Assignment 10

1. **(a)** Implement the Jacobi–Davidson method for the solution of a *generalized eigenvalue problem* $A\boldsymbol{x} = \lambda M\boldsymbol{x}$ in a MATLAB function. Assume that A and M are *real, symmetric* $n \times n$ matrices with M *positive definite.*

Input:

- Matrices A and M.
- Number p of eigenpairs to calculate.
- Starting vector $\boldsymbol{x}$.
- Target value τ.
- Tolerance ε for the stopping criterion $\|A\tilde{\mathbf{x}} - \tilde{\vartheta}M\tilde{\mathbf{x}}\| \leq \varepsilon$.
- 'Extreme' dimensions $j_{\min}$ and $j_{\max}$ of the search space.

Output:

- The p eigenvalues and eigenvectors.

Hint: Start from the code in `jacobi_davidson_qr.m` on the web page. In this m-file, a full-fledged Jacobi–Davidson algorithm is implemented for the special eigenvalue problem.

Hint: Solve the correction equation *explicitly* (Backslash operator).

(b) In `jacobi_davidson_qr.m`, the correction equation is solved in a crude way by means of the backslash operator. Improve the approach by using the backslash operator only for solving with $A - \vartheta I$.

(c) Test your method on the acoustic eigenfrequencies in the interior of a car, a problem we looked at before. The matrices A and M can be generated using the MATLAB function `generate_matrices`. It takes the required number of refinements of the mesh as an argument.

One refinement step is good enough. This gives an eigenvalue problem of size 298 (solving the correction equation explicitly does not allow big system sizes).

Compute the 8 smallest eigenvalues and associated eigenvectors, choose $\tau = 0.01$, $\varepsilon = 10^{-8}$, $j_{\min} = 10$, $j_{\max} = 20$ and a random starting vector.

2. Consider the Jacobi-Davidson method described in Algorithm 12.2 in the lecture notes. In the correction equation, choose a constant shift τ instead of θ_j.

Show that the columns of V_m are an orthogonal basis of the Krylov subspace $\mathcal{K}_j(\boldsymbol{t}, (A - \tau I)^{-1})$.

Please submit your solution via e-mail to Peter Arbenz (`arbenz@inf.ethz.ch`) by May 22, 2018. (12:00). Please specify the tag **EWP18-10** in the subject field.

Algorithm 1 The Jacobi–Davidson QR algorithm to compute p of the eigenvalues of a symmetric matrix closest to a target value τ

/ Initializations */*
$Q_0 := []; \quad k = 0.$
Choose $\mathbf{v}_1$ with $\|\mathbf{v}_1\| = 1$.
$\mathbf{w}_1 = A\mathbf{v}_1; \quad H_1 := \mathbf{v}_1^*\mathbf{w}_1; \quad V_1 := [\mathbf{v}_1]; \quad W_1 := [\mathbf{W}_1];$
$\tilde{\mathbf{q}} = \mathbf{v}_1; \quad \tilde{\theta} = \mathbf{v}_1^*\mathbf{w}_1; \quad \mathbf{r} := \mathbf{w}_1 - \tilde{\theta}\tilde{\mathbf{q}},\ j := 1;$

/ Iteration: increase V by one column vector each iteration */*
while $k < p$
Approximatively solve the correction equation

$$(I - \tilde{Q}_k\tilde{Q}_k^*)(A - \tilde{\theta}I)(I - \tilde{Q}_k\tilde{Q}_k^*)\boldsymbol{t} = -\boldsymbol{r}_j, \qquad \tilde{Q}_k^*\boldsymbol{t} = \mathbf{0}.$$

where $\tilde{Q}_k = [Q_k, \tilde{\mathbf{q}}]$.
$\tilde{\mathbf{v}}_j := (I - V_{j-1}V_{j-1}^*)\boldsymbol{t} \quad \boldsymbol{v}_j := \tilde{\mathbf{v}}_j/\|\tilde{\mathbf{v}}_j\|, \quad \tilde{Q}_k^*\boldsymbol{v}_j = \mathbf{0}, \quad V_{j-1}^*\boldsymbol{v}_j = \mathbf{0}.$
$V_j := [V_{j-1}, \boldsymbol{v}_j], \quad \boldsymbol{w}_j = A\boldsymbol{v}_j; \quad H_j = \begin{bmatrix} H_{j-1} & V_{j-1}^*\boldsymbol{w}_j \\ \boldsymbol{v}_j^*W_{j-1} & \boldsymbol{v}_j^*\boldsymbol{w}_j \end{bmatrix}; \quad W_j = [W_{j-1}, \boldsymbol{w}_j].$

Compute the eigenvalue decomposition of H_j:
$H_jS^{(j)} = S^{(j)}\mathrm{diag}(\theta_1^{(j)}, \ldots, \theta_j^{(j)}),$
with the Ritz values $\theta_i^{(j)}$ sorted according to their distance to τ.

/ Test for converged eigenpairs */*
repeat
$\tilde{\theta} = \theta_1^{(j)}; \quad \tilde{\mathbf{q}} = V_j\boldsymbol{s}_1^{(j)}; \quad \tilde{\mathbf{w}} = W_j\boldsymbol{s}_1^{(j)}; \quad \boldsymbol{r} = \tilde{\mathbf{w}} - \tilde{\theta}\tilde{\mathbf{q}}$
if $\|\mathbf{r}\| < \varepsilon$
$Q_{k+1} := [Q_k, \tilde{\mathbf{q}}]; \quad \lambda_{k+1} := \tilde{\theta}; \quad k := k+1;$
$V_{j-1} = V_j[\mathbf{s}_2^{(j)}, \ldots, \mathbf{s}_j^{(j)}]; \quad W_{j-1} = W_j[\mathbf{s}_2^{(j)}, \ldots, \mathbf{s}_j^{(j)}]; \quad H_{j-1} = \mathrm{diag}(\theta_2^{(j)}, \ldots \theta_j^{(j)});$
$S = I_{j-1}, \quad \theta_i^{(j)} = \theta_{i+1}^{(j)}, \quad i = 1, \ldots, j-1$
end if
until $\|\mathbf{r}\| > \varepsilon$

/ Deflate V */*
if $j = j_{\max}$
$V_{j_{\min}} := V_j[\mathbf{s}_1, \ldots, \mathbf{s}_{\min}]; \quad W_{j_{\min}} := W_j[\mathbf{s}_1, \ldots, \mathbf{s}_{\min}];$
$H_{j_{\min}} := \mathrm{diag}(\theta_1^{(j)}, \ldots, \theta_j^{(j)}) \quad j := j_{\min}$
end if
end while

Hints: Make sure H_j is *exactly* symmetric; otherwise, MATLAB may compute complex Ritz values. In the correction equation, use $(A - \tau I)$ instead of $(A - \tilde{\theta}I)$ until convergence starts, i.e. until $\|\mathbf{r}_j\| < 10^{-3}$.

Please submit your solution via e-mail to Peter Arbenz (`arbenz@inf.ethz.ch`) by May 22, 2018. (12:00). Please specify the tag **EWP18-10** in the subject field.