Accepted Manuscript

Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis

E.J. Evans, M.A. Scott, X. Li, D.C. Thomas

PII: S0045-7825(14)00180-7
DOI: http://dx.doi.org/10.1016/j.cma.2014.05.019
Reference: CMA 10253

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Please cite this article as: E.J. Evans, M.A. Scott, X. Li, D.C. Thomas, Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis, *Comput. Methods Appl. Mech. Engrg.* (2014), http://dx.doi.org/10.1016/j.cma.2014.05.019

Highlights

- We introduce hierarchical analysis-suitable T-splines (HASTS).
- We present a theoretical formulation of HASTS.
- We extend Bézier extraction to HASTS.
- HASTS are compared to local T-spline refinement algorithms.
- HASTS are utilized as a basis for adaptive isogeometric analysis.

Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis

E. J. Evans[a,*], M. A. Scott[b], X. Li[c], D. C. Thomas[d]

[a]*Department of Mathematics, Brigham Young University, Provo, Utah 84602, USA*
[b]*Department of Civil and Environmental Engineering, Brigham Young University, Provo, Utah 84602, USA*
[c]*University of Science and Technology of China, Hefei, Anhui Province 230026, P. R. China*
[d]*Department of Physics and Astronomy, Brigham Young University, Provo, Utah 84602, USA*

Abstract

In this paper hierarchical analysis-suitable T-splines (HASTS) are developed. The resulting spaces are a superset of both analysis-suitable T-splines and hierarchical B-splines. The additional flexibility provided by the hierarchy of T-spline spaces results in simple, highly localized refinement algorithms which can be utilized in a design or analysis context. A detailed theoretical formulation is presented. Bézier extraction is extended to HASTS simplifying the implementation of HASTS in existing finite element codes. The behavior of a simple HASTS refinement algorithm is compared to the local refinement algorithm for analysis-suitable T-splines demonstrating the superior efficiency and locality of the HASTS algorithm. Finally, HASTS are utilized as a basis for adaptive isogeometric analysis.

Keywords: isogeometric analysis, hierarchical splines, adaptive mesh refinement, T-splines

1. Introduction

In this work, a hierarchical extension of analysis-suitable T-splines is developed and utilized in the context of isogeometric design and analysis. We call this new spline description hierarchical analysis-suitable T-splines (HASTS). The class of HASTS is a strict superset of both analysis-suitable T-splines [1, 2, 3, 4, 5] and hierarchical B-splines [6, 7, 8, 9, 10, 11].

T-splines, introduced in the CAD community [12], are a generalization of non-uniform rational B-splines (NURBS) which address fundamental limitations in NURBS-based design. For example, a T-spline can model a complicated design as a single, watertight geometry and are also locally refinable [2, 13]. Since their advent they have

*Corresponding author
Email address: `ejevans@math.byu.edu` (E. J. Evans)

emerged as an important technology across multiple disciplines and can be found in several major commercial CAD products [14, 15].

Isogeometric analysis was introduced in [16] and described in detail in [17]. The isogeometric paradigm is simple: use the smooth spline basis that defines the geometry as the basis for analysis. As a result, exact geometry is introduced into the analysis, the smooth basis can be leveraged by the analysis [18, 19, 20], and new innovative approaches to model design [21, 22], analysis [7, 23, 24, 25], optimization [26], and adaptivity [9, 27, 28] are made possible. The use of T-splines as a basis for isogeometric analysis (IGA) has gained widespread attention across a number of application areas [2, 7, 25, 27, 29, 30, 31, 32, 23, 33, 34, 35, 36, 37, 38]. Particular focus has been placed on the use of T-spline local refinement in an analysis context [2, 30, 31, 32].

In the context of CAD, where a designer interacts directly with the geometry, T-spline local refinement is most useful if confined to a *single level*. In other words, all local refinement is done on one control mesh and all control points have similar influence on the shape of the surface. In this way, the geometric behavior of the surface is easily controlled through the manipulation of control points before and after refinement. In the context of analysis, however, where not all control points need to have a geometric interpretation, the single level restriction can be relaxed. This *hierarchical* point of view has important advantages:

1. Hierarchical local refinement remains completely local. Single level T-spline local refinement always entails a degree of nonlocal control point propagation [2].
2. Hierarchical local coarsening is achieved by simply removing higher levels of refinement where needed [9]. For T-splines this coarsening can be done locally using Bézier projection [39]. Local coarsening operations for single level T-spline descriptions are possible but their algorithmic complexity remains uncertain [13].
3. Hierarchical refinement and coarsening operations use a fixed control mesh which simplifies algorithmic developments, especially for parallel computations. Single level local refinement requires expensive mesh manipulation and modification operations.
4. Hierarchies of finite-dimensional subspaces are the natural setting for many optimized iterative solvers and preconditioning techniques for large-scale linear systems.

Initial investigations employing hierarchical B-spline refinement in the context of IGA have demonstrated the promise of the hierarchical approach [7, 40, 41, 42, 43].

HASTS inherit the design strengths of T-splines without the single level restriction. In this way, a complex T-spline design can be encapsulated in the first level of the hierarchy while higher levels can be leveraged to develop adaptive multiresolution schemes which are smooth, highly localized, geometrically exact, and appropriate for the analysis task at hand. We feel that this provides the appropriate mathematical foundation for the development of integrated isogeometric design and analysis methodologies for demanding applications in science and engineering. Note that, in this paper, we restrict our theoretical developments to HASTS defined over four-sided domains. However, extending HASTS to domains of arbitrary topological genus should be straightforward in the context of the recently introduced spline forest [9].

We note that in addition to T-splines, hierarchical B-splines, and NURBS a number of alternative spline technologies have been proposed as a basis for IGA with varying strengths and weaknesses. Truncated hierarchical B-splines (THB-splines) [11, 44, 45, 46] are a modification of hierarchical B-splines [6, 8, 41] which form a partition of unity and enhanced numerical conditioning. B-spline forests [9] are a generalization of hierarchical B-splines to surfaces and volumes of arbitrary topological genus. Polynomial splines over hierarchical T-meshes (PHT-splines) [47, 48, 49, 50], modified T-splines [51], and locally refined splines (LR-splines) [52, 53] are closely related to T-splines with varying levels of smoothness and approaches to local refinement. Generalized B-splines [54, 55] and T-splines [56] enhance a piecewise polynomial spline basis by including non-polynomial functions, typically trigonometric or hyperbolic functions. Generalized splines permit the exact representation of conic sections without resorting to rational functions. Generalized splines can also be used to represent solution features with known non-polynomial characteristics exactly in certain circumstances.

In Section 2 the T-mesh is described and appropriate notational conventions are introduced. Analysis-suitable T-splines are then described in Section 3. Hierarchical analysis-suitable T-splines are then defined in Section 4. In preparation for their use in design and analysis a Bézier extraction framework is introduced in Section 5. HASTS are then utilized as a basis for isogeometric analysis in Section 6. In Section 7 we draw conclusions.

2. The T-mesh

The T-mesh is used to define the topological structure of the associated T-spline space. In other words, the T-mesh defines the basis functions and their relationship to one another. We closely follow the notational conventions introduced in [3, 4, 5].

A T-mesh T in two dimensions is a rectangular partition of $\hat{\hat{\Omega}} = [1, m] \times [1, n]$ such that all vertices $V = \{i, j\} \in \mathsf{V}$ have integer coordinates. All cells $C \in \mathsf{C}$ are rectangular, non-overlapping, and open. An edge is a horizontal or vertical line segment between vertices which does not intersect any cell. The valence of a vertex $V \in \mathsf{V}$ is the number of edges coincident to that vertex. Since all cells are assumed rectangular, only valence three (i.e., T-junction) or four is allowed for all vertices $V \subset (1, m) \times (1, n)$. The sets of horizontal and vertical coordinates in the T-mesh are denoted by $h\mathsf{I} = \{1, 2, \ldots, m\}$ and $v\mathsf{I} = \{1, 2, \ldots, n\}$. The horizontal and vertical skeletons, $h\mathsf{S}$ and $v\mathsf{S}$, of a T-mesh are the union of all horizontal and vertical edges, respectively, and associated vertices. The entire skeleton is denoted by $\mathsf{S} = h\mathsf{S} \cup v\mathsf{S}$.

We split $\hat{\hat{\Omega}}$ into an active region AR and a frame region FR such that $\hat{\hat{\Omega}} = \mathsf{FR} \cup \mathsf{AR}$ and $\mathsf{AR} = \underline{[1 + \lfloor(p+1)/2\rfloor}, m - \lfloor(p+1)/2\rfloor] \times [1 + \lfloor(q+1)/2\rfloor, n - \lfloor(q+1)/2\rfloor]$, and $\mathsf{FR} = \overline{\hat{\hat{\Omega}}\backslash\mathsf{AR}}$ where p and q are polynomial degrees. Note that *both* FR and AR are closed. Further, all T-meshes considered in this work are admissible as described in [5], a mild restriction always adopted in practice. The notation $\mathsf{T}^1 \subseteq \mathsf{T}^2$ will indicate that T^2 can be created by adding vertices and edges to T^1. Fig. 1 shows an example T-mesh.

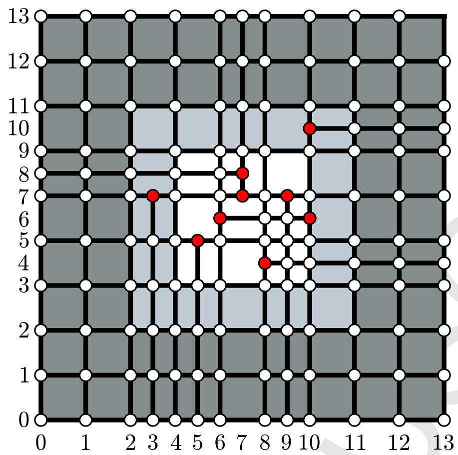


Figure 1: A bicubic T-mesh. The frame region (FR) is dark grey and the active region (AR) is the union of the light grey and white regions. Note that the region with zero parametric area (see Section 3.2 for a description of the parametric space of a T-spline) is the union of the dark and light grey regions.

2.1. Analysis-suitable T-meshes

Analysis-suitable T-splines (ASTS) were introduced in [1]. The analysis-suitability of a T-spline is dictated by the structure of the underlying T-mesh. We define face and edge extensions to be closed line segments that originate at T-junctions. For example, to define a horizontal face extension we trace out a horizontal line by moving in the direction of the missing edge until $\lfloor (p+1)/2 \rfloor$ vertical edges or vertices are intersected. To define an edge extension we trace out a horizontal line by moving in the direction opposite the face extension until $\lceil (p-1)/2 \rceil$ vertical edges or vertices are intersected. A T-junction extension includes both the face and edge extensions. Since extensions are defined as closed line segments they may intersect at their end points. An *extended T-mesh*, T_{ext}, is the T-mesh formed by adding the T-junction extensions to T. The collection of rectangular cells in T_{ext} is denoted by C_{ext}. We say a T-mesh is *analysis-suitable* if no horizontal T-junction extension intersects a vertical T-junction extension. Face and edge extensions (along with analysis-suitability) are illustrated in Figure 2.

2.2. Anchors

Anchors are used in the construction of T-spline blending functions. For an analysis-suitable T-mesh the anchors are located only in the active region and are defined as follows:

- if p and q are odd the anchors are vertices. It is written as $\{i\} \times \{j\}$ or equivalently $\{i, j\}$.
- if p is even and q is odd the anchors are horizontal edges. It is written as $(i_1, i_2) \times \{j\}$.

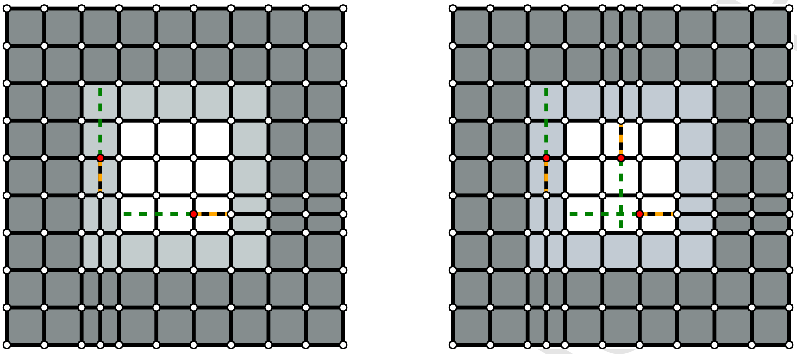


Figure 2: T-junction extension in two dimensions for a bicubic T-spline. Face extensions are shown in green and edge extensions are shown in orange. The T-mesh on the left is analysis-suitable whereas the T-mesh on the right is not due to intersecting face extensions.

- if p is odd and q is even the anchors are vertical edges. It is written as $\{i\} \times (j_1, j_2)$.
- if p and q are even the anchors are cells. It is written as $(i_1, i_2) \times (j_1, j_2)$.

The set of all anchors is denoted by A. The set of anchors for varying values of p and q are shown in Figure 3.

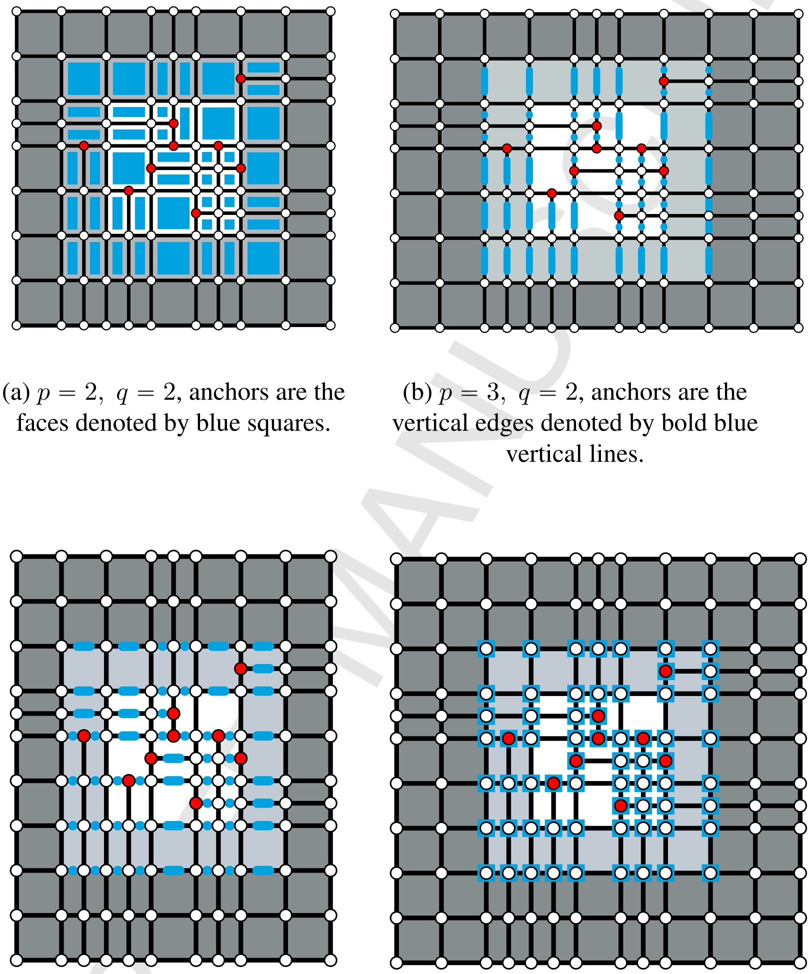
3. Analysis-suitable T-splines

Analysis-suitable T-splines form a useful subset of T-splines. ASTS maintain the important mathematical properties of the NURBS basis while providing an efficient and highly localized refinement capability.

3.1. Properties of analysis-suitable T-splines

Several important properties of ASTS have been proven:

- The blending functions are linearly independent for *any* choice of knots [1].
- The basis constitutes a partition of unity [5].
- Each basis function is non-negative.
- They can be generalized to arbitrary degree [4].



(a) $p = 2,\ q = 2$, anchors are the faces denoted by blue squares.

(b) $p = 3,\ q = 2$, anchors are the vertical edges denoted by bold blue vertical lines.

(c) $p = 2,\ q = 3$, anchors are the horizontal edges denoted by bold blue horizontal lines.

(d) $p = 3,\ q = 3$, anchors are the vertices denoted by small blue hollow squares.

Figure 3: The set of anchors for varying values of p and q. In this picture, the blue regions denote anchor locations. Note that no anchors reside inside the frame region (dark grey region).

- An affine transformation of an analysis-suitable T-spline is obtained by applying the transformation to the control points. We refer to this as affine covariance. This implies that all "patch tests" (see [57]) are satisfied *a priori*.
- They obey the convex hull property.
- They can be locally refined [2, 5, 12].
- A dual basis can be constructed [3, 4].
- Local linear independence [58].
- Optimal approximation [5].

The important properties of ASTS emanate directly from the topological properties of the underlying analysis-suitable T-mesh and resulting set of T-spline basis functions constructed from it.

3.2. T-spline basis functions, spaces, and geometry

Given a parametric domain $\hat{\Omega} = [0,1]^2$ we define global horizontal and vertical *open* knot vectors $h\mathsf{K} = \{s_1, s_2, \ldots, s_m\}$ and $v\mathsf{K} = \{t_1, t_2, \ldots, t_n\}$, respectively. In other words,

$$0 = s_1 = \ldots = s_{p+1} < s_{p+2} \leqslant \ldots \leqslant s_{m-p-1} < s_{m-p} = \ldots = s_m = 1$$

and

$$0 = t_1 = \ldots = t_{q+1} < t_{q+2} \leqslant \ldots \leqslant t_{n-q-1} < t_{n-q} = \ldots = t_n = 1.$$

As a result, every T-mesh vertex $V = \{i, j\} \in \hat{\hat{\Omega}}$ has the parametric representation $\{s_i, t_j\} \in \hat{\Omega}$. For reasons that will become apparent, we refer to a cell $C \in \mathsf{C}_{ext}$ with *positive parametric area* as a Bézier element. The parametric domain of a Bézier element is denoted by $\hat{\Omega}^e$. The set of all Bézier elements in a T-mesh is denoted by E.

For each anchor $A = a \times b \in \mathsf{A}$ we construct horizontal and vertical local index vectors $\{i_1, \ldots, i_{p+2}\}$ and $\{j_1, \ldots, j_{q+2}\}$ made up of increasing (but not necessarily consecutive) indices in $h\mathsf{I}$ and $v\mathsf{I}$, respectively. Note that for p odd, $\{i_{(p+3)/2}\} = a$, and for p even, $(i_{(p/2)+1}, i_{(p/2)+2}) = a$. Similar relationships hold for q. The procedure for determining local index vectors is shown in Fig. 4 for various polynomial degrees. To clarify this procedure we describe the anchors and associated local index vectors. In Figure 4a, $p = 2$ and $q = 2$ and thus the example anchor is the cell $(4, 8) \times (3, 7)$. The horizontal local index vector is $\{3, 4, 8, 10\}$ and the vertical local index vector is $\{2, 3, 7, 9\}$. We observe that subset of the vertical skeleton located at index 9 does not span the entire height of the anchor cell, hence it is not included in the horizontal local index vector; similarly since subset of the horizontal skeleton located at index 8 does not span the entire width of the cell it not included in the vertical local index vector. In Figure 4b, $p = 3$ and $q = 2$ thus the example anchor is the vertical edge $\{9\} \times (7, j)$. The horizontal local index vector is $\{5, 8, 9, 11, 12\}$ and the vertical local index vector is $\{6, 7, 9, 10\}$. Similar to the prior example the subset of the vertical skeleton at index

8 does not span the entire height of the anchor edge, hence it is not included in the horizontal local index vector. Figure 4c shows the case where $p = 2$ and $q = 3$ thus the example anchor is the horizontal edge $(4, 7) \times \{8\}$. The horizontal local index vector is $\{3, 4, 7, 8\}$ and the vertical local index vector is $\{3, 4, 8, 9, 10\}$. In the last case, shown in Figure 4d, $p = 3$ and $q = 3$ thus the example anchor is the vertex $\{8\} \times \{8\}$. The horizontal local index vector is $\{4, 5, 8, 9, 11\}$ and the vertical local index vector is $\{3, 4, 8, 9, 10\}$.

The T-spline blending function $N_A^{p,q}(s,t)$ is given by

$$N_A^{p,q}(s,t) := N_A^p[s_{i_1}, \ldots, s_{i_{p+2}}](s) N_A^q[t_{j_1}, \ldots, t_{j_{q+2}}](t) \quad \forall (s,t) \in \hat{\Omega} \tag{1}$$

where $N_A^p[s_{i_1}, \ldots, s_{i_{p+2}}](s)$ and $N_A^q[t_{j_1}, \ldots, t_{j_{q+2}}](t)$ are B-spline basis functions associated with the local knot vectors $[s_{i_1}, \ldots, s_{i_{p+2}}] \subset h\mathsf{K}$ and $[t_{j_1}, \ldots, t_{j_{q+2}}] \subset v\mathsf{K}$.

We define N to be the set of all basis functions associated with a T-mesh. Given a weight $w_A \in \mathbb{R}^+$ for each $A \in \mathsf{A}$ a *rational* T-spline basis function $R_A^{p,q} : \hat{\Omega} \to \mathbb{R}$ can be written as

$$R_A^{p,q}(s,t) = \frac{N_A^{p,q}(s,t)}{\sum_{A \in \mathsf{A}} w_A N_A^{p,q}(s,t)} \tag{2}$$

$$= \frac{N_A^{p,q}(s,t)}{w(s,t)} \tag{3}$$

where $w(s,t) : \hat{\Omega} \to \mathbb{R}$ is called a weight function. For clarity we will often suppress the dependence on the polynomial degrees p, q and write the basis function as R_A. Figure 5 shows several T-spline basis functions plotted in the parametric domain $\hat{\Omega}$. An ASTS space, denoted by $\mathcal{T}$, is the span of the blending functions in N constructed from an analysis-suitable T-mesh. Given vector valued control points, $\mathbf{P}_A \in \mathbb{R}^n$, $n = 2$ or 3, the geometry of a T-spline can be written as

$$\mathbf{x}(s,t) = \sum_{A \in \mathsf{A}} \mathbf{P}_A w_A R_A(s,t). \tag{4}$$

4. Hierarchical analysis-suitable T-splines

A hierarchical T-spline space is constructed from a finite sequence of N nested ASTS spaces, $\mathcal{T}^\alpha \subset \mathcal{T}^{\alpha+1}$, $\alpha = 1, \ldots, N-1$, and N bounded open index domains, $\hat{\hat{\Omega}}^N \subseteq \hat{\hat{\Omega}}^{N-1} \subseteq \cdots \subseteq \hat{\hat{\Omega}}^1$, which define the nested domains for the hierarchy. Two important theoretical results for ASTS will be used in the construction of hierarchical analysis-suitable T-splines:

Theorem 4.1. *Given two analysis-suitable T-meshes with non-overlapping T-junction extensions,* T^1 *and* T^2 *, if* $\mathsf{T}^1_{ext} \subseteq \mathsf{T}^2_{ext}$*, then* $\mathcal{T}^1 \subseteq \mathcal{T}^2$.

Theorem 4.2. *Analysis-suitable T-splines are locally linear independent.*

We note that to accommodate overlapping T-junction extensions requires a minor generalization of Theorem 4.1 which is not reproduced here to maintain clarity of exposition. For a complete description of the underlying theory we refer the interested reader to [5]. The local linear independence of ASTS is proven in [58].

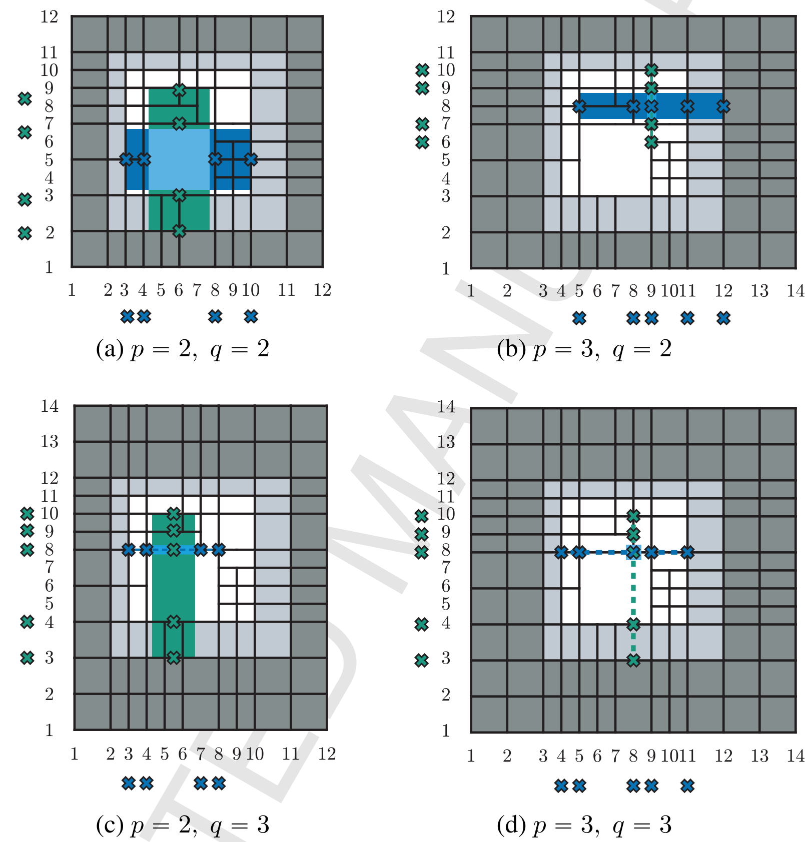


Figure 4: Examples for how local index vectors are constructed for T-spline basis functions of varying values of the polynomial degrees p and q. The function anchors are marked with light blue. The horizontal line used to determine the horizontal local index vector is indicated with a dark blue dashed line. The vertical line used to calculate the vertical local index vector is marked with a green dashed line. The indices that contribute to the local index vectors are marked with $\times$.

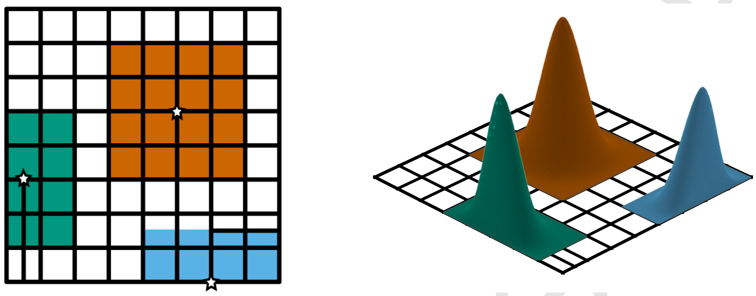


Figure 5: T-spline basis functions and supports in the parametric domain $\hat{\Omega}$ for $p = q = 3$. Anchor locations are denoted by a star.

4.1. *Sequences of analysis-suitable T-meshes*

We construct a sequence of N analysis-suitable T-meshes such that $\mathcal{T}^{\alpha} \subseteq \mathcal{T}^{\alpha+1}$, $\alpha = 1, \ldots, N-1$, as follows:

1. Create $\mathsf{T}^{\alpha+1}$ from T^{α} by subdividing each cell in E^{α} into four congruent cells.
2. Extend T-junctions in $\mathsf{T}^{\alpha+1}$ until it is analysis-suitable and $\mathsf{T}^{\alpha}_{ext} \subseteq \mathsf{T}^{\alpha+1}_{ext}$.

This algorithm is graphically demonstrated in Figure 6 for a particular T-mesh. For an efficient and general algorithm to produce nested analysis-suitable T-spline spaces see [2].

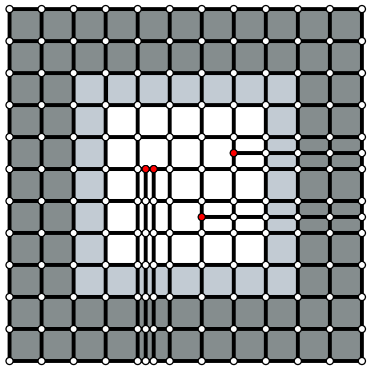
4.2. *Hierarchical T-spline spaces*

The hierarchical analysis-suitable T-spline basis can be constructed recursively in a manner analogous to that used for hierarchical B-splines [8]:

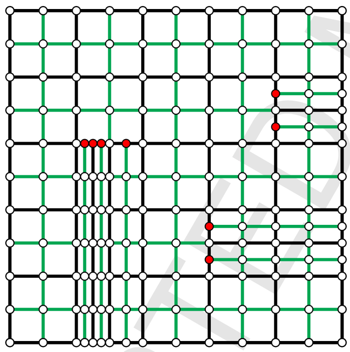
1. Initialize $\mathsf{H}^1 = \mathsf{N}^1$.
2. Recursively construct $\mathsf{H}^{\alpha+1}$ from H^{α} by setting
$$\mathsf{H}^{\alpha+1} = \mathsf{H}^{\alpha+1}_{\text{coarse}} \cup \mathsf{H}^{\alpha+1}_{\text{fine}}, \alpha = 1, \ldots, N-1,$$
where
$$\mathsf{H}^{\alpha+1}_{\text{coarse}} = \{N \in \mathsf{H}^{\alpha} : \text{supp}(N) \nsubseteq \hat{\Omega}^{\alpha+1}\},$$
and
$$\mathsf{H}^{\alpha+1}_{\text{fine}} = \{N \in \mathsf{N}^{\alpha+1} : \text{supp}(N) \subseteq \hat{\Omega}^{\alpha+1}\}.$$
3. Set $\mathsf{H} = \mathsf{H}^{N}$.

We denote the number of functions in H by n_f. We call the space spanned by the functions in H a hierarchical analysis-suitable T-spline space and denote it by $\mathcal{H}$. To make the ideas concrete a univariate hierarchical spline space is shown in Figure 7.

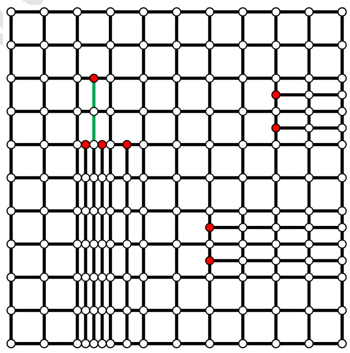
The linear independence of the functions in H follows immediately from the definition of hierarchical T-splines and the local linear independence of ASTS (see [58]).



(a) The initial T-mesh, T^{α}.



(b) Create $\mathsf{T}^{\alpha+1}$ from T^{α} by subdividing Bézier elements.



(c) Extend T-junctions until $\mathsf{T}^{\alpha+1}$ is analysis-suitable and $\mathsf{T}^{\alpha} \subseteq \mathsf{T}^{\alpha+1}$.

Figure 6: Generating nested analsysis-suitable T-meshes. In Figures (b) and (c) the zero parametric area region is omitted for clarity.

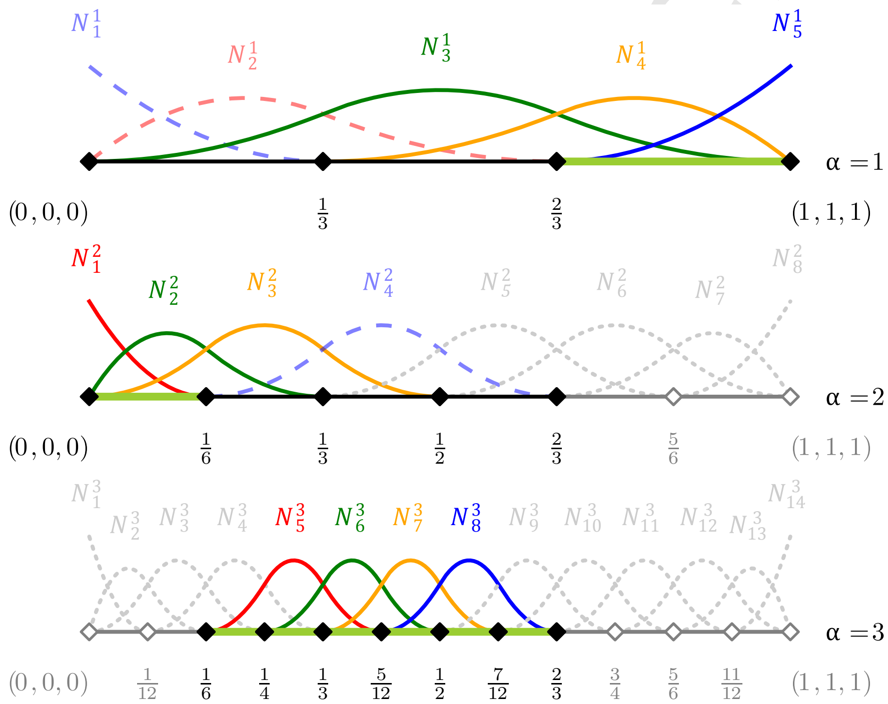


Figure 7: Basis functions for a univariate hierarchical spline space. Functions in H are indicated by solid colored lines. Functions dependent on higher-level functions are indicated by dashed colored lines. Functions which are not fully supported in the hierarchy of domains are indicated by grey dotted lines. Hierarchical domains are black and Bézier elements are green. Note that functions in H are supported entirely by Bézier elements.

Lemma 4.3. *The functions in the hierarchical basis* H *are linearly independent.*

Proof. See Lemma 2 in [8] □

Lemma 4.4. *Given* $\mathsf{H}^1, \ldots, \mathsf{H}^N$*, a sequence of hierarchical analysis-suitable T-spline bases,* $\operatorname{span} \mathsf{H}^\alpha \subseteq \operatorname{span} \mathsf{H}^{\alpha+1}$*,* $\alpha = 1, \ldots, N-1$*.*

Proof. See Lemma 3 in [8] □

By construction, $\mathcal{T}^1 \subseteq \mathcal{H}$, thus the approximation properties of analysis-suitable T-splines are inherited by their hierarchical counterpart. In particular, constants are exactly represented and all patch tests are satisfied [5, 57].

5. Bézier extraction of hierarchical analysis-suitable T-splines

The Bézier extraction framework [9, 29, 59] can be extended to HASTS in a straightforward fashion. Using Bézier extraction, the spline hierarchy is collapsed onto a single level finite element mesh which can then be processed by standard finite element codes without any explicit knowledge of HASTS algorithms or data structures.

5.1. Bernstein basis functions

The univariate Bernstein basis functions are written as

$$B_{i,p}(\xi) = \frac{1}{2^p}\binom{p}{i-1}(1-\xi)^{p-(i-1)}(1+\xi)^{i-1} \tag{5}$$

where $\xi \in [-1, 1]$ and the binomial coefficient $\binom{p}{i-1} = \frac{p!}{(i-1)!(p+1-i)!}$, $1 \leqslant i \leqslant p+1$. In CAGD, the Bernstein polynomials are usually defined over the unit interval $[0, 1]$, but in finite element analysis the biunit interval is preferred to take advantage of the usual domains for Gauss quadrature. The univariate Bernstein basis has the following properties:

- *Partition of unity.*
$$\sum_{i=1}^{p+1} B_{i,p}(\xi) = 1 \quad \forall \xi \in [-1, 1]$$
- *Pointwise nonnegativity.*
$$B_{i,p}(\xi) \geqslant 0 \quad \forall \xi \in [-1, 1]$$
- *Endpoint interpolation.*
$$B_{1,p}(-1) = B_{p+1,p}(1) = 1$$
- *Symmetry.*
$$B_{i,p}(\xi) = B_{p+1-i,p}(-\xi) \quad \forall \xi \in [-1, 1]$$

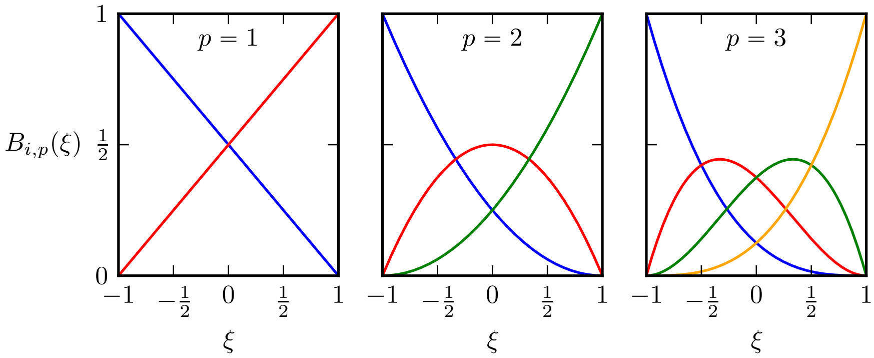


Figure 8: The Bernstein basis for polynomial degrees $p = 1, 2, 3$.

Figure 8 shows the Bernstein basis for polynomial degrees $p = 1, 2, 3$. We construct a bivariate Bernstein basis function of degree $\mathbf{p} = \{p, q\}$ by $B_{a,\mathbf{p}} : \bar{\Omega} \to \mathbb{R}^+ \cup 0$ where $a = 1, \ldots, n_b$, $n_b = (p+1)(q+1)$, and $\bar{\Omega} = [-1, 1]^2$, as the tensor product of univariate basis functions

$$B_{a(i,j),\mathbf{p}}(\xi, \eta) = B_{i,p}(\xi) B_{j,q}(\eta) \tag{6}$$

with

$$a\,(i, j) = (p+1)(j-1) + i. \tag{7}$$

5.2. The geometry of a hierarchical representation

In a single level T-spline, basis functions and control points have a one-to-one relationship and each control point influences the geometry in a similar manner. In a hierarchical context it is common to only associate control points with the functions in N^1. This is the convention adopted in this paper. Note that by construction every blending function in N^1 can be written in terms of basis functions in H (see Lemma 4.4). We call the functions in N^1 *geometric* blending functions. We use n_g to denote the number of geometric blending functions.

Given vector valued control points, $\mathbf{P}_G \in \mathbb{R}^n$, $n = 2$ or 3, and weights w_G, the geometry of a hierarchical representation $\mathbf{x} : \hat{\Omega} \to \Omega$ can be written as

$$\mathbf{x}(s, t) = \frac{\sum_{G=1}^{n_g} \mathbf{P}_G w_G N_G(s, t)}{\sum_{G=1}^{n_g} w_G N_G(s, t)} \tag{8}$$

$$= \frac{\sum_{G=1}^{n_g} \mathbf{P}_G w_G N_G(s, t)}{w(s, t)} \tag{9}$$

where $(s, t) \in \hat{\Omega}$, G is used to index the geometric blending functions, and $w(s, t)$ is the weight function. The decoupling of geometry from the basis functions in H is an additional complexity unique to hierarchical representations which is elegantly addressed via Bézier extraction.

5.3. Bézier Elements

The set of Bézier elements underlying a hierarchical T-spline are determined recursively in a manner similar to the basis. We denote the set of Bézier elements in a hierarchy by HE. We construct HE as follows:

1. Initialize $\mathsf{HE}^1 = \mathsf{E}^1$.
2. Recursively construct $\mathsf{HE}^{\alpha+1}$ from HE^{α} by setting

$$\mathsf{HE}^{\alpha+1} = \mathsf{HE}^{\alpha+1}_{coarse} \cup \mathsf{HE}^{\alpha+1}_{fine}, \alpha = 1, \ldots, N-1,$$

   where

$$\mathsf{HE}^{\alpha+1}_{coarse} = \{e \in \mathsf{HE}^{\alpha} : \hat{\Omega}^e \nsubseteq \hat{\Omega}^{\alpha+1}\},$$

   and

$$\mathsf{HE}^{\alpha+1}_{fine} = \{e \in \mathsf{E}^{\alpha+1} : \hat{\Omega}^e \subseteq \hat{\Omega}^{\alpha+1}\}.$$

3. Set $\mathsf{HE} = \mathsf{HE}^N$.

We denote the number of Bézier elements in HE by n_e.

5.4. Element localization

Using standard techniques [29, 59] it is possible to determine the set of functions in H which are nonzero over any element in HE. This gives rise to a standard element connectivity map which, given an element index e and local function index a, returns a global function index A. The reader is referred to [57] for additional details on common approaches to finite element localization and connectivity arrays. Note that A can indicate an anchor or a global function index.

We write a rational hierarchical T-spline basis function, restricted to element e, as

$$R^e_a(s,t) = \frac{N^e_a(s,t)}{w^e(s,t)} \tag{10}$$

where $(s,t) \in \hat{\Omega}^e$ and $w^e(s,t)$ is the element weight function restricted to element e. The element geometric map $\mathbf{x}^e : \hat{\Omega}^e \to \Omega^e$ is the restriction of $\mathbf{x}(s,t)$ to element e.

5.5. Bézier extraction

To present the basic ideas, Bézier extraction for a B-spline curve is shown graphically in Figure 9. Bézier extraction constructs a linear transformation defined by a matrix referred to as the extraction operator. The extraction operator maps a Bernstein polynomial basis defined on Bézier elements to the global spline basis. The transpose of the extraction operator maps the control points of the spline to the Bézier control points.

Each hierarchical basis function supported by element e can be written in Bézier form as

$$N^e_a(s(\xi), t(\eta)) = \sum_{b=1}^{n_b} c^a_b B_b(\xi, \eta) \tag{11}$$

$$= \bar{N}^e_a(\xi, \eta) \tag{12}$$

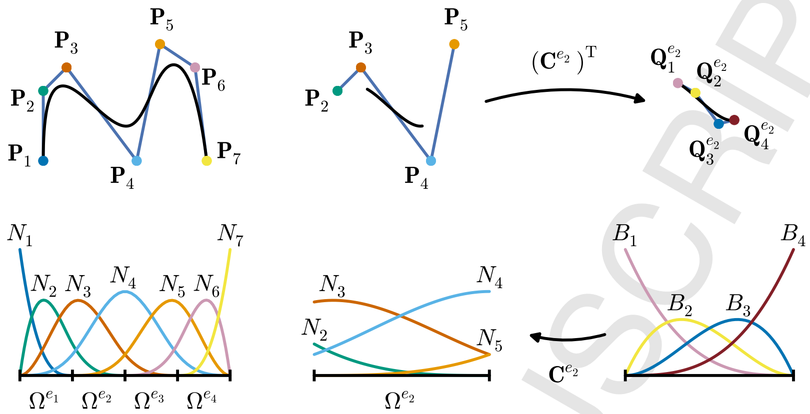


Figure 9: Illustration of the Bézier extraction operator $\mathbf{C}^e$ for a spline of degree 3.

where the dependence of the Bernstein polynomial $B_b(\xi,\eta)$ on the polynomial degrees p and q has been suppressed for clarity. The overbar will be used to denote a quantity written in terms of the Bernstein basis defined over the element domain $\bar{\Omega}$. The Bézier coefficients c^a_b are computed using standard knot insertion techniques [29]. We denote the vector of hierarchical basis functions supported by element e by $\mathbf{H}^e(\xi,\eta)$ and the vector of Bernstein basis functions by $\mathbf{B}(\xi,\eta)$. We then have that

$$\mathbf{N}^e(s(\xi),t(\eta)) = \mathbf{C}^e\mathbf{B}(\xi,\eta) \tag{13}$$

$$= \bar{\mathbf{N}}^e(\xi,\eta) \tag{14}$$

where $\mathbf{C}^e$ is the *element extraction operator* (see [9]). In other words, the element extraction operator is composed of the Bézier coefficients c^a_b.

We write the element weight function as

$$\begin{aligned} w^e(s(\xi),t(\eta)) &= \sum_{g=1}^{n^e_g} w^e_g N^e_g(s(\xi),t(\eta)) \\ &= \sum_{g=1}^{n^e_g} w^e_g \sum_{b=1}^{n_b} c^g_b B_b(\xi,\eta) \\ &= \sum_{b=1}^{n_b}\left(\sum_{g=1}^{n^e_g} w^e_g c^g_b\right) B_b(\xi,\eta) \\ &= \sum_{b=1}^{n_b} w^e_b B_b(\xi,\eta) \\ &= \bar{w}^e(\xi,\eta). \end{aligned}$$

where n^e_g is the number of geometric basis functions which are non-zero over element

e. We may also write the rational hierarchical basis functions as

$$\begin{aligned} R_a^e(s(\xi),t(\eta)) &= \frac{N_a^e(s(\xi),t(\eta))}{w^e(s(\xi),t(\eta))} \\ &= \frac{\bar{N}_a^e(\xi,\eta)}{\bar{w}^e(\xi,\eta)} \\ &= \bar{R}_a^e(\xi,\eta). \end{aligned}$$

Finally, the element geometric map can be written as

$$\begin{aligned} \mathbf{x}^e(s(\xi),t(\eta)) &= \frac{\sum_{g=1}^{n_g^e} \mathbf{P}_g^e w_g^e N_g^e(s(\xi),t(\eta))}{w^e(s(\xi),t(\eta))} \\ &= \frac{\sum_{g=1}^{n_g^e} \mathbf{P}_g^e w_g^e \sum_{b=1}^{n_b} c_b^g B_b(\xi,\eta)}{\bar{w}^e(\xi,\eta)} \\ &= \frac{\sum_{b=1}^{n_b} \left(\sum_{g=1}^{n_g^e} \mathbf{P}_g^e w_g^e c_b^g \right) B_b(\xi,\eta)}{\bar{w}^e(\xi,\eta)} \\ &= \frac{\sum_{b=1}^{n_b} \mathbf{Q}_b^e w_b^e B_b(\xi,\eta)}{\bar{w}^e(\xi,\eta)} \\ &= \bar{\mathbf{x}}^e(\xi,\eta). \end{aligned}$$

The implementation of a finite element framework based on Bézier extraction is described in detail in [29, 59].

6. Computational Results

We illustrate the use of hierarchical T-splines in the context of isogeometric analysis. We consider problems that highlight the unique attributes of both hierarchical refinement and T-splines. The examples used are inspired by those found in [2, 8, 9].

6.1. A comparison between ASTS and HASTS local refinement

We compare local refinement of ASTS to local refinement of HASTS. When working with ASTS all refinement is performed on a single level whereas when working with HASTS this constraint is relaxed. For additional algorithmic details on local refinement of ASTS see [2]. We locally refine the T-spline ship hull design shown in Fig. 10 using both methods. The geometry is constructed using the Autodesk T-spline plugin for Rhino3d [60]. T-splines are popular in ship hull design because an entire hull can be modeled by a single watertight surface with a minimal number of control points [61]. T-junctions can be used to efficiently model local features. Note that the initial T-spline of the hull contains just 75 control points and 36 Bézier elements.

We restrict the refinement region to the locations detailed in Fig. 11. It is assumed that the original design is too coarse to be used as a basis for analysis and additional

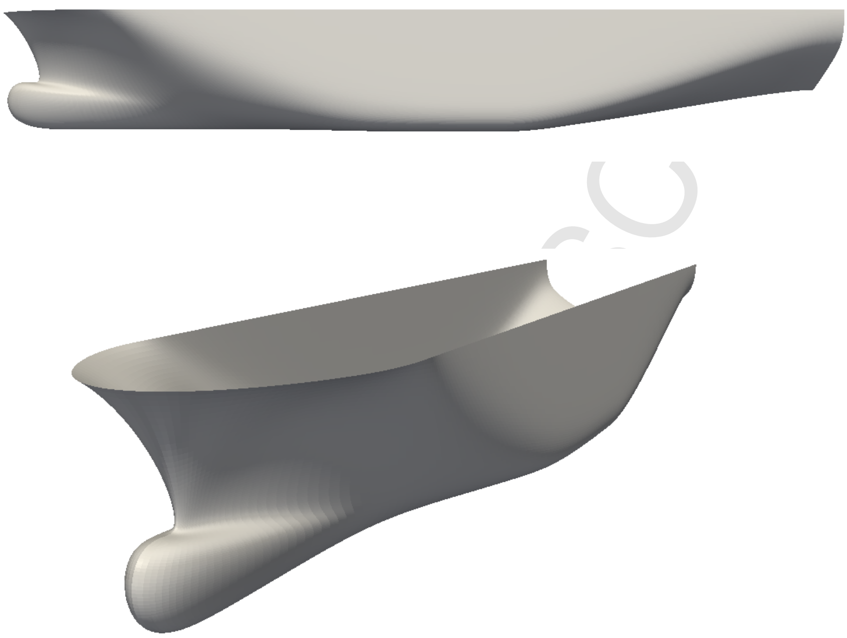


Figure 10: A T-spline container ship hull. The surface is C^2-continuous everywhere except along the line of symmetry where it is C^0.

resolution is required in the rectangular region followed by highly localized refinements along the region corresponding to the curve.

The HASTS refinement algorithm is based on the algorithm presented in [9] for spline forests. The algorithm is element-based, meaning refinement is driven by the subdivision of Bézier elements. The hierarchical basis is then reextracted into the new hierarchical T-mesh topology to generate the new set of Bézier elements. A detailed description of the underlying algorithms, in the context of HASTS, will be postponed to a future publication. Figure 12 shows three HASTS local refinements along the curve shown in Figure 11. The elements are colored according to their level, α, in the hierarchy. Note that *no* nonlocal propagation of local refinement occurs for HASTS. Only those elements specified for refinement are subdivided. This is possible due to the relaxation of the single level constraint inherent in ASTS. The refinements form a nested sequence of C^2-continuous hierarchical analysis-suitable T-spline spaces. The geometry of the hull is exactly preserved during refinement. The final HASTS is composed of 1857 Bézier elements and 1193 basis functions. However, only 75 geometric blending functions and control points are used to define the hull geometry.

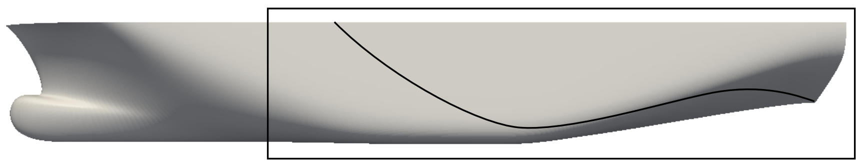


Figure 11: The regions of the container ship hull where local refinement will be performed. Refinement is first performed in the rectangular region followed by highly localized refinement along the curve.

As a comparison, Figure 13 shows the results of ASTS local refinement using the algorithm from [2]. The top figure shows the control points added during local refinement (black dots) along the curve. The region selected for refinement is shown in red. Observe the propagation of the control points away from the selected refinement region. The bottom figure shows the resulting Bézier elements after refinement. Superfluous control points and elements are added just to satisfy the single level constraint inherent in the definition of ASTS.

6.2. *HASTS as an adaptive basis*

We now consider HASTS as an adaptive basis for isogeometric analysis. In particular, the demands on the refinement algorithm are more stringent in this context. Additionally, we demonstrate more clearly the impact of T-junctions on hierarchical T-spline refinement. We choose as a benchmark the advection skew to the mesh problem shown in Figure 14. This problem is advection dominated, with diffusivity of 10^{-6}. Along the external boundary, the boundary conditions are selected such that sharp interior and boundary layers are present in the solution. In this case, $\theta = 45$ degrees.

6.2.1. *Problem Statement*

Let Ω be a bounded region in $\mathbb{R}^2$ and assume Ω has a piecewise smooth boundary Γ. Let $\mathbf{x} = \{x_i\}_{i=1}^2$ denote a general point in $\overline{\Omega}$, and let the temperature at a point $\mathbf{x} \in \Omega$ be denoted by $\phi(\mathbf{x}) \in \mathbb{R}$. Given Dirichlet boundary data, $g : \Gamma \to \mathbb{R}$, the steady-state advection-diffusion boundary value problem consists of finding the temperature ϕ such that

$$\begin{aligned} \mathbf{u} \cdot \nabla\phi - \nabla \cdot (\boldsymbol{\kappa}\nabla\phi) &= 0 \text{ on } \Omega \\ \phi &= g \text{ on } \Gamma \end{aligned} \tag{15}$$

where $\mathbf{u} : \Omega \to \mathbb{R}^2$ and $\boldsymbol{\kappa} : \Omega \to \mathbb{R}^{2\times2}$ are the spatially varying solenoidal velocity vector and symmetric, positive-definite, diffusivity tensor, respectively. Note that in this paper we define $\boldsymbol{\kappa} = \kappa\delta_{ij}$ where κ is a positive constant called the diffusivity coefficient. We employ SUPG [62] with a standard definition for the element stabilization parameter, τ^e.

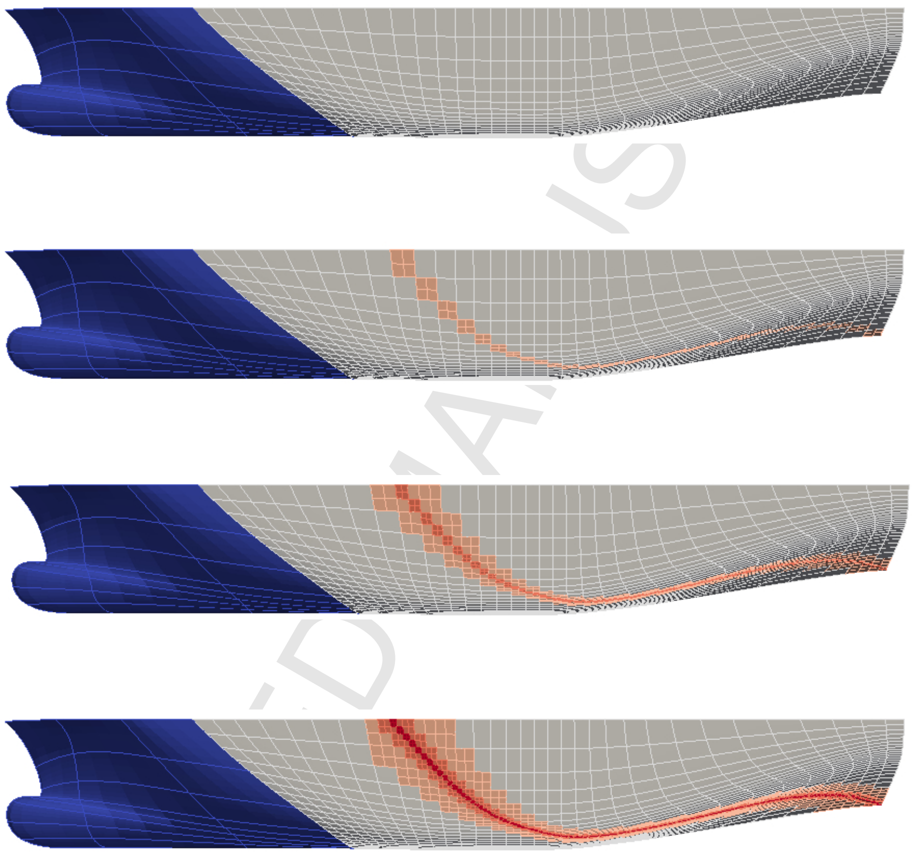


Figure 12: Three iterations of HASTS local refinement along the curve for the container ship hull in Figure 10. The elements are colored according to their level, α, in the hierarchy. The refinements form a nested sequence of C^2-continuous hierarchical analysis-suitable T-spline spaces. The geometry of the hull is exactly preserved during refinement.

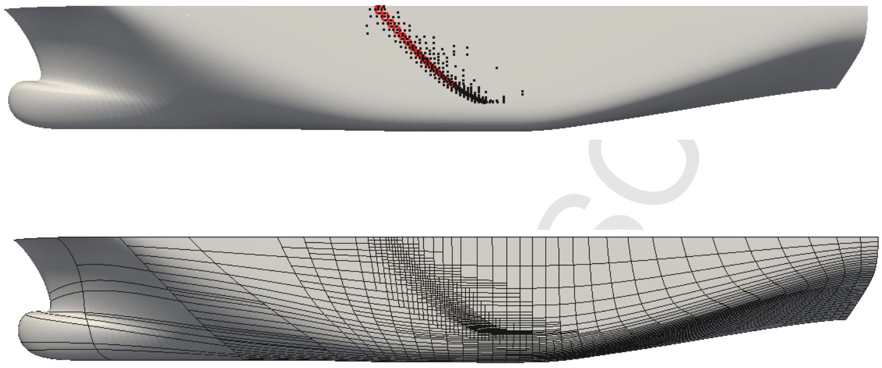


Figure 13: The results of ASTS local refinement using the algorithm from [2]. The top figure shows the control points added during local refinement (black dots) along the curve. Observe the propagation of the control points away from the selected refinement region. The bottom figure shows the resulting Bézier elements after refinement. Superfluous control points and elements are added just to satisfy the single level constraint inherent in the definition of ASTS.

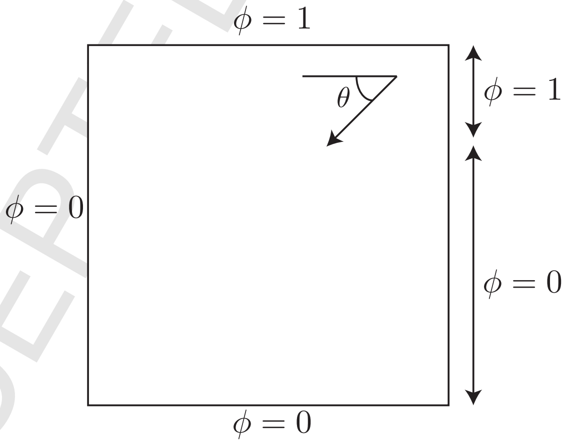


Figure 14: The advection skew to the mesh problem statement.

6.2.2. A residual based error estimator

To estimate the error we employ a simple residual-based explicit estimator based on the variational multiscale theory for fluids [63, 64, 65, 66]. It is given by

$$||\phi'||_{\Omega^e} \approx \tau^e ||\mathbf{u} \cdot \nabla\phi - \nabla \cdot \boldsymbol{\kappa}\nabla\phi||_{\Omega^e}. \tag{16}$$

Note that this error estimator underestimates the error for diffusion-dominated flows but is adequate for the advection-dominated benchmark presented in this paper. Using standard techniques [57] we use the element scaling

$$r = \frac{h^e_{\alpha+1}}{h^e_\alpha} = \left(\frac{tol}{||\phi'||_{\Omega^e}}\right)^{1/\beta} \tag{17}$$

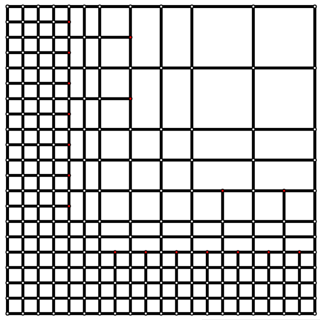
where $h^e_\alpha, h^e_{\alpha+1}$ are the mesh size distributions for T^α and $\mathsf{T}^{\alpha+1}$, respectively, and β is the order of convergence of the method. The element size, h^e, is the square root of the element area. We flag elements for refinement if $r < 1$. The adaptive process is repeated until a specified convergence tolerance is attained or a maximum number of hierarchical levels are introduced. While error estimation is not the focus of this paper the design and application of appropriate error estimators to T-spline-based refinement is an interesting area of future work.

6.2.3. Results

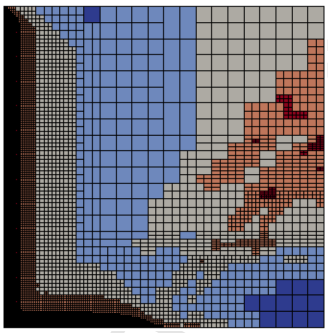
We solve the problem with C^1 biquadratic and C^2 bicubic hierarchical T-splines. The initial T-mesh for both cases is shown in Figure 15a and the final Bézier meshes for both the quadratic and cubic case are shown in Fig. 15b and Fig. 15c. A linear parameterization was employed in all cases and the level zero control points and blending functions define the geometry. Note that the *initial* T-mesh is locally refined and this refinement is *not* hierarchical. All future hierarchical refinements must respect the structure of the T-spline space on the first level. In general, we have found that judiciously performing local refinement of the first level of a T-spline hierarchy to accommodate geometric features or boundary conditions leads to smaller hierarchies and more efficient solution procedures. The solutions after one, three and five refinements for both the biquadratic and bicubic cases are shown in Figure 16. The converged solutions in both cases accurately captures the sharp boundary layers.

During each adaptive step the error is assessed as described in Section 6.2.2, elements are flagged for refinement and subdivided, and a new hierarchical basis is then extracted into the new hierarchical T-mesh topology. This generates a refined set of Bézier elements. This is illustrated in Figure 17. The sequence of biquadratic refinements form a nested sequence of C^1-continuous HASTS spaces, whereas the sequence of bicubic refinements form a nested sequence of C^2-continuous HASTS spaces. Observe that fewer elements are required for convergence as the smoothness and order of the basis increases [9].

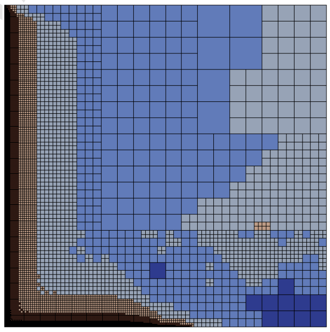
To illustrate the structure and distribution of the hierarchical basis in the presence of T-junctions the bicubic Greville abscissae [23] corresponding to the hierarchical T-spline basis after two refinements are shown in Figure 18. The dots are scaled and colored according to their level in the hierarchy; a larger dot denotes a lower level.



(a) Initial T-mesh

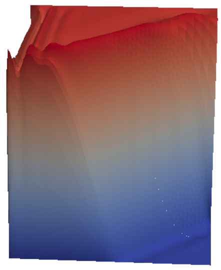


(b) Converged biquadratic Bézier mesh

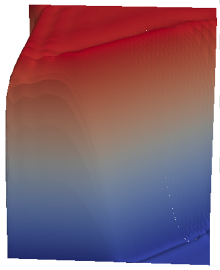


(c) Converged bicubic Bézier mesh

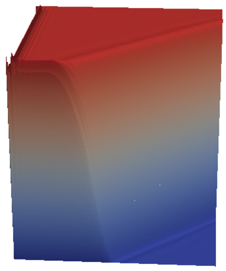
Figure 15: The initial mesh for the static advection skew to the mesh problem (a) and the converged quadratic (b) and cubic (c) Bézier meshes.



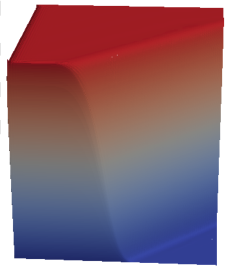
(a) Biquadratic solution 1



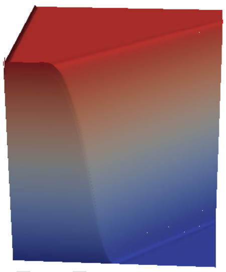
(b) Bicubic solution 1



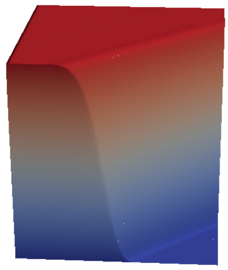
(c) Biquadratic solution 3



(d) Bicubic solution 3

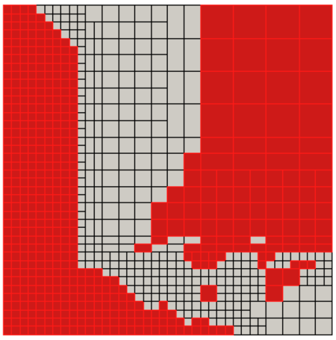


(c) Biquadratic solution 5

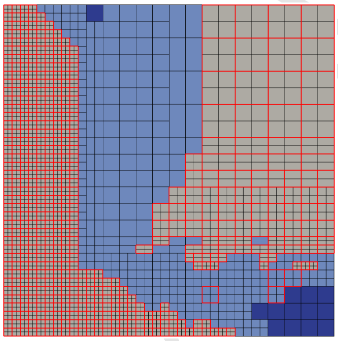


(d) Bicubic solution 5

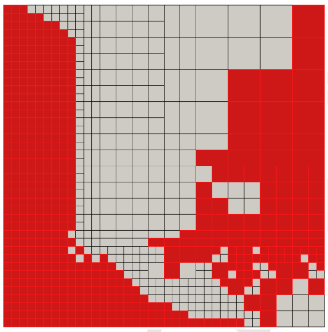
Figure 16: Biquadratic and bicubic solutions to the advection skew to the mesh problem after one, three, and five refinements. The final solutions correspond to the Bézier meshes in Figure 15.



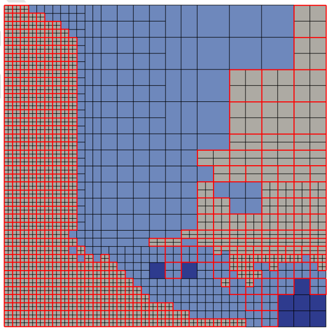
(a) Second biquadratic Bézier mesh with elements flagged for refinement



(b) Biquadratic Bézier mesh after refinement



(c) Second bicubic Bézier mesh with elements flagged for refinement



(d) Bicubic Bézier mesh after refinement

Figure 17: Error-indicator based refinement for the static skew advection problem. In the left column the second mesh is shown (i.e. the mesh after one refinement) with elements flagged for refinement in red. In the right column the refinement has been performed. In this column the elements are colored according to their level, α, in the hierarchy and the red outlines correspond to the boundaries of the selected element. Observe the locality of refinement.

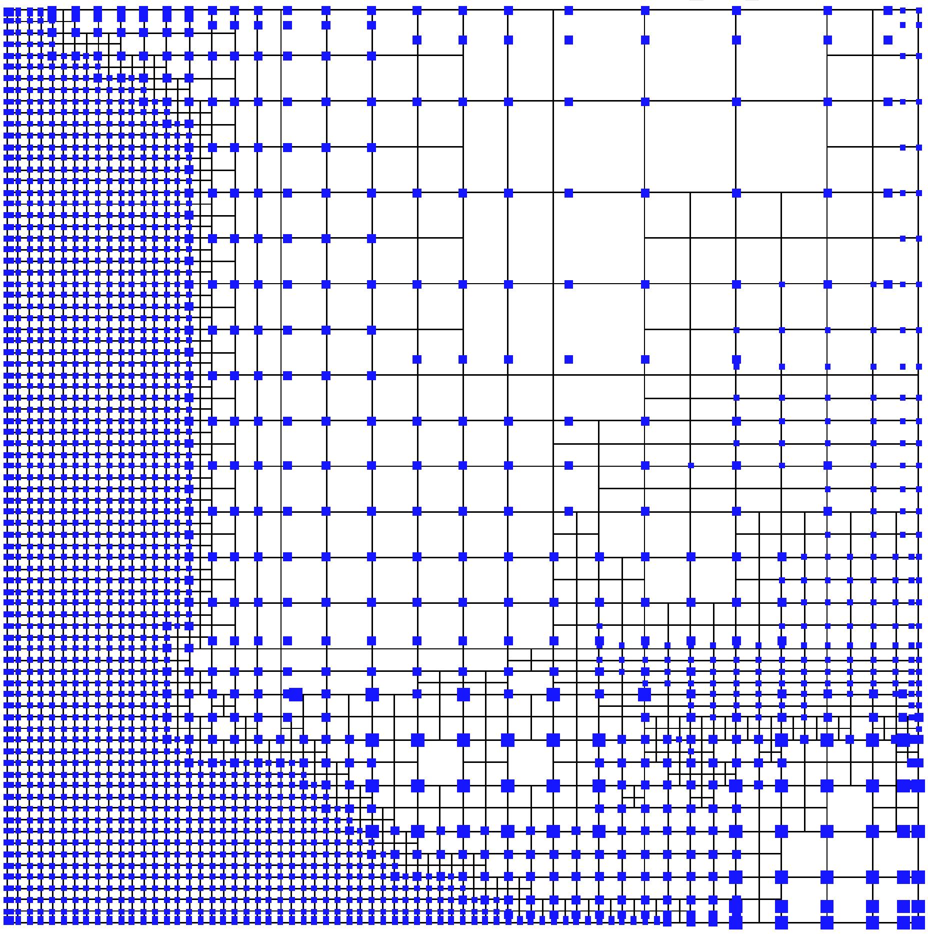


Figure 18: The Greville abscissae (blue dots) corresponding to the basis functions after two applications of HASTS local refinement for the bicubic advection skew to the mesh problem. The size of the dot corresponds to the basis function level with larger dots representing lower levels.

7. Conclusion

We have presented hierarchical analysis-suitable T-splines (HASTS). HASTS are a superset of both analysis-suitable T-splines and hierarchical B-splines. We developed a Bézier extraction framework for HASTS. We then demonstrated the potential of HASTS in the context of two demanding numerical computations. In future work, we will consider p and k refinement and coarsening of HASTS and extend HASTS to the spline forest setting [9].

References

[1] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, M. A. Scott, On linear independence of T-spline blending functions, Computer Aided Geometric Design 29 (2012) 63 – 76.

[2] M. A. Scott, X. Li, T. W. Sederberg, T. J. R. Hughes, Local refinement of analysis-suitable T-splines, Computer Methods in Applied Mechanics and Engineering 213 (2012) 206 – 222.

[3] L. Beirão da Veiga, A. Buffa, D. Cho, G. Sangalli, Analysis-suitable T-splines are dual-compatible, Computer Methods in Applied Mechanics and Engineering 249 – 252 (2012) 42 – 51.

[4] L. B. da Veiga, A. Buffa, G. Sangalli, R. Vázquez, Analysis-suitable T-splines of arbitrary degree: definition, linear independence, and approximation properties, Mathematical Models and Methods in Applied Sciences 23 (11) (2013) 1979 – 2003.

[5] X. Li, M. A. Scott, Analysis-suitable T-splines: characterization, refineability, and approximation, Mathematical Models and Methods in Applied Science 24 (06) (2014) 1141–1164.

[6] D. R. Forsey, R. H. Bartels, Hierarchical B-spline refinement, ACM SIGGRAPH Computer Graphics 22 (4) (1988) 205–212.

[7] D. Schillinger, L. Dedé, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, T. J. R. Hughes, An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces, Computer Methods in Applied Mechanics and Engineering 249 – 252 (2012) 116 – 150.

[8] A. Vuong, C. Giannelli, B. Jüttler, B. Simeon, A hierarchical approach to adaptive local refinement in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 200 (49 – 52) (2011) 3554 – 3567.

[9] M. A. Scott, D. C. Thomas, E. J. Evans, Isogeometric spline forests, Computer Methods in Applied Mechanics and Engineering 269 (2014) 222 – 264.

[10] C. Giannelli, B. Jüttler, Bases and dimensions of bivariate hierarchical tensor-product splines, Journal of Computational and Applied Mathematics 239 (0) (2013) 162 – 178.

[11] C. Giannelli, B. Jüttler, H. Speleers, THB–splines: The truncated basis for hierarchical splines, Computer Aided Geometric Design 29 (7) (2012) 485 – 498.

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

[13] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, T. Lyche, T-spline simplification and local refinement, ACM Trans. Graph. 23 (2004) 276–283.

[14] Autodesk, Autodesk T-Splines Plug-in for Rhino user manual, Autodesk (2012).

[15] Autodesk, Inc., Autodesk Fusion 360, Autodesk, Inc. (2014).

[16] T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (2005) 4135–4195.

[17] J. A. Cottrell, T. J. R. Hughes, Y. Bazilevs, Isogeometric analysis: Toward Integration of CAD and FEA, Wiley, Chichester, 2009.

[18] J. A. Evans, Y. Bazilevs, I. Babuška, T. J. R. Hughes, n-widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method, Computer Methods in Applied Mechanics and Engineering 198 (21-26) (2009) 1726–1741.

[19] T. J. R. Hughes, J. A. Evans, A. Reali, Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems, Computer Methods in Applied Mechanics and Engineering 272 (2014) 290 – 320.

[20] J. A. Cottrell, T. J. R. Hughes, A. Reali, Studies of refinement and continuity in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 196 (2007) 4160–4183.

[21] W. Wang, Y. Zhang, M. A. Scott, T. J. R. Hughes, Converting an unstructured quadrilateral mesh to a standard T-spline surface, Computational Mechanics, 48 (2011) 477 – 498.

[22] L. Liu, Y. Zhang, T. J. R. Hughes, M. A. Scott, T. W. Sederberg, Volumetric T-spline construction using boolean operations, in: J. Sarrate, M. Staten (Eds.), Proceedings of the 22nd International Meshing Roundtable, Springer International Publishing, 2014, pp. 405–424.

[23] M. A. Scott, R. N. Simpson, J. A. Evans, S. Lipton, S. P. A. Bordas, T. J. R. Hughes, T. W. Sederberg, Isogeometric boundary element analysis using unstructured T-splines, Computer Methods in Applied Mechanics and Engineering 254 (2013) 197 – 221.

[24] R. Schmidt, R. Wüchner, K.-U. Bletzinger, Isogeometric analysis of trimmed NURBS geometries, Computer Methods in Applied Mechanics and Engineering 241–244 (2012) 93 – 111.

[25] D. J. Benson, Y. Bazilevs, E. De Luycker, M. C. Hsu, M. A. Scott, T. J. R. Hughes, T. Belytschko, A generalized finite element formulation for arbitrary basis functions: From isogeometric analysis to XFEM, International Journal for Numerical Methods in Engineering 83 (2010) 765–785.

[26] W. A. Wall, M. A. Frenzel, C. Cyron, Isogeometric structural shape optimization, Computer Methods in Applied Mechanics and Engineering 197 (2008) 2976–2988.

[27] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, T. W. Sederberg, Isogeometric analysis using T-splines, Computer Methods in Applied Mechanics and Engineering 199 (5-8) (2010) 229–263.

[28] M. Dörfel, B. Jüttler, B. Simeon, Adaptive isogeometric analysis by local h-refinement with T-splines, Computer Methods in Applied Mechanics and Engineering 199 (5–8) (2009) 264–275.

[29] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, T. J. R. Hughes, Isogeometric Finite Element Data Structures based on Bézier Extraction of T-splines, International Journal for Numerical Methods in Engineering, 88 (2011) 126 – 156.

[30] C. V. Verhoosel, M. A. Scott, T. J. R. Hughes, R. de Borst, An isogeometric analysis approach to gradient damage models, International Journal for Numerical Methods in Engineering, 86 (2011) 115–134.

[31] C. V. Verhoosel, M. A. Scott, R. de Borst, T. J. R. Hughes, An isogeometric approach to cohesive zone modeling, International Journal for Numerical Methods in Engineering, 87 (2011) 336 – 360.

[32] M. J. Borden, M. A. Scott, C. V. Verhoosel, C. M. Landis, T. J. R. Hughes, A phase-field description of dynamic brittle fracture, Computer Methods in Applied Mechanics and Engineering 217 (2012) 77 – 95.

[33] R. N. Simpson, M. A. Scott, M. Taus, D. C. Thomas, H. Lian, Acoustic isogeometric boundary element analysis, Computer Methods in Applied Mechanics and Engineering 269 (2014) 265–290.

[34] R. Dimitri, L. D. Lorenzis, M. A. Scott, P. Wriggers, R. Taylor, G. Zavarise, Isogeometric large deformation frictionless contact using T-splines, Computer methods in applied mechanics and engineering 269 (2014) 394 – 414.

[35] S. Hosseini, J. J. Remmers, C. V. Verhoosel, R. de Borst, An isogeometric continuum shell element for non-linear analysis, Computer Methods in Applied Mechanics and Engineering 271 (2014) 1 – 22.

[36] Y. Bazilevs, M. C. Hsu, M. A. Scott, Isogeometric fluid-structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines, Computer Methods in Applied Mechanics and Engineering 249 - 252 (2012) 28 – 41.

[37] A. Buffa, G. Sangalli, R. Vázquez, Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations, Journal of Computational Physics 257, Part B (2014) 1291 – 1320.

[38] A. I. Ginnis, K. V. Kostas, C. G. Politis, P. D. Kaklis, K. A. Belibassakis, T. P. Gerostathis, M. A. Scott, T. J. R. Hughes, Isogeometric boundary-element analysis for the wave-resistance problem using T-splines, Computer Methods in Applied Mechanics and Engineering submitted.

[39] D. C. Thomas, M. A. Scott, J. A. Evans, K. Tew, E. J. Evans, Bézier projection: a unified approach for local projection and quadrature-free refinement and coarsening of NURBS and T-splines with particular application to isogeometric design and analysis, submitted.

[40] G. Kuru, C. Verhoosel, K. van der Zee, E. van Brummelen, Goal-adaptive isogeometric analysis with hierarchical splines, Computer Methods in Applied Mechanics and Engineering 270 (2014) 270 – 292.

[41] D. Schillinger, J. A. Evans, A. Reali, M. A. Scott, T. J. R. Hughes, Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations, Computer Methods in Applied Mechanics and Engineering 267 (2013) 170 – 232.

[42] E. Grinspun, P. Krysl, P. Schröder, CHARMS: a simple framework for adaptive simulation, ACM Transactions on Graphics 21 (3) (2002) 281–290.

[43] D. Schillinger, E. Rank, An unfitted hp-adaptive finite element method based on hierarchical B-splines for interface problems of complex geometry, Computer Methods in Applied Mechanics and Engineering 200 (47–48) (2011) 3358 – 3380.

[44] G. Kiss, C. Giannelli, B. Jüttler, Algorithms and data structures for truncated hierarchical B-splines, in: M. Floater, T. Lyche, M.-L. Mazure, K. Mørken, L. Schumaker (Eds.), Mathematical Methods for Curves and Surfaces, Vol. 8177 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2014, pp. 304–323.

[45] C. Giannelli, B. Jüttler, H. Speleers, Strongly stable bases for adaptively refined multilevel spline spaces, Advances in Computational Mathematics (2013) 1–32.

[46] D. Berdinsky, T. wan Kim, C. Bracco, D. Cho, B. Mourrain, M. Oh, S. Kiatpanichgij, Dimensions and bases of hierarchical tensor-product splines, Journal of Computational and Applied Mathematics 257 (2014) 86 – 104.

[47] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, Y. Feng, Polynomial splines over hierarchical T-meshes, Graphical Models 74 (2008) 76–86.

[48] X. Li, J. Deng, F. Chen, Surface modeling with polynomial splines over hierarchical T-meshes, The Visual Computer 23 (2007) 1027–1033.

[49] X. Li, J. Deng, F. Chen, Polynomial splines over general T-meshes, The Visual Computer 26 (2010) 277–286.

[50] X. Li, J. Deng, F. Chen, The dimension of spline spaces over 3d hierarchical T-meshes, Journal of Information and Computational Science 3 (2006) 487–501.

[51] H. Kang, F. Chen, J. Deng, Modified T-splines, Computer Aided Geometric Design 30 (9) (2013) 827 – 843.

[52] T. Dokken, T. Lyche, K. F. Pettersen, Polynomial splines over locally refined box-partitions, Computer Aided Geometric Design 30 (3) (2013) 331–356.

[53] A. Bressan, Some properties of LR-splines, Computer Aided Geometric Design 30 (8) (2013) 778 – 794.

[54] C. Manni, F. Pelosi, M. L. Sampoli, Generalized B-splines as a tool in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 200 (5–8) (2011) 867 – 881.

[55] P. Constantini, C. Manni, F. Pelosi, M. L. Sampoli, Quasi-interpolation in isogeometric analysis based on generalized B-splines, Computer Aided Geometric Design 27 (8) (2010) 656–668.

[56] C. Bracco, D. Berdinsky, D. Cho, M. Oh, T. wan Kim, Trigonometric generalized T-splines, Computer Methods in Applied Mechanics and Engineering 268 (2014) 540 – 556.

[57] T. J. R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Dover Publications, Mineola, NY, 2000.

[58] L. B. da Veiga, A. Buffa, G. Sangalli, R. Vázquez, Mathematical analysis of variational isogeometric methods, Acta Numerica 23 (2014) 157–287.

[59] M. J. Borden, M. A. Scott, J. A. Evans, T. J. R. Hughes, Isogeometric finite element data structures based on Bézier extraction of NURBS, International Journal for Numerical Methods in Engineering, 87 (2011) 15 – 47.

[60] Autodesk, Inc., http://www.tsplines.com/rhino/ (2011).

[61] M. T. Sederberg, T. W. Sederberg, T-splines: A technology for marine design with minimal control points, `http://www.tsplines.com/technicalpapers.html` (2010).

[62] A. N. Brooks, T. J. R. Hughes, Streamline upwind / Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, Computer Methods in Applied Mechanics and Engineering 32 (1982) 199–259.

[63] T. J. R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Computer Methods in Applied Mechanics and Engineering 127 (1995) 387–401.

[64] M. G. Larson, A. Målqvist, Adaptive variational multiscale methods based on a posteriori error estimation: Duality techniques for elliptic problems, in: B. Engquist, O. Runborg, P. Lötstedt (Eds.), Multiscale Methods in Science and Engineering, Vol. 44 of Lecture Notes in Computational Science and Engineering, Springer Berlin Heidelberg, 2005, pp. 181–193.

[65] G. Hauke, M. H. Doweidar, M. Miana, The multiscale approach to error estimation and adaptivity, Computer Methods in Applied Mechanics and Engineering 195 (13–16) (2006) 1573 – 1593.

[66] G. Hauke, M. H. Doweidar, S. Fuentes, Mesh adaptivity for the transport equation led by variational multiscale error estimators, International Journal for Numerical Methods in Fluids 69 (12) (2012) 1835–1850.