An introduction to Isogeometric Analysis

A. Buffa[1], G. Sangalli[1,2], R. Vázquez[1]

[1] Istituto di Matematica Applicata e Tecnologie Informatiche - Pavia
Consiglio Nazionale delle Ricerche

[2] Dipartimento di Matematica, Università di Pavia

Supported by the ERC Starting Grant: GeoPDEs n. 205004

1. An overview on IsoGeometric analysis (IGA)
   - B-splines and NURBS
   - Geometry description
   - Discretization in IGA
   - Local refinement: T-splines

2. Approximation of vector fields and differential forms
   - Construction of the discrete spaces
   - The commuting De Rham diagram
   - Maxwell eigenproblem: B-splines discretization
   - Maxwell eigenproblem: NURBS discretization

Part I

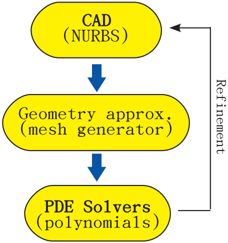
An overview on Isogeometric Analysis

IsoGeometric Analysis (IGA): an overview

Geometry is defined by Computer Aided Design (CAD) software.
CAD is based on Non Uniform Rational B-Splines (NURBS).

IsoGeometric Analysis (IGA): an overview

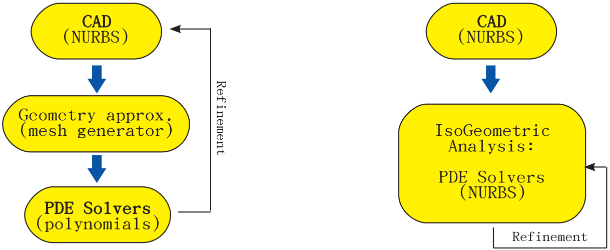
Geometry is defined by Computer Aided Design (CAD) software.
CAD is based on Non Uniform Rational B-Splines (NURBS).



- CAD and FEM use different descriptions for the geometry.
  - Iso-parametric description of the geometry.
  - Updating the geometry requires interface with CAD and remeshing.

IsoGeometric Analysis (IGA): an overview

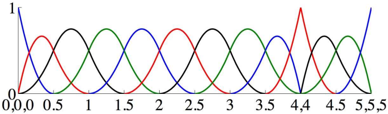
Geometry is defined by Computer Aided Design (CAD) software.
CAD is based on Non Uniform Rational B-Splines (NURBS).



- CAD and FEM use different descriptions for the geometry.
- CAD and IGA use the same geometry description.
  - Maintain the geometric description given by CAD (NURBS).
  - Iso-parametric approach: PDEs are numerically solved with NURBS.

Hughes, Cottrell, Bazilevs, CMAME, 2005

IsoGeometric Analysis: definition of B-splines

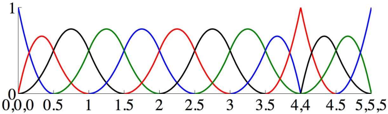


Let $\{\xi_1, \ldots, \xi_{n+p+1}\}$ be a non-uniform *knot vector* in the interval [0,1].
B-spline basis functions are defined recursively as

$$B_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise.} \end{cases}$$

$$B_{i,p}(x) = \frac{x - \xi_i}{\xi_{i+p} - \xi_i} B_{i,p-1}(x) + \frac{\xi_{i+p+1} - x}{\xi_{i+p+1} - \xi_{i+1}} B_{i+1,p-1}(x).$$

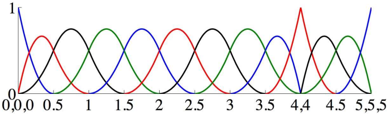
IsoGeometric Analysis: definition of B-splines



Let $\{\xi_1, \ldots, \xi_{n+p+1}\}$ be a non-uniform *knot vector* in the interval [0,1]. The main properties of **B-splines basis functions** are

- Piecewise polynomials of degree p, and regularity at most $p-1$.
- The regularity can be controlled by changing the knots multiplicity.
- The function $B_{i,p}$ is supported in the interval $[\xi_i, \xi_{i+p+1}]$.
- They are non-negative and form a partition of unity.

IsoGeometric Analysis: definition of B-splines



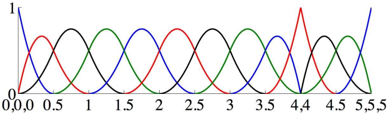
Let $\{\xi_1, \ldots, \xi_{n+p+1}\}$ be a non-uniform *knot vector* in the interval [0,1]. The main properties of **B-splines basis functions** are

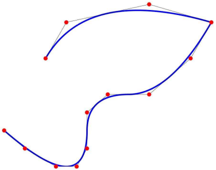
- Piecewise polynomials of degree p, and regularity at most $p-1$.
- The regularity can be controlled by changing the knots multiplicity.
- The function $B_{i,p}$ is supported in the interval $[\xi_i, \xi_{i+p+1}]$.
- They are non-negative and form a partition of unity.

S^p_α: space of B-splines of degree p and regularity α at the knots.

Their derivatives satisfy $\left\{ \frac{d}{dx} v : v \in S^p_\alpha \right\} \equiv S^{p-1}_{\alpha-1}$.

IsoGeometric Analysis: definition of B-splines

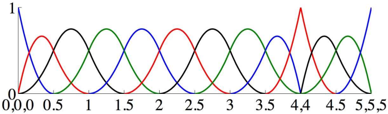


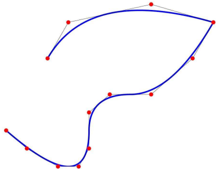


B-spline curves in $\mathbb{R}^d$

$\mathbf{F}(x) = \sum_{i=1}^{m} B_{i,p}(x)\mathbf{C}_i,$ $\mathbf{C}_i \in \mathbb{R}^d$ are the control points .

IsoGeometric Analysis: definition of B-splines





B-spline curves in $\mathbb{R}^d$

$$\mathbf{F}(x) = \sum_{i=1}^{m} B_{i,p}(x)\mathbf{C}_i, \qquad \mathbf{C}_i \in \mathbb{R}^d \text{ are the control points .}$$

Multivariate B-splines

The definition is generalized by tensor products:

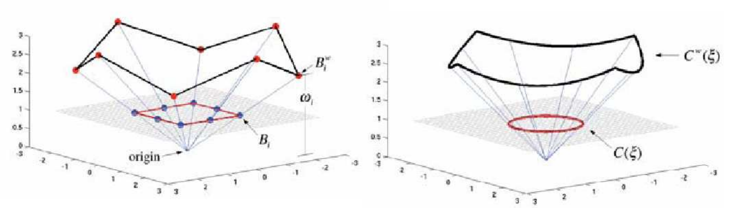
$$S^{p_1,p_2,p_3}_{\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}_3} := S^{p_1}_{\boldsymbol{\alpha}_1} \otimes S^{p_2}_{\boldsymbol{\alpha}_2} \otimes S^{p_3}_{\boldsymbol{\alpha}_3}, \qquad B_{ijk}(\mathbf{x}) := B_{i,p_1}(x)B_{j,p_2}(y)B_{k,p_3}(z).$$

B-spline volumes in $\mathbb{R}^3$

$$\mathbf{F}(\mathbf{x}) = \sum_{i,j,k=1}^{m_1,m_2,m_3} B_{ijk}(\mathbf{x})\mathbf{C}_{ijk}, \qquad \mathbf{C}_{ijk} \in \mathbb{R}^3 \text{ are the control points .}$$

IsoGeometric Analysis: definition of NURBS

NURBS in $\mathbb{R}^d$ are conic projections of B-splines in $\mathbb{R}^{d+1}$

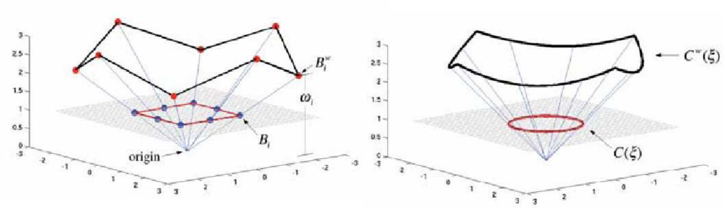


Weights, control points and basis functions:

$$w_i = (\mathbf{C}_i^w)_{d+1}, \qquad (\mathbf{C}_i)_j = \frac{(\mathbf{C}_i^w)_j}{w_i}, \qquad R_{i,p} = \frac{B_{i,p}(\xi) w_i}{\sum_{\ell=1}^m B_{\ell,p}(\xi) w_\ell}.$$

IsoGeometric Analysis: definition of NURBS

NURBS in $\mathbb{R}^d$ are conic projections of B-splines in $\mathbb{R}^{d+1}$



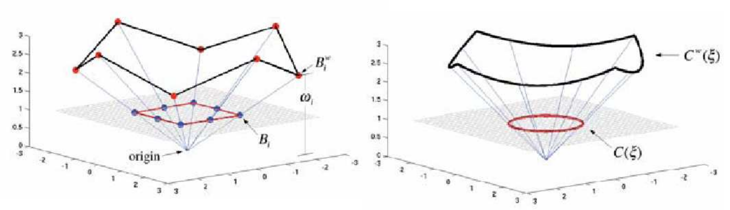
Weights, control points and basis functions:

$$w_i = (\mathbf{C}_i^w)_{d+1}, \qquad (\mathbf{C}_i)_j = \frac{(\mathbf{C}_i^w)_j}{w_i}, \qquad R_{i,p} = \frac{B_{i,p}(\xi) w_i}{\sum_{\ell=1}^m B_{\ell,p}(\xi) w_\ell}.$$

- Represent **exactly** a wide class of curves as, e.g., conic sections.
- They have most of the good properties of B-splines.

IsoGeometric Analysis: definition of NURBS

NURBS in $\mathbb{R}^d$ are conic projections of B-splines in $\mathbb{R}^{d+1}$



Weights, control points and basis functions:

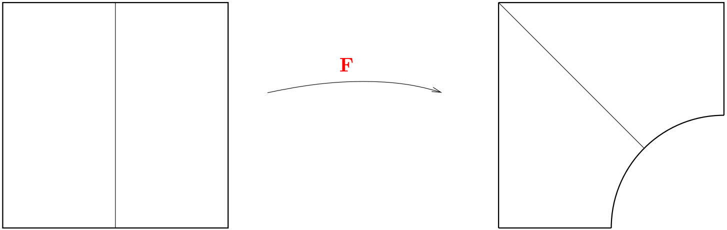
$$w_i = (\mathbf{C}_i^w)_{d+1}, \qquad (\mathbf{C}_i)_j = \frac{(\mathbf{C}_i^w)_j}{w_i}, \qquad R_{i,p} = \frac{B_{i,p}(\xi) w_i}{\sum_{\ell=1}^m B_{\ell,p}(\xi) w_\ell}.$$

- Represent **exactly** a wide class of curves as, e.g., conic sections.
- They have most of the good properties of B-splines.
- N_{α}^p: space of NURBS of degree p and regularity α at the knots.

IGA: geometry description and mesh refinement

Coarsest mesh: geometry description

"Patch", $\widehat{\Omega} = (0,1)^2$ | Physical domain Ω



$$\left\{ R_i = \frac{w_i B_i}{w} \right\}_{i=1,\dots,N_0} \qquad \left\{ \left(\frac{w_i B_i}{w} \right) \circ \mathbf{F}^{-1} \right\}_{i=1,\dots,N_0}$$

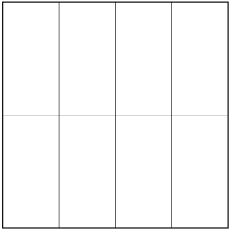
The approximation space $\mathcal{V}_h$ on Ω is obtained by push-forward:

$$\mathcal{V}_h = \mathrm{span}\{R_i \circ \mathbf{F}^{-1}, i = 1, \dots, N_0\}$$

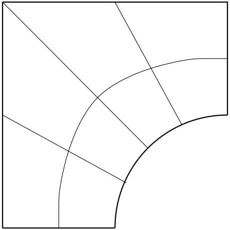
IGA: geometry description and mesh refinement

First refinement

"Patch", $\widehat{\Omega} = (0,1)^2$



Physical domain Ω



$$\left\{ R_i = \frac{w_i B_i}{w} \right\}_{i=1,\ldots,N_1}$$

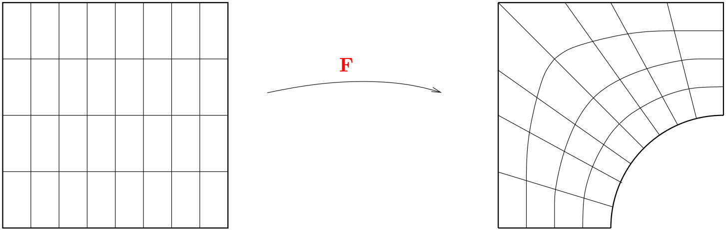
$$\left\{ \left(\frac{w_i B_i}{w} \right) \circ \mathbf{F}^{-1} \right\}_{i=1,\ldots,N_1}$$

The geometrical map **F** and the weight w are fixed at the coarsest level of discretization!

IGA: geometry description and mesh refinement

Second refinement... and so on

"Patch", $\widehat{\Omega} = (0,1)^2$ Physical domain Ω



$$\left\{ R_i = \frac{w_i B_i}{w} \right\}_{i=1,\dots,N_2} \qquad \left\{ \left(\frac{w_i B_i}{w} \right) \circ \mathbf{F}^{-1} \right\}_{i=1,\dots,N_2}$$

The main drawback is the tensor product structure.

Remark on refinement

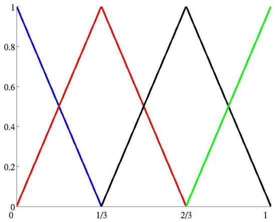
There are three possibilities for refinement:

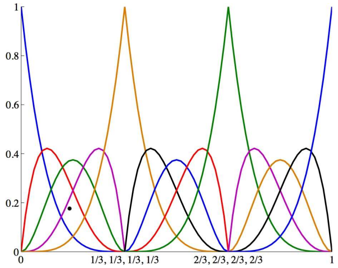
- h-refinement (by multiple knot insertion)

Remark on refinement

There are three possibilities for refinement:

- *h*-refinement (by multiple knot insertion)
- *p*-refinement (by degree elevation)

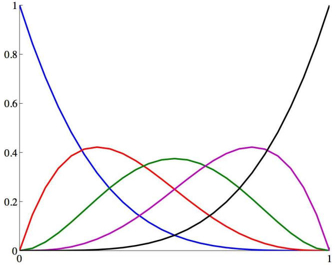


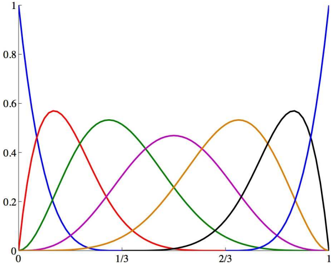


Remark on refinement

There are three possibilities for refinement:

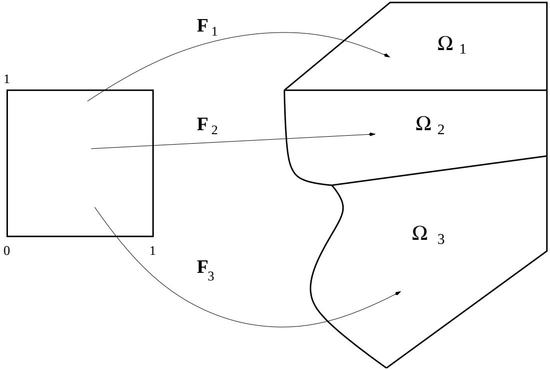
- *h*-refinement (by multiple knot insertion)
- *p*-refinement (by degree elevation)
- *k*-refinement (by degree and continuity elevation)





Multipatch domains

CAD : Geometries are described by mappings of several patches.



Patch interfaces are normally treated just imposing C^0 regularity

$\Downarrow$

domain decomposition type structure

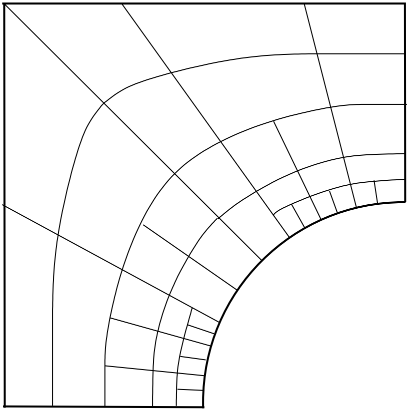
Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

CAD community (2004-): Definition of T-splines.

Based on *PB splines*, are associated with a T-mesh.

They are not based on tensor product structure.



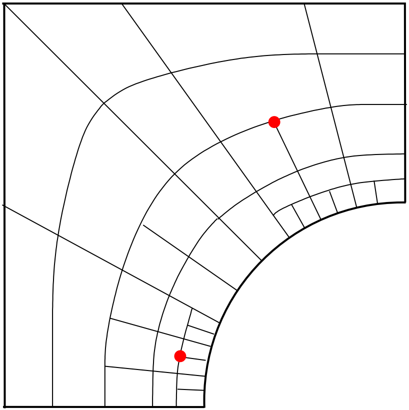
Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

CAD community (2004-): Definition of T-splines.

Based on *PB splines*, are associated with a T-mesh.

They are not based on tensor product structure.



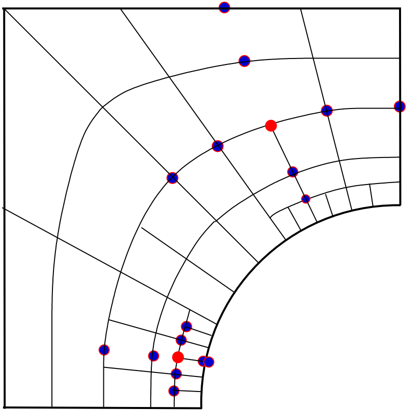
Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

CAD community (2004-): Definition of T-splines.

Based on *PB splines*, are associated with a T-mesh.

They are not based on tensor product structure.



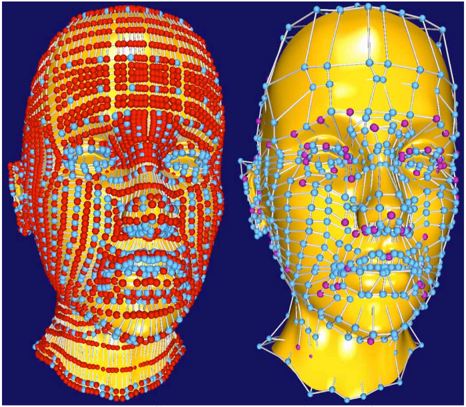
Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

CAD community (2004-): Definition of T-splines.

Based on *PB splines*, are associated with a T-mesh.

They are not based on tensor product structure.



a. NURBS   b. T-spline

Courtesy of Sederberg et al 2004

Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

CAD community (2004-): Definition of T-splines.
Based on *PB splines*, are associated with a T-mesh.
They are not based on tensor product structure.

- Refinement algorithm ensures $S_h \subseteq S_h^{\mathrm{ref}}$ (Seberberg et al)
  - Possible severe fill-in of the T-mesh
  - Expensive (cycle on many elements) and not local
  - There is no well defined de-refinement strategy!

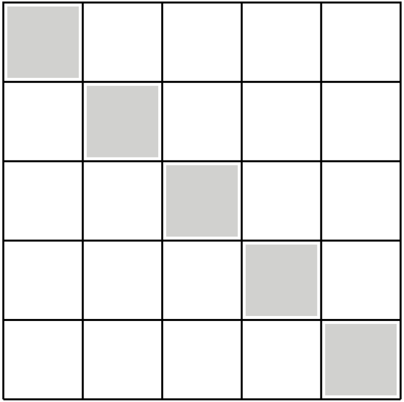
Breaking the tensor product structure: T-splines

Sederberg *et al.* 2004-, Buffa, Cho, Sangalli *et al.* 2010

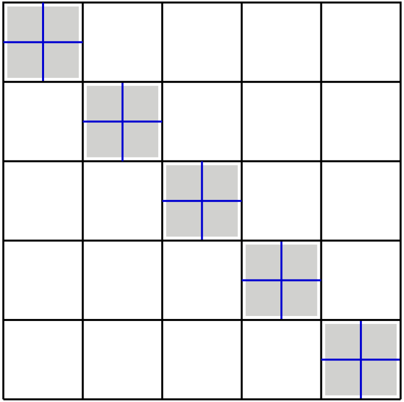
CAD community (2004-): Definition of T-splines.
Based on *PB splines*, are associated with a T-mesh.
They are not based on tensor product structure.

- Refinement algorithm ensures $S_h \subseteq S_h^{\mathrm{ref}}$ (Seberberg et al)
  - Possible severe fill-in of the T-mesh
  - Expensive (cycle on many elements) and not local
  - There is no well defined de-refinement strategy!
- Our contributions:
  - Locality analysis: C^2 versus C^1 cubic splines
  - Linear independence for fairly general meshes

Severe fill-in: the worst case scenario

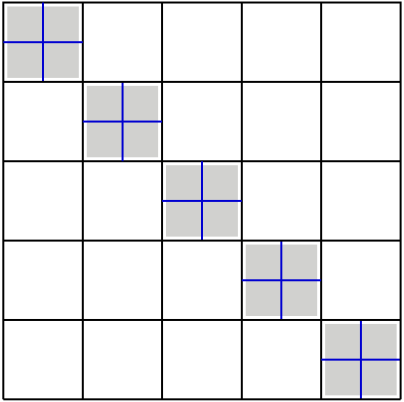


Want to refine the gray quads

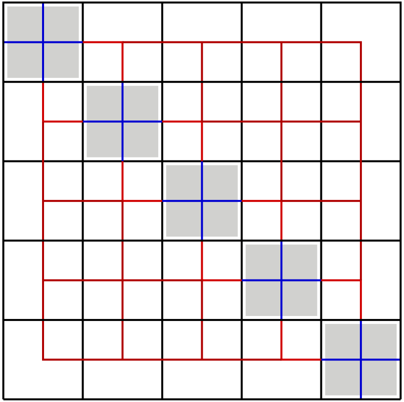


by split them in 4

Severe fill-in: the worst case scenario

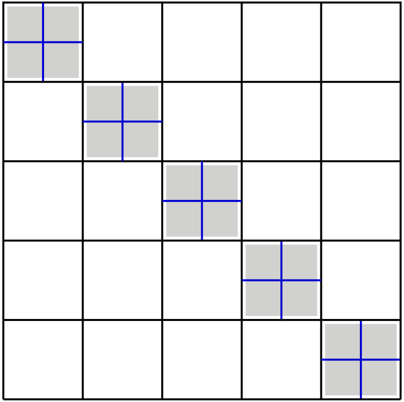


Want to refine the gray quads

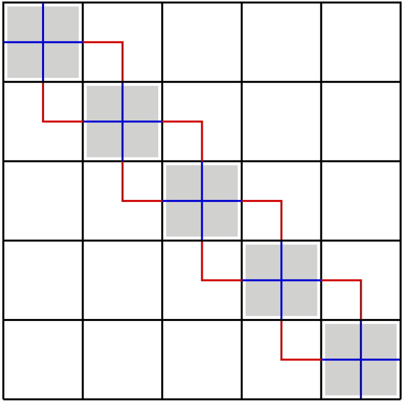


C^2 basis functions allover

Severe fill-in can be avoided

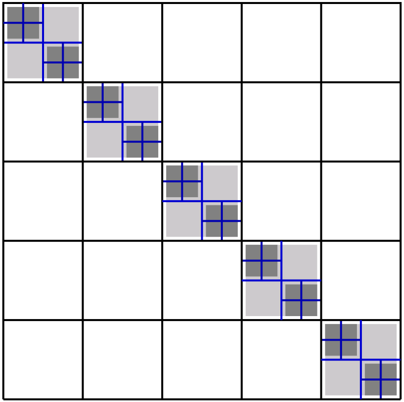


Want to refine the gray quads

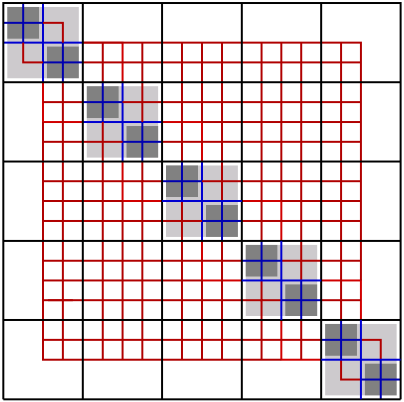


C^1 basis functions allover

Severe fill-in: second refinement step

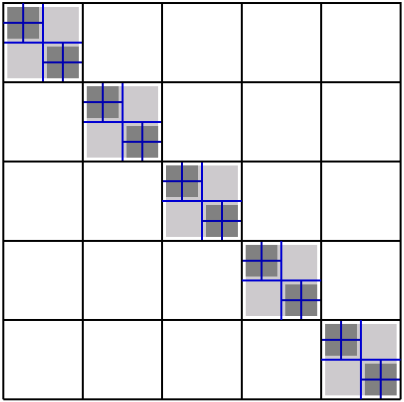


Want to refine the gray quads

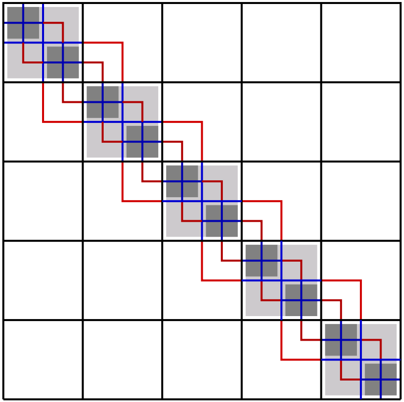


C^2 basis functions allover

Severe fill-in: second refinement step



Want to refine the gray quads

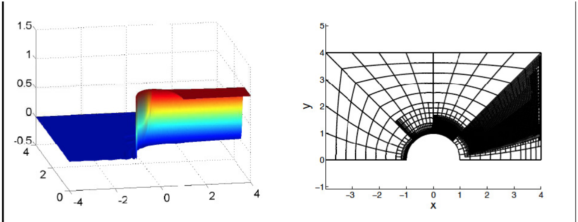


C^1 basis functions allover

Continuity in the use of T-splines

Buffa, Kumar, Sangalli. In preparation

The same behavior is observed in numerical simulations.



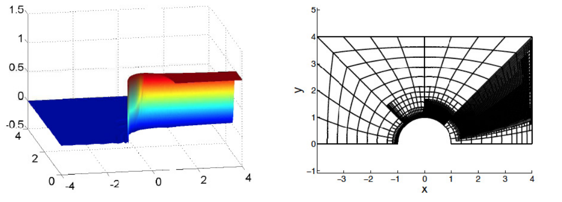
(a) Refined solution with 14875 DOF (b) T-mesh after the last step

C^2 continuity causes a fill-in of the mesh.

Continuity in the use of T-splines

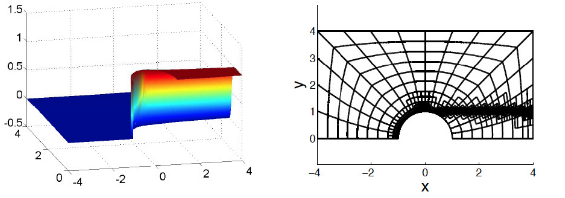
Buffa, Kumar, Sangalli. In preparation

The same behavior is observed in numerical simulations.



(a) Refined solution with 14875 DOF (b) T-mesh after the last step

C^2 continuity causes a fill-in of the mesh.



(a) Refined solution with 13129 DOF (b) T-mesh after the last step

The refinement with C^1 continuity is local.

Linearly Independent T-splines

Buffa, Cho, Sangalli, 2010

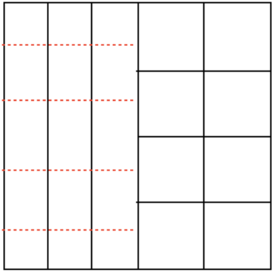
- There are examples of linearly **dependent** T-splines.
- In cases of interest, we have a result. Indeed, linear independence is guaranteed for all refinements obtained by the local refinement algorithm and that can be decomposed on successive insertion of new lines.

Linearly Independent T-splines

Buffa, Cho, Sangalli, 2010

- There are examples of linearly **dependent** T-splines.
- In cases of interest, we have a result. Indeed, linear independence is guaranteed for all refinements obtained by the local refinement algorithm and that can be decomposed on successive insertion of new lines.

E.g.,

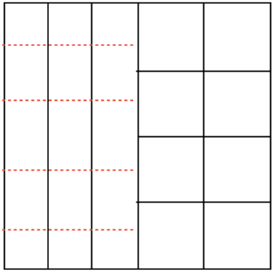


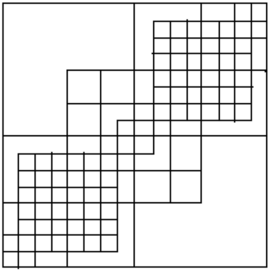
Linearly Independent T-splines

Buffa, Cho, Sangalli, 2010

- There are examples of linearly **dependent** T-splines.
- In cases of interest, we have a result. Indeed, linear independence is guaranteed for all refinements obtained by the local refinement algorithm and that can be decomposed on successive insertion of new lines.

E.g.,



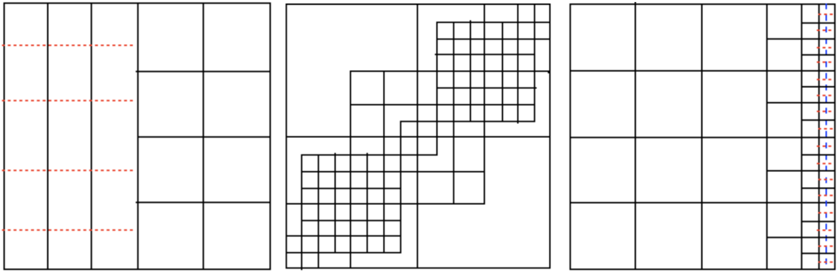


Linearly Independent T-splines

Buffa, Cho, Sangalli, 2010

- There are examples of linearly **dependent** T-splines.
- In cases of interest, we have a result. Indeed, linear independence is guaranteed for all refinements obtained by the local refinement algorithm and that can be decomposed on successive insertion of new lines.

E.g.,



Open issues

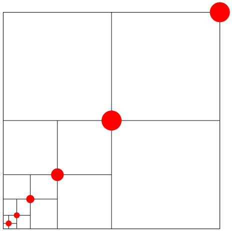
- Local or quasi local refinement algorithm, allowing for regular functions.
  The definition of "aligned" T-splines may help.
  Hughes, Scott et al. In preparation.

Open issues

- Local or quasi local refinement algorithm, allowing for regular functions.
- Linear independence on general T-meshes.

Open issues

- Local or quasi local refinement algorithm, allowing for regular functions.
- Linear independence on general T-meshes.
- Control on support intersection.



Open issues

- Local or quasi local refinement algorithm, allowing for regular functions.
- Linear independence on general T-meshes.
- Control on support intersection.
- Local approximation properties.

  Beirão da Veiga, Buffa, Cho, Sangalli. In preparation.

Open issues

- Local or quasi local refinement algorithm, allowing for regular functions.
- Linear independence on general T-meshes.
- Control on support intersection.
- Local approximation properties.
- Suitable derefinement strategy.

Open issues

- Local or quasi local refinement algorithm, allowing for regular functions.
- Linear independence on general T-meshes.
- Control on support intersection.
- Local approximation properties.
- Suitable derefinement strategy.

Some of these problems may be solved with LR-splines (locally refined).

Dokken, Lyche et al. In preparation