Efficient quasi-interpolation in hierarchical spline spaces

Carla Manni

University of Rome “Tor Vergata”
Department of Mathematics

joint work with Hendrik Speleers

SIMAI 2018
MS-02 Recent Advances in Quasi-Interpolations and Applications
Roma, July 6, 2018
Outline

1 Motivation
2 Hierarchical spline model
3 Properties of the THB-spline basis
4 Hierarchical quasi-interpolation
5 Closure
Quasi-Interpolation

quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

\[ Qf := \sum_i \lambda_i(f)B_i \]

- \( \lambda_i \) local linear functionals (easy to compute) on \( f \);
- \( \{B_i\} \) suitable set of functions:
  - compact support,
  - convex partition of unity,
- \( S := \text{span}(\ldots, B_i, \ldots) \). Usually, \( P_k \subset S \)
- to completely exploit the approximation power of \( S \)

\[ Qp = p, \quad \forall \ p \in P_k. \]
Quasi-Interpolation

- quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

\[ Qf := \sum \lambda_i(f)B_i \]

- \( \lambda_i \) local linear functionals (easy to compute) on \( f \);
- \( \{B_i\} \) suitable set of functions:
  - compact support,
  - convex partition of unity,
- \( S := \text{span}(\ldots, B_i, \ldots) \). Usually, \( \mathbb{P}_k \subset S \)
- to completely exploit the approximation power of \( S \)

\[ Qp = p, \quad \forall \ p \in \mathbb{P}_k. \]
Quasi-Interpolation

- quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

\[ Qf := \sum_{i} \lambda_i(f)B_i \]

- \( \lambda_i \) local linear functionals (easy to compute) on \( f \);
- \( \{B_i\} \) suitable set of functions:
  - compact support,
  - convex partition of unity,
- \( S := \text{span}\{\ldots, B_i, \ldots\} \). Usually, \( \mathbb{P}_k \subset S \)
- to completely exploit the approximation power of \( S \)

\[ Qp = p, \quad \forall \quad p \in \mathbb{P}_k. \]
Quasi-Interpolation

- quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

\[ Qf := \sum_i \lambda_i(f) B_i \]

- \( \lambda_i \) local linear functionals (easy to compute) on \( f \);
- \( \{B_i\} \) suitable set of functions:
  - compact support,
  - convex partition of unity,
- \( S := \text{span}(\ldots, B_i, \ldots) \). Usually, \( \mathbb{P}_k \subset S \)
- to completely exploit the approximation power of \( S \)

\[ Qp = p, \quad \forall \quad p \in \mathbb{P}_k. \]
Quasi-Interpolation

- quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

\[ Qf := \sum_i \lambda_i(f)B_i \]

- \( \lambda_i \) local linear functionals (easy to compute) on \( f \);
- \( \{B_i\} \) suitable set of functions:
  - compact support,
  - convex partition of unity,
- \( S := \text{span}(\ldots, B_i, \ldots) \). Usually, \( \mathbb{P}_k \subset S \)
- to completely exploit the approximation power of \( S \)

\[ Qp = p, \quad \forall \quad p \in \mathbb{P}_k. \]
quasi-interpolation (q.i.) is a general approach to construct, with low computational cost, efficient local approximants to a given set of data or a given function

$$Qf := \sum_i \lambda_i(f)B_i$$

- $\lambda_i$ local linear functionals (easy to compute) on $f$;
- $\{B_i\}$ suitable set of functions:
  - compact support,
  - convex partition of unity,
- $S := \text{span}(\{\ldots, B_i, \ldots\})$. Usually, $\mathbb{P}_k \subset S$
- to completely exploit the approximation power of $S$

$$Qp = p, \quad \forall \ p \in \mathbb{P}_k.$$
Motivation

Splines and B-splines are of interest in a wide range of areas:

- established tool: geometric modelling, approximation theory
- recently: isogeometric analysis (paradigm for solving PDEs)

In higher dimensions usually based on tensor-product topology

- computationally efficient, geometrically intuitive
- restriction to rectangular meshes
  → not well suited for adaptive refinement
**Motivation**

- Splines and B-splines are of interest in a wide range of areas:
  - established tool: geometric modelling, approximation theory
  - recently: isogeometric analysis (paradigm for solving PDEs)

- In higher dimensions usually based on tensor-product topology
  - ☺ computationally efficient, geometrically intuitive
  - 😞 restriction to rectangular meshes
    → not well suited for adaptive refinement
Motivation

- Splines and B-splines are of interest in a wide range of areas:
  - established tool: geometric modelling, approximation theory
  - recently: isogeometric analysis (paradigm for solving PDEs)

- In higher dimensions usually based on tensor-product topology
  - ☺ computationally efficient, geometrically intuitive
  - 😞 restriction to rectangular meshes
    → not well suited for adaptive refinement
Motivation

- Splines and B-splines are of interest in a wide range of areas:
  - established tool: geometric modelling, approximation theory
  - recently: isogeometric analysis (paradigm for solving PDEs)

- In higher dimensions usually based on tensor-product topology
  - ⌋ computationally efficient, geometrically intuitive
  - 😞 restriction to rectangular meshes
    → not well suited for adaptive refinement
Motivation

- Splines and B-splines are of interest in a wide range of areas:
  - established tool: geometric modelling, approximation theory
  - recently: isogeometric analysis (paradigm for solving PDEs)
- In higher dimensions usually based on tensor-product topology
  - 😊 computationally efficient, geometrically intuitive
  - 😞 restriction to rectangular meshes
    → not well suited for adaptive refinement
Motivation

- Splines and B-splines are of interest in a wide range of areas:
  - established tool: geometric modelling, approximation theory
  - recently: isogeometric analysis (paradigm for solving PDEs)

- In higher dimensions usually based on tensor-product topology
  - ☺ computationally efficient, geometrically intuitive
  - ☹ restriction to rectangular meshes
    → not well suited for adaptive refinement
Motivation

Splines and B-splines are of interest in a wide range of areas:
- established tool: geometric modelling, approximation theory
- recently: isogeometric analysis (paradigm for solving PDEs)

In higher dimensions usually based on tensor-product topology
- ☺ computationally efficient, geometrically intuitive
- ☹ restriction to rectangular meshes
  → not well suited for adaptive refinement
Motivation

Alternative spline spaces

- T-splines
- LR-splines
- Splines on T-meshes
- Splines on triangulations
- Splines on unstructured quad meshes
- ... 
- Hierarchical splines
  - local refinement
  - any degree, any smoothness, any dimension
  - basis: linear independence, convex partition of unity, stability
  - easy construction of quasi-interpolants
Motivation

Alternative spline spaces

- T-splines
- LR-splines
- Splines on T-meshes
- Splines on triangulations
- Splines on unstructured quad meshes
- ... 

Hierarchical splines

- local refinement
- any degree, any smoothness, any dimension
- basis: linear independence, convex partition of unity, stability
- easy construction of quasi-interpolants
Hierarchical B-splines: construction

[Forsey, Bartels, 1988]
[Kraft, 1997]

- sequence of $n$ nested (tensor-product) spline spaces on $\Omega^0$
  \[ \mathbb{V}^0 \subset \mathbb{V}^1 \subset \cdots \subset \mathbb{V}^{n-1} \]
  each spline space $\mathbb{V}^\ell$ spanned by normalized B-spline basis
  \[ \mathcal{B}^\ell = \{ B_{i,\ell}, \ i = 1, \ldots, N_\ell \} \]
- sequence of $n$ sets (domains)
  \[ \Omega^0 \supset \Omega^1 \supset \cdots \supset \Omega^{n-1} \]
Hierarchical B-splines: construction

[Forsey, Bartels, 1988]
[Kraft, 1997]

- sequence of \( n \) nested (tensor-product) spline spaces on \( \Omega^0 \)
  \[
  \mathbb{V}^0 \subset \mathbb{V}^1 \subset \cdots \subset \mathbb{V}^{n-1}
  \]
  each spline space \( \mathbb{V}^\ell \) spanned by normalized B-spline basis
  \[
  \mathcal{B}^\ell = \{ B_{i,\ell}, \ i = 1, \ldots, N_\ell \}
  \]
- sequence of \( n \) sets (domains)
  \[
  \Omega^0 \supset \Omega^1 \supset \cdots \supset \Omega^{n-1}
  \]

hierarchical domain
Hierarchical B-splines: construction

[Forsey, Bartels, 1988]
[Kraft, 1997]

- sequence of $n$ nested (tensor-product) spline spaces on $\Omega^0$
  
  $\forall 0 \subset \forall 1 \subset \cdots \subset \forall^{n-1}$

- each spline space $\forall^\ell$ spanned by normalized B-spline basis
  
  $B^\ell = \{B_{i,\ell}, \ i = 1, \ldots, N_\ell\}$

- sequence of $n$ sets (domains)
  
  $\Omega^0 \supset \Omega^1 \supset \cdots \supset \Omega^{n-1}$

**Note:**

$\text{supp}^0(f) = \text{supp}(f) \cap \Omega^0$
Hierarchical model

Hierarchical B-splines: 1D example

degree 2

refined region
Hierarchical model

Hierarchical B-splines: 1D example

degree 2
Hierarchical model

Hierarchical B-splines: 1D example

degree 2

refined region
Hierarchical model

Hierarchical B-splines: 1D example

degree 2

refined region
Hierarchical model

Hierarchical B-splines: construction

recursive definition of $\mathcal{H}$:

(I) initialization: $\mathcal{H}^0 = \{ B_{i,0} \in \mathcal{B}^0 : \text{supp}^0(B_{i,0}) \neq \emptyset \}$

(II) recursive case: construct $\mathcal{H}^{\ell+1}$ from $\mathcal{H}^\ell$

$$\mathcal{H}^{\ell+1} = \mathcal{H}^{\ell+1}_A \cup \mathcal{H}^{\ell+1}_B, \quad \ell = 0, \ldots, n - 2,$$

where

$$\mathcal{H}^{\ell+1}_A = \{ B_{i,j} \in \mathcal{H}^\ell : \text{supp}^0(B_{i,j}) \nsubseteq \Omega^{\ell+1} \}$$

$$\mathcal{H}^{\ell+1}_B = \{ B_{i,\ell+1} \in \mathcal{B}^{\ell+1} : \text{supp}^0(B_{i,\ell+1}) \subseteq \Omega^{\ell+1} \}$$

(III) $\mathcal{H} = \mathcal{H}^{n-1}$
Hierarchical B-splines: construction

recursive definition of \( \mathcal{H} \):

(I) initialization: \( \mathcal{H}^0 = \{ B_{i,0} \in \mathcal{B}^0 : \text{supp}^0(B_{i,0}) \neq \emptyset \} \)

(II) recursive case: construct \( \mathcal{H}^{\ell+1} \) from \( \mathcal{H}^{\ell} \)

\[
\mathcal{H}^{\ell+1} = \mathcal{H}^{\ell+1}_A \cup \mathcal{H}^{\ell+1}_B , \quad \ell = 0, \ldots, n - 2, 
\]

where

\[
\mathcal{H}^{\ell+1}_A = \{ B_{i,j} \in \mathcal{H}^{\ell} : \text{supp}^0(B_{i,j}) \not\subseteq \Omega^{\ell+1} \} 
\]

\[
\mathcal{H}^{\ell+1}_B = \{ B_{i,\ell+1} \in \mathcal{B}^{\ell+1} : \text{supp}^0(B_{i,\ell+1}) \subseteq \Omega^{\ell+1} \} 
\]

(III) \( \mathcal{H} = \mathcal{H}^{n-1} \)
Hierarchical model

**Hierarchical spline space**
- ✓ local refinement
- ✓ any degree, any smoothness, any dimension

**Hierarchical basis**
- ✓ linearly independent
- ✓ nonnegative
- ✗ no partition of unity
  
  [Kraft, 1997]
Hierarchical model

Hierarchical spline space

- ✓ local refinement
- ✓ any degree, any smoothness, any dimension

Hierarchical basis

- ✓ linearly independent
- ✓ nonnegative
- X no partition of unity
[ Kraft, 1997 ]

Normalization: truncated basis

- ✓ linearly independent
- ✓ nonnegative
- ✓ partition of unity
- ✓ smaller support
- ✓ improved stability
[ Giannelli, Jüttler, Speleers, 2012 ]
Hierarchical model

From hierarchical basis $\mathcal{H}$ to truncated basis $\mathcal{T}$

- by subdivision, we can write every $f \in \mathcal{V}^\ell \subset \mathcal{V}^\ell+1$

$$f = \sum_{i=1}^{N_{\ell+1}} c_{i,\ell+1} B_{i,\ell+1}$$

- truncation mechanism:

$$\text{trunc}^{\ell+1}(f) = \sum_{i : \text{supp}^0(B_{i,\ell+1}) \not\subset \Omega^\ell+1} c_{i,\ell+1} B_{i,\ell+1}$$
Hierarchical model

From hierarchical basis $\mathcal{H}$ to truncated basis $\mathcal{T}$

- by subdivision, we can write every $f \in \mathcal{V}^\ell \subset \mathcal{V}^{\ell+1}$

\[
f = \sum_{i=1}^{N_{\ell+1}} c_{i,\ell+1} B_{i,\ell+1}
\]

- truncation mechanism:

\[
\text{trunc}^{\ell+1}(f) = \sum_{i: \text{supp}^0(B_{i,\ell+1}) \not\subseteq \Omega^{\ell+1}} c_{i,\ell+1} B_{i,\ell+1}
\]
Hierarchical model

Truncated B-splines: 1D example

degree 2

refined region
Hierarchical model

Truncated B-splines: 1D example

degree 2

refined region
Hierarchical model

Truncated B-splines: 1D example

degree 2

refined region
Hierarchical model

Truncated B-splines: 1D example

degree 2

refined region
Hierarchical model

Truncated B-splines: 2D example

bidegree \([2, 2]\)

level \(\ell\)

level \(\ell + 1\)

level \(\ell + 2\)
Hierarchical model

Truncated B-splines: 2D example

bidegree \([2, 2]\)

level \(\ell\)
Hierarchical model

Truncated B-splines: 2D example

bidegree \([2, 2]\)

level \(\ell + 1\)
Hierarchical model

Truncated B-splines: 2D example

bidegree \([2, 2]\)

level \(\ell + 2\)
Truncated basis: properties

Truncated basis $\mathcal{T}$: another basis for hierarchical spline space

With respect to classical hierarchical basis $\mathcal{H}$:

1. $\text{span } \mathcal{H} = \text{span } \mathcal{T}$
2. the truncated basis functions are nonnegative and linearly independent
3. reduced support of coarse basis functions; reduced overlap of basis supports $\rightarrow$ sparser systems
Truncated basis: properties

Truncated basis $\mathcal{T}$: another basis for hierarchical spline space

With respect to classical hierarchical basis $\mathcal{H}$:

1. $\text{span } \mathcal{H} = \text{span } \mathcal{T}$
2. the truncated basis functions are nonnegative and linearly independent
3. reduced support of coarse basis functions; reduced overlap of basis supports $\rightarrow$ sparser systems
Truncated basis $\mathcal{T}$: another basis for hierarchical spline space

With respect to classical hierarchical basis $\mathcal{H}$:

1. $\text{span } \mathcal{H} = \text{span } \mathcal{T}$
2. the truncated basis functions are nonnegative and linearly independent
3. reduced support of coarse basis functions; reduced overlap of basis supports $\to$ sparser systems
Truncated basis: properties

Truncated basis $\mathcal{T}$: another basis for hierarchical spline space

(4) preservation of coefficients:

\[
\text{let } I_\ell := \{ i : B_{i,\ell} \in B^\ell \cap H \}, \text{ let } D^\ell := \Omega^\ell \setminus \Omega^{\ell+1}, \text{ and let }
\]
\[
f|_{D^\ell} = \sum_{k=0}^{n-1} \sum_{i \in I_k} c^\mathcal{T}_{i,k} B^\ell_{i,k}|_{D^\ell} = \sum_{j=1}^{N_\ell} c_{j,\ell} B_{j,\ell}|_{D^\ell}, \quad \forall \ell
\]

then 
\[
c^\mathcal{T}_{i,\ell} = c_{i,\ell}, \quad i \in I_\ell
\]

(5) the truncated basis forms a partition of unity on $\Omega^0$

⇒ convex partition of unity

[Giannelli, Jüttler, Speleers, 2012] [Giannelli, Jüttler, Speleers, 2014]
Truncated basis $\mathcal{T}$: another basis for hierarchical spline space

(4) preservation of coefficients:

let $\mathcal{I}_\ell := \{ i : B_{i,\ell} \in \mathcal{B}^\ell \cap \mathcal{H} \}$, let $D^\ell := \Omega^\ell \setminus \Omega^{\ell+1}$, and let

$$f|_{D^\ell} = \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_k} c_{i,k}^\mathcal{T} B_{i,k}^\mathcal{T}|_{D^\ell} = \sum_{j=1}^{N_\ell} c_{j,\ell} B_{j,\ell}|_{D^\ell}, \quad \forall \ell$$

then

$$c_{i,\ell}^\mathcal{T} = c_{i,\ell}, \quad i \in \mathcal{I}_\ell$$

(5) the truncated basis forms a partition of unity on $\Omega^0$

$\Rightarrow$ convex partition of unity

[Giannelli, Jüttler, Speleers, 2012] [Giannelli, Jüttler, Speleers, 2014]
Hierarchical model

Truncated Hierarchical B-splines: older than expected

[Speleers, Dierckx, Vandewalle, CAGD 2009]
Hierarchical quasi-interpolants [Speleers, Manni, 2016]

- QI based on truncated hierarchical basis
  - convex partition of unity
  - small support
  - preservation of coefficients

Consider a sequence of QIs in $\mathbb{V}^\ell$, $\ell = 0, \ldots, n-1$

$$Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}$$

then

$$Q(f) := \sum_{\ell=0}^{n-1} \sum_{i \in I^\ell} \lambda_{i,\ell}(f) B_{i,\ell}$$
Hierarchical quasi-interpolants [Speleers, Manni, 2016]

- QI based on truncated hierarchical basis
  - convex partition of unity $\rightarrow$ numerical stability
  - small support $\rightarrow$ local control
  - preservation of coefficients $\rightarrow$ easy construction

Consider a sequence of QIs in $V^\ell$, $\ell = 0, \ldots, n - 1$

$$Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}$$

then

$$Q(f) := \sum_{k=0}^{n-1} \sum_{i \in I_k} \lambda_{i,k}(f) B_{i,k}^T$$
Hierarchical quasi-interpolants [Speleers, Manni, 2016]

- QI based on truncated hierarchical basis
  - convex partition of unity $\rightarrow$ numerical stability
  - small support $\rightarrow$ local control
  - preservation of coefficients $\rightarrow$ easy construction

- Consider a sequence of QIs in $\mathcal{V}^\ell$, $\ell = 0, \ldots, n - 1$

\[
Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}
\]

then

\[
Q(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_k} \lambda_{i,k}(f) B_{i,k}^T
\]
Motivation  H-model  Properties  QI  Closure  Construction  Examples

Quasi-interpolation

Hierarchical quasi-interpolants [Speleers, Manni, 2016]

- QI based on truncated hierarchical basis
  - convex partition of unity $\rightarrow$ numerical stability
  - small support $\rightarrow$ local control
  - preservation of coefficients $\rightarrow$ easy construction

- Consider a sequence of QIs in $\mathbb{V}^\ell$, $\ell = 0, \ldots, n - 1$

\[
Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}
\]

then

\[
Q(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_k} \lambda_{i,k}(f) B^T_{i,k}
\]
Hierarchical quasi-interpolants

- Let \( Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}, \quad \ell = 0, \ldots, n - 1 \)
- then \( Q(f) := \sum_{k=0}^{n-1} \sum_{i \in I_k} \lambda_{i,k}(f) B_{i,k}^T \)

- Polynomial reproduction \((\mathbb{P}_p \subset \mathbb{V}^0)\):
  \[
  \text{if } Q^\ell(g) = g, \quad \forall g \in \mathbb{P}_p, \ \forall \ell \quad \text{then } Q(g) = g, \quad \forall g \in \mathbb{P}_p
  \]

- Spline reproduction (projector):
  \[
  \text{if } \begin{cases} 
  Q^\ell(s) = s, & \forall s \in \mathbb{V}^\ell, \ \forall \ell \\
  \lambda_{i,\ell} \text{ is supported in } \Omega^\ell \setminus \Omega^{\ell+1}
  \end{cases} \quad \text{then } Q(s) = s, \quad \forall s \in S
  \]
**Quasi-interpolation**

Hierarchical quasi-interpolants

- Let \( Q^\ell(f) := \sum_{i=1}^{N\ell} \lambda_{i,\ell}(f) B_{i,\ell}, \quad \ell = 0, \ldots, n - 1 \)

  then \( Q(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_k} \lambda_{i,k}(f) B_{i,k}^T \)

- **Polynomial reproduction** \((\mathbb{P}_p \subset \mathbb{V}^0)\):

  if \( Q^\ell(g) = g, \quad \forall g \in \mathbb{P}_p, \quad \forall \ell \) then \( Q(g) = g, \quad \forall g \in \mathbb{P}_p \)

- **Spline reproduction** (projector):

  if \( \begin{cases} Q^\ell(s) = s, & \forall s \in \mathbb{V}^\ell, \quad \forall \ell \\ \lambda_{i,\ell} \text{ is supported in } \Omega^\ell \setminus \Omega^{\ell+1} \end{cases} \)

  then \( Q(s) = s, \quad \forall s \in \mathbb{S} \)
Hierarchical quasi-interpolants

- Let \( Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell} \), \( \ell = 0, \ldots, n - 1 \)

  then \( Q(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_k} \lambda_{i,k}(f) B^T_{i,k} \)

- **Polynomial reproduction** (\( \mathbb{P}_p \subset \mathbb{V}^0 \)):
  
  if \( Q^\ell(g) = g, \ \forall g \in \mathbb{P}_p, \ \forall \ell \) then \( Q(g) = g, \ \forall g \in \mathbb{P}_p \)

- **Spline reproduction** (projector):
  
  if \( \left\{ \begin{array}{l} Q^\ell(s) = s, \ \forall s \in \mathbb{V}^\ell, \ \forall \ell \\ \lambda_{i,\ell} \text{ is supported in } \Omega^\ell \setminus \Omega^{\ell+1} \end{array} \right\} \) then \( Q(s) = s, \ \forall s \in \mathcal{S} \)
Hierarchical quasi-interpolants

- Let $Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}$, $\ell = 0, \ldots, n-1$

then $Q(f) := \sum_{k=0}^{n-1} \sum_{i \in I_\ell} \lambda_{i,k}(f) B_{i,k}^T$

- Spline reproduction (projector):

  if \[
  \begin{cases}
  Q^\ell(s) = s, & \forall s \in \mathbb{V}^{\ell}, \forall \ell \\
  \lambda_{i,\ell} \text{ is supported in } \Omega^{\ell} \setminus \Omega^{\ell+1}
  \end{cases}
  \]

  then $Q(s) = s, \forall s \in S$

- Projector $\rightarrow$ dual basis:

  $\lambda_{i,\ell}(B^T_{j,k}) = \delta_{i,j}\delta_{\ell,k}$  \hspace{1cm} ($\delta_{r,s}$: Kronecker delta)
Quasi-interpolation

Hierarchical quasi-interpolants

- Let $\gamma$ be a cell, and
  \[
  \Lambda_\gamma = \text{conv} \left( \bigcup_{(i,\ell) : \text{supp}^0(B_{i,\ell}) \cap \gamma \neq \emptyset} \Lambda_{i,\ell} \cup \gamma \right), \quad (\Lambda_{i,\ell} : \text{support of } \lambda_{i,\ell})
  \]

- Full approximation order [Speleers, Manni, 2016] [Speleers, 2017]:
  \[
  \text{if } f \in C^{p+1}(\Lambda_\gamma) \text{ and } Q(g) = g, \ \forall g \in P_p \text{ then }
  \]
  \[
  \|D^\alpha(f - Q(f))\|_{L_q(\gamma)} \leq K (\text{diam}(\Lambda_\gamma))^{p+1-|\alpha|} \sum_{|\beta|=p+1} \|D^\beta f\|_{L_q(\Lambda_\gamma)}
  \]

Quasi-interpolation

Numerical examples: setup

- $C^1$ quadratic bivariate tensor-product B-splines

  - Building block $\tilde{Q}^\ell(f) := \sum_{i=1}^{N_\ell} \tilde{\lambda}_{i,\ell}(f) B_{i,\ell}$:
    - choose a cell $\Upsilon_{i,\ell}$ in support of each $B_{i,\ell}$
    - choose $3 \times 3$ points $x_{j,i,\ell} \in \Upsilon_{i,\ell}, j = 1, \ldots, 9$
    - solve the system
      $$\sum_{k: \text{supp}(B_{k,\ell}) \cap \Upsilon_{i,\ell} \neq \emptyset} c_{k,\ell} B_{k,\ell}(x_{j,i,\ell}) = f(x_{j,i,\ell})$$
    - set $\tilde{\lambda}_{i,\ell} = c_{i,\ell}$

  - Set $\tilde{Q}(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_\ell} \tilde{\lambda}_{i,k}(f) B_{i,k}^\Upsilon$
Quasi-interpolation

Numerical examples: setup

- $C^1$ quadratic bivariate tensor-product B-splines
- Building block $\tilde{Q}^\ell(f) := \sum_{i=1}^{N_\ell} \tilde{\lambda}_{i,\ell}(f) B_{i,\ell}$:
  - choose a cell $\Upsilon_{i,\ell}$ in support of each $B_{i,\ell}$
  - choose $3 \times 3$ points $x_{j,i,\ell} \in \Upsilon_{i,\ell}$, $j = 1, \ldots, 9$
  - solve the system $\sum_{k: \text{supp}(B_{k,\ell}) \cap \Upsilon_{i,\ell} \neq \emptyset} c_{k,\ell} B_{k,\ell}(x_{j,i,\ell}) = f(x_{j,i,\ell})$
  - set $\tilde{\lambda}_{i,\ell} = c_{i,\ell}$

- Set $\tilde{Q}(f) := \sum_{k=0}^{n-1} \sum_{i \in \mathcal{I}_\ell} \tilde{\lambda}_{i,k}(f) B_{i,k}^T$
Quasi-interpolation

Numerical examples: setup

- \( C^1 \) quadratic bivariate tensor-product B-splines
- Building block \( \tilde{Q}^\ell(f) := \sum_{i=1}^{N_\ell} \tilde{\lambda}_{i,\ell}(f) B_{i,\ell} \):
  - choose a cell \( \Upsilon_{i,\ell} \) in support of each \( B_{i,\ell} \)
  - choose 3 \( \times \) 3 points \( x_{j,i,\ell} \in \Upsilon_{i,\ell}, \ j = 1, \ldots, 9 \)
  - solve the system \( \sum_{k: \text{supp}(B_{k,\ell}) \cap \Upsilon_{i,\ell} \neq \emptyset} c_{k,\ell} B_{k,\ell}(x_{j,i,\ell}) = f(x_{j,i,\ell}) \)
  - set \( \tilde{\lambda}_{i,\ell} = c_{i,\ell} \)
- Set \( \tilde{Q}(f) := \sum_{k=0}^{n-1} \sum_{i \in I_\ell} \tilde{\lambda}_{i,k}(f) B_{i,k}^T \)
Quasi-interpolation

Numerical examples

\[ f(x, y) = \frac{2}{3 \exp(\sqrt{(10x-3)^2+(10y-3)^2})} + \frac{2}{3 \exp(\sqrt{(10x+3)^2+(10y+3)^2})} + \frac{2}{3 \exp(\sqrt{(10x)^2+(10y)^2})} \]
Quasi-interpolation

Numerical examples

\[ f(x, y) = \frac{2}{3 \exp(\sqrt{(10x-3)^2+(10y-3)^2})} + \frac{2}{3 \exp(\sqrt{(10x+3)^2+(10y+3)^2})} + \frac{2}{3 \exp(\sqrt{(10x)^2+(10y)^2})} \]
Concluding message

Truncated hierarchical B-splines

- the hierarchical model: local refinement
- the truncation mechanism: construction of a basis for a hierarchical space with properties
  - ✓ linear independence
  - ✓ nonnegativity
  - ✓ small support
  - ✓ partition of unity
  - ✓ preservation of coefficients
  - ✓ strong $L_\infty$-stability
  - ✓ quasi-interpolants
Concluding message

Truncated hierarchical B-splines...not so convenient for QI

- the truncation mechanism: construction of a basis for a hierarchical space with nice properties
Concluding message

Truncated hierarchical B-splines...not so convenient for QI

- the truncation mechanism: construction of a basis for a hierarchical space with nice properties
- the truncation mechanism: requires some effort for implementation and data structures ... not really motivated for QI
Concluding message

Truncated hierarchical B-splines...not so convenient for QI

- the truncation mechanism: construction of a basis for a hierarchical space with nice properties
- the truncation mechanism: requires some effort for implementation and data structures ... not really motivated for QI

\[
Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}, \quad \ell = 0, \ldots, n - 1
\]

\[
Q(f) := \sum_{k=0}^{n-1} \sum_{i \in I_\ell} \lambda_{i,k}(f) B_{i,k}^T = \sum_{k=0}^{n-1} g^{(k)}
\]

\[
g^{(0)} := \sum_{i \in I_0} \lambda_{i,0}(f) B_{i,0}, \quad g^{(k)} := \sum_{i \in J_k} \lambda_{i,k}(f-g^{(0)}-\cdots-g^{(k-1)}) B_{i,k},
\]

\[
J_\ell := \{j : \text{supp}^0(B_{j,\ell}) \subset \Omega_\ell\}
\]

[Speleers, Manni, 2016]
Concluding message

Truncated hierarchical B-splines...not so convenient for QI

- the truncation mechanism: construction of a basis for a hierarchical space with nice properties
- the truncation mechanism: requires some effort for implementation and data structures ... not really motivated for QI

\[
Q^\ell(f) := \sum_{i=1}^{N_\ell} \lambda_{i,\ell}(f) B_{i,\ell}, \quad \ell = 0, \ldots, n - 1
\]

\[
Q(f) := \sum_{k=0}^{n-1} \sum_{i \in I_\ell} \lambda_{i,k}(f) B_{i,k}^T = \sum_{k=0}^{n-1} g^{(k)}
\]

\[
g^{(0)} := \sum_{i \in I_0} \lambda_{i,0}(f) B_{i,0}, \quad g^{(k)} := \sum_{i \in J_k} \lambda_{i,k}(f-g^{(0)}-\cdots-g^{(k-1)}) B_{i,k},
\]

\[
J_\ell := \{ j : \text{supp}^0(B_{j,\ell}) \subset \Omega_\ell \}
\]

[Speleers, Manni, 2016]

Thank you for your attention!
References


Bracco, C., Giannelli, C., Sestini, A.. *Adaptive scattered data fitting by extension of local approximations to hierarchical splines*. CAGD, . 2017