(T)HB- AND PATCHWORK B-SPLINES

Nora Engleitner

Institute of Applied Geometry, Johannes Kepler University
MTU Aero Engines AG, Munich

JOHANNES KEPLER
UNIVERSITY LINZ

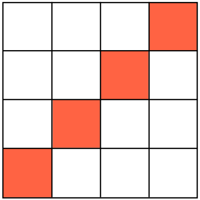
(T)HB-SPLINES

(T)HB-SPLINES

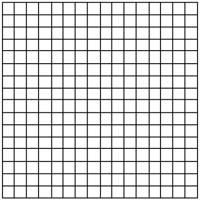
INTRODUCTION

Local spline refinement

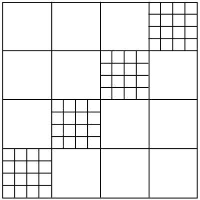
Problem: Classical tensor-product B-splines do not allow local refinement.



Task: Knot refinement in the marked areas.



Resulting mesh for B-splines.



How can we achieve a mesh like this?

Existing constructions

T-splines (Sederberg et al. 2003):
tensor-product B-splines defined on a mesh with T-junctions

PHT-splines (Falai Chen, Jiansong Deng 2008):
algebraically complete basis for splines on a mesh with T-junctions

HB-splines, THB-splines (Kraft 1997, Giannelli et al. 2012):
obtained by selecting B-splines from different levels in a hierarchy

LR-splines (Dokken et al. 2010):
constructed by repeatedly splitting tensor-product B-splines

This talk focuses on ***HB-splines***.

(T)HB-SPLINES

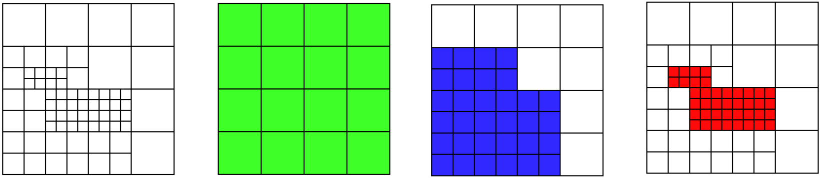
A HIERARCHICAL B-SPLINE BASIS

Hierarchical B-splines

Consider a finite sequence of

- nested spline spaces, $V^0 \subseteq V^1 \subseteq \ldots \subseteq V^N$, with $V^\ell = \mathrm{span} B^\ell$,
- corresponding nested domains, $\Omega = \Omega^0 \supseteq \Omega^1 \supseteq \ldots \supseteq \Omega^N$.

The index ℓ is called ***level***.



A hierarchical mesh and the domains Ω^0 (green), Ω^1 (blue) and Ω^2 (red).

Kraft's selection mechanism

$\mathrm{supp} f = \{\mathbf{x} : f(\mathbf{x}) \neq 0$ and $\mathbf{x} \in \Omega^0\}$

Recursive construction of HB-splines

1) Initialization: $H^0 = \{\beta \in B^0 : \mathrm{supp}\beta \neq \emptyset\}$
2) Recursion: $H^{\ell+1} = H_A^{\ell+1} \cup H_B^{\ell+1}$, for $\ell = 0, \dots, N-1$, with

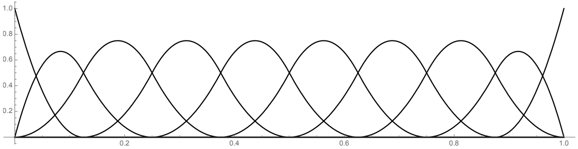
$$H_A^{\ell+1} = \{\beta \in H^\ell : \mathrm{supp}\beta \not\subseteq \Omega^{\ell+1}\},$$

   and

$$H_B^{\ell+1} = \{\beta \in B^{\ell+1} : \mathrm{supp}\beta \subseteq \Omega^{\ell+1}\}$$

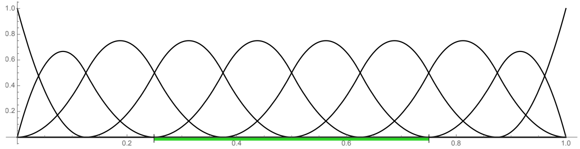
3) $H = H^N$

Hierarchical B-splines



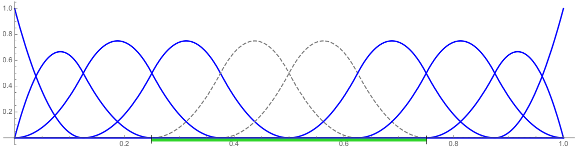
1) Initialization: $H^0 = B^0$

Hierarchical B-splines



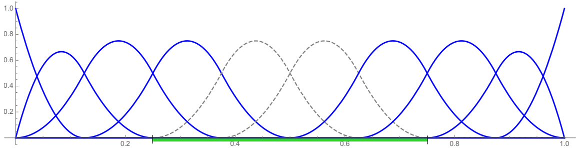
$\Omega^1 \subset \Omega^0$

Hierarchical B-splines

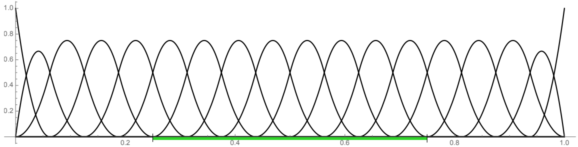


2a) Recursion: $H^1_A = \{\beta \in H^0 = B^0$ **:** $\mathrm{supp}\beta \not\subseteq \Omega^1\}$

Hierarchical B-splines

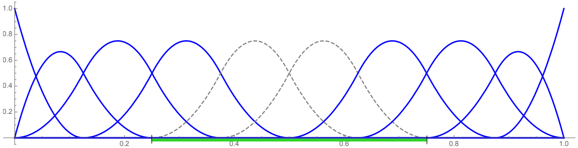


Selected B-splines of level 0

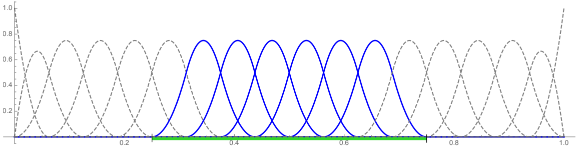


B-splines B^1 on Ω^0

Hierarchical B-splines



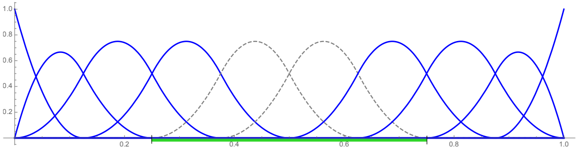
Selected B-splines of level 0



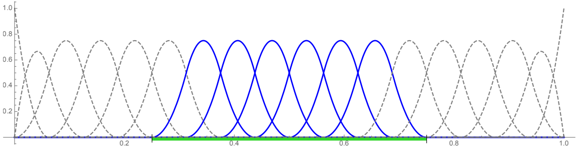
2b) Recursion: $H_B^1 = \{\beta \in B^1 : \mathrm{supp}\beta \subseteq \Omega^1\}$

JKU

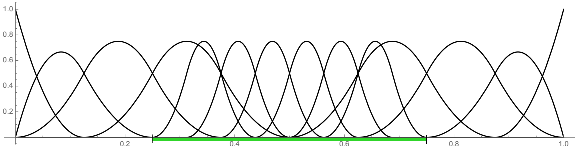
Hierarchical B-splines



Selected B-splines of level 0



Selected B-splines of level 1



3) Resulting hierarchical B-splines $H = H_A^1 \cup H_B^1$

(T)HB-SPLINES

TRUNCATION

Restoring partition of unity

HB-splines: no partition of unity $\rightarrow$ solution: ***truncation mechanism***
(cf. Giannelli et al. 2012)

Refinement relation: For $f \in V^\ell$ we have $f = \sum_{\beta \in B^{\ell+1}} c_\beta^{\ell+1}(f)\beta$.

Truncation:

$$\mathrm{trunc}^{\ell+1} f = \sum_{\beta \in B^{\ell+1}, \mathrm{supp}\beta \not\subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(f)\beta.$$

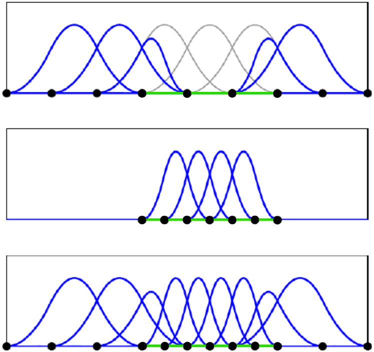
Truncated hierarchical B-spline basis:

1) Initialization: $T^0 = H^0$
2) Recursion: $T^{\ell+1} = T_A^{\ell+1} \cup T_B^{\ell+1}$, for $\ell = 0, \ldots, N-1$, with

$$T_A^{\ell+1} = \{\mathrm{trunc}^{\ell+1}\tau : \tau \in T^\ell \text{ and } \mathrm{supp}\tau \not\subseteq \Omega^{\ell+1}\}, \quad \text{and} \quad T_B^{\ell+1} = H_B^{\ell+1}$$

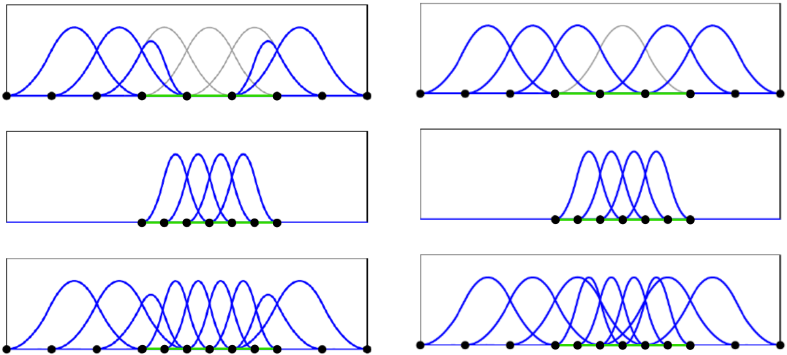
3) $T = T^N$

Comparing THB- and HB-splines



THB-splines

Comparing THB- and HB-splines



THB-splines HB-splines

(T)HB-SPLINES

PROPERTIES

Properties of (T)HB-splines

HB-splines:

- non-negativity
- linear independence
- under certain assumptions:
  $\text{span} H = \mathcal{H} = \{h : \Omega^0 \to \mathbb{R} : h|_{\Omega^0 \setminus \Omega^{\ell+1}} \in \mathcal{S}^\ell(M^\ell)\}$

Additionally for THB-splines:

- $\text{span} H = \text{span} T$
- preservation of coefficients
- partition of unity
- strongly stable under supremum norm

Quasi-interpolant for (T)HB-splines

References: Speleers et al. 2015, Speleers 2016

A ***one-level quasi-interpolant*** $\Pi^\ell : \mathcal{V}(\Omega^0) \mapsto V^\ell$

$$\Pi^\ell f = \sum_{i=1}^{n_\ell} \lambda_i^\ell(f)\beta_i^\ell, \quad \ell = 0, \ldots, N.$$

λ_i^ℓ is *supported on* Λ_i^ℓ if $f|_{\Lambda_i^\ell} = 0 \Rightarrow \lambda_i^\ell(f) = 0$. For a cell Q^ℓ in $\Omega^\ell \setminus \Omega^{\ell+1}$ define

$$\Lambda_{Q^\ell} = \operatorname{conv}\left(\bigcup_{(i,\ell):\operatorname{supp}\tau_i^\ell \cap Q^\ell \neq \emptyset} \Lambda_i^\ell \cup Q^\ell \right).$$

Hierarchical quasi-interpolant $\Pi : \mathcal{V}(\Omega^0) \mapsto \mathcal{H}$

$$\Pi f := \sum_{\ell=0}^{N} \sum_{\tau_i^\ell \in T} \lambda_i^\ell(f)\tau_i^\ell,$$

where $\tau_i^\ell = \operatorname{trunc}^N \left(\operatorname{trunc}^{N-1} \cdots \left(\operatorname{trunc}^{\ell+1}\beta_i^\ell\right)\cdots\right)$.

Quasi-interpolant for (T)HB-splines

Representation in terms of HB-splines (using telescoping argument):

Theorem: Π^ℓ and Π as defined before. If $\Pi^\ell s = s$ for all $s \in V^\ell$ then

$$\Pi f = \sum_{\ell=0}^{N} f^{(\ell)},$$

with

$$f^{(0)} = \sum_{\beta_i^0 \in H} \lambda_i^0 \beta_i^0, \quad f^{(\ell)} = \sum_{\beta_i^\ell \in H} \lambda_i^\ell (f - f^{(0)} - f^{(1)} - \cdots - f^{(\ell-1)}) \beta_i^\ell.$$

Error estimate:

$$\|D^\alpha (f - \Pi f)\|_{L^2(Q^\ell)} \le C h_\ell^{s-|\alpha|} |f|_{H^s(\Lambda_{Q^\ell})}.$$

(T)HB-SPLINES

SUMMARY

Summary

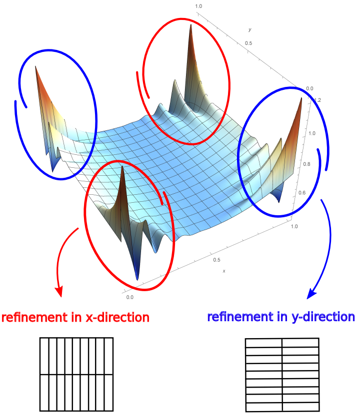
- Selection mechanism for hierarchical B-splines
- Restoring partition of unity with truncation mechanism
- (T)HB-splines have nice mathematical properties
- (T)HB-splines are a basis for the hierarchical spline space
- Quasi-interpolant and local approximation estimate

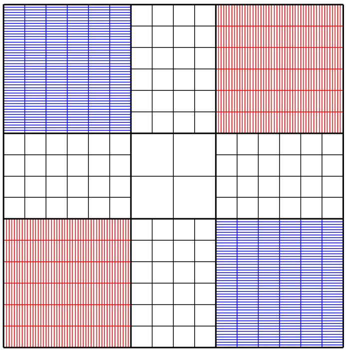
PATCHWORK B-SPLINES

PATCHWORK B-SPLINES

INTRODUCTION

Motivation





Independent refinement strategies $\rightsquigarrow$ cannot be achieved with HB-splines

$\rightsquigarrow$ hierarchical refinement leads to redundant dof

Motivation

State of the art: Hierarchical B-splines that use sequences of ***nested*** spline spaces, $V^0 \subseteq V^1 \subseteq \ldots \subseteq V^N$.

Limitation: Independent refinement strategies are not possible.

Possible application of independent refinement strategies:

- Modeling: designing objects with creases or similar features.
- IGA: using different refinement techniques (e.g., h- and p-refinement) in different parts of the domain.

Goal: Generalization of the selection mechanism for hierarchical B-splines to obtain sequences of ***partially nested*** hierarchical spline spaces, that use spline spaces such as $V^0 \subseteq V^1 \not\subseteq V^2 \subseteq V^3 \ldots$

Preliminaries

We consider:

- A finite sequence of bivariate tensor-product spline spaces:

$$V^\ell = \mathrm{span} B^\ell, \quad \ell = 1, \ldots, N.$$

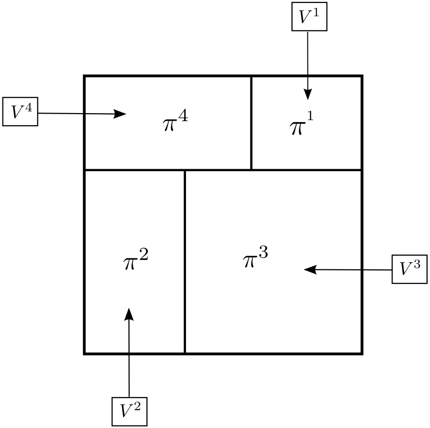
  - Note: V^ℓ not necessarily subspace of $V^{\ell+1}$
  - For simplicity: $d = 2$, uniform degrees, maximum smoothness

- An associated sequence of open sets

$$\pi^\ell \subseteq (0,1)^2, \quad \ell = 1, \ldots, N.$$

  - The sets are called ***patches***.
  - We assume that they are mutually disjoint, i.e., $\pi^\ell \cap \pi^k \neq \emptyset \;\Rightarrow\; \ell = k$.

Preliminaries



Patches and associated spline spaces.

The patchwork spline space

Collecting all patches results in the ***domain*** Ω, i.e.,

$$\Omega = \operatorname{int}\left(\bigcup_{\ell=1}^{N} \overline{\pi^\ell} \right) \subseteq (0,1)^2.$$

Now we define the ***patchwork spline space*** :

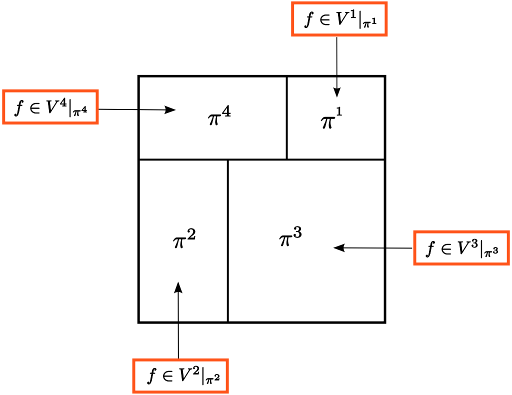
$$\mathcal{P} = \{ f \in \mathcal{C}^{\mathbf{s}}(\Omega) : f|_{\pi^\ell} \in V^\ell|_{\pi^\ell} \ \forall \ell = 1, \ldots, N \},$$

with maximal order of smoothness

$$\mathbf{s} = \mathbf{p} - 1.$$

The patchwork spline space

Definition: $\mathcal{P} = \{ \boxed{f \in \mathcal{C}^{\mathbf{s}}(\Omega)} : \boxed{f|_{\pi^\ell} \in V^\ell|_{\pi^\ell}} \ \forall \ell = 1, \ldots, N\}$



(1) $\boxed{C^{\mathbf{s}}\text{-smooth functions}}$

(2) $\boxed{\text{patch restriction belongs to assoc. spline space}}$

PATCHWORK B-SPLINES

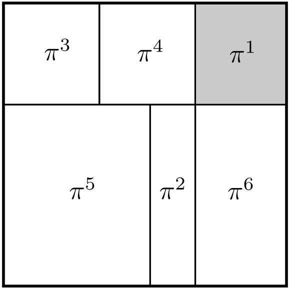
BASIS FUNCTIONS

Constraining boundaries

The ***constraining boundary*** of a patch

$$\Gamma^\ell = \bigcup_{k=1}^{\ell-1} \overline{\pi^k} \cap \overline{\pi^\ell},$$

is the part of the boundary shared with patches of a lower level.



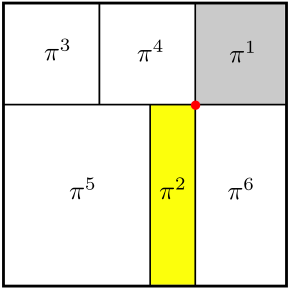
Constraining boundaries of π^1

Constraining boundaries

The ***constraining boundary*** of a patch

$$\Gamma^\ell = \bigcup_{k=1}^{\ell-1} \overline{\pi^k} \cap \overline{\pi^\ell},$$

is the part of the boundary shared with patches of a lower level.



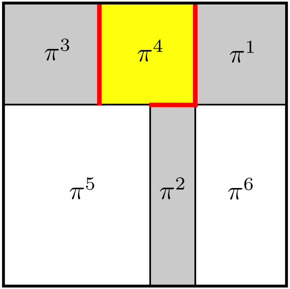
Constraining boundaries of π^2

Constraining boundaries

The ***constraining boundary*** of a patch

$$\Gamma^\ell = \bigcup_{k=1}^{\ell-1} \overline{\pi^k} \cap \overline{\pi^\ell},$$

is the part of the boundary shared with patches of a lower level.



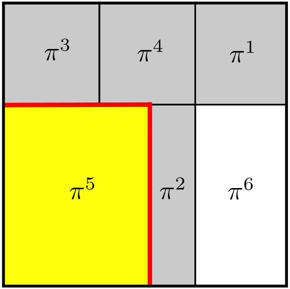
Constraining boundaries of π^4

Constraining boundaries

The ***constraining boundary*** of a patch

$$\Gamma^\ell = \bigcup_{k=1}^{\ell-1} \overline{\pi^k} \cap \overline{\pi^\ell},$$

is the part of the boundary shared with patches of a lower level.



Constraining boundaries of π^5

Selection mechanism

We generalize Kraft's selection mechanism:

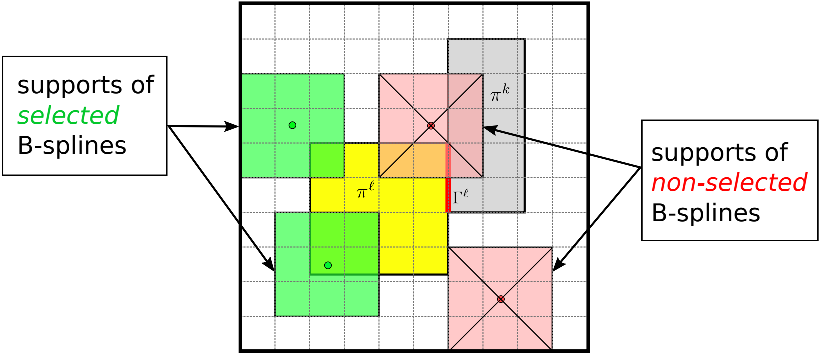
$$K^\ell = \{\beta^\ell \in B^\ell : \beta^\ell|_{\pi^\ell} \neq 0 \quad \text{and} \quad \beta^\ell|_{\Gamma^\ell} = 0\}.$$

Definition: The ***patchwork B-splines*** (PB-splines) are obtained by forming the union over all levels,

$$K = \bigcup_{\ell=1}^{N} K^\ell.$$

Selection mechanism

$$K^\ell = \{\beta^\ell \in B^\ell : \beta^\ell|_{\pi^\ell} \neq 0 \quad \text{and} \quad \beta^\ell|_{\Gamma^\ell} = 0\}.$$

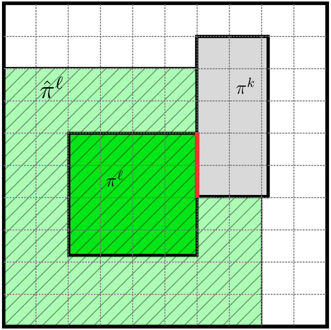


The selection mechanism for PB-splines ($k < \ell$).

Shadow

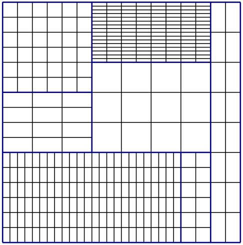
We define the ***shadow*** of a patch π^ℓ as the union of all supports of the selected basis functions,

$$\hat{\pi}^\ell = \mathrm{supp} K^\ell = \bigcup_{\beta^\ell \in K^\ell} \mathrm{supp} \beta^\ell.$$

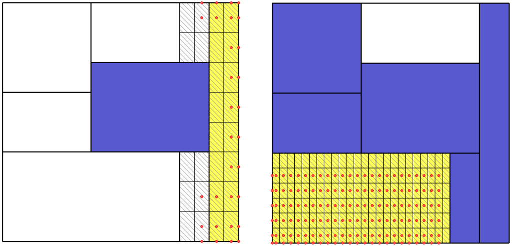


Example: Shadows and meshes

The knot lines of the spline space V^ℓ define a ***mesh*** M^ℓ of level ℓ.



A patchwork mesh.



Shadow and selected basis functions for two levels.
(points: selected B-splines, shadow: hatched area)

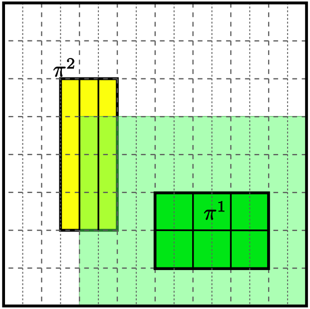
PATCHWORK B-SPLINES

CHARACTERIZING THE SPLINE SPACE

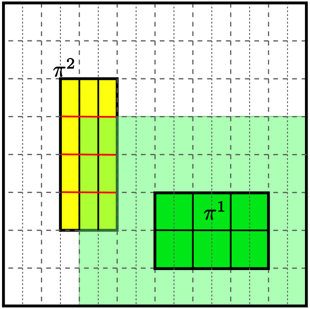
Shadow Compatibility Assumption (SCA)

Assumption If the shadow $\hat{\pi}^\ell$ of the patch of level ℓ intersects another patch π^k of a different level k, then the first level precedes the second one,

$$\hat{\pi}^\ell \cap \pi^k \neq \emptyset \quad \Rightarrow \quad \ell \leq k \quad \text{and} \quad V^\ell \subseteq V^k.$$



SCA not satisfied.

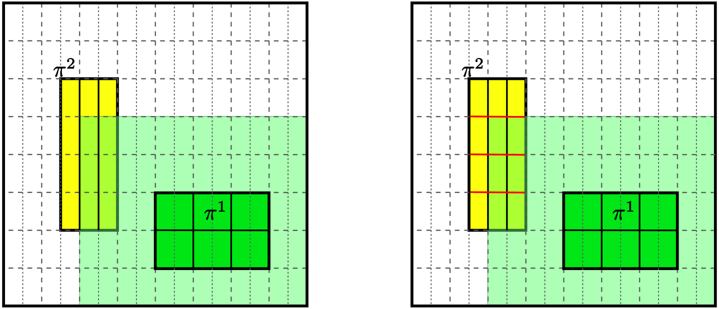


SCA satisfied.

Shadow Compatibility Assumption (SCA)

Assumption If the shadow $\hat{\pi}^\ell$ of the patch of level ℓ intersects another patch π^k of a different level k, then the first level precedes the second one,

$$\hat{\pi}^\ell \cap \pi^k \neq \emptyset \quad \Rightarrow \quad \ell \leq k \quad \text{and} \quad V^\ell \subseteq V^k.$$

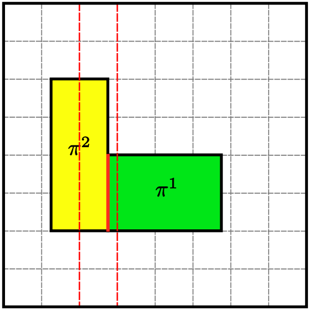


SCA not satisfied. SCA satisfied.

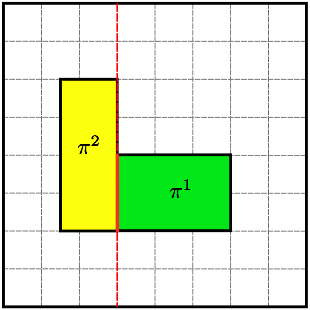
Theorem: SCA implies linear independence of PB-splines.

Constraining Boundary Alignment (CBA)

Assumption For each level ℓ, the constraining boundary Γ^ℓ of the patch π^ℓ is aligned with the knot lines of the spline space V^ℓ.



CBA not satisfied.



CBA satisfied.

Space characterization

Theorem The PB-splines span the patchwork spline space $\mathcal{P}$ if both SCA and CBA are satisfied.

Thus, we have ***two different characterizations*** of the patchwork spline space:

$$\mathcal{P} = \{f \in \mathcal{C}^{\mathbf{s}}(\Omega) : f|_{\pi^\ell} \in V^\ell|_{\pi^\ell} \ \forall \ell = 1, \ldots, N\},$$

(***"implicit"*** definition: space defined by properties of functions)

$$\mathcal{P} = \operatorname{span} \bigcup_{\ell=1}^{N} \{\beta^\ell \in B^\ell : \beta^\ell|_{\pi^\ell} \neq 0 \quad \text{and} \quad \beta^\ell|_{\Gamma^\ell} = 0\}$$

(***"constructive"*** definition: space defined as linear hull of basis functions)

Restoring partition of unity

Truncation mechanism

Recall: Hierarchical B-splines $\rightarrow$ Truncated Hierarchical B-splines

The recipe:

Truncated function: "original function ***minus*** contribution of selected basis functions from higher levels"

Is there a generalization to truncated PB-splines?

Restoring partition of unity

Truncation mechanism

Recall: Hierarchical B-splines $\rightarrow$ Truncated Hierarchical B-splines

The recipe:

Truncated function: "original function ***minus*** contribution of selected basis functions from higher levels"

Is there a generalization to truncated PB-splines? **Yes!**

Truncated PB-splines

- are linearly independent,
- form a **partition of unity**,
- are non-negative and
- span the patchwork spline space.

PATCHWORK B-SPLINES

PB-SPLINES IN SURFACE APPROXIMATION

Surface approximation problem

- Given: data set $(f_i, \mathbf{x}_i), i = 0, \ldots, m,$
  - coordinates of data points $f_i \in \mathbb{R}^3$,
  - associated parameters $\mathbf{x}_i \in [0,1]^2$.
- Choose a setting that generates the PB-splines K:
  - patches $\pi^1, \ldots, \pi^N$ and
  - spline spaces $V^1, \ldots, V^N$.
- Compute the least squares approximation $f = \sum_{\beta \in K} c_\beta \beta$, which minimizes

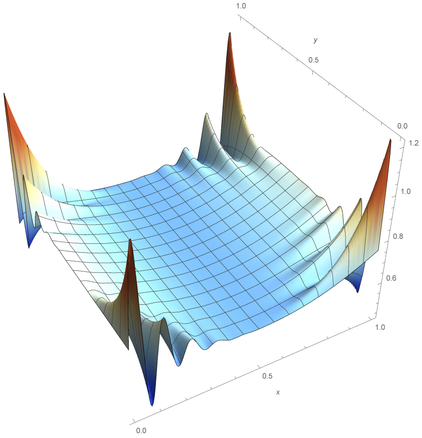
$$\sum_{i=0}^{m} \|f_i - f(\mathbf{x}_i)\|^2.$$

How to choose the patches and corresponding spline spaces?

- Manual construction
- Automatic refinement process

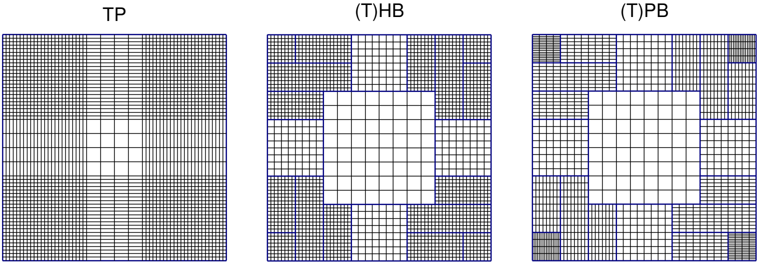
Example I

We want to approximate the following function:



Function for approximation.

Manual mesh generation



The meshes used for defining the approximating spline functions.

	no. of dof	% of dof	max. error	average error
tensor-product B-splines	2916	100%	3.08e-3	1.5e-4
HB-splines	2468	85 %	3.08e-3	1.02e-4
PB-splines	1572	54 %	1.03e-3	6.94e-5

Numerical results of the least-squares approximation.

Automatic mesh refinement

- Initial setting defining K_0
  - patches $\pi^1, \dots, \pi^{N_0}$ and
  - spline spaces $V_0^1, \dots, V_0^{N_0}$.
- Compute least squares approximation for K_0
- Marking process:
  - Identify those $\mathbf{x}_i$ with $\|f_i - f(\mathbf{x}_i)\| > \varepsilon$,
  - find the patches that contain $\mathbf{x}_i$,
  - mark them for refinement.
- Refinement process:
  - n-adic subdivision of the marked patches,
  - (poss. new marking process),
  - knot refinement in the corresponding spline spaces,
  - ensure that all assumptions are satisfied.
  - ***Challenge:*** Determine the **direction** of the refinement

Determining the refinement direction

Determining the refinement direction with a ***local fitting-based*** method:

- Perform local fitting on patches π^{ℓ}.
- Try different refinement strategies, e.g., uniform knot refinement in x- vs. in y-direction.
- The strategy that performs better, i.e., produces less error, determines the refinement direction.

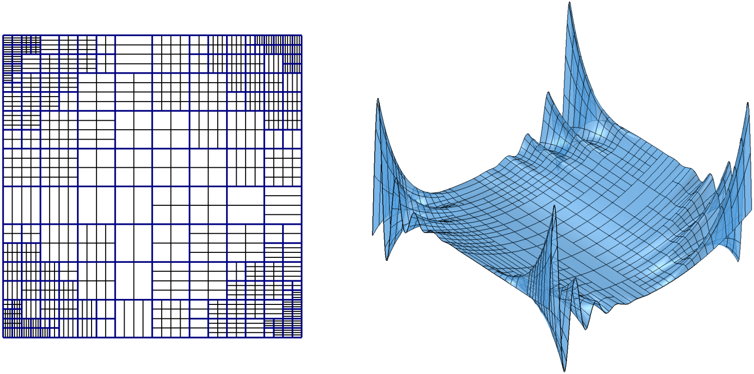
Advantages:

- No assumptions on data,
- simple.

Disadvantages:

- Could become slow if too many strategies are tested.

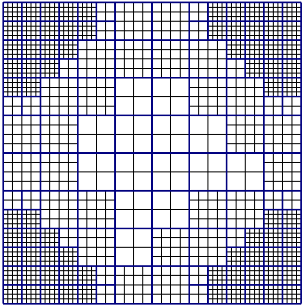
Automatic mesh refinement - results

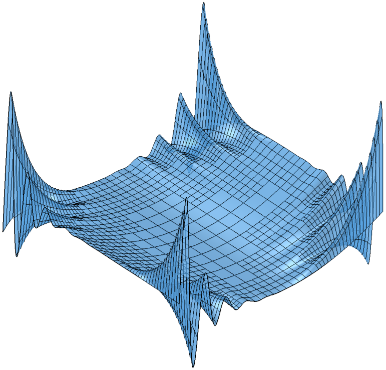


PB-spline mesh after 4 steps of adaptive refinement and resulting surface.

	no. of dof	% of dof	max. error	average error
HB-splines	1860	100 %	3.08e-3	1.42e-4
PB-splines	1106	59 %	1.12e-3	1.31e-4

Automatic mesh refinement - results

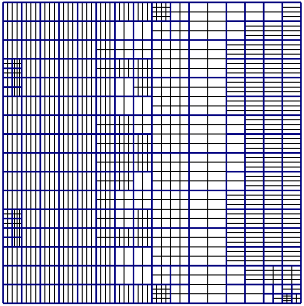


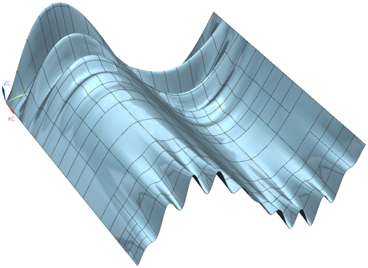


HB-spline mesh after 3 steps of adaptive refinement and resulting surface.

	no. of dof	% of dof	max. error	average error
HB-splines	1860	100 %	3.08e-3	1.42e-4
PB-splines	1106	59 %	1.12e-3	1.31e-4

Example II

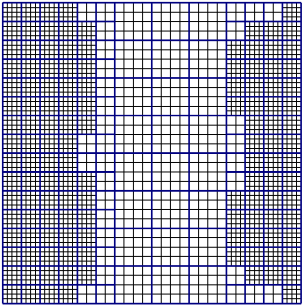


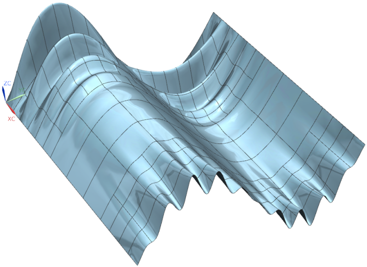


PB-spline mesh after 4 steps of adaptive refinement and resulting surface.

	no. of dof	% of dof	max. error	average error
HB-splines	2688	100 %	1.01e-3	1.56e-4
PB-splines	1169	43 %	1.06e-3	1.47e-4

Example II





HB-spline mesh after 3 steps of adaptive refinement and resulting surface.

	no. of dof	% of dof	max. error	average error
HB-splines	2688	100 %	1.01e-3	1.56e-4
PB-splines	1169	43 %	1.06e-3	1.47e-4

PATCHWORK B-SPLINES

SUMMARY AND OUTLOOK

Summary

- Generalization of the Kraft selection mechanism from hierarchical B-splines to PB-splines
- Characterization of the spline space spanned by the PB-splines
- Introduction of a truncation mechanism $\rightarrow$ partition of unity
- Application of PB-splines to surface approximation
- Automatic refinement algorithm for PB-spline meshes
- PB-splines enable the use of independent refinement strategies
- PB-splines need fewer degrees of freedom than HB-splines

Current work and outlook

- Generalizing the completeness result from HB-splines to PB-splines
- Generalizing the approximation error estimates from HB-splines to PB-splines
- Implementation of the truncation mechanism
- Development of further automatic mesh refinement strategies
- Application in industry
  - Fitting of structural components like airfoils $\rightarrow$ periodic fitting
  - Lofting