General instructions

All pages of your answer sheets must contain your name and your student number. Number each page of your answer sheets and indicate the total number of pages used. Write clearly and in English. Points for each question are in the left margin brackets.

This question sheet **must be returned** before leaving.

Questions

[25%] 1. Suppose you are working with a model for which the internal force vector can be written as

$$\mathbf{f}^{\text{int}} = \int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma} \, \mathrm{d}\Omega$$

Assume the existence of a linear stress-strain relation $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$ while $\boldsymbol{\varepsilon}$ is a nonlinear differentiable function of the vector with nodal degrees of freedom $\mathbf{a}$ with $\mathbf{B}(\mathbf{a}) = \partial\boldsymbol{\varepsilon}/\partial\mathbf{a}$.

Derive an eigenvalue problem of the shape $\det\left(\mathbf{K}_1 + \lambda\mathbf{K}_2\right) = 0$ that can be solved to identify the buckling load.

Solution: 5 steps, 5% each:

- State $\delta\mathbf{f}^{\text{int}} = 0$
- Definition of $\mathbf{K} = \frac{\partial\mathbf{f}^{\text{int}}}{\partial\mathbf{a}}$
- Product rule of differentiation to arrive at two parts
- Apply chain rule on $\boldsymbol{\sigma}$ $\quad\rightarrow\quad$ $\mathbf{K}_1 = \int \mathbf{B}^T\mathbf{D}\mathbf{B} \, \mathrm{d}\Omega$
- Unit force analysis $\boldsymbol{\sigma} = \lambda\hat{\boldsymbol{\sigma}}$ $\quad\rightarrow\quad$ $\mathbf{K}_2 = \int \frac{\partial\mathbf{B}}{\partial\mathbf{a}}\hat{\boldsymbol{\sigma}} \, \mathrm{d}\Omega$

2. The governing equations for the Timoshenko beam formulation are

$$-EI\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} - GA_{\mathrm{s}}\left(\frac{\mathrm{d}w}{\mathrm{d}x} - \phi\right) = 0$$
$$GA_{\mathrm{s}}\left(\frac{\mathrm{d}^2w}{\mathrm{d}x^2} - \frac{\mathrm{d}\phi}{\mathrm{d}x}\right) + q = 0$$

where ϕ and w are the unknown fields representing rotation and deflection, GA_{s} and EI are cross-sectional stiffnesses and q is a distributed load. The first equation originates from rotational equilibrium and the second equation from translational equilibrium.

$$\begin{bmatrix} \mathbf{K}_{\phi\phi} & \mathbf{K}_{\phi w} \\ \mathbf{K}_{w\phi} & \mathbf{K}_{ww} \end{bmatrix} \begin{bmatrix} \mathbf{a}_{\phi} \\ \mathbf{a}_{w} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{\phi} \\ \mathbf{f}_{w} \end{bmatrix}$$

Assume both fields ϕ and w are interpolated with a different set of shape functions $\phi = \mathbf{N}_\phi \mathbf{a}_\phi$ and $w = \mathbf{N}_w \mathbf{a}_w$.

[30%] (a) Derive an expression for each of the sub-matrices where the different interpolations are used in combination with the Bubnov-Galerkin method.

Solution: 6 steps, 5% each

- Multiply with weights ($\bar{\phi}$ for first equation, $\bar{w}$ for second) and integrate
- Integration by parts
- Introduce discretization, consistently use $\mathbf{N}_\phi$ and $\mathbf{N}_w$
- Define $\mathbf{B}_w = \frac{\partial \mathbf{N}_w}{\partial x}$ and $\mathbf{B}_\phi = \frac{\partial \mathbf{N}_\phi}{\partial x}$
- Take amplitudes and dofs out of the integrals
- Correct submatrices

$$\mathbf{K}_{\phi\phi} = \int_{\Omega^e} \mathbf{B}_\phi^{\mathrm{T}} EI \mathbf{B}_\phi + \mathbf{N}\phi^{\mathrm{T}} GA_s \mathbf{N}\phi \, \mathrm{d}\Omega \tag{1}$$

$$\mathbf{K}_{\phi w} = -\int_{\Omega^e} \mathbf{N}_\phi^{\mathrm{T}} GA_s \mathbf{B}_w \, \mathrm{d}\Omega \tag{2}$$

$$\mathbf{K}_{w\phi} = -\int_{\Omega^e} \mathbf{B}_w^{\mathrm{T}} GA_s \mathbf{N}_\phi \, \mathrm{d}\Omega \tag{3}$$

$$\mathbf{K}_{ww} = \int_{\Omega^e} \mathbf{B}_w^{\mathrm{T}} GA_s \mathbf{B}_w \, \mathrm{d}\Omega \tag{4}$$

The question was only about the matrix, boundary terms do not need to be handled

[5%] (b) What is the size of each of the matrices $\mathbf{K}^{\mathrm{e}}_{\phi\phi}, \mathbf{K}^{\mathrm{e}}_{\phi w}, \mathbf{K}^{\mathrm{e}}_{w\phi}, \mathbf{K}^{\mathrm{e}}_{ww}$ if an element has quadratic interpolation for w and linear interpolation for ϕ

Solution: $\mathbf{K}^{\mathrm{e}}_{\phi\phi} : [2 \times 2], \mathbf{K}^{\mathrm{e}}_{\phi w} : [2 \times 3], \mathbf{K}^{\mathrm{e}}_{w\phi} : [3 \times 2], \mathbf{K}^{\mathrm{e}}_{ww} : [3 \times 3]$

[5%] (c) Explain how this choice for mixed interpolation prevents shear locking

Solution: Shear locking exists when pure bending deformation is affected by the shear stiffness. For pure bending, we need a linear relation for ϕ. To then have zero shear strain ($dw/dx - \phi = 0$), the variation in w must be quadratic.

3. Consider the following code snippet from `pyJive`:

```
1
2  # ...
3
4          model.take_action(act.GETMATRIX0, params, globdat)
5
6          model.take_action(act.GETEXTFORCE, params, globdat)
7
8          model.take_action(act.GETCONSTRAINTS, params, globdat)
9
10         Kc, fc = c.constrain(K, f)
11
12         smat = sparse.csr_matrix(Kc)
13         u = linalg.spsolve(smat, fc)
14
15         globdat[gn.STATE0] = u
16
17 # ...
18
19
```

and the following table with models and modules:

Model/module	Functionality
`BarModel`	Assembly of 1D equilibrium elements in extension
`FrameModel`	Assembly of Timoshenko elements with extension
`SolidModel`	Assembly of isoparametric continuum equilibrium elements
`DirichletModel`	Handles Dirichlet-type boundary conditions
`NeumannModel`	Handles Neumann-type boundary conditions
`SolverModule`	Solves linear quasi-static FE systems
`NonlinModule`	Newton-Raphson solver in load/displacement control
`ExplicitTimeModule`	Solves dynamic systems with Central Difference
`NewmarkModule`	Solves linear dynamic systems with the Newmark time stepper
`ArclenModule`	Newton-Raphson solver with arc-length control
`LinBuckModule`	Computes buckling loads and vibration modes
`ModeShapeModule`	Computes natural frequencies and modes
`NLNewmarkModule`	A Newmark time stepper with a Newton-Raphson loop

[5%] (a) From the list of models and modules above, pick the one that you believe this snippet comes from. Provide a short motivation for your choice;

Solution: SolverModule, the code solves a linear system of equations with the external force vector as right hand side

[5%] (b) Imagine Line 10 of the code above was omitted and we directly used `K` and `f` for the rest of the function. What would the consequence be for the obtained solution? Provide a physical explanation for your answer.

> **Solution:** The stiffness matrix would be singular and line 13 would not result in a meaningful result. Without Dirichlet boundary conditions (i.e. without any supports) there is no unique solution to the mechanical problem.

4. Consider again the list of `pyJive` models and modules from the previous question. For each of the modeling applications below, pick an appropriate set of models and modules to combine into a FEM simulation. Provide a short motivation for your choices.

[5%] (a) A quasi-static simulation of a simply-supported beam under distributed load, linear-elastic material;

> **Solution:** `FrameModel`, `DirichletModel` + `SolverModule` or
> `SolidModel`, `DirichletModel` + `SolverModule`

[5%] (b) An L-shaped domain in plane stress with two edges fixed and a point load applied to one of the corners, hyperelastic material;

> **Solution:** `SolidModel`, `NeumannModel`, `DirichletModel` + `NonlinModule`

[5%] (c) A simulation of a fiber pullout test: a very long fiber is pulled with a point load from a domain that can be modeled as an elastic support;

> **Solution:** `BarModel`, `DirichletModel`, `NeumannModel` + `SolverModule`

[5%] (d) The formation of plastic hinges along a column due to the propagation and reflection of stress waves;

> **Solution:** `FrameModel`, `DirichletModel`, (`NeumannModel`) + `NLNewmarkModule`

[5%] (e) The complete equilibrium path of a square-shaped domain in plane strain undergoing a combination of snap-throughs and snap-backs due to fracture. All DOFs are fixed on one end of the domain and a load is applied on the opposite end;

> **Solution:** `SolidModel`, `DirichletModel`, `NeumannModel` + `ArclenModule`