Invariant subspace methods for CAREs

X solves CARE $A^*X + XA + Q = XGX$ iff

$$\begin{bmatrix} A & -G \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} I \\ X \end{bmatrix} = \begin{bmatrix} I \\ X \end{bmatrix} \mathcal{R}, \quad \mathcal{R} = A - GX.$$

One can find X through an invariant subspace of the Hamiltonian.

```
>> [A,G,Q] = carex(4) %if test suite is installed
>> n = length(A);
>> H = [A -G; -Q -A'];
>> [U, T] = schur(H);
>> [U, T] =ordschur(U, T, 'lhp');
>> X = U(n+1:2*n, 1:n) / U(1:n, 1:n);
```

Recall: backward stability

QR-like algorithms based on successive orthogonal transformations are backward stable: the local error ΔM_i at each step (from machine arithmetic + truncation of 'numerical zeros') is mapped back to a "global error" $Q_1^T Q_2^T \dots Q_i^T (\Delta M_i) Q_i \dots Q_2 Q_1$ of the same norm.

In particular, the Schur method computes a true invariant subspace of $\mathcal{H} + \Delta\mathcal{H}$, with $\|\Delta\mathcal{H}\|$ small.

However, this method is not structured backward stable: the error $\Delta\mathcal{H}$ is not Hamiltonian.

Among the consequences, eigenvalues close to the imaginary axis can be 'mixed up'. Try `carex(14)` for instance: the Schur method produces an invariant subspace $\mathcal{U}$ that does not give a symmetric X, because it is the wrong invariant subspace.

Symplectic transformations

What preserves Hamiltonianity? Symplectic transformations do:

Definition

$S \in \mathbb{C}^{2n\times 2n}$ is symplectic, i.e., orthogonal w.r.t the scalar product $J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$, if $S^* J S = J$.

Lemma

If $\mathcal{H}$ is Hamiltonian and S is symplectic, then $S^{-1}\mathcal{H}S$ is Hamiltonian.

Proof: $(S^{-1}\mathcal{H}S)^* J = J(S^{-1}\mathcal{H}S) \iff$
$(S^{-1}\mathcal{H}S)^* S^* J S = S^* J S (S^{-1}\mathcal{H}S) \iff S^*\mathcal{H}^* J S = S^* J \mathcal{H} S$.

Remark: symplectic transformations do not automatically ensure stability: $\|v\|$ small does not imply $\|Sv\|$ small.

Orthosymplectic transformations

Ideal setting: construct successive changes of bases $\mathcal{H} \mapsto S^{-1}\mathcal{H}S$ where S is both orthogonal (for stability reasons) and symplectic (for structure preservation reasons).

For instance:

- If $Q \in \mathbb{R}^{n \times n}$ is any orthogonal matrix, then $\mathrm{blkdiag}(Q, Q)$ is orthogonal and symplectic.
- A Givens matrix that acts on entries k and $n+k$ (i.e., `G = eye(2*n); G([k,n+k], [k,n+k]) = [c s; -s c];`) is orthogonal and symplectic.

The “Laub trick”

There is a certain orthogonal and symplectic matrix that reduces $\mathcal{H}$ to a special form.

Let $U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}$ the unitary matrix produced by `schur(H)`.

Then,

- $\left[\begin{smallmatrix} U_{11} \\ U_{21} \end{smallmatrix}\right]$ spans the stable subspace and has orthonormal columns ($U_{11}^* U_{11} + U_{21}^* U_{21} = I$);
- we have proved earlier that $U_{21}^* U_{11} - U_{11}^* U_{21} = 0$.

These two properties imply that $V = \begin{bmatrix} U_{11} & -U_{21} \\ U_{21} & U_{11} \end{bmatrix}$ is orthogonal and symplectic.

Then, $V^T \mathcal{H} V = \begin{bmatrix} T_{11} & T_{12} \\ 0 & -T_{11}^* \end{bmatrix}$, with T_{11} upper triangular and T_{12} symmetric (Hamiltonian Schur form).

An orthogonal symplectic algorithm

It is a nice trick, but numerically it is not any more effective than the "non-structured" Schur method, because in the end the computed invariant subspace is the same $\begin{bmatrix} U_{11} \\ U_{21} \end{bmatrix}$.

But the existence of this nice structured factorization suggests that maybe there is a structure-preserving method to compute it.

Problem ("curse of Van Loan")

Is there a structure-preserving QR method that produces the Hamiltonian Schur form via a sequence of orthosymplectic transformations applied to $\mathcal{H}$?

Roadblock: some Hamiltonian matrices $\mathcal{H}$ (coming from problems with non-controllable (A, B)) do not admit a Hamiltonian Schur form $\implies$ algorithms to compute a HSF must become unstable for nearby matrices.

Chu–Liu–Mehrmann algorithm [Chu-Liu-Mehrmann '98]

A solution comes from another, different decomposition:
$\mathcal{H} = URV^T$, with U, V orthosymplectic and

$$R = \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}$$

with R_{11}, R_{22}^* upper triangular.

(Reminds a little of an SVD.)

It can be computed 'almost' directly in $O(n^3)$.

Note that this R is not Hamiltonian.

URV — simpler version (produces Hessenberg R_{22})

- Left-multiply by blkdiag(Q, Q) to get $\begin{matrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$
- Left-multiply by a Givens on $(1, n+1)$ to get $\begin{matrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$
- Left-multiply by blkdiag(Q, Q) to get $\begin{matrix} * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$
- Right-multiply by blkdiag(Q, Q) to get $\begin{matrix} * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$
- Right-multiply by a Givens on $(2, n+2)$ to get $\begin{matrix} * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & 0 & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$
- Right-multiply by blkdiag(Q, Q) to get $\begin{matrix} * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & 0 & * & * & 0 \\ 0 & * & * & * & * & * \\ 0 & * & * & * & * & * \end{matrix}$

Using URV

Note that $\mathcal{H} = URV$ together with symplecticity implies

$$\mathcal{H} = V \begin{bmatrix} -R_{22}^T & R_{12}^T \\ 0 & -R_{11}^T \end{bmatrix} U^T.$$

Hence

$$\mathcal{H}^2 = V \begin{bmatrix} -R_{11} R_{22}^T & * \\ 0 & -R_{22} R_{11}^T \end{bmatrix} V^T$$

can be used to compute eigenvalues (easily) and eigenvectors of $\mathcal{H}$ (for instance: the columns of V cause breakdown at step 2 in Arnoldi).