



ECOLE
POLYTECHNIQUE
DE BRUXELLES

UNIVERSITÉ LIBRE DE BRUXELLES

SUMMARY

Structural Analysis & Finite Elements

MECA-H421

Author:
Enes ULUSOY

Professors:
Pyl LINCY
Peter BERKE

Year 2016 - 2017

Appel à contribution

Synthèse Open Source



Ce document est grandement inspiré de l'excellent cours donné par Pyl LINCY et Peter BERKE à l'EPB (École Polytechnique de Bruxelles), faculté de l'ULB (Université Libre de Bruxelles). Il est écrit par les auteurs susnommés avec l'aide de tous les autres étudiants et votre aide est la bienvenue ! En effet, il y a toujours

moyen de l'améliorer surtout que si le cours change, la synthèse doit être changée en conséquence. On peut retrouver le code source à l'adresse suivante

<https://github.com/nenglebert/Syntheses>

Pour contribuer à cette synthèse, il vous suffira de créer un compte sur *Github.com*. De légères modifications (petites coquilles, orthographe, ...) peuvent directement être faites sur le site ! Vous avez vu une petite faute ? Si oui, la corriger de cette façon ne prendra que quelques secondes, une bonne raison de le faire !

Pour de plus longues modifications, il est intéressant de disposer des fichiers : il vous faudra pour cela installer L^AT_EX, mais aussi *git*. Si cela pose problème, nous sommes évidemment ouverts à des contributeurs envoyant leur changement par mail ou n'importe quel autre moyen.

Le lien donné ci-dessus contient aussi un README contenant de plus amples informations, vous êtes invités à le lire si vous voulez faire avancer ce projet !

Licence Creative Commons

Le contenu de ce document est sous la licence Creative Commons : *Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)*. Celle-ci vous autorise à l'exploiter pleinement, compte- tenu de trois choses :



1. *Attribution* ; si vous utilisez/modifiez ce document vous devez signaler le(s) nom(s) de(s) auteur(s).
2. *Non Commercial* ; interdiction de tirer un profit commercial de l'œuvre sans autorisation de l'auteur
3. *Share alike* ; partage de l'œuvre, avec obligation de rediffuser selon la même licence ou une licence similaire

Si vous voulez en savoir plus sur cette licence :

<http://creativecommons.org/licenses/by-nc-sa/4.0/>

Merci !

Contents

2	Isoparametric elements	1
2.1	Partition of the domain	1
2.2	Classical elements	1
2.3	Reference element and real elements	2
2.4	Approximation on the reference element	4
2.5	Construction of the shape function	4

Chapter 2

Isoparametric elements

2.1 Partition of the domain

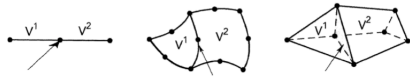


Figure 2.1

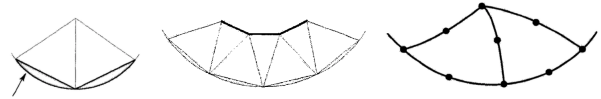


Figure 2.2

As we have seen, any domain V must undergo a sub-domain V^e . In dimension 1 the elements are separated by a point, 2 a curve and 3 a surface. The connection of all elements should be as closely as possible similar to the real domain. Discretization errors can occur for complex geometries. We can solve the issue by decreasing the size of the elements (mesh refreshment) or using elements with curved boundaries (Figure 2.2).

Nodes and elements must satisfy:

1. each element is defined by the coordinates of the nodes located in the element. In the majority of elements available in the litterature, the nodes are on the boundary of the elements.
2. A portion of the boundain between two elements must be identically defined for both elements.

2.2 Classical elements

The order of the element is directly related to the number of nodes on its boundary and vice-versa. Most elements are first, second or third-order, enabling to model linear quadratic or cubic boundaries.

Order Dimension	Linear	Quadratic	Cubic
1D			
2D (triangular)			
2D (quadrangular)			
3D (tetrahedral)			
3D (hexahedral)			
3D (prismatic)			

Figure 2.3

2.3 Reference element and real elements

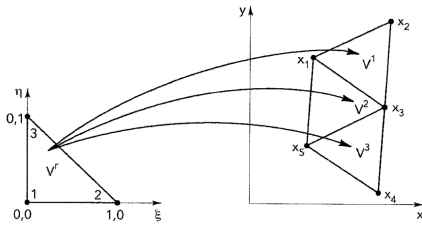


Figure 2.4

The finite element is the basic element on which the governing equations will be transposed. To avoid complex analytical operations, we will use a **reference element** on which a mapping of the **real element** (parent element) has to be performed. By this approach, the shape functions are written for the reference and then converted by linear transformation to the real. In 2D, the reference element is defined over a (ξ, η) -space, while the real elements are defined over the usual (x, y) -space.

To perform this operation, a transformation \mathcal{T} is required:

$$\mathcal{T} : \xi \rightarrow x\xi = \bar{N}^e(\xi)x^e \quad (2.1)$$

where x^e are the coordinates of the nodes of the element V^e and $\bar{N}^e(\xi)$ are the **geometric transformation functions** which must follow the same rules as shape functions. This transformation allows to write the analytical definition for each V^e with respect to ξ on a simple element V^r . The function $u(x)$ to be approximated can be written as $u(\xi)$.

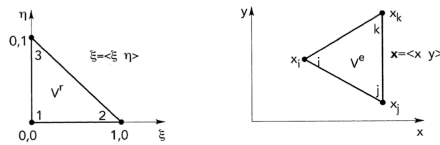


Figure 2.5

Let's make an example with a triangular element. On the figure is represented the reference and the parent element of a three-node triangular element. The reference is defined as:

$$\xi \geq 0, \quad \eta \geq 0, \quad \xi + \eta \leq 1. \quad (2.2)$$

Let us consider a real element defined by three nodes i, j, k and the transformation \mathcal{T} such that:

$$x(\xi, \eta) = [1 - \xi - \eta \quad \xi \quad \eta] \begin{bmatrix} x_i \\ x_j \\ x_k \end{bmatrix}, \quad y(\xi, \eta) = [1 - \xi - \eta \quad \xi \quad \eta] \begin{bmatrix} y_i \\ y_j \\ y_k \end{bmatrix}. \quad (2.3)$$

\mathcal{T} verifies the following properties:

1. the nodes $(0,0)$, $(0,1)$ and $(1, 0)$ from V^r are converted into (x_i, y_i) , (x_j, y_j) and (x_k, y_k) :
2. each boundary of V^r is transformed into a boundary of V^e . For example, the line between $(1, 0)$ and $(0, 1)$ is transformed into $1 - \xi - \eta = 0$ or $\eta = 1 - \xi$:

$$x = [0 \quad \xi \quad 1 - \xi] \begin{bmatrix} x_i \\ x_j \\ x_k \end{bmatrix}, \quad y = [0 \quad \xi \quad 1 - \xi] \begin{bmatrix} y_i \\ y_j \\ y_k \end{bmatrix}. \quad (2.4)$$

3. The transformation is **bijective**. This is valid only if the Jacobian matrix \mathbf{J} is not singular:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} x_j - x_i & y_j - y_i \\ x_k - x_i & y_k - y_i \end{bmatrix}. \quad (2.5)$$

This is equal to 0 only if the vertices are aligned.

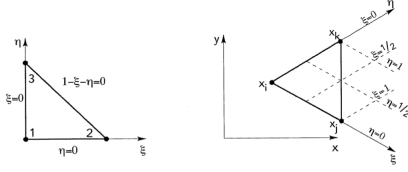


Figure 2.6

Geometrically, we can see the ξ -variables as a local coordinate system. The process of generating meshes on surfaces is called **tessellation**. Reference elements are listed on Figure 2.7 and remark that for the 3D case, we have a system (ξ, η, ζ) .

Order	Linear	Quadratic	Cubic
Dimension	Linear	Quadratic	Cubic
1D			
2D (triangular)			
2D (quadrangular)			
3D (tetrahedral)			
3D (hexahedral)			
3D (prismatic)			

Figure 2.7

2.4 Approximation on the reference element

Remind that we want to approximate a quantity of interest $u(\mathbf{x})$:

$$u(\mathbf{x}) = [N_1(\mathbf{x}) \quad N_2(\mathbf{x}) \dots] \begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = \mathbf{N}^e(\mathbf{x}) \mathbf{q}^e \quad (2.6)$$

This expression is in the real space, using the \mathcal{T} transformation:

$$u(\boldsymbol{\xi}) = \mathbf{N}^e(\boldsymbol{\xi}) \mathbf{q}^e, \quad \mathcal{T} : \boldsymbol{\xi} \rightarrow \mathbf{x}(\boldsymbol{\xi}) = \bar{\mathbf{N}}^e \mathbf{x}^e. \quad (2.7)$$

We have 4 properties:

1. the approx is **interpolant**: $u(\xi_i) = u_i$, related to $N_j(\xi_i) = \delta_{ij}$;
2. **continuity in the element**: the N_j and their derivatives up to the order s must be continuous;
3. **continuity between elements**: the approx of u and its derivatives up to the order s must depend univocally on the nodal variables appearing on the boundary between the elements and nothing else;
4. when the size of the element tends to 0, the error on the approx and its derivatives vanish.

It is also possible to demonstrate that $\sum_i N_i(\boldsymbol{\xi}) = 1$. Approx of C^0 - or C^s -type if u or u and its derivatives up to the order s are continuous. Finally, if $\bar{\mathbf{N}} = \mathbf{N}$ the elements are **isoparametric**. If not:

- **sub-parametric** if the order of polynomials of the geometric transformation is lower than for the shape functions;
- **super-parametric** if the order is higher.

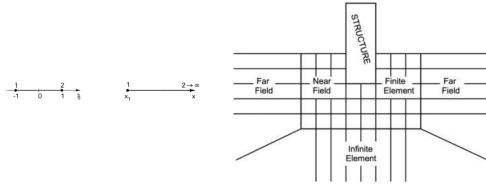


Figure 2.8

There can also be **infinite elements**. They can be devised in 1D by the following mapping:

$$x = x_1 + \alpha \frac{1 + \xi}{1 - \xi} \quad (2.8)$$

varying from $x = x_1$ to ∞ for ξ from -1 to 1. Used for infinite domains like the soil.

2.5 Construction of the shape function

This will be shown for the example of a four-node quadrangular element. We are searching for \mathbf{N}^e such that $u^e(\mathbf{x}) = \mathbf{N}^e(\mathbf{x}) \mathbf{q}^e$. First a polynomial basis must be set up with $u(\boldsymbol{\xi}) = \mathbf{p}(\boldsymbol{\xi})^T \mathbf{a}$, then by $u(\boldsymbol{\xi}) = \mathbf{p}(\boldsymbol{\xi})^T \mathbf{a} = \mathbf{N}^e \mathbf{q}^e$ we can retrieve the shape functions. A suitable polynomial is the one containing four terms and symmetric wrt ξ and η :

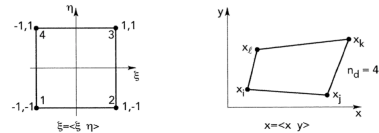


Figure 2.9

$$\mathbf{p}(\mathbf{x}) = [1 \quad \xi \quad \eta \quad \xi\eta]. \quad (2.9)$$

Then the interpolation:

$$\mathbf{q}^e = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \begin{bmatrix} p_1(\xi_1) & p_2(\xi_1) & p_3(\xi_1) & p_4(\xi_1) \\ p_1(\xi_2) & p_2(\xi_2) & p_3(\xi_2) & p_4(\xi_2) \\ \vdots & & & \vdots \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \quad (2.10)$$

By inverting the matrix \mathbf{P} , we get the coefficients: $\mathbf{a} = \mathbf{P}^{-1}\mathbf{q}^e$. Finally, the shape functions can be obtained as follows:

$$u(\boldsymbol{\xi}) = \mathbf{N}^e(\boldsymbol{\xi})\mathbf{q}^e = \mathbf{p}(\boldsymbol{\xi})^T \mathbf{a} = \mathbf{p}(\boldsymbol{\xi})^T \mathbf{P}^{-1} \mathbf{q}^e \quad \Rightarrow \quad \mathbf{N}^e(\boldsymbol{\xi}) = \mathbf{p}(\boldsymbol{\xi})^T \mathbf{P}^{-1}. \quad (2.11)$$