (skew-)Hamiltonian Schur form. Real skew-Hamiltonian matrices, and Hamiltonian matrices without purely imaginary eigenvalues have Hamiltonian Schur forms.

One basic computation in optimal and robust control systems analysis and design is the evaluation of the L_∞-norm. An example is specifying upper bounds on the weighted and/or mixed sensitivity transfer-function matrices in the H_∞ design problems. More generally, L_∞-norms are used to quantify the trade-off between performance and robust stability. Consider a linear time-invariant generalized system, described by its state-space matrices and the associated transfer-function matrix

$$G(\lambda) = C(\lambda E - A)^{-1}B + D, \tag{2}$$

where $A, E \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{n\times m}$, $C \in \mathbb{C}^{p\times n}$, $D \in \mathbb{C}^{p\times m}$, and λ is a complex variable replacing the Laplace variable, s, for a continuous-time system, and the Z-transform variable, z, for a discrete-time system. For convenience, assume that E is nonsingular. Briefly speaking, the L_∞-norm for (2) is defined as the peak gain of the frequency response of $G(\lambda)$. This is finite if and only if the matrix pencil (A, E) has no eigenvalue on the boundary of the stability domain, B_s, where $B_s = \{s \,|\, \Re(s) = 0\}$, for a continuous-time system, $B_s = \{z \,|\, \|z\|_2 = 1\}$, for a discrete-time system, and $\Re(\cdot)$ denotes the real part of a complex number. In this case, which includes standard systems ($E = I_n$) with A stable, the L_∞-norm, also then called H_∞-norm, can be expressed by the least upper bound,

$$\|G\|_\infty := \sup_{\lambda\in B_s} \sigma_{\max}(G(\lambda)),$$

where $\sigma_{\max}(M)$ denotes the largest singular value of the matrix M. Quadratically convergent algorithms [Bruinsma and Steinbuch, 1990] for the computation of the L_∞/H_∞-norm use the purely imaginary eigenvalues of a structured, Hamiltonian or symplectic, matrix or matrix pencil at each iteration. (Actually, the pencils arising in the continuous-time case are skew-Hamiltonian/Hamiltonian.) The detection of purely imaginary eigenvalues is a delicate numerical problem if an unstructured algorithm is used. Several simple examples are given in Section 3.

Another fundamental computation in control systems design is the solution of continuous-time and discrete-time algebraic Riccati equations (CAREs and DAREs). CAREs and DAREs arise in many applications, such as, stabilization and linear-quadratic regulator problems, Kalman filtering, LQG—linear-quadratic Gaussian (H_2-) optimal control problems, computation of (sub)optimal H_∞ controllers, model reduction techniques based on stochastic, positive or bounded real LQG balancing, factorization procedures for transfer functions. Usually, the *stabilizing solution* is required, which can be used to stabilize the closed-loop system matrix or matrix pencil. A very important class of CARE/DARE solvers makes use of stable invariant or deflating subspaces of some structured, Hamiltonian or symplectic matrices or pencils, assuming certain nonsingularity and eigenvalue dichotomy properties [Laub, 1979]. The explicit need of matrix inversion in the CARE/DARE solvers (for instance, of the system matrix, for symplectic DARE solvers) can ruin the accuracy of the results, if the matrix to be inverted is ill-conditioned. Better results can be obtained using stable deflating subspaces of extended matrix pencils, with no inversion involved [Bender and Laub, 1987*a*; Bender and Laub, 1987*b*; Lancaster and Rodman, 1995; Mehrmann, 1991; Van Dooren, 1981].

The solvers currently available, e.g., in MATLAB® Control System Toolbox [MATLAB, 2010], and SLICOT [Benner et al., 1999; Benner et al., 2010], are using the standard QZ algorithm for reordering the eigenvalues, to determine the stable deflating subspaces. The special structure of the matrix pencils involved is not exploited, but it should be exploited in order to improve the numerical properties of the Riccati solvers. Recently, structure-exploiting techniques have been investigated for solving skew-Hamiltonian/Hamiltonian eigenproblems, see, e.g., [Benner et al., 2002; Benner et al., 2007], and the references therein. These techniques can be employed for CARE solvers. For solving DAREs, it is possible to preprocess the pencils by an extended Cayley transformation, which only involves matrix additions and subtractions [Xu, 2006], to obtain equivalent skew-Hamiltonian/Hamiltonian pencils.

In the sequel, the pencils $\lambda M - N$ will be represented in the numerically better form $\alpha M - \beta N$, with $\lambda = \alpha/\beta$ (possibly ∞).

2 Computation of Eigenvalues and Stable Deflating Subspaces of complex skew-Hamiltonian/Hamiltonian matrix pencils

Let $\alpha\mathcal{S} - \beta\mathcal{H}$ be a skew-Hamiltonian/Hamiltonian pencil, i.e., $(\mathcal{S}\mathcal{J})^H = -\mathcal{S}\mathcal{J}$, $(\mathcal{H}\mathcal{J})^H = \mathcal{H}\mathcal{J}$. By definition, these pencils have even size, $2n$. Therefore, odd-order pencils, which can appear, e.g., in optimal control, should be extended to an even size, to apply the techniques summarized below. Moreover, sometimes, permutation and scaling are needed to transform the original pencils to the skew-Hamiltonian/Hamiltonian form.

For some problems, including linear-quadratic optimization applications, $\mathcal{S}$ can be given in a factored form, the so-called *skew-Hamiltonian Cholesky factorization*, defined by $\mathcal{S} = \mathcal{J}\mathcal{Z}^H\mathcal{J}^T\mathcal{Z}$. Such a matrix S is said to be *$\mathcal{J}$-semidefinite*. For instance, for a block-diagonal matrix $\mathcal{S} = \mathrm{diag}(E, E^H)$, a factor $\mathcal{Z}$ can be written as $\mathcal{Z} = \mathrm{diag}(I_n, E^H)$.

Some properties of skew-Hamiltonian/Hamiltonian pencils are proven, e.g., in [Benner et al., 2002]. Let $\alpha\mathcal{S} - \beta\mathcal{H}$ be a skew-Hamiltonian/Hamiltonian pencil with nonsingular $\mathcal{J}$-semidefinite skew-Hamiltonian part $\mathcal{S} = \mathcal{J}\mathcal{Z}^H\mathcal{J}^T\mathcal{Z}$. Under certain conditions (see [Benner et al., 2002]), then there are a unitary matrix $\mathcal{Q}$ and a unitary symplectic matrix $\mathcal{U}$, such that

$$\begin{aligned} \mathcal{U}^H\mathcal{Z}\mathcal{Q} &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ \mathcal{J}\mathcal{Q}^H\mathcal{J}^T\mathcal{H}\mathcal{Q} &= \begin{bmatrix} H_{11} & H_{12} \\ 0 & -H_{11}^H \end{bmatrix}, \end{aligned} \qquad (3)$$

where Z_{11}, Z_{22}^T, and H_{11} are $n \times n$ upper triangular. Similarly, if $\imath\mathcal{H}$ is also nonsingular $\mathcal{J}$-semidefinite, i.e., $\imath\mathcal{H} = \mathcal{J}\mathcal{W}^H\mathcal{J}^T\mathcal{W}$, then there are a unitary matrix $\mathcal{Q}$ and unitary symplectic matrices $\mathcal{U}$ and $\mathcal{V}$, such that

$$\mathcal{U}^H\mathcal{Z}\mathcal{Q} = \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \quad \mathcal{V}^H\mathcal{W}\mathcal{Q} = \begin{bmatrix} W_{11} & W_{12} \\ 0 & W_{22} \end{bmatrix},$$

where Z_{11}, Z_{22}^T, W_{11}, and W_{22}^T are $n \times n$ upper triangular.

Another property refers to *real skew-Hamiltonian/ skew-Hamiltonian* pencils, in factored form. Let $\alpha\mathcal{S} - \beta\mathcal{N}$ be a real regular skew-Hamiltonian/skew-Hamiltonian pencil with $\mathcal{S} = \mathcal{J}\mathcal{Z}^T\mathcal{J}^T\mathcal{Z}$. Then, there are a real orthogonal matrix $\mathcal{Q}$ and a real orthogonal symplectic matrix $\mathcal{U}$, such that

$$\mathcal{U}^T\mathcal{Z}\mathcal{Q} = \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \qquad \mathcal{J}\mathcal{Q}^T\mathcal{J}^T\mathcal{N}\mathcal{Q} = \begin{bmatrix} N_{11} & N_{12} \\ 0 & N_{11}^T \end{bmatrix}, \tag{4}$$

where Z_{11}, Z_{22}^T are upper triangular, N_{11} is upper quasi-triangular, and $N_{12} = -N_{12}^T$. Moreover, the Schur form

$$\mathcal{J}\mathcal{Q}^T\mathcal{J}^T(\alpha\mathcal{S} - \beta\mathcal{N})\mathcal{Q} = \alpha\begin{bmatrix} Z_{22}^T Z_{11} & Z_{22}^T Z_{12} - Z_{12}^T Z_{22} \\ 0 & Z_{11}^T Z_{22} \end{bmatrix} - \beta\begin{bmatrix} N_{11} & N_{12} \\ 0 & N_{11}^T \end{bmatrix}$$

is a $\mathcal{J}$-congruent skew-Hamiltonian/skew-Hamiltonian matrix pencil. Consequently, the spectra of such pencils have eigenvalues with multiplicity at least 2.

An algorithm for computing the eigenvalues and bases for the stable deflating subspace (corresponding to the eigenvalues with strictly negative real part), and for a companion subspace, of a complex skew-Hamiltonian/Hamiltonian pencil is summarized below, based on Algorithm 1 in [Benner et al., 2002]:

Algorithm 1. Let $\alpha\mathcal{S} - \beta\mathcal{H}$ be a complex regular skew-Hamiltonian/Hamiltonian pencil with $\mathcal{S} = \mathcal{J}\mathcal{Z}^H\mathcal{J}^T\mathcal{Z}$.

1. Let $\mathcal{T} = \imath\mathcal{H}$, which is skew-Hamiltonian. Define the "embedded" matrices $\mathcal{B}_\mathcal{Z} = \mathrm{diag}(\mathcal{Z}, \bar{\mathcal{Z}})$, $\mathcal{B}_\mathcal{T} = \mathrm{diag}(\mathcal{T}, \bar{\mathcal{T}})$, and the real matrices of order $4n$

$$\widehat{\mathcal{B}}_\mathcal{Z} = (\mathcal{Y}\mathcal{P})^H\mathcal{B}_\mathcal{Z}(\mathcal{Y}\mathcal{P}), \quad \widehat{\mathcal{B}}_\mathcal{T} = (\mathcal{Y}\mathcal{P})^H\mathcal{B}_\mathcal{T}(\mathcal{Y}\mathcal{P}),$$

where

$$\mathcal{Y} = \frac{\sqrt{2}}{2}\begin{bmatrix} I_{2n} & \imath I_{2n} \\ I_{2n} & -\imath I_{2n} \end{bmatrix}, \ \mathcal{P} = \begin{bmatrix} I_n & 0 & 0 & 0 \\ 0 & 0 & I_n & 0 \\ 0 & I_n & 0 & 0 \\ 0 & 0 & 0 & I_n \end{bmatrix}.$$

Compute the corresponding decompositions (4), for the real skew-Hamiltonian/skew-Hamiltonian pencil $\alpha\mathcal{J}_2\widehat{\mathcal{B}}_\mathcal{Z}^T\mathcal{J}_2^T\widehat{\mathcal{B}}_\mathcal{Z} - \beta\widehat{\mathcal{B}}_\mathcal{T}$ (where $\mathcal{J}_2$ is the matrix $\mathcal{J}$ with blocks of order $2n$), using a real orthogonal matrix $\mathcal{Q}$ and a real orthogonal symplectic matrix $\mathcal{U}$,

$$\widetilde{\mathcal{B}}_\mathcal{Z} = \mathcal{U}^T\widehat{\mathcal{B}}_\mathcal{Z}\mathcal{Q} = \begin{bmatrix} \mathcal{Z}_{11} & \mathcal{Z}_{12} \\ 0 & \mathcal{Z}_{22} \end{bmatrix}, \qquad \widetilde{\mathcal{B}}_\mathcal{T} = \mathcal{J}\mathcal{Q}^T\mathcal{J}^T\widehat{\mathcal{B}}_\mathcal{T}\mathcal{Q} = \begin{bmatrix} \mathcal{T}_{11} & \mathcal{T}_{12} \\ 0 & \mathcal{T}_{11}^T \end{bmatrix},$$

where $\mathcal{Z}_{11}$, $\mathcal{Z}_{22}^T$, are upper triangular, $\mathcal{T}_{11}$ is upper quasi-triangular, and $\mathcal{T}_{12} = -\mathcal{T}_{12}^T$. Transform the 2×2 real diagonal blocks into 1×1 complex blocks.

2. Find a unitary matrix $\widehat{\mathcal{Q}}$ and a unitary symplectic matrix $\widehat{\mathcal{U}}$, such that

$$\breve{\mathcal{B}}_\mathcal{Z} := \widetilde{\mathcal{U}}^H\widetilde{\mathcal{B}}_\mathcal{Z}\widetilde{\mathcal{Q}} = \begin{bmatrix} \widetilde{\mathcal{Z}}_{11} & \widetilde{\mathcal{Z}}_{12} \\ 0 & \widetilde{\mathcal{Z}}_{22} \end{bmatrix}, \qquad \breve{\mathcal{B}}_\mathcal{H} := \mathcal{J}\widetilde{\mathcal{Q}}^H\mathcal{J}^T\big(-\imath\widetilde{\mathcal{B}}_\mathcal{T}\big)\widetilde{\mathcal{Q}} = \begin{bmatrix} \mathcal{H}_{11} & \mathcal{H}_{12} \\ 0 & -\mathcal{H}_{11}^H \end{bmatrix},$$

where $\widetilde{\mathcal{Z}}_{11}$, $\widetilde{\mathcal{Z}}_{22}^T$, and $\mathcal{H}_{11}$ are upper triangular, such that $\Lambda_{\text{-}}\big(\breve{\mathcal{B}}_\mathcal{H}, \mathcal{J}\breve{\mathcal{B}}_\mathcal{Z}^H\mathcal{J}^T\breve{\mathcal{B}}_\mathcal{Z}\big)$ is contained in the spectrum of the leading $2p \times 2p$ principal subpencil of $\alpha\widetilde{\mathcal{Z}}_{22}^H\widetilde{\mathcal{Z}}_{11} - \beta\mathcal{H}_{11}$. The notation $\Lambda_{\text{-}}(N, M)$ denotes the stable finite spectrum of the pencil $\alpha M - \beta N$, and p is the number of eigenvalues in $\Lambda_{\text{-}}(\mathcal{H}, \mathcal{S})$.

3. Set

$$V = \begin{bmatrix} I_{2n} & 0 \end{bmatrix}\mathcal{Y}\mathcal{P}\mathcal{Q}\widetilde{\mathcal{Q}}\begin{bmatrix} I_{2p} \\ 0 \end{bmatrix}, \qquad U = \begin{bmatrix} I_{2n} & 0 \end{bmatrix}\mathcal{Y}\mathcal{P}\mathcal{U}\widetilde{\mathcal{U}}\begin{bmatrix} I_{2p} \\ 0 \end{bmatrix},$$

and compute an orthonormal basis of the stable deflating subspace, and an orthonormal basis of the companion subspaceas, P_U^-, [Benner et al., 2002], as a basis of range V and range U, respectively.

Step 1 of the algorithm uses an RQ decomposition with changed elimination order (compared to the classical RQ decomposition), to triangularize $\widehat{\mathcal{B}}_\mathcal{Z}$ in the form stated for $\widetilde{\mathcal{B}}_\mathcal{Z}$. (Actually, both QR and RQ decompositions are used.) Skew-symmetric updates of the off-diagonal blocks of order $2n$ of $\widehat{\mathcal{B}}_\mathcal{T}$ are made, and the (1,1) diagonal block is also updated for the transformations applied to the $\widehat{\mathcal{B}}_\mathcal{Z}$ part. Then, the transformed $\widehat{\mathcal{B}}_\mathcal{T}$ part is reduced to the skew-Hamiltonian Hessenberg form, using real plane rotations (i.e., the (1,1) block is upper Hessenberg, and the (2,1) block is zero), and $\widetilde{\mathcal{B}}_\mathcal{Z}$ is updated while maintaining its form. Finally, the real periodic QZ algorithm [Bojanczyk et al., 1992; Sreedhar and Van Dooren, 1994], applied to the pencil $\alpha\mathcal{Z}_{22}^T\mathcal{Z}_{11} - \beta\mathcal{T}_{11}$ is used, to reduce the obtained upper Hessenberg matrix $\mathcal{T}_{11}$ to upper quasi-triangular form, while preserving the other factors upper triangular. The 2×2 diagonal blocks are transformed to equivalent complex 1×1 blocks using the complex periodic QZ algorithm for 2×2 pencils.

Note that if only the eigenvalues are desired, then they are returned by the periodic QZ algorithm called in Step 1 of the algorithm, and the transformations should not be accumulated.

Step 2 needs to reorder the eigenvalues of the pencil. A special strategy is used. First, the eigenvalues in the subpencil $\alpha\mathcal{Z}_{22}^H\mathcal{Z}_{11} - \beta\mathcal{T}_{11}$ are reordered, in two substeps: reorder the eigenvalues with negative real parts to the top; then, reorder the eigenvalues with positive real parts to the bottom. Second, the remaining eigenvalues with negative real parts are reordered, also in two substeps: the eigenvalue of the last diagonal block in the current subpencil $\alpha\widetilde{\mathcal{Z}}_{22}^H\widetilde{\mathcal{Z}}_{11} - \beta\mathcal{H}_{11}$ is interchanged with the eigenvalue of the last diagonal block in the current pencil $\alpha\mathcal{J}_2\check{\mathcal{B}}_{\mathcal{Z}}^H\mathcal{J}_2^T\check{\mathcal{B}}_{\mathcal{Z}} - \beta\check{\mathcal{B}}_{\mathcal{H}}$; finally, the eigenvalue in the $2n$-th place is moved to the $(p+1)$-th place, where p denotes the current number of eigenvalues with negative real parts in the subpencil $\alpha\widetilde{\mathcal{Z}}_{22}^H\widetilde{\mathcal{Z}}_{11} - \beta\mathcal{H}_{11}$. All these exchanges essentially involve 2×2 pencils and are performed using complex plane rotations.

The structure can be exploited in the all steps of the algorithm. For instance, $\mathcal{H}_{12} = \mathcal{H}_{12}^T$, and so, only its upper triangular part should be computed. Also, as mentioned, skew-symmetric updates are used, whenever possible.

A similar algorithm can be written for an unfactored matrix $\mathcal{S}$, based on Algorithm 2 in [Benner et al., 2002], and the called algorithms. Extended embedded matrices $\widehat{\mathcal{B}}_{\mathcal{S}}$ (instead of $\widehat{\mathcal{B}}_{\mathcal{Z}}$), and $\widehat{\mathcal{B}}_{\mathcal{T}}$ are used. Both have the same structure.

Below is a summary about the related software:

- Fortran and MATLAB software for eigenvalues and deflating subspaces have just been developed.
- Both real and complex cases are considered.
- Factored or unfactored versions are covered.
- Auxiliary routines for problems (of even order) with (quasi-)triangular structure are included.
- Optimized kernels for problems of order 2, 3, or 4, called by the general solvers, are available.

3 Numerical Results

This section presents some preliminary numerical results, based on Fortran implementation of the algorithms and corresponding MATLAB MEX-files. These results have been obtained on a portable Intel Dual Core computer at 2 GHz, with 2 GB RAM, and relative machine precision $\epsilon \approx 1.11 \times 10^{-16}$, using Windows XP (Service Pack 2) operating system, Intel Visual Fortran 11.1 compiler, and MATLAB 7.11.0.584 (R2010b).

3.1 Computation of Eigenvalues

Many numerical tests have been performed, to assess the correct behavior of the developed solvers.

Several small skew-Hamiltonian/Hamiltonian examples are used below to illustrate the limitations of the standard, unstructured approach. The generalized eigenvalues computed by a structure-preserving algorithm and the standard QZ algorithm, optimally implemented in the MATLAB function `eig`, have been compared with those delivered by symbolic calculations, using the following MATLAB commands[1]

```
Ss = sym( S );  Hs = sym( H );
evs = double( eig( Ss \ Hs ) );
```

Based on the symmetry properties of the eigenvalues of complex $(\mathcal{H},\mathcal{S})$ pencils, for every eigenvalue λ, $-\bar{\lambda}$ is also an eigenvalue. (This does not mean that a purely imaginary eigenvalue is necessarily multiple, but that $-\bar{\lambda} = \lambda$ in this case.) Consequently, a quality measure which has been used was the (relative) *deviation from symmetry* of the eigenvalues, defined as $\|(\lambda - P(\lambda, -\bar{\lambda}))\|_2$, where $P(v,w)$ is a permutation which makes the vector w be as close as possible to the vector v. (The *relative deviation* is obtained by dividing the deviation to the norm of λ.) Let

$$\mathcal{S} = \begin{bmatrix} 2-3\imath & 3\imath \\ -5\imath & 2+3\imath \end{bmatrix}, \; \mathcal{H} = \begin{bmatrix} -19-3\imath & -4 \\ -1 & 19-3\imath \end{bmatrix}.$$

The structured algorithm found the eigenvalues

$$-2.526134862339483\imath,\; 74.02613486233942\imath\,,$$

the MATLAB function `eig` returned

$$-2.526134862339483\imath\,,$$
$$3.22768506974576\cdot 10^{-13} + 74.02613486233955\imath\,,$$

and the symbolic MATLAB function `eig` computed

$$-2.526134862339484\imath,\; 74.02613486233949\imath\,.$$

The relative error norms of the first two solvers, compared to the symbolic solver, have the values $9.59\cdot 10^{-16}$ and $4.42\cdot 10^{-15}$, respectively. The first value is about 5 times smaller than the second one. The eigenvalues are plotted in Fig. 1.

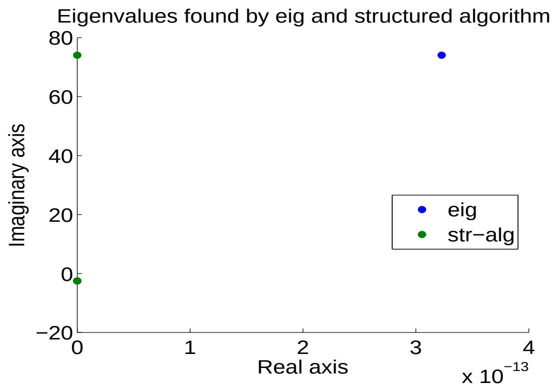

Figure 1. Eigenvalue scatter plot for an example of order 2. The smallest eigenvalue computed by `eig` coincides to that computed by the structured solver.

Several similar examples have been built. The standard solver gives possible large errors in the real parts for problems with purely imaginary eigenvalues. Fig. 2 shows another eigenvalue comparison for an example of order 4.

[1] Unfortunately, there is no MATLAB generalized symbolic eigensolver, so the `mldivide` (or `mrdivide`) operator has been used, but the condition numbers of the tried skew-Hamiltonian matrices were small.

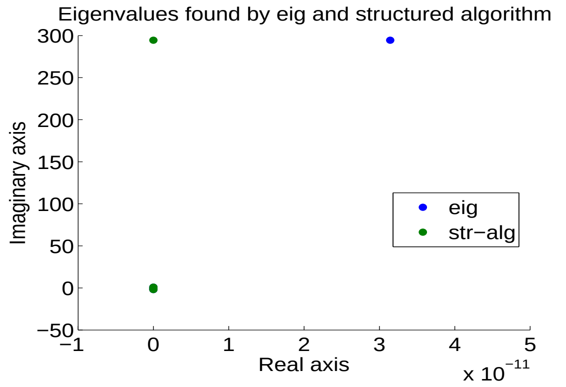

Figure 2. Eigenvalue scatter plot for an example of order 4. Three eigenvalue computed by `eig` are very close to those computed by the structured solver, but one eigenvalue has a large real part.

In a set of 60 test problems (with 20 problems for each value $m := n/2 \in \{1, 2, 3\}$), the sum of the relative errors for the structured and standard solver (compared to the symbolic solver) had the values $1.19 \cdot 10^{-13}$ and $1.67 \cdot 10^{-13}$, respectively. The important fact is that the symmetry of the spectrum is ensured by the structured solver.

It was not possible to symbolically solve problems with $m \geq 5$. For larger matrices, the differences between the results produced by the structured solver and by `eig` were more pronounced.

The deviation from symmetry is usually nonzero, but small, even for structured solvers, due to the complex eigenvalue calculation. For one example, the deviation was $6.28 \cdot 10^{-15}$, for the structured solver, but $1.4 \cdot 10^{-10}$, for `eig`.

In a set of 1200 test problems (with 20 problems for each value $m \in \{1, \ldots, 30\}$, and matrices generated randomly using both uniform and normal distributions), the maximum relative deviation was $3 \cdot 10^{-16}$, for the structured solver, and $5.4 \cdot 10^{-11}$, for MATLAB `eig`.

3.2 Computation of Right Deflating Subspaces

Thousands of tests have been performed with random matrices for computing the eigenvalues and right deflating subspaces of skew-Hamiltonian/Hamiltonian matrix pencils. The results computed by the structured solvers have been in good agreement to those obtained by the standard, unstructured solver, `eig`. As mentioned before, the structured solvers return better computed eigenvalues, satisfying the symmetry property. This might come with a greater cost, due to the increased complexity of the implementations, compared to the standard solver. Note that some initial performance results for eigenvalue computations using the unfactored version of the solver, in comparison with `eig`, have been reported in [Sima, 2010]. Unfortunately, the speed-up values were wrong. Actually, `eig` was about two to over six fold faster than the structured solver. The current implementation is significanly improved, being over 2.5 times faster than the previous one, but it still does not outperform `eig` for pencils of order larger than, say, 400. It is planned to investigate the bottlenecks and further speed-up the structured solvers, e.g., by a better use of block algorithms. Note that `eig` recorded many improvements during its long life.

Fig. 3 presents the CPU times needed by the structured and unstructured solvers for computing the eigenvalues in the unfactored case. Fig. 4 presents the ratios of the CPU times needed by the structured and unstructured solvers for computing the eigenvalues. The structured solver is about 20% faster than `eig` for pencils of order less than 300.

Similarly, Fig. 5 and Fig. 6 show the same parameters when computing both the eigenvalues and a basis for the stable deflating subspace. The structured solver is about 10% faster than `eig` for orders less than 300.

The timing values are mean values: five problems of each order were solved. The list of problem orders has an increment 10. The factored solver needed about twice as much CPU time compared to the unfactored solver, but can offer better accuracy, by avoiding the computation of $\mathcal{J}\mathcal{Z}^H\mathcal{J}^T\mathcal{Z}$. The CPU ratios behaved similarly on a computer at 3 GHz, with 1 GB RAM, but the CPU times were over 2.5 fold larger for all solvers.

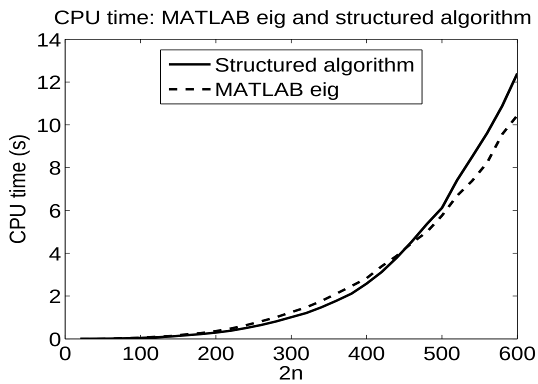

Figure 3. CPU times needed by the structured and unstructured solvers for computing the eigenvalues.

4 Conclusion

Main issues related to the structure-preserving algorithms for computing the eigenvalues and stable deflating subspaces of complex skew-Hamiltonian/Hamiltonian matrix pencils, with applications in control systems analysis and design, are presented. The techniques use specialized algorithms to exploit the structure of such matrix pencils. The structured solvers ensure the symmetry of the spectra, while the standard algorithm can deliver eigenvalues with large deviation from symmetry, even for problems of order 2. The current implementations are slower than the standard, un-

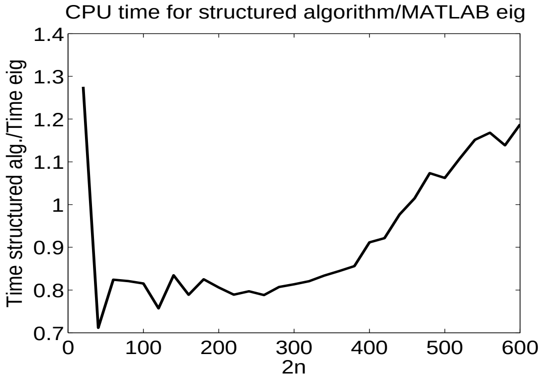

Figure 4. Ratios of the CPU times needed by the structured and unstructured solvers for computing the eigenvalues.

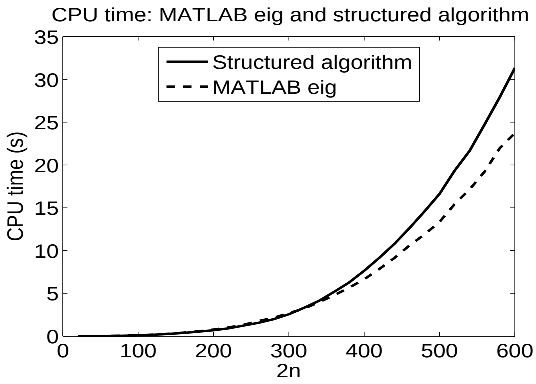

Figure 5. CPU times needed by the structured and unstructured solvers for computing the eigenvalues and the stable deflating subspace.

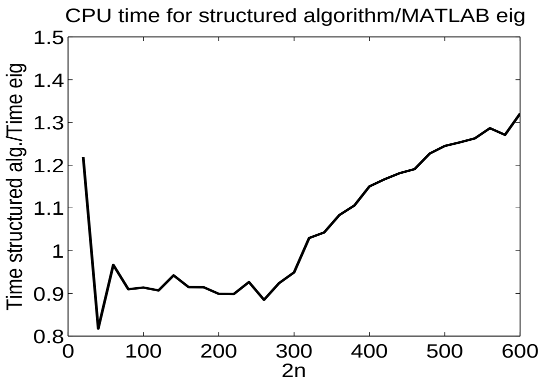

Figure 6. Ratios of the CPU times needed by the structured and unstructured solvers for computing the eigenvalues and the stable deflating subspace.

structured algorithm for pencils with orders larger than few hundreds, especially when computing the deflating subspaces. Improved, faster solvers are investigated.

Acknowledgements

The collaboration with Peter Benner, Matthias Voigt, and Daniel Kressner is highly acknowledged.

References

Bender, D. J. and Laub, A. J. (1987*a*). The linear-quadratic optimal regulator for descriptor systems, *IEEE Trans. Automat. Contr.* **AC-32**(8), pp. 672–688.

Bender, D. J. and Laub, A. J. (1987*b*). The linear-quadratic optimal regulator for descriptor systems: Discrete-time case, *Automatica* **23**(1), pp. 71–85.

Benner, P., Byers, R., Losse, P., Mehrmann, V. and Xu, H. (2007). Numerical solution of real skew-Hamiltonian/Hamiltonian eigenproblems, Technical report, Technische Universität Chemnitz, Chemnitz.

Benner, P., Byers, R., Mehrmann, V. and Xu, H. (2002). Numerical computation of deflating subspaces of skew Hamiltonian/Hamiltonian pencils, *SIAM J. Matrix Anal. Appl.* **24**(1), pp. 165–190.

Benner, P., Kressner, D., Sima, V. and Varga, A. (2010). Die SLICOT-Toolboxen für Matlab, *at—Automatisierungstechnik* **58**(1), pp. 15–25.

Benner, P., Mehrmann, V., Sima, V., Van Huffel, S. and Varga, A. (1999). SLICOT — A subroutine library in systems and control theory. In *Applied and Computational Control, Signals, and Circuits*, Vol. 1, chapter 10, Birkhäuser, Boston, pp. 499–539.

Bojanczyk, A. W., Golub, G. and Van Dooren, P. (1992). The periodic Schur decomposition: algorithms and applications. In *Proc. of the SPIE Conference*, Vol. 1770, pp. 31–42.

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

Lancaster, P. and Rodman, L. (1995). *The Algebraic Riccati Equation*, Oxford University Press, Oxford.

Laub, A. J. (1979). A Schur method for solving algebraic Riccati equations, *IEEE Trans. Automat. Contr.* **AC–24**(6), pp. 913–921.

MATLAB (2010). The MATLAB Control System Toolbox. Version 9.

Mehrmann, V. (1991). *The Autonomous Linear Quadratic Control Problem. Theory and Numerical Solution*, Springer-Verlag, Berlin.

Sima, V. (2010). Structure-preserving algorithms for discrete-time algebraic matrix Riccati equations. In *Proceedings of the "7th International Conference on Informatics in Control, Automation and Robotics" (ICINCO 2010)*, Funchal, Madeira - Portugal, 15–18 June, 2010, Vol. 1, SciTePress – Science and Technology Publications, Portugal, pp. 187–192.

Sreedhar, J. and Van Dooren, P. (1994). Periodic Schur form and some matrix equations. In *Proc. of the Symposium on the Mathematical Theory of Networks and Systems (MTNS'93)*, Vol. 1, Regensburg, Germany, pp. 339–362.

Van Dooren, P. (1981). A generalized eigenvalue approach for solving Riccati equations, *SIAM J. Sci. Stat. Comput.* **2**(2), pp. 121–135.

Xu, H. (2006). On equivalence of pencils from discrete-time and continuous-time control, *Lin. Alg. Appl.* **414**, pp. 97–124.