Always first add the auxiliary routines to your Matlab path. More precisely, under the directory containing the main routines, type the following.

>> addpath eigopt

>> addpath auxiliary

**UNSTRUCTURED STABILITY RADII**

**COMPUTATION OF r(R;B,C) = r(J;B,C)**

**(WHEN IS AVAILABLE RATHER THAN )**

% This data set is available under the directory data/randomdense800x800

% It consists of J,R,Q,B,C all of which are dense, random

% J,R,Q are 800x800, B is 800x2, and C is 2x800

>> load randomdense1.mat;

>> [f,z,info] = DHradiiJR\_nonHermit(J,Q,R,B,C,1,25,10,intval);

% *f* contains 1/r(R;B,C) = 1/r(J;B,C)

% *z* contains the maximizer of sigma\_max(CQ(iwI – (J-R)Q B) over w

**COMPUTATION OF r(Q;B,C)**

>> load randomdense1.mat;

>> [f,z,info] = DHradiiQ\_nonHermit(J,Q,R,B,C,1,25,10,intval);

% *f* contains 1/r(Q;B,C)

% *z* contains the maximizer of sigma\_max(C(iwI – (J-R) Q (J-R)B) over w

**COMPUTATION OF r(R;B,C) = r(J;B,C)**

**(WHEN IS AVAILABLE RATHER THAN )**

% This is the brake-squealing problem (for details see the motivating

% example in Section 1 in [1]; also see Section 3.3.3 in [1], this concerns

% exactly the example considered over there.)

% The data set is available under the directory data/BrakeSqueal

>> load BrakeSqueal9338x9338.mat;

>> omega = 2.5;

>> J = [-omega\*DG -KE-omega^2\*Kg; (KE+omega^2\*Kg)' zeros(size(DG))];

>> Zn = zeros(4669,4669);

>> R = [DM + (1/omega)\*DR Zn; Zn Zn];

>> Qinv = [M Zn; Zn KE+omega^2\*Kg];

>> [f,z,info] = DHradiiJR\_nonHermit\_Qinv(J,Qinv,R,B,B',2,15,15,intval,0,1,'lu\_brake');

**STRUCTURED STABILITY RADII**

**(SUBJECT TO HERMITIAN PERTURBATIONS OF R)**

**SMALL-SCALE COMPUTATION OF (R;B)**

% The data is available under the directory data/structured,small

% This is also the small-scale example in Section 5.1 of [1].

% J,R,Q are 20x20, B is 20x2, rank (R) = 5

>> load random20by20.mat;

>> [f, z, parsout] = DHradiiJR\_Hermit\_ss(J,Q,R,B,intval);

% *f* contains (R;B)

% *z* is such that i\*z is the first point on the imaginary axis that is attained under

% smallest Hermitation perturbation of R of the form R + .

**LARGE-SCALE COMPUTATION OF (R;B)**

**(WHEN IS AVAILABLE RATHER THAN )**

% The data set is available under the directory data/structured,large

% J,R,Q are 2000x2000, B is 2000x2

>> load random2000x2000\_1.mat;

>> [f,z,info] = DHradiiJR\_Hermit(J,Q,R,B,1,5,5,intval);

% *f* and *z* are the same as the for the small-scale problems.

**LARGE-SCALE COMPUTATION OF (R;B)**

**(WHEN IS AVAILABLE RATHER THAN )**

% Once again the brake-squealing problem

>> load BrakeSqueal9338x9338.mat;

>> omega = 2.5;

>> J = [-omega\*DG -KE-omega^2\*Kg; (KE+omega^2\*Kg)' zeros(size(DG))];

>> Zn = zeros(4669,4669);

>> R = [DM + (1/omega)\*DR Zn; Zn Zn];

>> Qinv = [M Zn; Zn KE+omega^2\*Kg];

>>[f,z,info] = DHradiiJR\_Hermit\_Qinv(J,Qinv,R,B,2,15,15,intval,0,1,'lu\_brake');

**References:**

**[1]** N. Aliyev, V. Mehrmann and E. Mengi. Computation of Stability Radii for Large-Scale Dissipative Hamiltonian Systems. arXiv preprint arXiv:1808.03574v2 [math.NA]