Linear Elasticity

Computer Session B5

Linear Elastostatics

The linear elastic problem for static equilibrium of a homogeneous isotropic body $\Omega \subset \mathbb{R}^2$ under the assumption of small deformations and strains reads: find the symmetric stress tensor $\boldsymbol{\sigma} = [\sigma_{ij}]_1^2$ and the displacement vector $\boldsymbol{u} = [u_i]_1^2$, such that

$$
\begin{aligned}
-\nabla \cdot \boldsymbol{\sigma} &= \boldsymbol{f}, && \text{in } \Omega && \text{(1a)}\\
\boldsymbol{\sigma} &= 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda(\nabla \cdot \boldsymbol{u})\boldsymbol{I}, && \text{in } \Omega && \text{(1b)}\\
\boldsymbol{u} &= \boldsymbol{0}, && \text{on } \Gamma_D && \text{(1c)}\\
\boldsymbol{\sigma} \cdot \boldsymbol{n} &= \boldsymbol{g}, && \text{on } \Gamma_N && \text{(1d)}
\end{aligned}
$$

Here, $\boldsymbol{f}$ is a given body force, and $\boldsymbol{g}$ a given traction load acting along a segment Γ_N of the boundary, which has outward unit normal $\boldsymbol{n}$. Along the rest of the boundary Γ_D the body is clamped and can not be displaced. The elastic properties of the body are governed by the positive constants λ and μ called the Lamé parameters. We imagine Ω to be the cross section of a long slender structure aligned along the x_3-axis. For such structures a state of plane strain is applicable, which essentially means that all loads and are confined to the x_1x_2-plane and that no quantities depend on x_3. Further, $\boldsymbol{\varepsilon}(\boldsymbol{u}) = [\varepsilon_{ij}]_1^2$ is the strain tensor with components

$$
\varepsilon_{ij}(\boldsymbol{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \quad i, j = 1, 2 \tag{2}
$$

The divergence of the 2×2 tensor $\boldsymbol{\sigma}$ and the 2×1 vector $\boldsymbol{u}$ is defined by

$$\nabla \cdot \boldsymbol{\sigma} = \left[\sum_{j=1}^{2} \frac{\partial \sigma_{ij}}{\partial x_j} \right]_{i=1}^{2}, \qquad \nabla \cdot \boldsymbol{u} = \sum_{i=1}^{2} \frac{\partial u_i}{\partial x_i} \tag{3}$$

Finally, $\boldsymbol{I}$ is the 2×2 identity matrix.

Problem 1. Given the stress field $\sigma_{11} = x_1 x_2$, $\sigma_{12} = (1 - x_2^2)/2$, and $\sigma_{22} = 0$. Determine if this corresponds to a state of equilibrium under a zero body force.

Problem 2. Show the vector identity $2\nabla \cdot \boldsymbol{\varepsilon}(\boldsymbol{v}) = \Delta \boldsymbol{v} + \nabla(\nabla \cdot \boldsymbol{v})$ for $\boldsymbol{v} = [v_1,\ v_2]$.

Problem 3. Use the previous result to rewrite (1) as the single equation $\mu \Delta \boldsymbol{u} + (\lambda + \mu)\nabla(\nabla \cdot \boldsymbol{u}) + \boldsymbol{f} = \boldsymbol{0}$. mit16.20 module 4!

Problem 4. Show that the strain tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$ is zero under the deformation

$$\boldsymbol{u} = \begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} 0 & -\theta \\ \theta & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where a, b, and θ are constants. Can you give a physical interpretation of $\boldsymbol{u}$, assuming that θ is small?

Variational Formulation

To obtain a variational formulation of (1) we multiply $-\nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f}$ with a test function $\boldsymbol{v} = (v_1, v_2)$ and integrate over Ω.

$$\int_\Omega -(\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, dx = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \, dx \tag{4}$$

Introducing the notation $\boldsymbol{q}_1 = (\sigma_{11}, \sigma_{12})$ and $\boldsymbol{q}_2 = (\sigma_{21}, \sigma_{22})$ we can write

$$(\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} = \sum_{i,j=1}^{2} \frac{\partial \sigma_{ij}}{\partial x_j} v_i = \nabla \cdot \boldsymbol{q}_1 v_1 + \nabla \cdot \boldsymbol{q}_2 v_2 \tag{5}$$

Integration by parts then yields

$$\int_{\Omega} (\nabla \cdot \boldsymbol{q}_1 v_1 + \nabla \cdot \boldsymbol{q}_2 v_2)\, dx$$
$$= \int_{\partial\Omega} \boldsymbol{n} \cdot (\boldsymbol{q}_1 v_1 + \boldsymbol{q}_2 v_2)\, ds - \int_{\Omega} (\boldsymbol{q}_1 \cdot \nabla v_1 + \boldsymbol{q}_2 \cdot \nabla v_2)\, dx \tag{6}$$

We next note that

$$\boldsymbol{n} \cdot \sum_{i=1}^{2} \boldsymbol{q}_i v_i = \sum_{i,j=1}^{2} n_j \sigma_{ij} v_i = (\boldsymbol{\sigma} \cdot \boldsymbol{n}) \cdot \boldsymbol{v} \tag{7}$$

Using the symmetry of the stress tensor $\boldsymbol{\sigma}$ we have

$$\begin{aligned}
\boldsymbol{q}_1 \cdot \nabla v_1 + \boldsymbol{q}_2 \cdot \nabla v_2 &= \sigma_{11}\frac{\partial v_1}{\partial x_1} + \sigma_{12}\frac{\partial v_1}{\partial x_2} + \sigma_{21}\frac{\partial v_2}{\partial x_1} + \sigma_{22}\frac{\partial v_2}{\partial x_2} && (8)\\
&= \sigma_{11}\frac{1}{2}\left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_1}{\partial x_1}\right) && (9)\\
&\quad + 2\sigma_{12}\frac{1}{2}\left(\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1}\right)\\
&\quad + \sigma_{22}\frac{1}{2}\left(\frac{\partial v_2}{\partial x_2} + \frac{\partial v_2}{\partial x_2}\right)\\
&= \sum_{i,j=1}^{2} \sigma_{ij}(\boldsymbol{u})\varepsilon_{ij}(\boldsymbol{v}) && (10)\\
&\equiv \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) && (11)
\end{aligned}$$

Thus we end up with the variational equation

variational formulation
$$\underline{\int_{\Omega} \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\, dx = \int_{\Omega} \boldsymbol{f} \cdot \boldsymbol{v}\, dx + \int_{\partial\Omega} \boldsymbol{g} \cdot \boldsymbol{v}\, ds} \tag{12}$$

Formally, (12) reads: find $\boldsymbol{u} \in \mathcal{V} = \{\boldsymbol{v} \in [H^1(\Omega)]^2 : \boldsymbol{v}|_{\Gamma_D} = \boldsymbol{0}\}$, such that

$$a(\boldsymbol{u}, \boldsymbol{v}) = l(\boldsymbol{v}), \quad \forall \boldsymbol{v} \in \mathcal{V} \tag{13}$$

where we have introduced the linear forms $a(\cdot,\cdot)$ and $l(\cdot)$, defined by

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_{\Omega} \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\, dx \tag{14}$$
$$l(\boldsymbol{v}) = \int_{\Omega} \boldsymbol{f} \cdot \boldsymbol{v}\, dx + \int_{\Gamma_N} \boldsymbol{g} \cdot \boldsymbol{v}\, ds \tag{15}$$

Problem 5. Show that $\boldsymbol{\varepsilon}(\boldsymbol{v}) : I = \nabla \cdot \boldsymbol{v}$.

Problem 6. Show that the bilinear form (14) can be written

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) + \lambda(\nabla \cdot \boldsymbol{u})(\nabla \cdot \boldsymbol{v})\, dx$$

might be helpful later

Problem 7. Verify that the conditions for the Lax-Milgram lemma are satisfied for the variational equation (13). For simplicity, you only have to consider the case of homogeneous Dirichlet boundary conditions $\boldsymbol{u} = \boldsymbol{0}$ on the whole boundary $\partial\Omega$. *Hint:* Korn's inequality is useful.

Finite Element Approximation

The finite element approximation to (13) reads: find $\boldsymbol{U} \in \mathcal{V}_h$, such that

$$a(\boldsymbol{U}, \boldsymbol{v}) = l(\boldsymbol{v}), \quad \forall \boldsymbol{v} \in \mathcal{V}_h \tag{16}$$

where $\mathcal{V}_h$ is a finite dimensional subspace of $\mathcal{V}$.

To be more specific, let $\mathcal{K}_h = \{K\}$ be a partition of Ω into shape regular triangles K of size h_K, and let $\mathcal{V}_h$ be the space of continuous piecewise linear vector polynomials on $\mathcal{K}$. That is,

$$\mathcal{V}_h = \{\boldsymbol{v} \in \mathcal{V} : \ \boldsymbol{v}|_K = [\mathcal{P}_1(K)]^2, \ \forall K \in \mathcal{K}\} \tag{17}$$

degree! = hat

where $\mathcal{P}_1(K)$ denotes the space of polynomials of degree 1 on K. With this choice of finite element space $\mathcal{V}_h$, a set of basis functions is given by the columns of the matrix

in 2D so… rows corresponds to different spatial axis

$$\boldsymbol{\varphi} = \begin{bmatrix} \varphi_1 & 0 & \varphi_2 & 0 & \dots & \varphi_N & 0 \\ 0 & \varphi_1 & 0 & \varphi_2 & \dots & 0 & \varphi_N \end{bmatrix} \tag{18}$$

where φ_i, $i = 1, 2, \dots, N$, are the usual hat functions and N the number of nodes within the mesh. In other words there are $2N$ bases and each base is a 2×1 vector. Moreover, every second base takes the form $[\varphi_i, \ 0]^T$ and every other $[0, \ \varphi_i]^T$.

Taking a linear combination of these bases we can write $\boldsymbol{U}$ as the matrix-vector multiplication

$$\boldsymbol{U} = \begin{bmatrix} \varphi_1 & 0 & \varphi_2 & 0 & \dots & \varphi_N & 0 \\ 0 & \varphi_1 & 0 & \varphi_2 & \dots & 0 & \varphi_N \end{bmatrix} \begin{bmatrix} \xi_1 \\ \eta_1 \\ \xi_2 \\ \eta_2 \\ \vdots \\ \xi_N \\ \eta_N \end{bmatrix} = \boldsymbol{\varphi d} \tag{19}$$

where the components d_i, $i = 1, 2, \dots, 2N$, of the vector $\boldsymbol{d}$ are the unknown nodal displacements that we wish to determine. Needless to say, this is just a simple way of writing $\boldsymbol{U} = [U_1,\ U_2]^T$ with $U_1 = \sum_{i=1}^N \xi_i \varphi_i$ and $U_2 \sum_i \eta_{i=1}^N \varphi_i$.

Computer Implementation

In finite element programming it is customary to rewrite the bilinear form (14) using matrix algebra since this simplifies the book keeping of the hat basis functions.

The starting point is to rearrange the components of the strain tensor into a vector $\boldsymbol{\epsilon}(\boldsymbol{u}) = [\varepsilon_{11},\ \varepsilon_{22},\ \gamma_{12}]^T$, where $\gamma_{12} = 2\varepsilon_{12}$ is called the engineering shear strain. Obviously,

$$\boldsymbol{\epsilon}(\boldsymbol{u}) = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \gamma_{12} \end{bmatrix} = \begin{bmatrix} \partial/\partial x_1 & 0 \\ 0 & \partial/\partial x_2 \\ \partial/\partial x_2 & \partial/\partial x_1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{20}$$

Using $\boldsymbol{\epsilon}(\boldsymbol{u})$ we can also express the components of the stress tensor $\boldsymbol{\sigma}$ as

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \begin{bmatrix} \lambda + 2\mu & \lambda & 0 \\ \lambda & \lambda + 2\mu & 0 \\ 0 & 0 & \mu \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \gamma_{12} \end{bmatrix} = \boldsymbol{D\epsilon}(\boldsymbol{u}) \tag{21}$$

Hence, we have

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \boldsymbol{\epsilon}^T(\boldsymbol{v}) \boldsymbol{D\epsilon}(\boldsymbol{u})\, dx \tag{22}$$

Now, introducing the notation

$$\boldsymbol{B} = \begin{bmatrix} \partial/\partial x_1 & 0 \\ 0 & \partial/\partial x_2 \\ \partial/\partial x_2 & \partial/\partial x_1 \end{bmatrix} \boldsymbol{\varphi} \tag{23}$$

$$= \begin{bmatrix} \partial\varphi_1/\partial x_1 & 0 & \partial\varphi_2/\partial x_1 & 0 & \dots & \partial\varphi_N/\partial x_1 & 0 \\ 0 & \partial\varphi_1/\partial x_2 & 0 & \partial\varphi_2/\partial x_2 & \dots & 0 & \partial\varphi_N/\partial x_2 \\ \partial\varphi_1/\partial x_2 & \partial\varphi_1/\partial x_1 & \partial\varphi_2/\partial x_2 & \partial\varphi_2/\partial x_1 & \dots & \partial\varphi_N/\partial x_2 & \partial\varphi_N/\partial x_1 \end{bmatrix} \tag{24}$$

we can write the strain of the finite element solution as $\boldsymbol{\epsilon}(\boldsymbol{U}) = \boldsymbol{B}\boldsymbol{d}$.

Collecting these results the finite element method (16) can now be written as the matrix equation

$$\left(\int_\Omega \boldsymbol{B}^T \boldsymbol{D} \boldsymbol{B} \, dx \right) \boldsymbol{d} = \int_\Omega \boldsymbol{\varphi}^T \boldsymbol{f} \, dx + \int_{\Gamma_N} \boldsymbol{\varphi}^T \boldsymbol{g} \, ds \tag{25}$$

or simply

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{F} \tag{26}$$

where $\boldsymbol{K}$ is the $2N \times 2N$ stiffness matrix

$$\boldsymbol{K} = \int_\Omega \boldsymbol{B}^T \boldsymbol{D} \boldsymbol{B} \, dx \tag{27}$$

and $\boldsymbol{F}$ is the $2N \times 1$ load vector

$$\boldsymbol{F} = \int_\Omega \boldsymbol{\varphi}^T \boldsymbol{f} \, dx + \int_{\Gamma_N} \boldsymbol{\varphi}^T \boldsymbol{g} \, ds \tag{28}$$

Assembly of the Stiffness Matrix and the Load Vector. As usual, the stiffness matrix $\boldsymbol{K}$ and the load vector $\boldsymbol{F}$ are assembled by summing elemental contributions from each element. Let us therefore consider a single triangle K with nodes $N_i = (x_1^{[i]}, x_2^{[i]})$, $i = 1, 2, 3$.

On K the element displacements are given by

$$\boldsymbol{U}^e = \begin{bmatrix} \varphi_1 & 0 & \varphi_2 & 0 & \varphi_3 & 0 \\ 0 & \varphi_1 & 0 & \varphi_2 & 0 & \varphi_3 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \eta_1 \\ \xi_2 \\ \eta_2 \\ \xi_3 \\ \eta_3 \end{bmatrix} = \boldsymbol{\varphi}^e \boldsymbol{d}^e \tag{29}$$

where φ_i, $i = 1, 2, 3$ are the usual hat functions

$$\varphi_i = \frac{1}{2|K|}(a_i + b_i x_1 + c_i x_2) \tag{30}$$

Recall that $|K|$ is the area of K and that $a_i = x_1^{[j]} x_2^{[k]} - x_1^{[k]} x_2^{[j]}$, $b_i = x_2^{[j]} - x_2^{[k]}$, and $c_i = x_1^{[k]} - x_1^{[j]}$, where the indices i, j, k are obtained by cyclic permutation of $1, 2, 3$, respectively.

The element strains are obtained by differentiating the hat functions with respect to x_1 and x_2.

$$\boldsymbol{\epsilon}^e = \begin{bmatrix} \partial/\partial x_1 & 0 \\ 0 & \partial/\partial x_2 \\ \partial/\partial x_2 & \partial/\partial x_1 \end{bmatrix} \boldsymbol{U}^e = \frac{1}{2|K|} \begin{bmatrix} b_1 & 0 & b_2 & 0 & b_3 & 0 \\ 0 & c_1 & 0 & c_2 & 0 & c_3 \\ c_1 & b_1 & c_2 & b_2 & c_3 & b_3 \end{bmatrix} \boldsymbol{d}^e = \boldsymbol{B}^e \boldsymbol{d}^e \tag{31}$$

Note that the matrix $\boldsymbol{B}^e$ is constant. As a consequence both the element strains $\boldsymbol{\epsilon}^e = \boldsymbol{B}^e \boldsymbol{U}^e$, and the element stresses $\boldsymbol{\sigma}^e = \boldsymbol{D}\boldsymbol{\epsilon}^e$ are constant on K.

The 6×6 element stiffness matrix is given by

$$\boldsymbol{K}^e = \int_K \boldsymbol{B}^{eT} \boldsymbol{D} \boldsymbol{B}^e \, dx \tag{32}$$

which, since the integrand is constant, simplifies to $\boldsymbol{K}^e = \boldsymbol{B}^{eT} \boldsymbol{D} \boldsymbol{B}^e |K|$.

In Matlab this is easy to compute:

```
function Ke = stiffness(x,y,mu,lambda)
area=polyarea(x,y); % x and y are 3 x 1 and hold node coordinates
b=[y(2)-y(3); y(3)-y(1); y(1)-y(2)];
```

```
c=[x(3)-x(2); x(1)-x(3); x(2)-x(1)];
D=mu*[2 0 0; 0 2 0; 0 0 1] + lambda*[1 1 0; 1 1 0; 0 0 0];
Be=[b(1)   0  b(2)   0  b(3)   0 ;
      0  c(1)   0  c(2)   0  c(3);
    c(1) b(1) c(2) b(2) c(3) b(3)]/2/area;
Ke=Be'*D*Be*area;
```

For homogeneous materials the Lamé parameters are given by $\lambda = E\nu/((1+\nu)(1-2\nu))$ and $\mu = E/(2(1+\nu))$, where E is Young's elastic modulus and ν Poisson's ratio.

The 6×1 element load vector is given by

$$\boldsymbol{F}^e = \int_K \boldsymbol{\varphi}^{\boldsymbol{e}T} \boldsymbol{f} \, dx = \int_K \begin{bmatrix} \varphi_1 & 0 \\ 0 & \varphi_1 \\ \varphi_2 & 0 \\ 0 & \varphi_2 \\ \varphi_3 & 0 \\ 0 & \varphi_3 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} dx \tag{33}$$

where we for simplicity consider only contributions from the body force $\boldsymbol{f} = [f_1,\ f_2]^T$. In the simplest case f_1 and f_2 are taken to be constant over the element, yielding

$$\boldsymbol{F}^e = \frac{|K|}{3} \begin{bmatrix} f_1 & f_2 & f_1 & f_2 & f_1 & f_2 \end{bmatrix}^T \tag{34}$$

The Matlab realization of this is also simple.

```
function Fe = load(x,y)
area=polyarea(x,y);
f=force(mean(x),mean(y));
Fe=(f(1)*[1 0 1 0 1 0]'+f(2)*[0 1 0 1 0 1]')*area/3;
```

Here, `force` is a subroutine defining the body force.

```
function f = force(x,y)
f=[1 0];
```

When performing the assembly of the global system of equations, one needs to recall that there are two displacements per node (i.e., triangle vertex). This makes the mapping between the nodes and the degrees of freedom a bit more complicated than usual. For example, if triangle K has the nodes 2, 5, and 7 then we should add the six entries of the element load vector $\boldsymbol{F}^e$ to positions $\boldsymbol{F}_i$ with $i = 2 \cdot 2 - 1 = 3$, $2 \cdot 2 = 4$, $2 \cdot 5 - 1 = 9$, $2 \cdot 5 = 10$, $2 \cdot 7 - 1 = 13$, and $2 \cdot 7 = 14$.

Below we list a routine for assembling $\boldsymbol{K}$ and $\boldsymbol{F}$:

```
function [K,F] = assemble(p,e,t)
ndof=2*size(p,2);
K=sparse(ndof,ndof);
F=zeros(ndof,1);
dofs=zeros(6,1);
E=1;  nu=0.3;
lambda=E*nu/((1+nu)*(1-2*nu));  mu=E/(2*(1+nu));
for i=1:size(t,2)
  nodes=t(1:3,i);
  x=p(1,nodes);  y=p(2,nodes);
  dofs(1:2:end)=2*nodes-1;  dofs(2:2:end)=2*nodes;
  Ke=stiffness(x,y,mu,lambda);
  Fe=load(x,y);
  K(dofs,dofs)=K(dofs,dofs)+Ke;
  F(dofs)=F(dofs)+Fe;
end
```

Of course we must also add suitable boundary conditions to $\boldsymbol{K}$ and $\boldsymbol{F}$. For example, homogeneous boundary conditions $\boldsymbol{u} = \boldsymbol{0}$ can be enforced by adding a large number (e.g., 10^6) to diagonal entries of $\boldsymbol{K}$ corresponding to degrees of freedom on the boundary.

```
for i=1:size(p,2)
  x=p(1,i); y=p(2,i);
  if x < 0.001 % or wherever the boundary is
    K(2*i-1,2*i-1)=1.e+6;  K(2*i,2*i)=1.e+6;
  end
```

```
end
```

Problem 8. Compute $\boldsymbol{K}^e$ for the triangle with corners at $(0,0)$, $(3,1)$, and $(2,2)$. Let

$$\boldsymbol{D} = \begin{bmatrix} 4 & 1 & 0 \\ 1 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

Verify that $\boldsymbol{K}^e$ has three zero eigenvalues. Can you explain why?

Problem 9. Write a code and solve (1) for a console occupying the unit square $\Omega = [0,1]^2$ and clamped along the boundary. Assume $E = 1$, $\nu = 0.3$, and body force

$$\boldsymbol{f} = \begin{bmatrix} (\lambda+\mu)(1-2x)(1-2y) \\ -2\mu y(1-y) - 2(\lambda+2\mu)x(1-x) \end{bmatrix}$$

The analytical solution is given by $\boldsymbol{u} = [0,\ -x(1-x)y(1-y)]$. Plot the displacement components (e.g., with `pdesurf(p,t,d(2:2:end))`). Validate your code by computing the energy norm $a(\boldsymbol{u}_h, \boldsymbol{u}_h) = \boldsymbol{d}^T \boldsymbol{F}$. It should converge to $(\lambda + 3\mu)/90$.

Linear Elastodynamics

Newton's second law states that the net force acting on a particle equals the mass of the particle times its acceleration. For a material particle within a material body this can be written

$$\rho \ddot{\boldsymbol{u}} - \nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f} \tag{35}$$

where ρ is the density of the material and $\ddot{u} = \partial^2 \boldsymbol{u}/\partial t^2$ the second derivative of $\boldsymbol{u}$ with respect to time t.

Thus the fundamental problem of linear elastodynamics is to find the time-dependent symmetric stress tensor $\boldsymbol{\sigma}$ and the time-dependent displacement vector $\boldsymbol{u}$ such that

$$
\begin{aligned}
\rho\ddot{\boldsymbol{u}} - \nabla\cdot\boldsymbol{\sigma} &= \boldsymbol{f}, && \text{in } \Omega\times I && \text{(36a)}\\
\boldsymbol{\sigma} &= 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda(\nabla\cdot\boldsymbol{u})\boldsymbol{I}, && \text{in } \Omega\times I && \text{(36b)}\\
\boldsymbol{u} &= \boldsymbol{0}, && \text{on } \Gamma_D\times I && \text{(36c)}\\
\boldsymbol{\sigma}\cdot\boldsymbol{n} &= \boldsymbol{g}, && \text{on } \Gamma_N\times I && \text{(36d)}\\
\boldsymbol{u}(\cdot,0) &= \boldsymbol{u}_0, && \text{in } \Omega && \text{(36e)}\\
\dot{\boldsymbol{u}}(\cdot,0) &= \boldsymbol{v}_0, && \text{in } \Omega && \text{(36f)}
\end{aligned}
$$

where $I = (0,T)$ is the time interval and $\boldsymbol{u}_0$ and $\boldsymbol{v}_0$ is the initial displacement and velocity of the body.

Semi-Discretization

Applying a finite element discretization in space to (36) we end up with a system of $2N$ ordinary differential equations

$$\boldsymbol{M}\ddot{\boldsymbol{d}}(t) + \boldsymbol{K}\boldsymbol{d}(t) = \boldsymbol{F}, \quad t\in I \tag{37}$$

where $\boldsymbol{d}(t)$ is a $2N\times 1$ vector of time-dependent nodal displacements. Moreover, $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{K}$, is the stiffness matrix, and $\boldsymbol{F}$ is the load vector, defined by

$$\boldsymbol{M} = \int_\Omega \rho\boldsymbol{\varphi}^T\boldsymbol{\varphi}\,dx \tag{38}$$

$$\boldsymbol{K} = \int \boldsymbol{B}^T\boldsymbol{D}\boldsymbol{B}\,dx \tag{39}$$

$$\boldsymbol{F} = \int_\Omega \boldsymbol{\varphi}^T\boldsymbol{f}\,dx + \int_{\Gamma_N}\boldsymbol{\varphi}^T\boldsymbol{g}\,ds \tag{40}$$

Modal Analysis

We note that (37) resembles a wave equation and it is therefore natural to look for a solution of the form

$$\boldsymbol{d}(t) = \boldsymbol{\phi}\sin\omega t \tag{41}$$

where $\boldsymbol{\phi}$ is a vector independent of time and ω a number to be determined. From a physical point this ansatz represents a vibration with frequency ω and the natural shape of the vibrational mode given by $\boldsymbol{\phi}$. Inserting (41) into $\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} = \boldsymbol{F}$ assuming $\boldsymbol{F} = \mathbf{0}$ we obtain a generalized eigenvalue problem for ω and $\boldsymbol{\phi}$

$$\boldsymbol{K}\boldsymbol{\phi} = \omega^2 \boldsymbol{M}\boldsymbol{\phi} \tag{42}$$

The vectors ϕ are called eigenmodes and the numbers $\lambda = \omega^2$ are called the eigenvalues. By inspecting the eigenvalues ω it is easy determine if a given load of the form $\boldsymbol{F} = \boldsymbol{F}_0 \sin \omega_0 t$ will give dangerous resonance effects for a particular value of ω_0. Consequently, this type of *modal analysis* is of great importance to structural engineers.

We can compute eigenvalues and eigenmodes with Matlab using the build-in routine `eigs`. In doing so we can speed up the computation by making use of the fact that the involved matrices are real and symmetric. In the code below this information is passed on to `eigs` via the `opts` field.

```
[p,e,t]=initmesh(geom,'hmax',0.1);
% assemble stiffness and mass matrix
[K,M]=assemble(p,e,t);
% compute 10 eigenvalues and eigenmodes to K phi=omega^2 M phi
opts.isreal=1; opts.issym=1;
[phi,omega2]=eigs(K,M,10,1.e-6,opts);
% visualize the 7:th eigenmode
p(1,:)=p(1,:)+0.1*phi(1:2:end,7)';
p(2,:)=p(2,:)+0.1*phi(2:2:end,7)';
pdemesh(p,e,t)
```

The mass matrix $\boldsymbol{M}$ is assembled similarly to the stiffness matrix $\boldsymbol{K}$. The element mass matrix is given by

$$\boldsymbol{M}^e = \int_K \rho \boldsymbol{\varphi}^{\boldsymbol{e}T} \boldsymbol{\varphi}^e \, dx \tag{43}$$

which translates into code viz.,

```
Me=rho*[2 0 1 0 1 0;
        0 2 0 1 0 1;
        1 0 2 0 1 0;
        0 1 0 2 0 1;
        1 0 1 0 2 0;
        0 1 0 1 0 2]*dx/12;
```

Problem 10. Calculate by hand the element mass matrix $\boldsymbol{M}^e$ assuming a unit density on the reference triangle with vertices at origo, $(1, 0)$, and $(0, 1)$.

Problem 11. A mesh of the famous L-shaped domain is obtained by typing `[p,e,t]=initmesh('lshapeg')`. Compute and plot the ten lowest eigenmodes on this domain. Assume elastic constants $\rho = 1$, $E = 1$, and $\nu = 0.3$.