Problem set 9 - solutions

Problem 1

The corresponding integral formulation of the governing equation is given by

$$\int_0^\ell \left[\frac{\partial}{\partial x}\left(GI_p\frac{\partial\varphi}{\partial x}\right)\right]\delta\varphi\,dx = \int_0^\ell \rho I_p\ddot{\varphi}\,\delta\varphi\,dx \qquad \forall\,\delta\varphi,$$

where $\delta\varphi$ denotes the virtual rotation. Applying integration by parts to the left-hand side yields

$$-\int_0^\ell GI_p\left(\frac{\partial\varphi}{\partial x}\right)\left(\frac{\partial\delta\varphi}{\partial x}\right)dx + \left[GI_p\left(\frac{\partial\varphi}{\partial x}\right)\delta\varphi\right]_0^\ell = \int_0^\ell \rho I_p\ddot{\varphi}\,\delta\varphi\,dx \qquad \forall\,\delta\varphi.$$

By imposing natural boundary conditions that eliminate the boundary term, the expression simplifies to

$$\int_0^\ell GI_p\left(\frac{\partial\varphi}{\partial x}\right)\left(\frac{\partial\delta\varphi}{\partial x}\right)dx + \int_0^\ell \rho I_p\ddot{\varphi}\,\delta\varphi\,dx = 0 \qquad \forall\,\delta\varphi.$$

This yields the weak form of the problem: find $\varphi \in \mathcal{U}$ such that

$$\int_0^\ell GI_p\left(\frac{\partial\varphi}{\partial x}\right)\left(\frac{\partial\delta\varphi}{\partial x}\right)dx + \int_0^\ell \rho I_p\ddot{\varphi}\,\delta\varphi\,dx = 0 \qquad \forall\,\delta\varphi \in \mathcal{V},$$

with the function spaces defined as

$$\mathcal{U} = \{\varphi(\cdot,t) \in H^1(]0,\ell[)\ \forall t \in]0,T[\},$$
$$\mathcal{V} = \{\delta\varphi \in H^1(]0,\ell[)\}.$$

The entries k_{ij} and m_{ij} of the elemental stiffness and mass matrices $\mathbf{K}$ and $\mathbf{M}$, respectively, are given by:

$$k_{ij} = \int_0^\ell GI_p\left(\frac{dh_i}{dx}\right)\left(\frac{dh_j}{dx}\right)dx,$$
$$m_{ij} = \int_0^\ell \rho I_p h_i h_j\,dx,$$

where $h_1(x) = 1 - x/\ell$ and $h_2(x) = x/\ell$ are the standard linear shape functions defined over the interval $[0,\ell]$. For a two-node finite element of length ℓ, these integrals lead to the following matrices:

$$\mathbf{K} = \frac{GI_p}{\ell}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} \quad \text{and} \quad \mathbf{M} = \frac{\rho\ell I_p}{6}\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix}.$$

The corresponding Rayleigh quotient is expressed as

$$\mathcal{R}(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{K}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{M}\mathbf{w}} = \frac{3G}{\rho\ell^2}\frac{(w_1-w_2)^2}{w_1^2+w_1w_2+w_2^2},$$

where $\mathbf{w} = [w_1,\, w_2]^T$ denotes the vector of nodal values, either of the rotation field φ or the virtual rotation $\delta\varphi$. To reduce the number of variables, we normalize by setting $w_1 = 1$, which is admissible since the Rayleigh quotient is homogeneous. This leads to the reduced form:

$$\mathcal{R}\left(\begin{bmatrix}1\\ w_2\end{bmatrix}\right) = \frac{3G}{\rho\ell^2}\frac{(1-w_2)^2}{w_2^2+w_2+1}.$$

An approximation of the natural frequencies can be obtained by identifying the stationary points of $\mathcal{R}(w_2)$. Solving

$$\frac{d\mathcal{R}}{dw_2} = \frac{3G}{\rho\ell^2}\left(\frac{2(w_2-1)}{w_2^2+w_2+1} - \frac{(2w_2+1)(w_2-1)^2}{(w_2^2+w_2+1)^2}\right) = 0,$$

yields two critical points: $w_2 = 1$ and $w_2 = -1$, corresponding to the Rayleigh quotient values:

$$\mathcal{R}\left(\begin{bmatrix}1\\ 1\end{bmatrix}\right) = 0, \qquad \mathcal{R}\left(\begin{bmatrix}1\\ -1\end{bmatrix}\right) = \frac{12G}{\rho\ell^2}.$$

The vanishing of the first computed frequency is consistent with the presence of a rigid-body rotation mode in a free-free torsional shaft. This result confirms that the formulation correctly captures the underlying physical behavior, including the absence of restoring torque for uniform rotation. The second frequency provides an estimate of the first non-zero torsional natural frequency:

$$\omega_2^{\text{approx}} = \frac{\sqrt{12}}{\ell}\sqrt{\frac{G}{\rho}},$$

to be compared with the exact value for a free-free shaft:

$$\omega_2^{\text{exact}} = \frac{\pi}{\ell}\sqrt{\frac{G}{\rho}}.$$

The relative error between the approximate and exact values is given by

$$\text{Relative error} = \left|\frac{\sqrt{12}-\pi}{\pi}\right| \approx 0.102,$$

which corresponds to approximately 10.2%. This level of accuracy is typical for low-order finite element approximations and validates the adopted formulation and shape functions.

Problem 2

1. Formulation of the eigenvalue problem. We consider the longitudinal vibrations of a uniform free-free bar. The bar is discretized using a single quadratic finite element with three equally spaced nodes. Its dynamic behavior is described by the generalized eigenvalue problem:

$$\mathbf{K}\mathbf{p} = \lambda\mathbf{M}\mathbf{p},$$

where $\lambda = \omega^2$ are the squared natural frequencies and $\mathbf{p}$ are the corresponding mode shapes. The stiffness and lumped mass matrices are defined as:

$$\mathbf{K} = \frac{EA}{3\ell}\begin{bmatrix}7 & -8 & 1\\ -8 & 16 & -8\\ 1 & -8 & 7\end{bmatrix}, \quad \mathbf{M} = \frac{\rho A\ell}{6}\cdot\text{diag}(1,4,1).$$

```
% Parameters (symbolic for generality, assign numerical values as needed)
clear
syms E A rho l real

% Exact eigenvalues
Lambda_exact = [2*E / (rho * l^2); 14*E / (rho * l^2)];

% Exact eigenvectors (modes)
p1 = [-1; -1; -1];
p2 = [-1; 0; 1];

P_exact = [p1, p2];

% Stiffness matrix K
K = (E*A)/(3*l) * [  7  -8   1;
        -8  16  -8;
        1  -8   7 ];

% Lumped mass matrix M
M = (rho*A*l)/6 * diag([1, 4, 1]);
```

2. Spectral shift for rigid-body mode. To improve convergence, we apply a spectral shift to eliminate the rigid-body mode:

$$\mathbf{K}_\sigma = \mathbf{K} + \sigma \mathbf{M}, \quad \text{with} \quad \sigma = \frac{2E}{\rho \ell^2}.$$

```
% Spectral shift
sigma = 2*E / (rho*l^2);

% Shifted stiffness matrix K_sigma
K_sigma = K + sigma * M;
```

The symbolic matrices are evaluated numerically for specific material and geometric properties.

```
% Numerical values
E_val = 210e9;
A_val = 4e-4;
rho_val = 7850;
l_val = 0.5;

% Numerical evaluation
K_num = double(subs(K_sigma, {E, A, rho, l}, {E_val, A_val, rho_val, l_val}));
M_num = double(subs(M, {E, A, rho, l}, {E_val, A_val, rho_val, l_val}));
Lambda_exact_num = double(subs(Lambda_exact, {E, A, rho, l}, {E_val, A_val,
        rho_val, l_val}));
P_exact_num = double(subs(P_exact, {E, A, rho, l}, {E_val, A_val, rho_val, l_val}));
```

3. Subspace iteration algorithm. The subspace iteration algorithm computes a subspace spanned by approximate eigenvectors. We perform three iterations of the subspace iteration algorithm starting from an initial guess $\mathbf{P}_0$. Each iteration involves:

1. Solving $\mathbf{K}_\sigma \overline{\mathbf{P}} = \mathbf{M}\mathbf{P}$
2. Projecting the problem onto the subspace: $\mathbf{K}_{proj}$ and $\mathbf{M}_{proj}$.
3. Solving the reduced eigenproblem: $\mathbf{\Lambda}_\sigma$ and $\mathbf{Z}$,
4. Reconstructing Ritz vectors: $\mathbf{P}$.

Results are stored in cells P, Lambda, etc. The vectors are sorted to maintain order, and normalization is applied in the mass norm.

```
    % Initial guess matrix
P0 = (1/sqrt(rho*A*l)) * [1 0; 0 1; 0 0];
P0_num = double(subs(P0, {E, A, rho, l}, {E_val, A_val, rho_val, l_val}));

% Definiton of matrices needed in the subspace iteration
P = cell(1, 3);             % Store the approximated modal matrices
Pbar = cell(1, 3);          % Store the approximated non-orthogonal matrices
Lambda_sigma = cell(1, 3);  % Store the approximated offset spectral matrices
Lambda = cell(1, 3);        % Store the approximated spectral matrices
Z = cell(1, 3);             % Store the approximated projected modal matrices
K_proj = cell(1, 3);        % Store the projected stiffness matrices
M_proj = cell(1, 3);        % Store the projected mass matrices
% Start subspace iteration
for i = 1:3
    if i == 1
        Pbar{i} = K_num \ (M_num * P0_num);
    else
        Pbar{i} = K_num \ (M_num * P{i-1});
    end

    % Rayleigh matrices
    K_proj{i} = transpose(Pbar{i}) * K_num * Pbar{i};
    M_proj{i} = transpose(Pbar{i}) * M_num * Pbar{i};

    % Solve reduced eigenvalue problem
    [Z{i}, Lambda_sigma{i}] = eig(K_proj{i}, M_proj{i});

    % Update P (Ritz vectors)
    P{i} = Pbar{i} * Z{i};

    % Normalize mode shapes
    P{i} = P{i} / sqrt(transpose(P{i}) * M_num * P{i});  % M_num-norm normalization
```

```
    % Undo the spectral shift and consider only the real part
    Lambda{i} = real(diag(Lambda_sigma{i}) - double(subs(sigma, {E, rho, l},
            {E_val, rho_val, l_val})));
    Lambda_sigma{i} = real(diag(Lambda_sigma{i}));

    % Sort eigenvalues and apply permutation to eigenvectors
    [Lambda{i}, idx] = sort(Lambda{i});
    Lambda_sigma{i} = sort(Lambda_sigma{i});
    P{i} = real(P{i}(:, idx));
end
```

4) Comparison with exact solutions. Exact eigenvalues and eigenvectors (up to scale) are known. We compare:

- The eigenvalue approximation using relative error,
- The eigenvector approximation using the error in the **M**-norm.
- The Ritz values converge to the exact ones and the approximate modes align with the analytical solutions.

```
    %Initialize error arrays
rel_error_lambda = cell(1,3);
modal_error = cell(1,2);

% Loop over each iteration
for j = 1:3
    % --- Relative error in eigenvalues ---
    rel_error_lambda{j} = abs(Lambda_sigma{j} - Lambda_exact_num)
            ./ abs(Lambda_exact_num);

    % Loop over modes
    for m = 1:2
        % --- Modal error in M-norm ---
        p_diff = P{j}(:,m) - P_exact_num(:,m);
        modal_error{j}(m) = sqrt(p_diff' * M_num * p_diff);
    end
end

for j = 1:3
    fprintf('\n=======================================================\n');
    fprintf('\n--- Iteration %d ---\n', j);

    fprintf('Relative Error in Eigenvalues:\n');
    disp(rel_error_lambda{j});

    fprintf('Modal Error (M-norm):\n');
    disp(modal_error{j});
end
```

The first eigenmode corresponds to a rigid-body motion, which has a zero natural frequency. The associated mode shape is a constant vector (e.g., $[1; 1; 1]$ or $[-1; -1; -1]$), and it spans a one-dimensional eigenspace. Since eigenvectors are only defined up to a multiplicative constant—including sign—the subspace iteration algorithm may converge to either the positive or the negative of the exact mode shape.

When computing the modal error using the $\mathbf{M}$-norm (i.e., comparing the approximated and exact vectors directly), this ambiguity can result in misleading error values. Even though the vectors represent the same physical mode, their difference can have a nonzero norm if the signs differ. Specifically, the error between a normalized vector and its negative is twice the norm of the vector. As a result, the modal error for the zero-frequency mode may appear to oscillate across iterations or fail to decrease monotonically, even though the convergence is mathematically correct. This behavior is not a flaw of the numerical method but a consequence of normalization and sign ambiguity. To mitigate this, one can minimize the error with respect to sign or explicitly align the direction of the approximated eigenvector with the reference one.