Approximation and Finite Elements in Isogeometric Problems

L. Carlon[1] Supervisor: Prof. Maria Morandi Cecchi[2]

[1]Department of Computer Engineering

[2]Department of Pure and Applied Mathematics

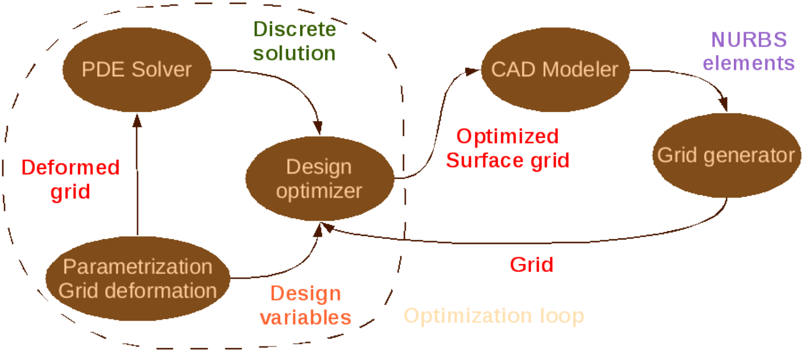
Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

Outline

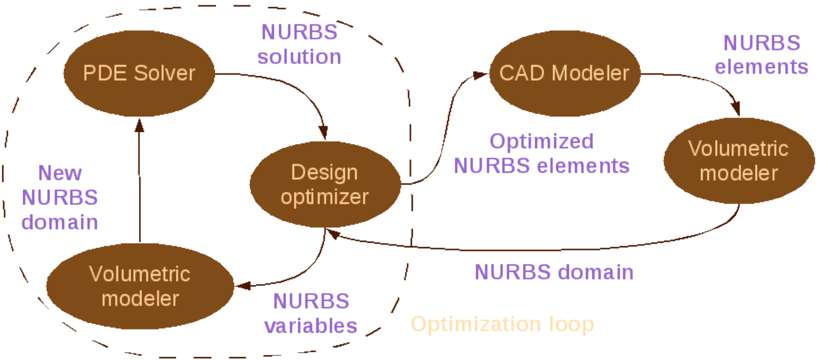
1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

CURRENT AUTOMATED DESIGN LOOP



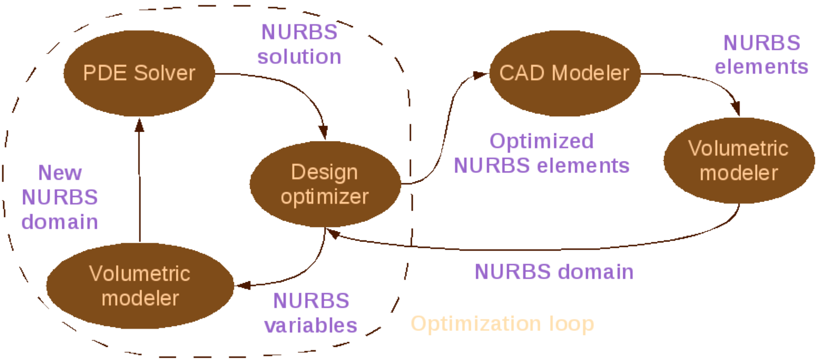
- Various tools each being complex and belonging to a specific field, e.g.:
  - CAD tools;
  - simulation tools...

ISOGEOMETRIC AUTOMATED DESIGN LOOP



- Loop entirely built on a *CAD basis*;
- all the processes can be *merged*.

ISOGEOMETRIC AUTOMATED DESIGN LOOP



- Loop entirely built on a *CAD basis*;
- all the processes can be *merged*.

Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

PARTIAL DIFFERENTIAL EQUATIONS

- Assuming $\boldsymbol{z} = [\boldsymbol{x}, t] = [z_1, \dots, z_n]^T$, $a_{i,j} = a_{i,j}(\boldsymbol{z}), \dots$
- the form of a *generic second-order PDE* is:

$$-\sum_{i,j=1}^{n} \frac{\partial}{\partial z_i}\left(a_{ij}\frac{\partial u}{\partial z_j}\right)+\sum_{i=1}^{n}\left(\frac{\partial}{\partial z_i}(b_i u) + c_i \frac{\partial u}{\partial z_i}\right)+a_0 u = f, \ \forall \boldsymbol{z} \in \mathcal{O},$$

- equipped with adequate *boundary conditions*.

Notice!

The *strong* or *classical solution* of this problem is a solution $u(z)$ which satisfies all the equations.
It is *not possible* to find a solution in general.

PARTIAL DIFFERENTIAL EQUATIONS

- Assuming $\boldsymbol{z} = [\boldsymbol{x}, t] = [z_1, \ldots, z_n]^T$, $a_{i,j} = a_{i,j}(\boldsymbol{z}), \ldots$
- the form of a *generic second-order PDE* is:

$$-\sum_{i,j=1}^{n} \frac{\partial}{\partial z_i}\left(a_{ij}\frac{\partial u}{\partial z_j}\right)+\sum_{i=1}^{n}\left(\frac{\partial}{\partial z_i}(b_i u) + c_i \frac{\partial u}{\partial z_i}\right)+a_0 u = f, \ \forall \boldsymbol{z} \in \mathcal{O},$$

- equipped with adequate *boundary conditions*.

Notice!

The *strong* or *classical solution* of this problem is a solution $u(z)$ which satisfies all the equations.
It is *not possible* to find a solution in general.

THERMAL CONDUCTION PROBLEM

Strong form of the problem

Find $u : \bar{\Omega} \to \mathbb{R}$ so that

$$\begin{cases} \nabla \left(\kappa \left(\boldsymbol{x} \right) \nabla u \left(\boldsymbol{x} \right) \right) = f \left(\boldsymbol{x} \right), & \forall \boldsymbol{x} \in \Omega, \text{ given } f : \Omega \to \mathbb{R} \\ u \left(\boldsymbol{x} \right) = g_D \left(\boldsymbol{x} \right), & \forall \boldsymbol{x} \in \Gamma_D, \ g_D : \Gamma_D \to \mathbb{R} \\ \kappa \left(\boldsymbol{x} \right) \dfrac{\partial u}{\partial \boldsymbol{\nu}} \left(\boldsymbol{x} \right) = g_N \left(\boldsymbol{x} \right), & \forall \boldsymbol{x} \in \Gamma_N, \ g_N : \Gamma_N \to \mathbb{R} \end{cases} .$$

THERMAL CONDUCTION PROBLEM

Weak form of the problem

Find u so that

$$\begin{cases} a\left(\upsilon,\varphi\right) = l\left(\varphi\right), & \forall \varphi \in H_0^1\left(\Omega\right),\ \upsilon \in H_0^1\left(\Omega\right) \\ \upsilon\left(\boldsymbol{x}\right) = 0, & \forall \boldsymbol{x} \in \Gamma_D \\ \gamma\left(\boldsymbol{x}\right) = g_N\left(\boldsymbol{x}\right), & \forall \boldsymbol{x} \in \Gamma_D,\ g_D : \Gamma_D \to \mathbb{R} \\ \kappa\left(\boldsymbol{x}\right) \dfrac{\partial\left(\upsilon + \gamma\right)}{\partial \boldsymbol{\nu}}\left(\boldsymbol{x}\right) = g_N\left(\boldsymbol{x}\right), & \forall \boldsymbol{x} \in \Gamma_N,\ g_N : \Gamma_N \to \mathbb{R} \\ u\left(\boldsymbol{x}\right) = \upsilon\left(\boldsymbol{x}\right) + \gamma\left(\boldsymbol{x}\right), & \forall \boldsymbol{x} \in \Omega \end{cases},$$

$$a\left(\upsilon,\varphi\right) = \iint_\Omega \kappa \nabla \upsilon \nabla \varphi d\boldsymbol{x},$$

$$l\left(\varphi\right) = \iint_\Omega \left(f\varphi - \kappa \nabla \gamma \nabla \varphi\right) d\boldsymbol{x} - \int_{\Gamma_N} \kappa g_N \varphi d\boldsymbol{\Gamma}.$$

Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

GALERKIN METHOD

- Approximate the space: $V = H_0^1(\Omega) \rightsquigarrow V_{h,p}$.
- We rewrite the equation $a(\upsilon, v) = l(v)\,,\ \forall v \in V$ with

  $a(\upsilon_{h,p}, v) = l(v)\,,\ \forall v \in V_{h,p} = \text{span}\left\{v_n, n = 0, \ldots, N_{h,p} - 1\right\} \subset V$

  where $\upsilon(x) = \sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i$.
- In this space any function can be written as the linear combination of the basis functions as

$$a\left(\sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i, v_j\right) = \sum_{i=0}^{N_{h,p}-1} a(v_i, v_j)\,\bar{\upsilon}_i = l(v_j)\,, j = 0, \ldots, N_{h,p}-1.$$

- The result is a linear system of equations which can be written in the matrix form $\boldsymbol{S}_{h,p}\bar{\boldsymbol{\Upsilon}}_{h,p} = \boldsymbol{F}_{h,p}$.

GALERKIN METHOD

- Approximate the space: $V = H_0^1(\Omega) \rightsquigarrow V_{h,p}$.
- We rewrite the equation $a(\upsilon, v) = l(v)$, $\forall v \in V$ with

  $a(\upsilon_{h,p}, v) = l(v)$, $\forall v \in V_{h,p} = \text{span}\{v_n, n = 0, \ldots, N_{h,p} - 1\} \subset V$

  where $\upsilon(\boldsymbol{x}) = \sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i$.
- In this space any function can be written as the linear combination of the basis functions as

$$a\left(\sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i, v_j\right) = \sum_{i=0}^{N_{h,p}-1} a(v_i, v_j)\,\bar{\upsilon}_i = l(v_j),\, j = 0, \ldots, N_{h,p}-1.$$

- The result is a linear system of equations which can be written in the matrix form $\boldsymbol{S}_{h,p}\bar{\boldsymbol{\Upsilon}}_{h,p} = \boldsymbol{F}_{h,p}$.

GALERKIN METHOD

- Approximate the space: $V = H_0^1(\Omega) \rightsquigarrow V_{h,p}$.
- We rewrite the equation $a(\upsilon, v) = l(v)$, $\forall v \in V$ with

  $a(\upsilon_{h,p}, v) = l(v)$, $\forall v \in V_{h,p} = \text{span}\left\{v_n, n = 0, \ldots, N_{h,p} - 1\right\} \subset V$

  where $\upsilon(\boldsymbol{x}) = \sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i$.
- In this space any function can be written as the linear combination of the basis functions as

$$a\left(\sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i, v_j\right) = \sum_{i=0}^{N_{h,p}-1} a(v_i, v_j)\,\bar{\upsilon}_i = l(v_j), j = 0, \ldots, N_{h,p}-1.$$

- The result is a linear system of equations which can be written in the matrix form $\boldsymbol{S}_{h,p}\bar{\boldsymbol{\Upsilon}}_{h,p} = \boldsymbol{F}_{h,p}$.

GALERKIN METHOD

- Approximate the space: $V = H_0^1(\Omega) \rightsquigarrow V_{h,p}$.
- We rewrite the equation $a(\upsilon, v) = l(v)$, $\forall v \in V$ with

  $a(\upsilon_{h,p}, v) = l(v)$, $\forall v \in V_{h,p} = \text{span}\left\{v_n, n = 0, \ldots, N_{h,p} - 1\right\} \subset V$

  where $\upsilon(\boldsymbol{x}) = \sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i$.
- In this space any function can be written as the linear combination of the basis functions as

$$a\left(\sum_{i=0}^{N_{h,p}-1} \bar{\upsilon}_i v_i, v_j\right) = \sum_{i=0}^{N_{h,p}-1} a(v_i, v_j)\,\bar{\upsilon}_i = l(v_j), j = 0, \ldots, N_{h,p}-1.$$

- The result is a linear system of equations which can be written in the matrix form $\boldsymbol{S}_{h,p}\bar{\boldsymbol{\Upsilon}}_{h,p} = \boldsymbol{F}_{h,p}$.

Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

ISOPARAMETRIC CONCEPT

Both Isogeometric Analysis and the Finite Element Method are based on the

Isoparametric concept

The same basis functions used to define the geometry are used to define the approximated solution.

FINITE ELEMENT METHOD (1)

Step 1: *Approximation* of the domain with a *piecewise-polynomial* boundary.

Step 2: The approximated domain is subdivided into N_{el} elements forming the *mesh*.

Step 3: As a consequence of the approximation, the domains of the functions defining the boundary conditions need to be modified.

Step 4: The space V of the solutions has to be redefined with a new space $V_{h,p}$ where the basis functions are associated with the nodes of the elements and are piecewise-polynomials of degree p.

FINITE ELEMENT METHOD (1)

Step 1: *Approximation* of the domain with a *piecewise-polynomial* boundary.

Step 2: The approximated domain is subdivided into N_{el} elements forming the *mesh*.

Step 3: As a consequence of the approximation, the domains of the functions defining the boundary conditions need to be modified.

Step 4: The space V of the solutions has to be redefined with a new space $V_{h,p}$ where the basis functions are associated with the nodes of the elements and are piecewise-polynomials of degree p.

FINITE ELEMENT METHOD (1)

Step 1: *Approximation* of the domain with a *piecewise-polynomial* boundary.

Step 2: The approximated domain is subdivided into N_{el} elements forming the *mesh*.

Step 3: As a consequence of the approximation, the domains of the functions defining the boundary conditions need to be modified.

Step 4: The space V of the solutions has to be redefined with a new space $V_{h,p}$ where the basis functions are associated with the nodes of the elements and are piecewise-polynomials of degree p.

FINITE ELEMENT METHOD (1)

Step 1: *Approximation* of the domain with a *piecewise-polynomial* boundary.

Step 2: The approximated domain is subdivided into N_{el} elements forming the *mesh*.

Step 3: As a consequence of the approximation, the domains of the functions defining the boundary conditions need to be modified.

Step 4: The space V of the solutions has to be redefined with a new space $V_{h,p}$ where the basis functions are associated with the nodes of the elements and are piecewise-polynomials of degree p.

FINITE ELEMENT METHOD (2)

Step 5: The weak formulation can be rewritten using the approximations presented so far:

$$\sum_{i=0}^{N_{h,p}-1} \bar{v}_i \iint_{\Omega_h} \kappa \nabla v_i \nabla v_j d\mathbf{x} =$$

$$\iint_{\Omega_h} \left(f v_j - \kappa \nabla \gamma_{h,p} \cdot \nabla v_j \right) d\mathbf{x} + \int_{\Gamma_{N,h}} \kappa g_{N,h} v_j d\mathbf{S},$$

$j = 0, \ldots, N_{h,p} - 1.$

ISOPARAMETRIC ELEMENTS

Definition

Let $\boldsymbol{x}_{K_m} : \tilde{K} \to K_m$ be of the form

$$\boldsymbol{x}_{K_m}(\boldsymbol{\xi}) = \sum_{i=0}^{N_{h,p}-1} v_{\tilde{K}}^{i} \boldsymbol{x}_i^{(m)}$$

where $\boldsymbol{x}_i$ is the i^{th} node of the element $\mathcal{K}_m$. If the element interpolation function can be written as

$$\upsilon_{h,p}(\boldsymbol{\xi}) = \sum_{i=0}^{N_{h,p}-1} v_{\tilde{K}}^{i} \bar{\upsilon}_i$$

the element is said to be isoparametric.

RESULTING MODEL (1)

The *left-hand side* of the equation of the linear system can be written as:

$$\sum_{i=0}^{N_{h,p}-1} \bar{v}_i \sum_{m=1}^{M_{h,p}} \iint_{K_m} \kappa \nabla v_j \nabla v_i d\mathbf{x} =$$

$$= \sum_{j=0}^{N_{h,p}-1} \bar{v}_j \sum_{m=1}^{M_{h,p}} \sum_{n=1}^{2} \iint_{\tilde{K}} J_{K_m} \tilde{\kappa}^{(m)} \cdot \left(\sum_{r=1}^{2} \frac{\partial \tilde{v}_j^{(m)}}{\partial \xi_r} \cdot \frac{\partial \xi_r^{(m)}}{\partial x_n} \right) \cdot \ldots$$

$$\ldots \cdot \left(\sum_{s=1}^{2} \frac{\partial \tilde{v}_i^{(m)}}{\partial \xi_s} \cdot \frac{\partial \xi_s^{(m)}}{\partial x_n} \right) d\boldsymbol{\xi}.$$

where:

$$\left\{ \frac{\partial \xi_r^{(m)}}{\partial x_n} \right\}_{r,n=1}^{2} = \left(\frac{D\mathbf{x}_{K_m}}{D\boldsymbol{\xi}} \right)^{-1}.$$

RESULTING MODEL (2)

The *right-hand side* turns out to be:

$$\sum_{m=1}^{M_{h,p}} \iint_{K_m} \left(f v_j - \kappa \nabla \gamma_{h,p} \nabla v_j \right) d\mathbf{x} + \sum_{m=1}^{M_{h,p}} \int_{\Gamma_{N,h} \cap \bar{K}_m} \kappa g_{N,h} v_j d\mathbf{\Gamma} =$$

$$\sum_{m=1}^{M_{h,p}} \iint_{\tilde{K}} J_{K_m} \left(\tilde{f}^{(m)} \tilde{v}_j^{(m)} - \tilde{\kappa}^{(m)} \nabla \tilde{\gamma}_{h,p}^{(m)} \nabla \tilde{v}_j^{(m)} \right) d\boldsymbol{\xi} +$$

$$+ \sum_{m=1}^{M_{h,p}} \int_{\bar{\tilde{K}} \cap \tilde{\Gamma}_{N,h}} \left(\left(a_1 g_{N,h} v_j \right) \circ \mathbf{y}_{\bar{K}_m \cap \Gamma_{D,h}} \right) (t)\, dt.$$

Outline

1 Introduction
- Automated design process
- Partial Differential Equations
- Galerkin method
- Isoparametric concept

2 Finite Element Method

3 Isogeometric Analysis
- CAD basis functions
- Analysis

4 Numerical examples
- Numerical example (1)
- Numerical example (2)
- Numerical example (3)

B-SPLINES basis functions (1)

Definition

A B-spline basis functions is defined as:

$$N_i^0(\xi) = \begin{cases} 1, & \xi_i \le \xi < \xi_{i+1} \\ 0, & \text{otherwise} \end{cases} \quad ,$$

$$N_i^p(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} \cdot N_i^{p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} \cdot N_{i+1}^{p-1}(\xi),$$

where $\Xi = [\xi_0, \ldots, \xi_m]$ is the knot vector.

B-SPLINES basis functions (2)

Definition

A knot vector of the form

$$\Xi = \left[\underbrace{a, \dots, a}_{p+1}, \xi_{p+1}, \dots, \xi_{m-p-1}, \underbrace{b, \dots, b}_{p+1} \right]$$

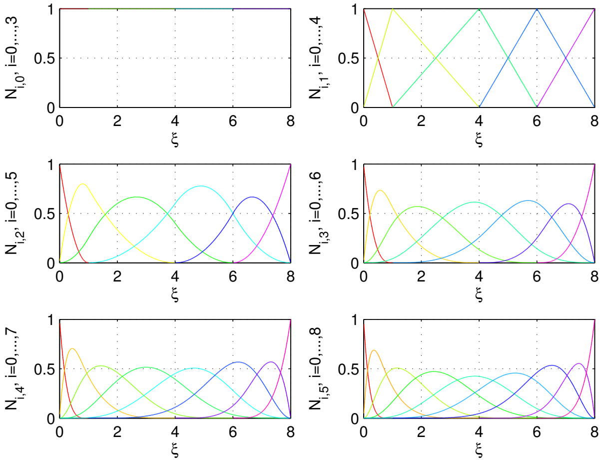
where p is the degree and m is the number of elements of the knot vector, is said to be *non-periodic* or *clamped* or *open*.

Derivatives of B-spline basis functions:

$$N_i^{p,(k)}(\xi) = p \left(\frac{N_i^{p-1,(k-1)}(\xi)}{\xi_{i+p} - \xi_i} - \frac{N_{i+1}^{p-1,(k-1)}(\xi)}{\xi_{i+p+1} - \xi_{i+1}} \right).$$

B-SPLINES basis functions (3)

Example of basis functions of degrees $p = 0, 1, 2, 3, 4, 5$



B-SPLINES

Definition

B-spline *curve*:

$$\boldsymbol{C}(\xi) = \sum_{i=0}^{n} N_i^p(\xi)\,\boldsymbol{P}_i, \;\; a \leq \xi \leq b.$$

Definition

B-spline *surface*:

$$\boldsymbol{S}(\xi,\eta) = \sum_{i=0}^{n}\sum_{j=0}^{m} N_i^p(\xi)\,N_j^q(\eta)\,\boldsymbol{P}_{i,j}, \;\; \boldsymbol{a} \leq \boldsymbol{\xi} \leq \boldsymbol{b}.$$

NURBS basis functions (1)

Definition

A NURBS basis function is defined as:

$$R_i^p(\xi) = \frac{N_i^p(\xi)\, w_i}{\sum_{\hat{i}=0}^{n} N_{\hat{i}}^p(\xi)\, w_{\hat{i}}}, \quad \boldsymbol{a} \leq \boldsymbol{\xi} \leq \boldsymbol{b},$$

$$R_{i,j}^{p,q}(\xi,\eta) = \frac{N_i^p(\xi)\, N_j^q(\eta)\, w_{i,j}}{\sum_{\hat{i}=0}^{n}\sum_{\hat{j}=0}^{m} N_{\hat{i}}^p(\xi)\, N_{\hat{j}}^q(\eta)\, w_{\hat{i},\hat{j}}}, \quad \boldsymbol{a} \leq \boldsymbol{\xi} \leq \boldsymbol{b}.$$

NURBS basis functions (2)

Derivatives of NURBS basis functions:

$$\frac{\partial R_i^p(\xi)}{\partial \xi} = \frac{w_i \dfrac{\partial N_i^p(\xi)}{\partial \xi} \cdot W(\xi) - N_i^p(\xi)\, w_i \cdot \displaystyle\sum_{\hat{i}=0}^{n} \dfrac{\partial N_{\hat{i}}^p(\xi)}{\partial \xi} w_{\hat{i}}}{(W(\xi))^2},$$

$$W(\xi) = \sum_{\hat{i}=0}^{n} N_{\hat{i}}^p(\xi)\, w_{\hat{i}}.$$

NURBS'S (1)

Definition

NURBS *curve*:

$$\boldsymbol{C}(\xi) = \sum_{i=0}^{n} R_i^p(\xi)\, \boldsymbol{P}_i, \;\; a \leq \xi \leq b.$$

Definition

NURBS *surface*:

$$\boldsymbol{S}(\xi, \eta) = \sum_{i=0}^{n} \sum_{j=0}^{m} R_{i,j}^{p,q}(\xi, \eta)\, \boldsymbol{P}_{i,j}, \;\; \boldsymbol{a} \leq \boldsymbol{\xi} \leq \boldsymbol{b}.$$

NURBS'S (2)

Definition

NURBS *solid*:

$$\boldsymbol{S}(\xi,\eta,\zeta) = \sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{k=0}^{l} R_{i,j,k}^{p,q,l}(\xi,\eta,\zeta)\,\boldsymbol{P}_{i,j,k},\ \ \boldsymbol{a} \le \boldsymbol{\xi} \le \boldsymbol{b}.$$

T-SPLINES

- *Generalization* of NURBS's;
- *local refinement*;
- *backwards-compatible* with NURBS;

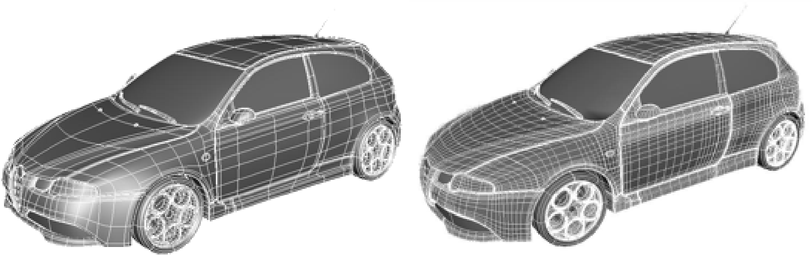


Figure: On the left car model designed using T-splines, on the right the same exact model is converted to a NURBS-based model.

Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

OVERVIEW OF ISOGEOMETRIC ANALYSIS

- The mesh is defined by the *product of knot vectors*.
  - Knot spans subdivide the *patch* into elements.
- The *support* of the basis functions is small (just like basis functions used in FEM). This leads to *sparse matrices*.
- The *isoparametric concept* is invoked: the solution field is defined by using the same basis functions used to describe the geometry.
  - The *DOFs* are the control points.
- *Boundary conditions* are applied like in FEM.

OVERVIEW OF ISOGEOMETRIC ANALYSIS

- The mesh is defined by the *product of knot vectors*.
  - Knot spans subdivide the *patch* into elements.
- The *support* of the basis functions is small (just like basis functions used in FEM). This leads to *sparse matrices*.
- The *isoparametric concept* is invoked: the solution field is defined by using the same basis functions used to describe the geometry.
  - The *DOFs* are the control points.
- *Boundary conditions* are applied like in FEM.

OVERVIEW OF ISOGEOMETRIC ANALYSIS

- The mesh is defined by the *product of knot vectors*.
  - Knot spans subdivide the *patch* into elements.
- The *support* of the basis functions is small (just like basis functions used in FEM). This leads to *sparse matrices*.
- The *isoparametric concept* is invoked: the solution field is defined by using the same basis functions used to describe the geometry.
  - The *DOFs* are the control points.
- *Boundary conditions* are applied like in FEM.

OVERVIEW OF ISOGEOMETRIC ANALYSIS

- The mesh is defined by the *product of knot vectors*.
  - Knot spans subdivide the *patch* into elements.
- The *support* of the basis functions is small (just like basis functions used in FEM). This leads to *sparse matrices*.
- The *isoparametric concept* is invoked: the solution field is defined by using the same basis functions used to describe the geometry.
  - The *DOFs* are the control points.
- *Boundary conditions* are applied like in FEM.

TRANSFORMATION TO THE PARAMETRIC SPACE

According to the isoparametric concept, the same basis functions used for the geometry (we used B-splines for instance) are used to build the *solution field*. Therefore, it can be expressed with

$$\upsilon_h(x,y) = \sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,j}^{p,q}(x,y)\,\bar{\upsilon}_{i,j},$$

and the *geometrical map* used to map points from the physical space to the parametric space is:

$$\tilde{\boldsymbol{x}}(\xi,\eta) = \sum_{i=0}^{n}\sum_{j=0}^{m} \tilde{N}_{i,j}^{p,q}(\xi,\eta)\,\boldsymbol{P}_{i,j} \in \mathcal{S}_{\Xi,H}^{p,q},$$

where $\boldsymbol{\xi} = [\xi,\eta]^T \in \tilde{\Omega} = [0,1]\times[0,1] \subset \mathbb{R}^2$.

$\{h, p, hp, ph\}$-REFINEMENTS

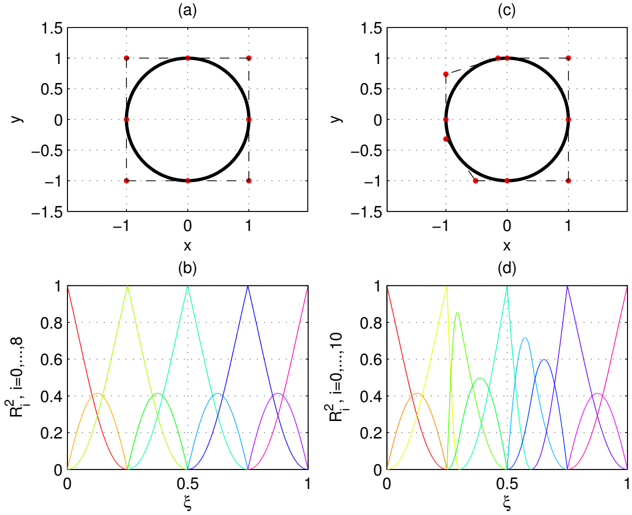
h-refinement: this kind of refinement is performed by the knot insertion procedure;

p-refinement: this kind of refinement is performed in IGA by the degree elevation procedure;

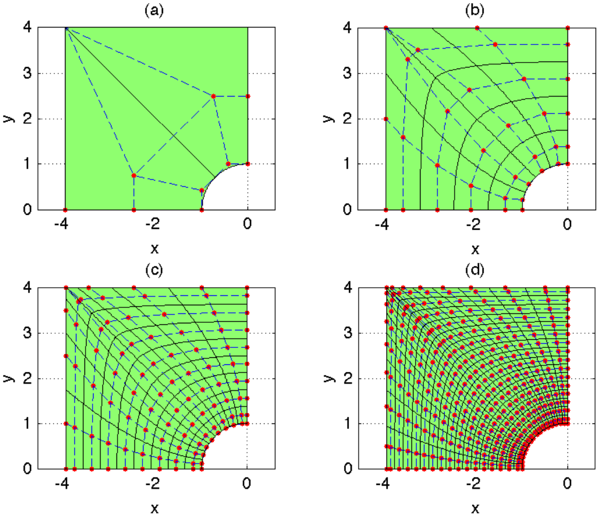
hp-refinement: this kind of refinement is performed by by knot insertion followed by degree elevation;

ph-refinement: this kind of refinement is performed by degree elevation followed by knot insertion.

EXAMPLE OF *h*-REFINEMENT ON CURVES



EXAMPLE OF *h*-REFINEMENT ON SURFACES



Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

NUMERICAL EXAMPLE (1)

Example

Example of solutions using FEM and IGA of the problem

$$\begin{cases} \nabla\left(\nabla u\left(\boldsymbol{x}\right)\right)=1, & \forall \boldsymbol{x}\in\Omega,\ \text{given } u:\bar{\Omega}\to\mathbb{R} \\ u\left(\boldsymbol{x}\right)=0, & \forall \boldsymbol{x}\in\partial\Omega \end{cases},$$

where the domain is first a square plate, and then the same plate with a hole in a corner.

SOLUTION ON SQUARE PLATE

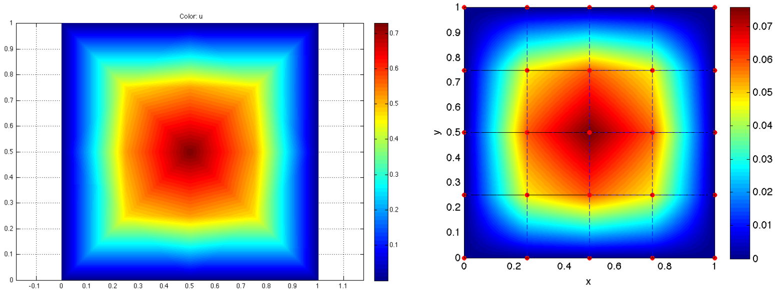
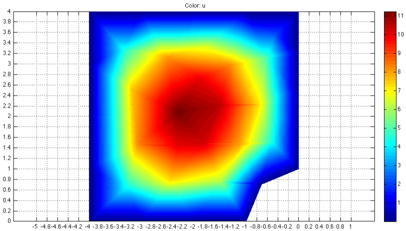
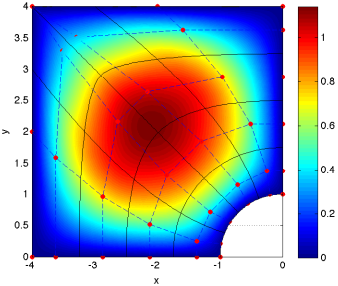
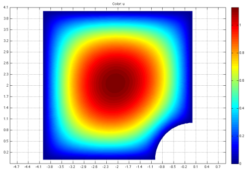


Figure: Solution using FEM with 25 nodes on the left, IGA with 25 nodes on the right.

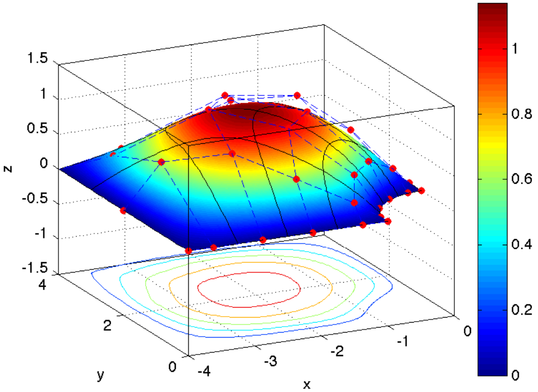
SOLUTION ON SQUARE PLATE WITH HOLE







SOLUTION ON SQUARE PLATE WITH HOLE WITH IGA



Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

DEFINITION OF THE PROBLEM

Example

Example of solutions using FEM and IGA of the problem

$$\begin{cases} \nabla\left(\nabla u\left(\boldsymbol{x}\right)\right) = 1, & \forall \boldsymbol{x} \in \Omega, \text{ given } u : \bar{\Omega} \to \mathbb{R} \\ u\left(\boldsymbol{x}\right) = 0, & \forall \boldsymbol{x} \in \Gamma_{D,0} = \left\{\boldsymbol{x} | \tilde{\boldsymbol{x}}\left(\xi,\eta\right) = \boldsymbol{x}, \xi = 0, \eta \in [0,1]\right\} \\ u\left(\boldsymbol{x}\right) = 1, & \forall \boldsymbol{x} \in \Gamma_{D,1} = \left\{\boldsymbol{x} | \tilde{\boldsymbol{x}}\left(\xi,\eta\right) = \boldsymbol{x}, \xi = 1, \eta \in [0,1]\right\} \\ u\left(\boldsymbol{x}\right) = 2, & \forall \boldsymbol{x} \in \Gamma_{D,2} = \left\{\boldsymbol{x} | \tilde{\boldsymbol{x}}\left(\xi,\eta\right) = \boldsymbol{x}, \xi \in [0,1], \eta = 1\right\} \\ \dfrac{\partial u}{\partial \boldsymbol{\nu}} = 9x, & \forall \boldsymbol{x} \in \Gamma_N = \left\{\boldsymbol{x} | \tilde{\boldsymbol{x}}\left(\xi,\eta\right) = \boldsymbol{x}, \xi \in [0,1], \eta = 0\right\} \end{cases} \tag{1}$$

where the domain is first a square plate, and then the same plate with a hole in a corner.

SOLUTION ON SQUARE PLATE

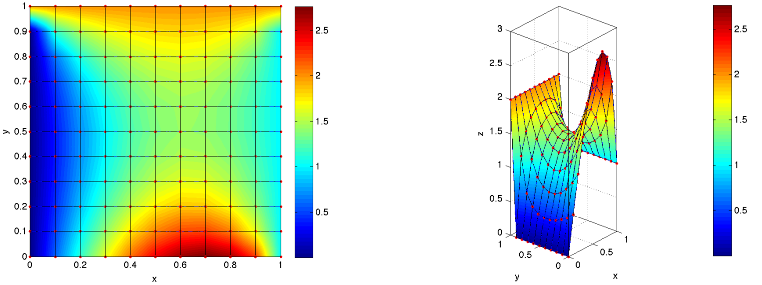


Figure: Solution using 121 nodes.

SOLUTION ON SQUARE PLATE WITH HOLE

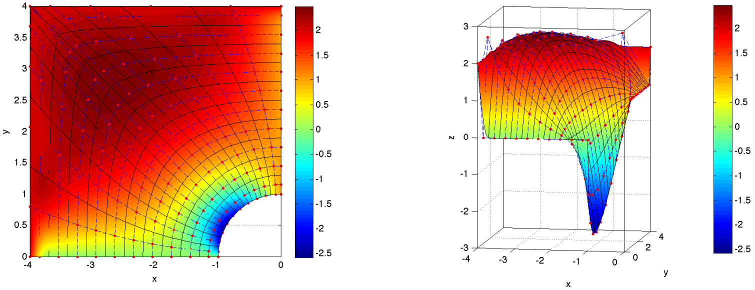


Figure: Solution using 144 nodes.

Outline

1. Introduction
   - Automated design process
   - Partial Differential Equations
   - Galerkin method
   - Isoparametric concept
2. Finite Element Method
3. Isogeometric Analysis
   - CAD basis functions
   - Analysis
4. Numerical examples
   - Numerical example (1)
   - Numerical example (2)
   - Numerical example (3)

DEFINITION OF THE PROBLEM

Example

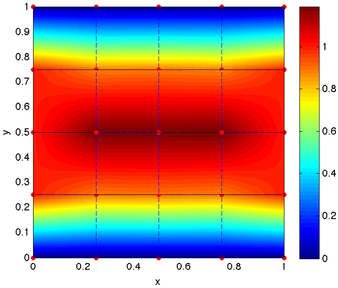
Example of solutions using FEM and IGA of the problem

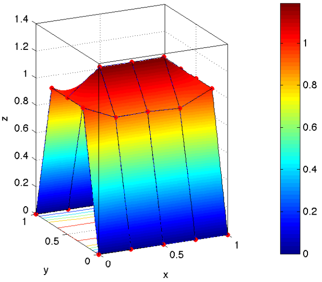
$$\begin{cases} \nabla\left(2x\mathbf{I}_2\nabla u\left(\boldsymbol{x}\right)\right) = 20x, & \forall \boldsymbol{x} \in \Omega \\ u\left(\boldsymbol{x}\right) = 1, & \forall \boldsymbol{x} \in \Gamma_D \end{cases},$$

where the domain is first a square plate, and then the same plate with a hole in a corner.

h-REFINEMENT BY KNOT INSERTION (9 NODES)

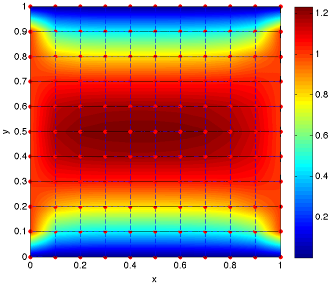
h-refinement on square plate

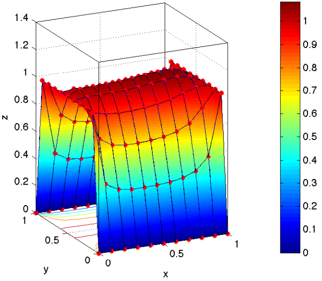




h-REFINEMENT BY KNOT INSERTION (81 NODES)

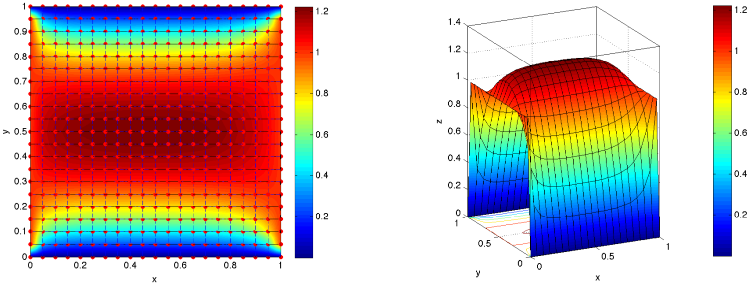
h-refinement on square plate





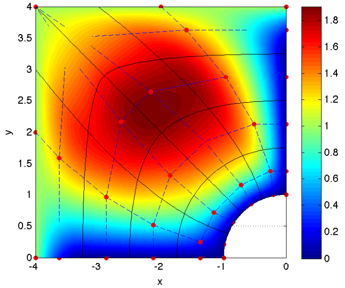
h-REFINEMENT BY KNOT INSERTION (361 NODES)

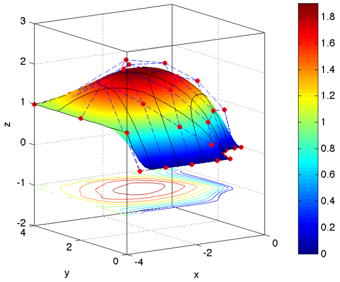
h-refinement on square plate



h-REFINEMENT BY KNOT INSERTION (16 NODES)

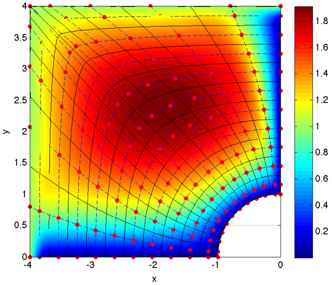
h-refinement on square plate with hole

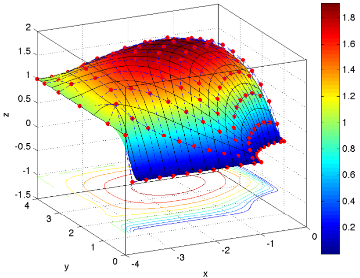




h-REFINEMENT BY KNOT INSERTION (100 NODES)

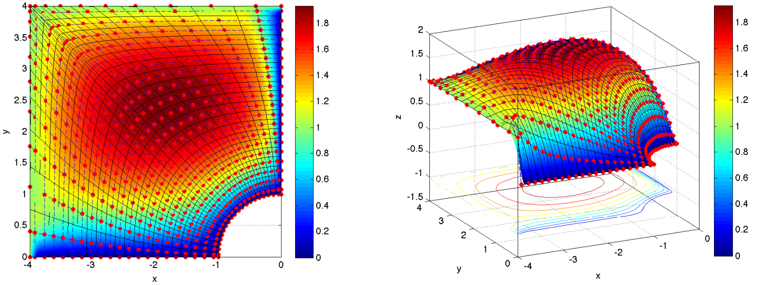
h-refinement on square plate with hole





h-REFINEMENT BY KNOT INSERTION (400 NODES)

h-refinement on square plate with hole



END

THANK YOU FOR YOUR ATTENTION!

For Further Reading I

- M. Aigner, C. Heinrich, B. Jüttler, E. Pilgerstorfer, B. Simeon, and A.-V. Vuong.
  Swept Volume Parameterization for Isogeometric Analysis.
  Proceedings of the 13th IMA International Conference on Mathematics of Surfaces XIII, 2009.

- Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg.
  Isogeometric analysis using T-splines.
  Computer Methods in Applied Mechanics and Engineering, In Press, Corrected Proof:–, 2009.

For Further Reading II

- Y. Bazilevs, L. Beirão da Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli.
  Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes.
  Mathematical Models and Methods in Applied Sciences (M3AS), 16(7):1031–1090, July 2006.

- T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl.
  Meshless methods: An overview and recent developments.
  Computer Methods in Applied Mechanics and Engineering, 139(1-4):3 – 47, 1996.

For Further Reading III

- E. Bendìto, Á. Carmona, A.M. Encinas, and J.M. Gesto.
  Estimation of Fekete Points.
  Journal of Computational Physics, 225(2):2354–2376, 2007.
- M. Bertsch and R. Dal Passo.
  Elementi di Analisi matematica.
  Aracne, 2001.
- N. Bourbaki.
  Elements of Mathematics.
  Springer-Verlag.
- A. Campanaro and M.J. Collavo.
  Modellazione agli elementi finiti di disegni CAD multidimensionali utilizzando le NURBS, 2008.

For Further Reading IV

F. Cesari.
Comportamento non lineare delle strutture col metodo degli elementi finiti.
Pitagora, 1985.

J.A. Cottrell, A. Reali, Y. Bazilevs, and T.J.R. Hughes.
Isogeometric analysis of structural vibrations.
Computer Methods in Applied Mechanics and Engineering, 195(41-43):5257 – 5296, 2006.
John H. Argyris Memorial Issue. Part II.

For Further Reading V

T. Dokken, V. Skytt, J. Haenisch, and K. Bengtsson.
Isogeometric representation and analysis - bridging the gap between CAD and analysis.
47th AIAA Aerospace Sciences Meeting inlcuding The New Horizons Forum and Aerospace Exposition, 2009.

M.R. Dörfel, B. Jüttler, and B. Simeon.
Adaptive isogeometric analysis by local h-refinement with T-splines.
Computer Methods in Applied Mechanics and Engineering, In Press, Corrected Proof:–, 2008.

P.J. Frey and P.-L. George.
Mesh Generation.
Hermes Science Publishing, 2000.

For Further Reading VI

- Weimin Han and Xueping Meng.
  Error analysis of the reproducing kernel particle method.
  Computer Methods in Applied Mechanics and Engineering, 190(46-47):6157 – 6181, 2001.
- K. Höllig.
  Finite Element Methods with B-Splines.
  Society for Industrial and Applied Mathemaics, Philadelphia, PA, USA, 2003.
- A. Huerta and J. Donea.
  Finite Element Methods for Flow Problems.
  Wiley, April 2003.

For Further Reading VII

- T.J.R. Hughes.
  The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.
  Mineola, 2000.
- T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs.
  Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement.
  Computer Methods in Applied Mechanics and Engineering, 194(39-41):4135 – 4195, 2005.
- T.J.R. Hughes, A. Reali, and G. Sangalli.
  Efficient quadrature for nurbs-based isogeometric analysis.
  Computer Methods in Applied Mechanics and Engineering, In Press, Corrected Proof:–, 2008.

For Further Reading VIII

B.M. Irons.
Engineering Application of Numerical Integration in Stiffness Method.
Journal of the American Institute of Aeronautics and Astronautics, 14:2035–2037, 1966.

Y.W. Kwon and H. Bang.
The Finite Element Method Using Matlab.
CRC Press, 1997.

F. Marcuzzi and M. Morandi Cecchi.
The Best-Apprximation Weighted-Residuals Method for Finite Element Approximations.
MASCOT08-IMACS/ISGG.

For Further Reading IX

A. Panizzo, L. Agostini, and L. Carlon.
Implementazione in C (C++) di un codice agli elementi finiti per problemi ellittici bidimensionali.
Project for the course "Calcolo Numerico", Jenuary 2008.

L. Piegl and W. Tiller.
The NURBS Book.
Springer-Verlag, 2nd edition edition, 1997.

A.D. Polyanin.
Handbook of linear partial differential equations for engineers and scientists.
CRC Press, 2002.

For Further Reading X

W. Rudin.
Functional Analysis.
McGraw-Hill Science, 1991.

S. Salsa.
Partial Differential Equations in Action: From Modelling to Theory.
Springer, 2008.

T.W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri.
T-splines and T-NURCCs.
ACM Trans. Graph., 22(3):477–484, 2003.

For Further Reading XI

- Amit Shaw and D. Roy.
  NURBS-based parametric mesh-free methods.
  Computer Methods in Applied Mechanics and Engineering, 197(17-18):1541 – 1567, 2008.
- P. Šolín.
  Partial Differential Equations and the Final Element Method.
  John Wiley & Sons, Inc., 2006.
- P. Šolín, I. Dolezel, and K. Segeth.
  Higher-Order Finite Element Methods.
  CRC Pr I Llc, 2003.

For Further Reading XII

- J. Stoer and R. Bulirsch.
  Introduction to Numerical Analysis, volume 2nd.
  Springer-Verlag, 1992.
- I.C. Taig.
  Structural Analysis by the Matrix Displacement Method.
  English Electric Aviation Report, (S017), 1961.
- R.L. Taylor and O.C. Zienkiewicz.
  The Finite Element Method, volume 1.
  Butterworth-Heinemann, fifth edition, 2000.
- The MathWorks.
  Optimization toolbox User Guide, 2008.

For Further Reading XIII

M.E. Tylor.
Partial Differential Equations I.
Springer.

W. Gander and W. Gautschi.
Adaptive Quadrature - Revisited.
BIT, 40(1):84–101, March 2000.

Y. Zhang, Y. Bazilevs, S. Goswami, C.L. Bajaj, and T.J.R. Hughes.
Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow.
Computer Methods in Applied Mechanics and Engineering, 196(29-30):2943 – 2959, 2007.

For Further Reading XIV

O.C. Zienkiewicz, J.Z. Zhu, and N.G. Gong.
Effective and practical h-p-version adaptive analysis procedures for the finite element method.
International Journal for Numerical Methods in Engineering, (28):879–891, 1989.