NGCM6001 Simulation and Modelling Computer Lab 2

Aims

The aim of this computer laboratory is to gain a practical insight into the fundamental steps required to program a simple finite element code to solve a problem of elasticity. In doing so, you will learn:

- How to recast the PDE problem into a variational form
- To derive the formulation of a 2D finite element
- To create a simple structured mesh generator to mesh a 2D domain
- To create an assembly procedure for the stiffness matrix of a 2D structure
- How to numerically solve the variational problem for a practical example

The numerical environment chosen for this laboratory is IPython (http://ipython.org).

Basic problem: linear elasticity

Let's consider a 2D solid body Ω at rest, subjected to body forces $\mathbf{b}$, prescribed displacements $\overline{\mathbf{d}}$ and tractions $\overline{\mathbf{t}}$. The problem of elasticity is defined as follows:

$$\begin{array}{ll} \mathrm{div}\boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} & \boldsymbol{x}(\mathrm{x},\mathrm{y}) \text{ in } \Omega \\ \mathbf{d}(u,v) = \overline{\mathbf{d}}(u,v) & \boldsymbol{x}(\mathrm{x},\mathrm{y}) \text{ on } \partial\Omega_u \\ \mathbf{t}(\boldsymbol{x}) = \overline{\mathbf{t}}(\boldsymbol{x}) & \boldsymbol{x}(\mathrm{x},\mathrm{y}) \text{ on } \partial\Omega_\sigma \end{array} \tag{1}$$

$\boldsymbol{\sigma}$ is the symmetric Cauchy stress tensor.

Question 1:

Apply integration by parts to the PDE $\mathrm{div}\boldsymbol{\sigma} + \mathbf{b} = \mathbf{0}$ with the divergence theorem (relation between the flow (flux) of a vector field though a surface and the behavior inside the surface) recast it into its variational form. You will also take into account the conditions about the **test function** v on the boundary where $\mathbf{d}$ is prescribed.

To relate stress and strain one must define what is called a **constitutive relationship**. In other branches of physics other types of relationships would be used (e.g. relation between strain and temperature).

In the case of isotropic linear elasticity (Hooke's elasticity) the Cauchy stress tensor is related to the strain tensor ε through the following relationship:

$$\boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon} \tag{2}$$

where $\mathbb{C}$ is the elasticity tensor.

Equation (2) is a **tensor equation** (valid in any coordinate system). If one decides to use a Cartesian coordinate system (2) can be written in index notation as:

$$\left(\boldsymbol{\sigma}\right)_{ij} = \sigma_{ij} = \left(\mathbb{C} : \boldsymbol{\varepsilon}\right)_{ij} = \mathbb{C}_{ijkl}\varepsilon_{kl} \qquad \left\{2\mathrm{D} : i,j,k,l : 1..2;\ \ 3\mathrm{D} : i,j,k,l : 1..3\right\} \tag{3}$$

Please note that we use the following convention: bold characters represent vectors, matrices or tensors.

Because the Cauchy, strain and elasticity tensors are symmetric and the material isotropic one can simplify the tensorial constitutive equation into a matrix equation involving vectors and asecond-order matrix.

$$\left\{\boldsymbol{\sigma}\right\} = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix} = \underbrace{\frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \dfrac{1-\nu}{2} \end{bmatrix}}_{\left[\mathbb{C}\right]} \underbrace{\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ 2\varepsilon_{xy} \end{Bmatrix}}_{\left\{\boldsymbol{\varepsilon}\right\}} \tag{4}$$

$\left\{E,\nu\right\}$ are material properties, respectively the Young's modulus and Poisson's ratio which characterise the stiffness and compressibility of the material.

Let's consider a 4-noded bi-linear interpolation element (**Figure 1**).

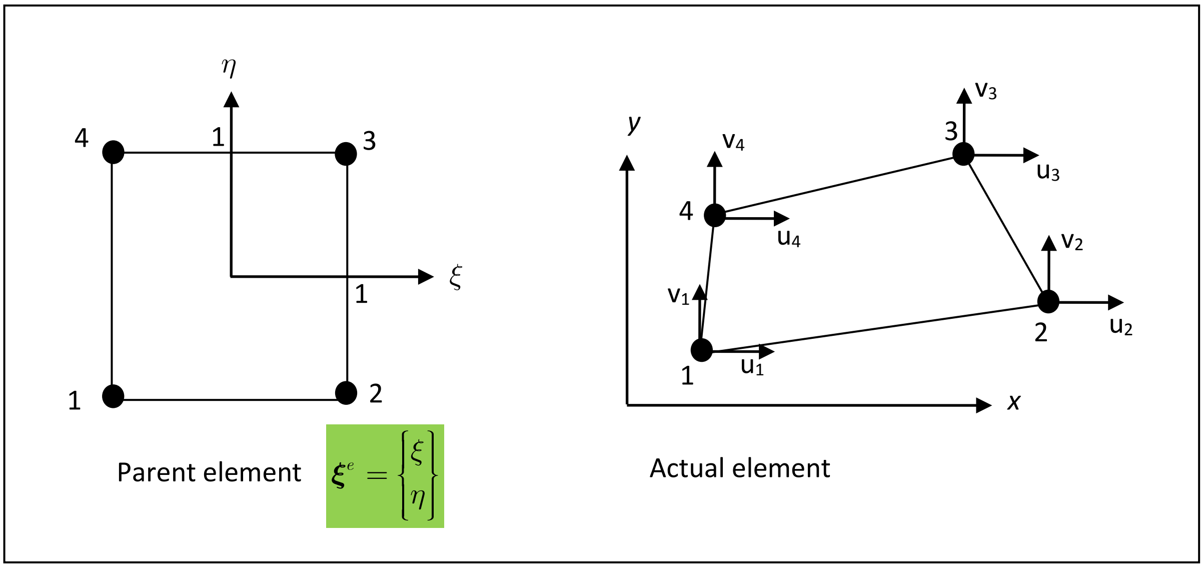


Figure 1. Four-noded quadrilateral element in the parent and actual configuration.

The shape functions chosen here are simple linear Lagrange polynomials but other forms could be used (quadratic Lagrange, Hermite, NURBS, etc):

$$N_i = \frac{1}{4}\,(1+\xi_i\xi)(1+\eta_i\eta) \qquad \xi_i,\eta_i = -1,1 \tag{5}$$

There are as many shape functions as nodes in the element, i.e. 4:

$$\begin{Bmatrix} N_1 = \frac{1}{4}\left(1-\xi\right)\left(1-\eta\right) \\ N_2 = \frac{1}{4}\left(1+\xi\right)\left(1-\eta\right) \\ N_3 = \frac{1}{4}\left(1+\xi\right)\left(1+\eta\right) \\ N_4 = \frac{1}{4}\left(1-\xi\right)\left(1+\eta\right) \end{Bmatrix} \tag{6}$$

We are using an **isoparametric formulation** which means that the **position vector** and **displacement vector** are interpolated using the same order of shape (or interpolation) functions.

Interpolation of the position vector (discretisation)

$$\mathbf{x}\left(\boldsymbol{\xi}^e,t\right)=\mathbf{x}_1(t)N_1\left(\boldsymbol{\xi}^e\right)+\mathbf{x}_2(t)N_2\left(\boldsymbol{\xi}^e\right)+\mathbf{x}_3(t)N_3\left(\boldsymbol{\xi}^e\right)+\mathbf{x}_4(t)N_4\left(\boldsymbol{\xi}^e\right) \tag{7}$$

where

$$\mathbf{x}_A(t)=\text{position}\left(node\,A\right)=\begin{Bmatrix}x_A\\ y_A\end{Bmatrix} \tag{8}$$

$$\mathbf{x}\left(\boldsymbol{\xi}^e,t\right)=\begin{Bmatrix}x(t)\\ y(t)\end{Bmatrix}_{node\,1}N_1\left(\boldsymbol{\xi}^e\right)+\begin{Bmatrix}x(t)\\ y(t)\end{Bmatrix}_{node\,2}N_2\left(\boldsymbol{\xi}^e\right)+\begin{Bmatrix}x(t)\\ y(t)\end{Bmatrix}_{node\,3}N_3\left(\boldsymbol{\xi}^e\right)+\begin{Bmatrix}x(t)\\ y(t)\end{Bmatrix}_{node\,4}N_4\left(\boldsymbol{\xi}^e\right) \tag{9}$$

Interpolation of the displacement vector (discretisation)

$$\mathbf{d}\left(\boldsymbol{\xi}^e,t\right)=\mathbf{d}_1(t)N_1\left(\boldsymbol{\xi}^e\right)+\mathbf{d}_2(t)N_2\left(\boldsymbol{\xi}^e\right)+\mathbf{d}_3(t)N_3\left(\boldsymbol{\xi}^e\right)+\mathbf{d}_4(t)N_4\left(\boldsymbol{\xi}^e\right) \tag{10}$$

where

$$\mathbf{d}_A(t)=\text{displacement}\left(node\,A\right)=\begin{Bmatrix}u_A\\ v_A\end{Bmatrix} \tag{11}$$

We can arrange all the degrees of freedom of the element in one vector:

$$\mathbf{d}=\left\{u_1,v_1,u_2,v_2,u_3,v_3,u_4,v_4\right\}^T \tag{12}$$

The discretised versions of the position and displacement vector can be rewritten in matrix form:

$$\mathbf{x}\left(\boldsymbol{\xi}^e,t\right)=N_I\left(\boldsymbol{\xi}^e\right)\mathbf{x}_I(t)=\mathbf{N}^T\left(\boldsymbol{\xi}^e\right)\mathbf{x}_I(t) \tag{13}$$

$$\mathbf{d}\left(\boldsymbol{\xi}^e,t\right)=N_I\left(\boldsymbol{\xi}^e\right)\mathbf{d}_I(t)=\mathbf{N}^T\left(\boldsymbol{\xi}^e\right)\mathbf{d}_I(t) \tag{14}$$

where

$$\mathbf{N}^T\left(\boldsymbol{\xi}^e\right)=\left\{N_1(\boldsymbol{\xi}^e),N_2(\boldsymbol{\xi}^e),N_3(\boldsymbol{\xi}^e),N_4(\boldsymbol{\xi}^e)\right\} \tag{15}$$

(16)

The test function used for the formulation of the weak form is also discretised:

$$\boldsymbol{v}\left(\boldsymbol{\xi}^e,t\right)=N_I\left(\boldsymbol{\xi}^e\right)\boldsymbol{v}_I(t)=\mathbf{N}^T\left(\boldsymbol{\xi}^e\right)\boldsymbol{v}_I(t) \tag{17}$$

Question 2:

Inject the discretised version of the displacement into the weak form established in **Question 1**. You will have to express the strain matrix using the **B** matrix (derivatives of shape functions). You should obtain 3 integral matrix/vector forms: the **element stiffness matrix**, the **nodal load vector due to body forces** and the **nodal load vector due to boundary forces**.

The integral expressions obtained in **Question 2** will have to be numerically integrated using, in our case, Gauss integration. We have integrals defined over the element domain Ω^e and its boundary $\partial\Omega^e$.
There are well known integration formulas over parametric domains that are exact for polynomials like the Lagrange polynomials used in the shape functions. The integral of a function is transformed into the sum of **weighted values that the function takes at specific points**, the so-called **integration points** or **Gauss points**.

$$\int_{\square} f(\boldsymbol{\xi})\ d\square = w_i f(\boldsymbol{\xi}_i) \tag{18}$$

These integration formulas apply to the **parent element** (**Figure 1**) which is the element in the parametric domain $\square = \{\{-1,1\},\{-1,1\}\}$ so it is necessary to map back the parent element domain to the current (deformed) domain. This is a simple change of variables:

$$d\Omega_e = dxdy = \det(\mathbf{J}) d\xi d\eta \tag{19}$$

$$\int_{\Omega^e} f(\mathbf{x})\ d\Omega = \int_{\square} f(\mathbf{x}(\boldsymbol{\xi}^e)) \underbrace{\det\left(\frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}^e}\right)}_{J_{\boldsymbol{\xi}^e} = \det(\mathbf{J})} d\square = w_i J_{\boldsymbol{\xi}^e}({\boldsymbol{\xi}^e}_i) f({\boldsymbol{\xi}^e}_i) \tag{20}$$

Here, we will use a 2 points Gauss integration rule: i.e. 4 integration points for the quadrilateral element:

$$\begin{aligned}
\mathbf{G}_1 &= \left\{-\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right\} & w_1 &= 1\\
\mathbf{G}_2 &= \left\{\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right\} & w_2 &= 1\\
\mathbf{G}_3 &= \left\{\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right\} & w_3 &= 1\\
\mathbf{G}_4 &= \left\{-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right\} & w_4 &= 1
\end{aligned} \tag{21}$$

Question 3:

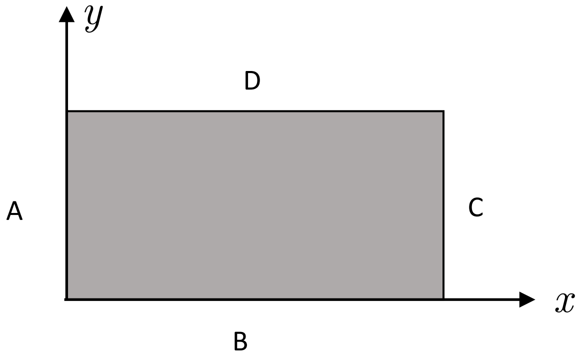
1/ Inject the integration formula into the integral matrix equations obtained in **Question 2**.
2/ **Write a Python programme:**

a/ to calculate the stiffness matrix of the element and nodal force vector
b/ to generate a structured grid over a 2D rectangle domain (mesh generator).
c/ to specify a uniform force at each node of one side of the rectangle domain
d/ to specify a uniform displacement at each node of one side of the rectangle domain
e/ to assemble the global stiffness matrix and force vector of the whole rectangular structure

Question 4:

Using the Python programme that you developed in Question 3, solve the following problem.

Let's consider a 2D rectangular solid domain (length L_x = 1 m; height L_y = 0.5 m) with material properties $\{E = 210 \text{ GPa}; \ \nu = 0.3\}$ and sides labelled as A, B, C and D.



The boundary conditions are as follows:

$$u(\text{side A}) = v(\text{side B}) = 0 \tag{22}$$

The load condition is:

$$\mathbf{F}(\text{side C})[\text{Newton}] = \begin{Bmatrix} F_x(\text{side C}) \\ F_y(\text{side C}) \end{Bmatrix} = \begin{Bmatrix} 10^6 \\ 0 \end{Bmatrix} \tag{23}$$

The boundary and loading conditions are homogeneous and therefore do not introduce shear coupling. The Cauchy stress tensor has only one non-null component, σ_{xx}.

NB: the total load of 10^6 N must be uniformly distributed over all the nodes of side C.

a/ Solve the boundary-value problem of elasticity described above to calculate the displacement of faces D and C. **You will consider 2 mesh sizes:** 1 for which the whole structure is meshed *with one element only* and one size of your choice.

b/Compare the solutions to the exact analytical solutions given by $\sigma_{xx} = F_x / L_y$.
The strains can be calculated as $\varepsilon_{xx} = \sigma_{xx} / E = F_x / EL_y = -\varepsilon_{yy} / \nu; \qquad \varepsilon_{xy} = 0$ and the displacements as $u = \varepsilon_{xx} L_x; \qquad v = \varepsilon_{yy} L_y$. Do the results surprise you? Why?