Strategies for local refinement and coarsening of hierarchical splines with particular application to phase-field models of prostate tumor growth

G. Lorenzo[a,*], M.A. Scott[b], K. Tew[c], Thomas J.R. Hughes[d], H. Gomez[a]

[a]*Departamento de Métodos Matemáticos e de Representación, Universidade da Coruña, Campus de Elviña s/n, 15071 A Coruña, Spain*
[b]*Department of Civil and Environmental Engineering, Brigham Young University, Provo, Utah 84602, USA*
[c]*Department of Information Technology, Brigham Young University, Provo, Utah 84602, USA*
[d]*Institute for Computational Engineering and Sciences, The University of Texas at Austin, 201 East 24th Street, C0200, Austin, TX 78712-1229, USA*

Abstract

We present a set of algorithms for local h-adaptivity of hierarchical B-splines for application in isogeometric analysis. We extend Bézier projection, an efficient quadrature-free local projection technique, to the hierarchical setting. In this case, extraction operators may not be invertible. To address this issue we develop a multi-level reconstruction operator which maintains the locality properties of the projection. We also introduce a balance parameter to control the overlap of hierarchical functions leading to improved numerical conditioning. We apply our algorithms to the simulation of localized prostate cancer growth. We model this disease using the phase-field method and a set of diffusion-reaction equations to account for the dynamics of nutrients and a key biomarker termed Prostate Specific Antigen. Our results include examples on simple 2D and 3D domains and a more compelling tissue-scale, patient-specific simulation, which is run over a prostate anatomy extracted from medical images. Our methods for local h-adaptivity efficiently capture the evolving interface between the tumor and the neighboring healthy tissue with remarkable accuracy in all cases.

Keywords: isogeometric analysis, Bézier projection, local refinement and coarsening, hierarchical spline spaces, phase field, tumor growth

1. Introduction

Isogeometric analysis (IGA) is a computational technology that tightly connects computer aided design (CAD) and finite element analysis (FEA) [1, 2]. The central idea of IGA is to adopt the smooth basis functions which define the CAD geometry as the basis for analysis. Thus, for properly constituted CAD models, isogeometric methods bypass the mesh generation and geometry clean-up steps which characterize classical FEA. CAD descriptions which are suitable for analysis are called analysis-suitable geometry (ASG).

---

*Corresponding author
Email address: guillermo.lorenzo@udc.es (G. Lorenzo)

The higher-order smooth ASG bases that are used in IGA have been shown to produce superior results when compared to standard C^0 discretizations. Another important property of IGA that has been recently proven is that Galerkin solutions can be obtained with only one evaluation per degree of freedom using the concept of variational collocation [3]. Methodologies such as Bézier extraction [4, 5] simplify the implementation of isogeometric basis function technology into existing finite element frameworks and provide a rigorous foundation for the localized projection procedures presented in this paper. The advantages of IGA have been demonstrated in a wide variety of fields of science and engineering, including structural analysis [6, 7, 8, 9], fluids [10, 11, 12], fluid-structure interaction [13, 14, 15, 16], fracture and damage [17, 18, 19], phase-field analysis [19, 20, 21], shape optimization [22], electromagnetics [23], and biomedicine [24, 25, 26, 27, 28, 29, 30].

Mesh adaptivity is a rich subject in IGA. The possibilities include element subdivision or merging (*h*-adaptivity), degree elevation or reduction (*p*-adaptivity), basis roughening or smoothing (*k*-adaptivity), and reparameterization (*r*-adaptivity). These operations can be combined to produce sophisticated *hpkr*-adaptive schemes. These methodologies have been recently cast into a unified framework based on Bézier projection [31].

Traditional approaches to isogeometric mesh adaptivity based on non-uniform rational B-splines (NURBS) suffer from global propagation of refinement [32, 33, 34]. Coarsening is much more difficult to apply accurately and efficiently and it is rarely considered. Several ASG descriptions enable local adaptivity. Hierarchical B-splines [35, 36, 37, 38] are constructed in a multi-level fashion where higher levels are composed of finer B-spline bases. A selection mechanism is used to construct a global basis from functions in the hierarchy. Due to interactions between coarse and fine basis functions the conditioning of the basis can deteriorate. Truncated hierarchical B-splines (THB-splines) address this issue to a certain degree [39, 40, 41, 42]. Spline forests [43] are an extension of hierarchical B-splines and NURBS to complex surfaces and solids of arbitrary topological genus. T-splines [44] are a generalization of NURBS that are locally refinable and capable of modeling complicated designs as a single, watertight geometry. Analysis-suitable T-splines [45, 46, 47, 48, 49, 50] are a subset of T-splines with the same mathematical properties as NURBS. Recently, hierarchical analysis-suitable T-splines [51] have been shown to combine the local refinement properties of T-splines, which can be used to precisely design the geometry, with the power of the hierarchical concept, which accommodates efficient local adaptivity for analysis. Polynomial splines over hierarchical T-meshes (PHT-splines) [52, 53, 54, 55], modified T-splines [56], locally refined splines (LR-splines) [57, 58], and generalized T-splines [59] are other locally refinable ASG representations.

In the context of phase-field models, local adaptivity is essential to accurately and efficiently resolve evolving interfaces. In particular, hierarchical B-splines are an attractive adaptive technology for several reasons:

- Simple and efficient.

- Preserves many of the mathematical properties of B-splines.

- Generates smooth, localized, and geometrically exact discretizations.

- Enables design on the first level while higher levels can be leveraged for analysis.

45   • Compatible with quadrilateral and hexahedral meshing technology.

We present a set of algorithms for coupled local hierarchical h-refinement and h-coarsening and apply it in the context of adaptive IGA. Our algorithms are capable of handling evolving interfaces efficiently. The framework is based on an extension of Bézier extraction and projection to the hierarchical setting. The coarsening projection is local and quadrature-free. We introduce a *balance* parameter to control the overlap of hierarchical basis functions resulting in superior conditioning of the resulting linear systems. Coupling the localized projection and the balance parameter enables fine-grain control over the behavior of the adaptive algorithms.

In order to illustrate our methods we address the simulation of localized prostate cancer growth using a phase-field model that we previously presented in [30]. This problem has an evolving interface which separates the tumor from the surrounding healthy tissue. To resolve this interface we use our local adaptivity approach. First, we describe and analyze the behavior of our adaptive algorithms in simple 2D and 3D settings. Then, we apply them to a tissue-scale, patient-specific simulation, where we solve our equations of tumor growth over the patient's prostate anatomy extracted from computerized tomography images, as in [30]. Finally, we demonstrate that our algorithms for local adaptivity are a cornerstone in the successful implementation of our model for localized prostate cancer growth as a predictive tool for the clinical management of the disease.

1.1. Modeling and simulation of prostate cancer

Approximately 95% of prostate cancer cases are adenocarcinomas [60]. This is a form of cancer that originates in an epithelial tissue, like the one present in the majority of the prostate. The development of a prostate adenocarcinoma requires a gradual accumulation of alterations to the genes within the cells' nucleus, which varies from patient to patient. The accumulation of successive genetic changes through years leads to the phenomenon of tumor progression, whereby an initial mild disorder of cell behavior evolves gradually into an advanced cancer. In this process, prostate cancer begins as a localized tumor within the prostate, which is the stage when this disease is normally detected, diagnosed and treated. Then, the tumor escapes from this organ invading the surrounding tissues and, finally, some malignant cells leave the original tumor, get into the bloodstream and migrate to distant tissues, which they colonize and invade (metastasis) [61].

3

Predictive medicine [62, 63, 64, 30] is a new interdisciplinary trend of medical practice that uses mathematical models and computer simulations in order to predict outcomes of diseases and design optimal treatments on a patient-specific basis. This approach complements the statistical and experiential methods that dominate diagnosis and treatment planning in current medical practice. Regarding cancer research, mathematical oncology [65, 66, 67] is the field of predictive medicine aimed at developing models to simulate cancer progression and treatment.

In mathematical oncology, localized prostate cancer growth can be understood as a moving interface problem that aims at defining the regions occupied by the evolving tumor within this organ. As for other moving interface problems, there have been several fruitful attempts to model the phenomena involved in cancer growth with the phase field method [68, 69, 70, 71, 72, 73, 74, 75]. Though this technique has its origin in materials science [76], it has been extended to a broad variety of areas of science and engineering [14, 20, 77, 78, 79] for one reason: the possibility to capture the interface implicitly without solving a moving boundary problem.

Mathematical models representing tumor growth include one or several cancerous cell species and the host tissue as different interacting phases. Tumoral and healthy cells compete in order to obtain nutrients, proliferate and control cell density so as to thrive in the harsh tumoral environment − where nutrients are scarce and cells are tightly packed. The dynamics of each cellular species is modeled by means of a combination of the following processes: diffusion, convection, reaction and gradient-oriented migration, also known as taxis. Reactive terms normally include cell proliferation, metabolism, death or interactions with other cell species and substances included in the model. Additionally, there are different types of gradients that promote cancerous cell migration: in the concentration of a substance dissolved in the interstitial fluid (chemotaxis), in the concentration of a compound attached to the extracellular matrix (haptotaxis), and in the mechanical properties of the extracellular matrix or in the forces exerted on the cells (mechanotaxis). The dynamics of the nutrients, as well as any other substance with a major role in cancer progression, is included by means of further diffusion-reaction equations. Moreover, it is possible to include the effect of some treatments by means of additional reactive terms in the equations of cell dynamics (e.g., radiation therapy or chemotherapy), although sometimes additional diffusion-reaction equations are required (e.g., chemotherapy).

1.2. Structure and contents of the paper

Section 2 introduces the fundamental concepts of univariate and multivariate Bernstein, B-spline and NURBS spaces in order to perform IGA. We proceed to present hierarchical splines in Section 3. Then, we describe the concept of the Bézier mesh and the extraction operator for hierarchical splines in Section 4, where we also introduce the spline reconstruction operator for hierarchical splines. We present our algorithms for h-refinement and h-coarsening of hierarchical splines in Section 5 and 6, respectively. Section 7 discusses

4

the operation of balancing and introduces the new function support balance parameter, which we also compare to the previous ring balance parameter described in [43]. In Section 8, we use our methodology to accommodate local h-adaptivity in IGA to simulate localized prostate cancer tumor growth. First, we describe our model, the details for numerical analysis, and how we implement automatic adaptivity. We proceed to present the results corresponding to the 2D and 3D academic scenarios to discuss the performance of our algorithms. Then, we describe and discuss the results of a tissue-scale, patient-specific simulation, which demonstrates the potential of our technology to efficiently address compelling problems in complex geometries. Finally, we draw conclusions in Section 9.

2. Notation and preliminaries

2.1. Univariate Bernstein basis

The univariate Bernstein basis functions are defined as

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

where $\xi \in [-1, 1]$, p is the polynomial degree, and the binomial coefficient $\binom{p}{i-1} = \frac{p!}{(i-1)!(p+1-i)!}$, $1 \le i \le p+1$. We choose to define the Bernstein basis over the biunit interval to facilitate Gaussian quadrature in finite element analysis rather than use the convention of the computer-aided geometric design (CAGD) community where the Bernstein polynomials are defined over the unit interval $[0, 1]$. The Bernstein basis functions for polynomial degrees $p = 1, 2, 3$ are shown in Fig. 1. It is often useful to define a vector of Bernstein basis
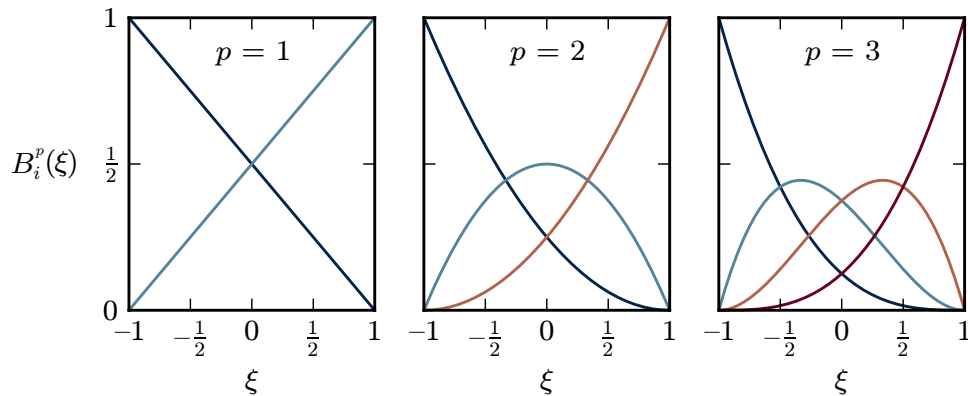

Figure 1: The Bernstein basis for polynomial degrees $p = 1, 2, 3$.

functions $\mathbf{B}^p(\xi) = \{B_i^p(\xi)\}_{i=1}^{p+1}$. The degree superscript is suppressed when unnecessary. We denote the space of functions over the biunit interval spanned by the Bernstein basis of degree p by $\mathcal{B}^p$. A useful review of Bernstein polynomials and their properties is provided in [80].

2.2. Multivariate Bernstein basis

We define a multivariate Bernstein basis over the box of dimension d_p, $[-1,1]^{d_p}$ by the tensor product. The polynomial degree may be different in each direction and so we define the vector of degrees $\mathbf{p} = \{p_\ell\}_{\ell=1}^{d_p}$. The vector of multivariate Bernstein basis functions is defined by the Kronecker product

$$\mathbf{B^p} = \mathbf{B}^{p_{d_p}}(\xi_{d_p}) \otimes \cdots \otimes \mathbf{B}^{p_1}(\xi_1). \tag{2}$$

Thus, there are $\prod_{\ell=1}^{d_p}(p_\ell + 1)$ basis functions in the vector. All of the properties of the univariate Bernstein basis are inherited by the multivariate Bernstein basis.

2.3. Relations between Bernstein polynomials over different intervals

The Bernstein polynomials over the interval $[a,b]$ are

$$B_i^p(t) = \binom{p}{i-1} \frac{(b-t)^{p-(i-1)}(t-a)^{i-1}}{(b-a)^p}. \tag{3}$$

Given another interval $[\tilde{a}, \tilde{b}]$, the Bernstein polynomials are

$$\tilde{B}_i^p(t) = \binom{p}{i-1} \frac{(\tilde{b}-t)^{p-(i-1)}(t-\tilde{a})^{i-1}}{(\tilde{b}-\tilde{a})^p}. \tag{4}$$

A polynomial function f of degree p can be represented by a linear combination of the Bernstein polynomials over $[a,b]$ or by a combination of the Bernstein polynomials over $[\tilde{a}, \tilde{b}]$

$$f(t) = \sum_{i=1}^{p+1} c_i B_i^p(t) = \sum_{i=1}^{p+1} \tilde{c}_i \tilde{B}_i^p(t). \tag{5}$$

As shown by [81], the coefficient vectors $\mathbf{c} = \{c_i\}_{i=1}^{p+1}$ and $\tilde{\mathbf{c}} = \{\tilde{c}_i\}_{i=1}^{p+1}$ can be related by the transformation matrix $\mathbf{A}$

$$\tilde{\mathbf{c}} = \mathbf{A}\mathbf{c} \tag{6}$$

where the entries of $\mathbf{A}$ are given by

$$A_{jk} = \sum_{i=\max(1,j+k-p+1)}^{\min(j,k)} B_i^{j-1}(\tilde{b}) B_{k-i}^{p-j-1}(\tilde{a}) \quad \text{for } j,k = 1,2,\ldots,p+1. \tag{7}$$

This can be extended to multiple dimensions by a tensor product. The elements of the inverse of $\mathbf{A}$ are given by

$$[\mathbf{A}^{-1}]_{jk} = \sum_{i=\max(1,j+k-p+1)}^{\min(j,k)} \tilde{B}_i^{j-1}(b) \tilde{B}_{k-i}^{p-j-1}(a) \quad \text{for } j,k = 1,2,\ldots,p+1. \tag{8}$$

The inverse matrix provides a relationship between the basis functions over each interval

$$\tilde{\mathbf{B}}^p = \mathbf{A}^{-\mathrm{T}}\mathbf{B}^p. \tag{9}$$

Both Eqs. (7) and (8) are defined using one-based indexing for both the matrix entries and the Bernstein basis as opposed to the zero-based indexing used in [81].

6

2.4. The Gramian of the Bernstein basis and its inverse

When computing the projection of an arbitrary function onto the Bernstein polynomials, an expression for the Gramian matrix $\mathbf{G}^p$ for the basis of degree p is required. The entries in the matrix are

$$G_{jk}^p = \int_{-1}^{1} B_j^p(\xi) B_k^p(\xi) d\xi \quad \text{for } j, k = 1, 2, \ldots, p+1. \tag{10}$$

Expressions for products and integrals of the Bernstein polynomials in [80, 82] permit Eq. (10) to be written in closed form as

$$G_{jk}^p = \frac{2}{2p+1} \binom{2p}{j+k-2}^{-1} \binom{p}{j-1} \binom{p}{k-1}. \tag{11}$$

The Gramian matrix for a multivariate Bernstein basis of dimension d_p and with the vector of polynomial degrees $\mathbf{p} = \{p_1, \ldots, p_{d_p}\}$ is obtained from a Kronecker product

$$\mathbf{G}^{\mathbf{P}} = \mathbf{G}^{p_{d_p}} \otimes \cdots \otimes \mathbf{G}^{p_1}. \tag{12}$$

An expression for the inverse of the Gramian of the Bernstein basis can be obtained by considering the Bernstein-Bézier representation of the dual basis in [83]. The Bézier coefficients of the dual basis are precisely the entries in the inverse of the Gramian and so the expression for the dual basis can be used to obtain

$$[(\mathbf{G}^p)^{-1}]_{jk} = \frac{(-1)^{j+k}}{2} \left[\binom{p}{j-1} \binom{p}{k-1} \right]^{-1} \sum_{i=1}^{\min(j,k)} (2i-1) \binom{p-i+1}{p-j+1} \binom{p-i+1}{p-k+1} \binom{p+i}{p-j+1} \binom{p+i}{p-k+1} \tag{13}$$

after modification to use one-based indexing and the Bernstein basis over the biunit interval. The inverse of a Kronecker product of matrices is given by the Kronecker product of the inverses and so Eq. (13) can be used to compute the inverse of a multivariate Gramian matrix

$$(\mathbf{G}^{\mathbf{P}})^{-1} = (\mathbf{G}^{p_{d_p}})^{-1} \otimes \cdots \otimes (\mathbf{G}^{p_1})^{-1}. \tag{14}$$

2.5. Splines

A univariate spline is defined by the polynomial degree of the spline p and the knot vector G, a set of non-decreasing parametric coordinates $\mathsf{G} = \{s_A\}_{A=1}^{n+p+1}$, $s_A \leq s_{A+1}$ where n is the number of spline basis functions [32, 33, 34]. The multiplicity of a knot value $s_A \in \mathsf{G}$ is $0 \leq \mu(\mathsf{G}, s_A) \leq p+1$ and it will be zero for any $s_A \notin \mathsf{G}$. We require that the knot vector be open, that is, $\mu(\mathsf{G}, s_1) = \mu(\mathsf{G}, s_{n+p+1}) = p+1$. We call $\hat{\Omega} = [s_1, s_{n+p+1}]$ the parameter space, which is partitioned into knot spans $[s_A, s_{A+1}]$ by the knots in G. The Ath spline basis function over the knot vector can be defined using the Cox-de Boor recursion formula [84, 85]:

$$N_A^0(s) = \begin{cases} 1 & s_A \leq s < s_{A+1} \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

$$N_A^p(s) = \frac{s - s_A}{s_{A+p} - s_A} N_A^{p-1}(s) + \frac{s_{A+p+1} - s}{s_{A+p+1} - s_{A+1}} N_{A+1}^{p-1}(s). \tag{16}$$

The spline space spanned by the spline basis is denoted by $\mathcal{N}^p$. We denote the set of functions in the basis by N. A spline curve of dimension d_s is a function mapping $\hat{\Omega} \subset \mathbb{R}$ to $\mathbb{R}^{d_s}$. The curve $\mathbf{x}(s)$ is defined by a set of d_s dimensional control points $\mathbf{P}_A$ as

$$\mathbf{x}(s) = \sum_{A=1}^{n} \mathbf{P}_A N_A^p(s). \tag{17}$$

An alternate form of Eq. (17) can be obtained by defining the vector of control points $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^{n}$ and the vector of basis functions $\mathbf{N}^p(s) = \{N_A^p(s)\}_{A=1}^{n}$ so that

$$\mathbf{x}(s) = \mathbf{P}^{\mathrm{T}} \mathbf{N}^p(s). \tag{18}$$

160   The vector of control points $\mathbf{P}$ can be interpreted as a matrix of dimension $n \times d_s$. Due to the variation diminishing property of the spline basis, the curve will generally only interpolate the control points at the ends of the curve or at locations where the spline basis is C^0. A multivariate spline basis is defined from a tensor product of univariate spline bases [32, 33, 34]. The properties of multivariate spline basis functions follow from the corresponding properties of their univariate counterparts. We can build spline surfaces and 165   solids using a bivariate or trivariate spline basis in Eq. (17), respectively.

2.6. Rational splines

The spline basis defined in the previous section provides a flexible means to represent curves. However, this polynomial basis cannot represent certain curves of interest such as circular arcs. A rational spline basis can be used to remedy this deficiency [32, 33, 34]. The rational basis is defined by associating a weight 170 $w_A \in \mathbb{R}^+$ with each basis function N_A^p and introducing the weight function

$$w(s) = \sum_{A=1}^{n} w_A N_A^p(s) \tag{19}$$

The rational basis functions are then defined as

$$R_A^p(s) = \frac{w_A N_A^p(s)}{w(s)} \tag{20}$$

and a rational curve is defined as

$$\mathbf{x}(s) = \sum_{A=1}^{n} \mathbf{P}_A R_A^p(s) \tag{21}$$

or, alternatively, as

$$\mathbf{x}(s) = \mathbf{P}^{\mathrm{T}} \mathbf{R}^p(s). \tag{22}$$

where $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^{n}$ and $\mathbf{R}^p(s) = \{R_A^p(s)\}_{A=1}^{n}$. We can define multivariate rational spline bases and describe multiparametric geometries with them following the same approach as in Section 2.5.

3. Hierarchical splines

A hierarchical spline space is constructed from a finite sequence of n_h nested spline spaces, $\mathcal{N}^\alpha \subset \mathcal{N}^{\alpha+1}$, $\alpha = 1, \ldots, n_h - 1$, and n_h open bounded subdomains of the parametric space $\hat{\Omega}$ such that $\hat{\Omega}^{n_h} \subseteq \hat{\Omega}^{n_h-1} \subseteq \ldots \subseteq \hat{\Omega}^1$ [35]. We denote by N^α the spline basis at each hierarchical level. To ensure nestedness of the spline spaces $\mathcal{N}^\alpha \subset \mathcal{N}^{\alpha+1}$ we impose the following restrictions:

- $p_\ell^{\alpha+1} \geq p_\ell^\alpha$, $\ell = 1, \ldots, d_p$, $\alpha = 1, \ldots, n_h - 1$

- $\mathsf{G}_\ell^\alpha \subset \mathsf{G}_\ell^{\alpha+1}$, $\ell = 1, \ldots, d_p$, $\alpha = 1, \ldots, n_h - 1$

- $\mu(\mathsf{G}_\ell^{\alpha+1}, s_\ell) - \mu(\mathsf{G}_\ell^\alpha, s_\ell) \geq p_\ell^{\alpha+1} - p_\ell^\alpha$, $\forall \mathbf{s} \in \hat{\Omega}$, $\ell = 1, \ldots, d_p$, $\alpha = 1, \ldots, n_h - 1$

We will set $\hat{\Omega}^1 = \hat{\Omega}$ and require that the boundaries $\partial\hat{\Omega}^\alpha$, $\alpha = 1, \ldots, n_h$, are aligned with knot lines in $\mathcal{N}^{\alpha-1}$. This is called a *strong boundary condition* [35].

3.1. Hierarchical spline spaces

The hierarchical spline basis H is constructed recursively:

1. Initialize $\mathsf{H}^1 = \mathsf{N}^1$.

2. Recursively construct $\mathsf{H}^{\alpha+1}$ from H^α by setting

$$\mathsf{H}^{\alpha+1} = \mathsf{H}_{coarse}^{\alpha+1} \cup \mathsf{H}_{fine}^{\alpha+1}, \quad \alpha = 1, \ldots, n_h - 1$$

where

$$\mathsf{H}_{coarse}^{\alpha+1} = \{N \in \mathsf{H}^\alpha : \mathrm{supp}\,(N) \not\subseteq \hat{\Omega}^{\alpha+1}\}$$

and

$$\mathsf{H}_{fine}^{\alpha+1} = \{N \in \mathsf{N}^{\alpha+1} : \mathrm{supp}\,(N) \subseteq \hat{\Omega}^{\alpha+1}\}$$

3. Set $\mathsf{H} = \mathsf{H}^{n_h}$.

We denote the number of hierarchical basis functions in H by n_f, which span a hierarchical spline space $\mathcal{H}$. During each step of the previous algorithm, we first build $\mathsf{H}_{coarse}^{\alpha+1}$ by dropping from H^α those functions whose support is contained in $\hat{\Omega}^{\alpha+1}$. Then, we add to the hierarchical basis through $\mathsf{H}_{fine}^{\alpha+1}$ all the functions from $\mathsf{N}^{\alpha+1}$ that are fully supported by $\hat{\Omega}^{\alpha+1}$. The rationale for excluding functions from H^α when building $\mathsf{H}_{coarse}^{\alpha+1}$ is that those functions can be represented by a linear combination of the functions in $\mathsf{H}_{fine}^{\alpha+1}$. Consequently, the functions in the resulting hierarchical basis H are linearly independent. Fig. 2 shows an example of a univariate quadratic hierarchical spline basis. Notice also that given $\mathsf{H}^1, \ldots, \mathsf{H}^N$, a sequence of hierarchical spline bases, span $\mathsf{H}^\alpha \subseteq$ span $\mathsf{H}^{\alpha+1}$. For a thorough exposition of the these and other properties of hierarchical spline bases, the interested reader is referred to [35].

By construction, $\mathcal{N}^1 \subseteq \mathcal{H}$, thus the approximation properties of B-splines and NURBS are inherited by their hierarchical counterparts. In particular, constants are exactly represented and all patch tests are satisfied [43, 86].
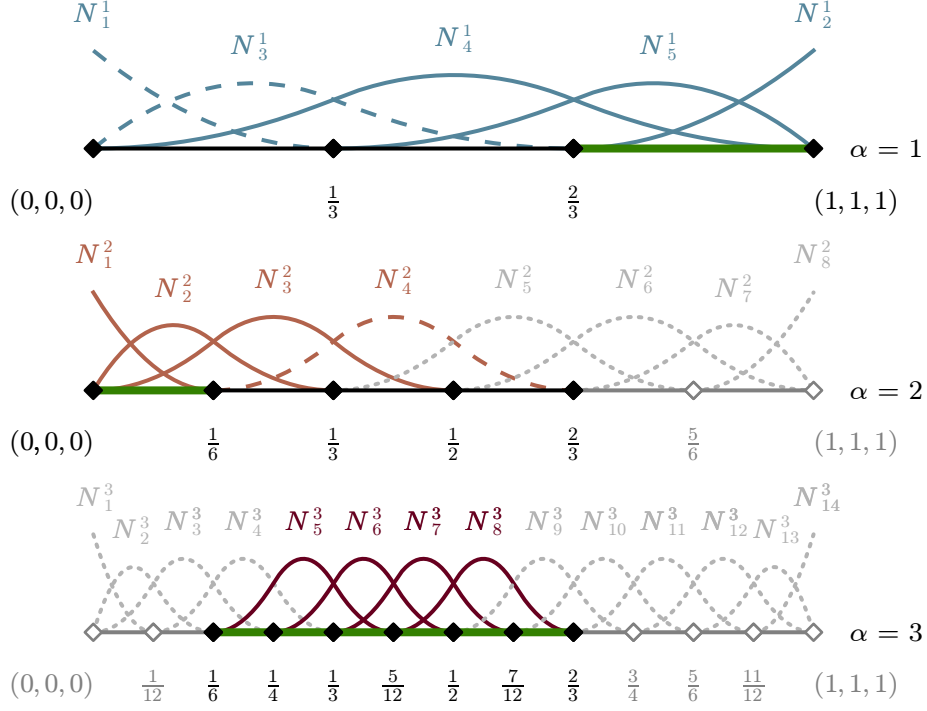

Figure 2: Basis functions for a univariate quadratic hierarchical spline space. Functions in H are represented with solid lines and colored with a different hue for each level. Functions in N^α dependent on higher-level functions are depicted with dashed lines in the corresponding color for each level. Functions that are not fully supported by the hierarchy of subdomains $\hat{\Omega}^\alpha$ are represented in gray dotted lines. In each level, the Bézier elements in E^α (see Section 4) that are contained in the subdomain $\hat{\Omega}^\alpha$ are shown in black and the remainder in dark gray. Bézier elements in HE (see Section 4) are highlighted in green across the hierarchy. Note that the functions in H are supported entirely by Bézier elements.

3.2. The geometric map for hierarchical splines

The geometric map $\mathbf{x} : \hat{\Omega} \subset \mathbb{R}^{d_p} \mapsto \Omega \subset \mathbb{R}^{d_s}$ for hierarchical splines can be constructed in various ways. Let n_g denote the number of geometric blending functions. In this paper we will associate a d_s dimensional control point $\mathbf{P}_A$ to each function $N_A^1 \in \mathsf{N}^1$ and write the geometric map as

$$\mathbf{x}(\mathbf{s}) = \sum_{A=1}^{n_g} \mathbf{P}_A N_A^1(\mathbf{s}). \tag{23}$$

or, alternatively,

$$\mathbf{x}(\mathbf{s}) = \mathbf{P}^{\mathrm{T}} \mathbf{N^1}(\mathbf{s}) \tag{24}$$

where $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^{n_g}$ and $\mathbf{N^1}(\mathbf{s}) = \left\{ N_A^1(\mathbf{s}) \right\}_{A=1}^{n_g}$. We can also use the functions in N^1 to represent the weight function $w(\mathbf{s})$ if we are working with a rational basis [2, 43].

10

4. Bézier extraction and spline reconstruction

It is not immediately obvious how to integrate hierarchical splines into existing finite element tools. A primary challenge is converting the hierarchy into a finite element mesh. A simple and elegant solution to this problem is based on Bézier extraction. This technique was initially introduced in [4] for NURBS and in [5] for T-splines. Then it was extended for hierarchical splines [43] and hierarchical analysis-suitable T-splines [51]. Bézier extraction collapses the spline hierarchy onto a single level finite element mesh. Both the hierarchical basis H and geometry are described with the same set of Bernstein basis functions.

4.1. Bézier elements

We denote a Bézier element by e and the set of all Bézier elements on each hierarchical level by E^α. The set of Bézier elements HE corresponding to a spline hierarchy can be determined recursively by selecting specific Bézier elements from each E^α, $\alpha = 1, \ldots, n_h$. We will use the index e to enumerate the elements in HE and we denote by n_e the number of elements in HE. The parametric domain of an element e is denoted by $\hat{\Omega}^e$ and we use Ω^e to denote the physical domain corresponding to $\hat{\Omega}^e$ under the geometric map (23). The set of all Bézier elements is also called the Bézier mesh. We build HE as follows:

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

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

where

$$\mathsf{HE}^{\alpha+1}_{coarse} = \{\mathsf{e} \in \mathsf{HE}^\alpha : \hat{\Omega}^e \not\subseteq \hat{\Omega}^{\alpha+1}\}$$

and

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

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

The previous algorithm bears much in common with the selection mechanism to build the hierarchical basis. We exclude from $\mathsf{HE}^\alpha_{coarse}$ the elements $\mathsf{e} \in \mathsf{HE}^\alpha$ whose parametric domain is also contained in $\hat{\Omega}^{\alpha+1}$ because there exists a union of finer elements in that subdomain that represents $\hat{\Omega}^e$. Indeed, $\mathsf{HE}^{\alpha+1}_{fine}$ brings in those finer elements. Notice that only the elements contained in the subdomains $\hat{\Omega}^\alpha$, $\alpha = 1, \ldots, N$, are eligible to be part of the Bézier mesh. In Fig. 2, the elements in HE are colored in green across the hierarchy. Fig. 3 shows the Bézier mesh HE for a two-dimensional hierarchical spline space together with the Bézier meshes E^α corresponding to each hierarchical level.

11
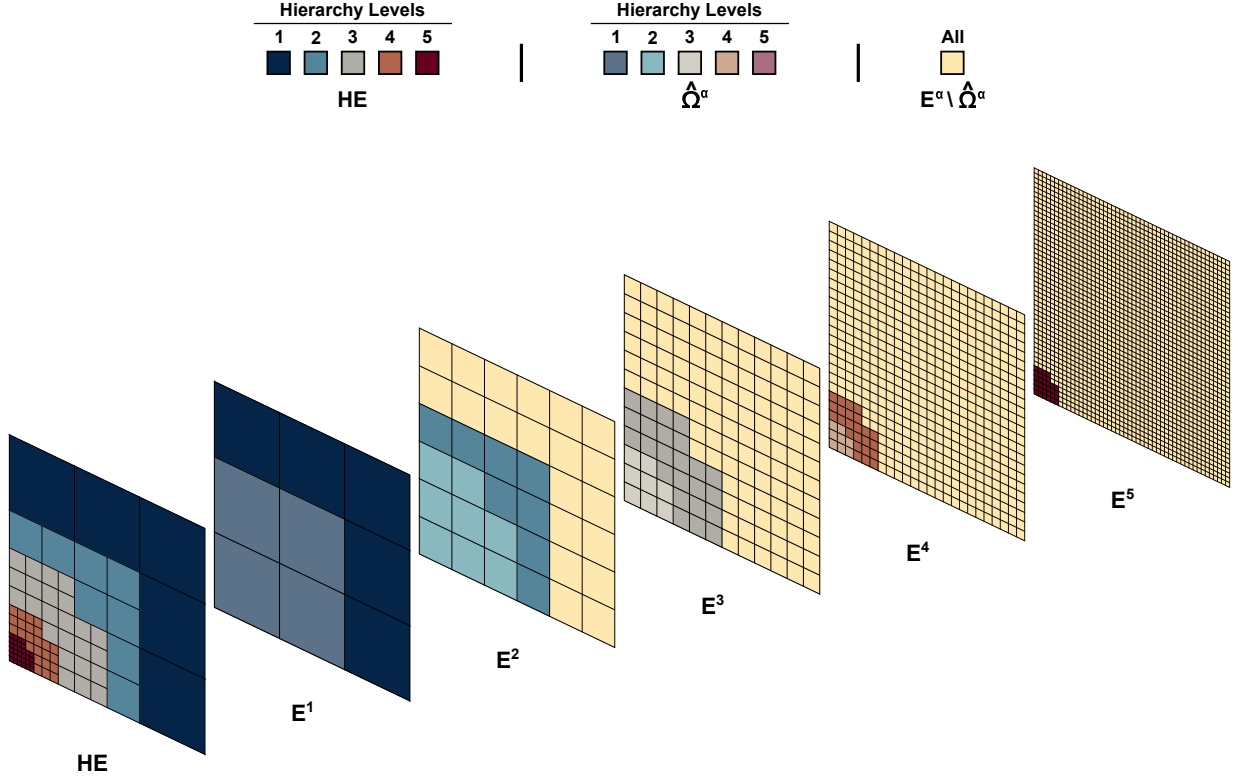
Figure 3: The Bézier mesh for a two-dimensional hierarchical spline space and the distribution of the elements in HE across the hierarchical levels. E^α denotes the Bézier mesh at level α. The elements in HE at each level α are depicted with a darker shade of the same color used to represent the Bézier elements from E^α included in Ω^α. Notice that $\mathsf{HE} \cap \mathsf{E}^\alpha \subseteq \Omega^\alpha$.

4.2. Hierarchical extraction operator

Over each Bézier element it is possible to determine the Bernstein-Bézier representation of the hierarchical basis functions. This relationship can be written as

$$\mathbf{H}^e = \mathbf{C}^e \mathbf{B} \tag{25}$$

where $\mathbf{H}^e$ is the vector containing the hierarchical basis functions with support over the element, $\mathbf{C}^e$ is the element extraction operator whose entries are computed using standard knot insertion techniques [5], and $\mathbf{B}$ is the vector of Bernstein basis functions.

4.3. Hierarchical reconstruction operator

The element extraction operator $\mathbf{C}^e$ for single-level spline representations like B-splines, NURBS and analysis-suitable T-splines is invertible. The inverse of $\mathbf{C}^e$ is called the *reconstruction operator* [31] and is denoted by $\mathbf{R}^e$. For hierarchical splines there are elements where $\mathbf{C}^e$ is not invertible since the hierarchical

12

basis, restricted to the element domain, is not *linearly independent.* To overcome this limitation we define the element reconstruction operator as

$$\mathbf{R}^e = (\mathbf{L}^e \mathbf{C}^e)^{-1} \mathbf{L}^e \tag{26}$$

where $\mathbf{L}^e$ is called a *transmission refinement operator* and it is computed following Algorithm 4.1. Fig. 4 illustrates the action of $\mathbf{L}^e$.

Algorithm 4.1. *Computation of the transmission refinement operator* $\mathbf{L}^e$ *and the reconstruction operator* $\mathbf{R}^e$ *for an element* e.

1. *Compute* $\mathbf{H}^e$ *and* $\mathbf{C}^e$ *over element* e.

2. *Search through the functions in* $\mathbf{H}^e$ *finding the function from the coarsest level* α_c.

3. *Find* e_c *in* E^{α_c} *such that* $\hat{\Omega}^{e_c} \supset \hat{\Omega}^e$.

4. *Construct a vector* $\mathbf{N}^{e_c}$ *of all the functions in* N^{α_c} *that have support over the element* e_c. *These functions form a complete, locally linearly independent basis at level* α_c *over* e_c.

5. *Build the transmission refinement operator* $\mathbf{L}^e$ *from element* e *to element* e_c *mapping from the functions in* $\mathbf{H}^e$ *to* $\mathbf{N}^{e_c}$, *that is*

$$\mathbf{N}^{e_c} = \mathbf{L}^e \mathbf{H}^e. \tag{27}$$

This can be done using standard knot insertion techniques [1, 2, 32].

6. *Build the reconstruction operator* $\mathbf{R}^e$ *using* $\mathbf{L}^e$ *and* $\mathbf{C}^e$ *in Eq. (26).*

5. Hierarchical refinement

Local refinement for hierarchical splines is straightforward due to the multilevel construction of the basis.

Algorithm 5.1. *Refinement of a hierarchical B-spline.*

1. *Create the refined hierarchical destination mesh from a hierarchical source mesh by defining the refined subdomains* $\hat{\Omega}_r^\alpha$ *such that* $\hat{\Omega}^\alpha \subseteq \hat{\Omega}_r^\alpha$, $\alpha = 1, \dots, n_h$.

2. *Build the refinement operator* $\mathbf{M}^{a,b}$ *using knot insertion [1, 2, 32] such that*

$$\mathbf{N}^b = \mathbf{M}^{a,b} \mathbf{N}^a \tag{28}$$

3. *Construct the destination spline coefficients* $\mathbf{P}^b$ *from the source spline coefficients* $\mathbf{P}^a$ *as*

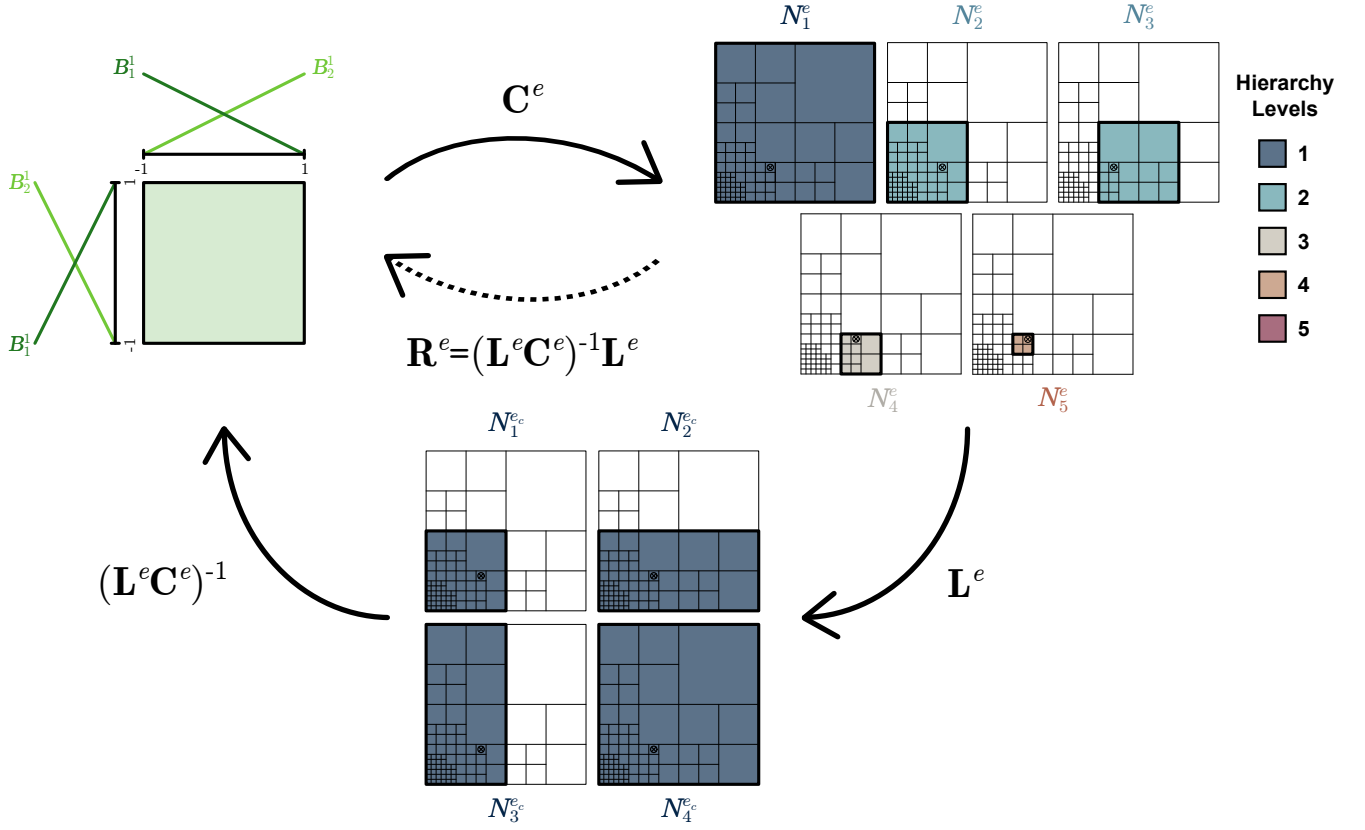$$\mathbf{P}^b = (\mathbf{M}^{a,b})^{\mathrm{T}} \mathbf{P}^a \tag{29}$$

Figure 4: Bézier extraction operator $\mathbf{C}^e$ and reconstruction operator $\mathbf{R}^e$ for a bilinear hierarchical spline. The element marked with a circled cross supports 5 hierarchical basis functions, i.e., $\mathbf{H}^e = \{N_1^e, N_2^e, N_3^e, N_4^e, N_5^e\}$, which we represent throughout their support. However, the bilinear Bernstein basis contains only 4 functions, so $\mathbf{C}^e$ is not invertible. $\mathbf{L}^e$ maps the functions in $\mathbf{H}^e$ to a local independent basis on level 1 over the element, $\mathbf{N}^{e_c} = \{N_1^{e_c}, N_2^{e_c}, N_3^{e_c}, N_4^{e_c}\}$. The functions in $\mathbf{N}^{e_c}$ are also represented throughout their support. Now, we can map the functions in $\mathbf{N}^{e_c}$ to the Bernstein basis throughout the spline reconstruction operator $(\mathbf{L}^e\mathbf{C}^e)^{-1}$. Then, the hierarchical reconstruction operator over the element is given by $\mathbf{R}^e = (\mathbf{L}^e\mathbf{C}^e)^{-1}\mathbf{L}^e$.

6. Hierarchical coarsening

Bézier projection will be used to perform local coarsening. For a complete exposition on the subject the interested reader is referred to [31], which also addresses Bézier projection onto and between analysis-suitable T-spline spaces, shows the optimal convergence rates of the methodology, and demonstrates its applicability to *hpkr*-adaptivity in IGA. Of particular importance is the possibility to use Bézier projection to develop quadrature-free coarsening algorithms.

Using Eq. (26) as the reconstruction operator we can employ Bézier projection to coarsen hierarchical splines spaces and to project fields defined over a source mesh onto a coarsened destination mesh.

Algorithm 6.1. *Projection of the control values from m source elements $\{\mathbf{e}_i\}$ onto the control values for a*

14

single coarse element ē in the destination mesh.

1. *Use the element extraction operators for the elements* $\{e_i\}$ *to convert the control values on each element to Bézier form*

$$\mathbf{Q}^{e_i} = (\mathbf{C}^{e_i})^{\mathrm{T}} \mathbf{P}^{e_i}.$$ (30)

2. *The vector of Bézier control values on* $\bar{e}$ *is given by*

$$\mathbf{Q}^{\bar{e}} = \sum_{i=1}^{m} \phi_i \mathbf{G}^{-1} \mathbf{A}_i^{-\mathrm{T}} \mathbf{G} \mathbf{Q}^{e_i}$$ (31)

where $\phi_i = vol\,(\hat{\Omega}^{e_i})/vol\,(\hat{\Omega}^{\bar{e}})$.

3. *Compute the element reconstruction operator for* $\bar{e}$ *using Algorithm 4.1 to convert the Bézier control values to spline control values*

$$\mathbf{P}^{\bar{e}} = (\mathbf{R}^{\bar{e}})^{\mathrm{T}} \mathbf{Q}^{\bar{e}}.$$ (32)

Algorithm 6.2. *Coarsening of a hierarchical B-spline.*

1. *Create the coarsened hierarchical destination mesh from a hierarchical source mesh by defining the coarsened subdomains* $\hat{\Omega}_c^{\alpha}$ *such that* $\hat{\Omega}^{\alpha} \supseteq \hat{\Omega}_c^{\alpha}$, $\alpha = 1, \ldots, n_h$.

2. *Perform the Bézier projection in Algorithm 6.1 to obtain the set of local control values for each element in the destination mesh.*

3. *Use the new local control values and the weighting scheme provided by*

$$w_A^e = \frac{\int_{\Omega^e} N_A d\Omega}{\sum_{e' \in \mathsf{E}_A} \int_{\Omega^{e'}} N_A d\Omega},$$ (33)

where $\mathsf{E}_A = \mathsf{HE} \cap \mathrm{supp}(N_A)$, *to compute the new global control values for the Ath function on the coarsened mesh*

$$\mathbf{P}_A = \sum_{e \in \mathsf{E}_A} \omega_A^e \mathbf{P}_A^e.$$ (34)

7. Function support balancing

Balancing controls the overlap of functions across the hierarchy. In other words, it prevents the elements in level α from supporting hierarchical basis functions from level β where $\alpha \gg \beta$. In the context of IGA, such a situation would result in poorly conditioned system matrices. Balancing is performed every time the hierarchical mesh HE changes. The balancing behavior is controlled by a parameter called the *balance parameter*.

To simplify the exposition, we introduce the concept of *parent element*, *grandparent element*, and *children element*. Given an element $e \in \mathsf{E}^{\alpha}$, we say that the parent element of e is another element $p \in \mathsf{E}^{\alpha-1}$ such that $\hat{\Omega}^p \supset \hat{\Omega}^e$. Likewise, the k grandparent of element e is an element $g \in \mathsf{E}^{\alpha-k-1}$ such that $\hat{\Omega}^g \supset \hat{\Omega}^e$. The children of element e are the set of elements $\{e_i\} \in \mathsf{E}^{\alpha+1}$ such that $\hat{\Omega}^{e_i} \subset \hat{\Omega}^e$.

Algorithm 7.1 outlines ring balancing, a simple strategy which was introduced in [43].

15

Algorithm 7.1. *Balancing of a hierarchical spline for a given value of the ring balance parameter, $n_{rb} \in \mathbb{N}$.*

1. *Partition* HE *by sorting the elements according to their hierarchical level, hence building the subsets*

   $$\mathsf{pHE}^\alpha = \mathsf{E}^\alpha \cap \mathsf{HE}$$

2. *For $\alpha = n_h, n_h - 1, \ldots, 1$, do the following for each element $\mathsf{e} \in \mathsf{pHE}^\alpha$:*

   (a) *Construct a set $\mathsf{P}_1 \subset \mathsf{E}^{\alpha-1}$ consisting of the parent of element e, p, and n_{rb} rings of neighbors of p that lie immediately around p.*

   (b) *Construct a set $\mathsf{P}_2 \subset \mathsf{E}^{\alpha-2}$ consisting of the parents of the elements in P_1.*

   (c) *Update P_1 including all the children elements of the elements in P_2.*

   (d) *Define the new subdomain $\hat{\Omega}_b^{\alpha-1} = \mathsf{P}_1 \cup \hat{\Omega}^{\alpha-1}$.*

   (e) *Define the new subdomain $\hat{\Omega}_b^{\alpha-2} = \mathsf{P}_2 \cup \hat{\Omega}^{\alpha-2}$.*

As depicted in Fig. 5, a reduction in basis overlap is achieved with higher values of the ring balance parameter. However, some elements might still support hierarchical basis functions from distant levels after balancing. For instance, in Fig. 5(e), element e supports a function from level 1 after balancing with either $n_{rb} = 1$ or $n_{rb} = 2$.
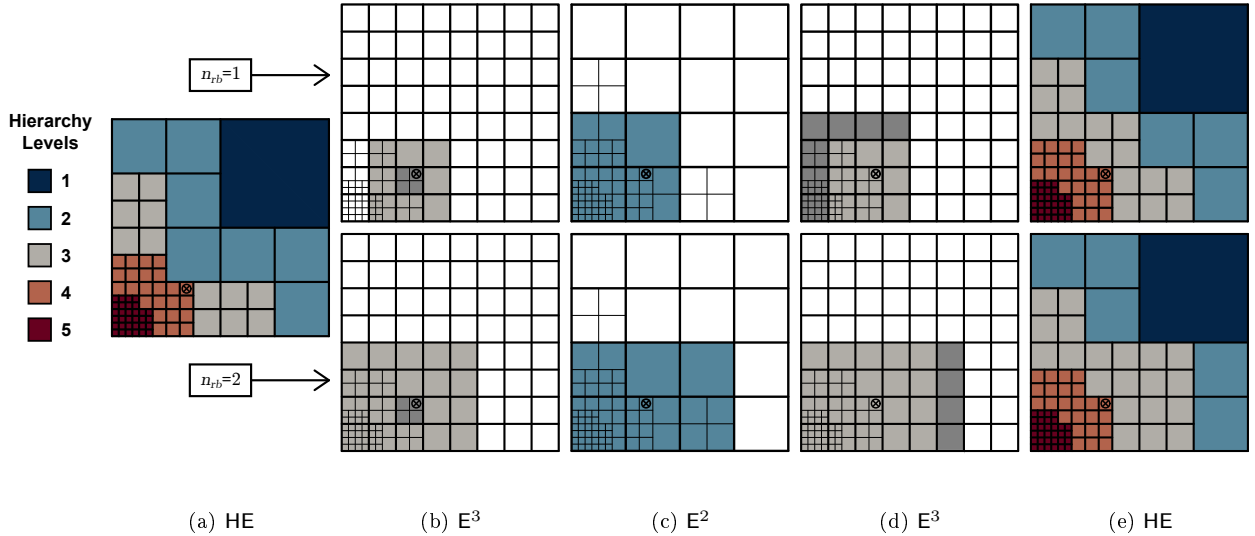

(a) HE   (b) E^3 (c) E^2 (d) E^3 (e) HE

Figure 5: Ring balancing of a bilinear hierarchical mesh over an element e for $n_{rb} = 1$ and $n_{rb} = 2$. The considered element e is marked with a circled cross. (a) Original hierarchical mesh. (b) Construction of P_1 over E^3 with the parent of element e, p (shaded in darker gray), and n_{rb} rings of elements surrounding p. (c) Construction of P_2 over E^2 with the parents of the elements in P_1. (d) Update of P_1 over E^3 with all the children of the elements in P_2. The new elements added to P_1 are shown in darker gray. (e) Final mesh, after redefining the subdomains $\hat{\Omega}_b^3 = \mathsf{P}_1 \cup \hat{\Omega}^3$ and $\hat{\Omega}_b^2 = \mathsf{P}_2 \cup \hat{\Omega}^2$. In subfigures (b)-(d), elements in higher levels are shown with lighter contours.

To overcome this limitation, we developed *function support balancing*. Function support balancing is an elementwise balancing procedure that specifically accounts for the level of each element e, the levels of the

16

hierarchical basis functions in $\mathbf{H}^e$, and the extent of the support of these functions. Algorithm 7.2 outlines this new balancing strategy.

Algorithm 7.2. *Balancing of a hierarchical spline for a given value of the function support balance parameter, $n_{fsb} \in \mathbb{N}$.*

1. *Partition* HE *by sorting the elements according to their hierarchical level, hence building the subsets* $\mathsf{pHE}^\alpha = \mathsf{E}^\alpha \cap \mathsf{HE}$

2. *For $\alpha = n_h, n_h - 1, \ldots, 1$, do the following for each element $\mathsf{e} \in \mathsf{pHE}^\alpha$:*

   (a) *Construct a set $\mathsf{P} \in \mathsf{E}^{\alpha - n_{fsb} - 1}$ consisting of the n_{fsb} grandparent of element e, g, and the neighbors of g that support e's hierarchical basis functions.*

   (b) *Construct a set $\mathsf{C} \in \mathsf{E}^{\alpha - n_{fsb}}$ consisting of the children of the elements in P.*

   (c) *Define the new subdomain $\hat{\Omega}_b^{\alpha - n_{fsb} - 1} = \mathsf{P} \cup \hat{\Omega}^{\alpha - n_{fsb} - 1}$.*

   (d) *Define the new subdomain $\hat{\Omega}_b^{\alpha - n_{fsb}} = \mathsf{C} \cup \hat{\Omega}^{\alpha - n_{fsb}}$.*

   (e) *If $n_{fsb} > 1$, ring balance the current element with $n_{rb} = 1$.*

The function support balance parameter enables us to set a maximum level for the hierarchical basis functions with support over an element as follows: an element in level α can only support functions in levels $\alpha, \alpha - 1 \ldots \alpha - n_{fsb}$. Should a function from level $\beta < \alpha - n_{fsb}$ be nonzero over this element, Algorithm 7.2 will refine its support until the balancing constraint is satisfied. Thus, function support balancing sets an explicit limit to the overlapping of functions over the elements in HE that we can control directly. The performance of n_{fsb} is illustrated in Fig. 6 on a simple two-dimensional scenario. Additional analysis of function support balancing will be presented in Section 8.6.1.

Notice that function support balancing also brings an improved control over the computation of $\mathbf{L}^e$: n_{fsb} enables us to explicitly define the distance from the level α of each element to the coarse level α_c in Algorithm 4.1. This also implies an enhanced control of the computation of the reconstruction operator $\mathbf{R}^e$ and the performance of coarsening as described in Algorithms 6.1 and 6.2.

8. Applications in mathematical oncology

In this section, we present a series of numerical examples that highlight the advantages of hierarchical adaptivity over uniform meshes in a promising, cutting-edge field: the modeling and simulation of cancer. In particular, we will focus on localized prostate cancer growth, as introduced in Section 1.1. We will show that the use of the techniques described in Sections 5, 6 and 7 are fundamental in order to perform patient-specific tissue-scale simulations in an accurate and efficient way − a requisite for the implementation of predictive models of tumor progression on a clinical scenario.
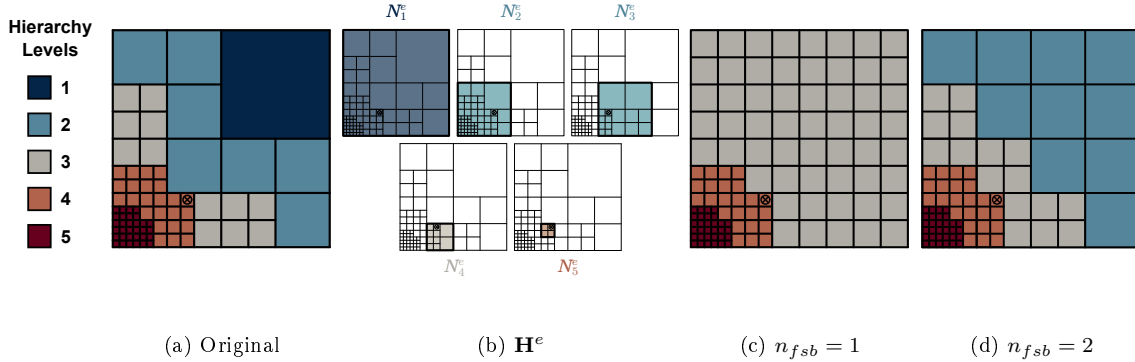
17

(a) Original   (b) $\mathbf{H}^e$ (c) $n_{fsb} = 1$ (d) $n_{fsb} = 2$

Figure 6: Function support balancing of a bilinear hierarchical mesh over an element e for $n_{fsb} = 1$ and $n_{fsb} = 2$. The considered element e is marked with a circled cross. (a) Original hierarchical mesh. (b) Hierarchical basis functions in $\mathbf{H}^e$, represented throughout their support with lighter shades for each level. The support of the only hierarchical basis function in $\mathbf{H}^e$ from level 1 covers $\hat{\Omega}$ completely. (c) Resulting hierarchical mesh after balancing with $n_{fsb} = 1$ over e. Following Algorithm 7.2, $\mathsf{P} = \mathsf{E}^2$ and, then, $\mathsf{C} = \mathsf{E}^3$. Consequently, $\hat{\Omega}_b^2 = \mathsf{E}^2$ and $\hat{\Omega}_b^3 = \mathsf{E}^3$, so that no function from any level $\alpha < 3$ enters in $\mathbf{H}^e$. (d) Resulting hierarchical mesh after balancing with $n_{fsb} = 2$ over e. Now, Algorithm 7.2 renders $\mathsf{P} = \mathsf{E}^1$ and $\mathsf{C} = \mathsf{E}^2$. Hence, the balancing constraint only applies to hierarchical basis functions in level 1. Finally, we ring balance the element e with $n_{rb} = 1$.

.

8.1. Tumor growth model

The model described herein comes from our prior work on the modeling and simulation of localized prostate cancer [30]. We will only consider two cell species: cancerous and non-cancerous. Prostate cancer growth is driven by a complex array of nutrients, hormones and proteins that interact leading to very different behaviors of the disease. Following other approaches to model cancer growth, we will simply consider a generic nutrient σ. We will assume that σ follows the dynamics of glucose. However, the effect of other regulatory substances on cancer growth could be included as an equivalent in terms of glucose. This model also accounts for the dynamics of a prostate cancer biomarker known as Prostate Specific Antigen (PSA), which is measured with a blood test and plays a crucial role during the diagnosis, treatment selection, and follow-up of the patients [60]. Both healthy and tumoral cells produce PSA, but the latter generally secrete it at a much higher rate than healthy cells, which is the key idea behind PSA testing. In particular, we will include an equation for the dynamics of the tissue PSA p, defined as the serum PSA[1] concentration leaked to the bloodstream per unit volume of prostatic tissue.

Leveraging the phase field method, we define an order parameter ϕ that measures the cell microstructure: it will vary from the value of 0, corresponding to the healthy host tissue, to the value of 1, representing a

---

[1] Serum PSA is the concentration of PSA in the patient's blood.

tumor region. The level set $\phi = 0.5$ implicitly defines the interface between healthy and cancerous tissue, with no need to explicitly track it. The dynamics of the tumor is governed by the equation

$$\frac{\partial \phi}{\partial t} = \lambda \Delta \phi - \frac{1}{\tau} \frac{\mathrm{d} F(\phi)}{\mathrm{d} \phi} + \chi \sigma - A\phi \tag{35}$$

where $F(\phi) = 16\phi^2(1-\phi)^2$ is a double-well potential, a typical function within the phase-field method that makes possible the coexistence of both phases in the model. The third and fourth terms in Eq. (35) model the nutrient-driven growth and apoptosis (i.e., programmed cell death), respectively. We assume that the tumor grows linearly at a rate χ in the presence of the nutrient [87]. We further suppose that apoptosis follows first order kinetics with a rate A, as this is the natural response in the prostatic tissue.

The nutrient σ will follow the diffusion-reaction dynamics described by

$$\frac{\partial \sigma}{\partial t} = \epsilon \Delta \sigma + s - \delta \phi - \gamma \sigma \tag{36}$$

where we have assumed that convection has a negligible effect on nutrient transport. This is consistent with the results found in [88]. Furthermore, we will consider a constant nutrient supply s, we will assume that the tumor consumes the nutrient at a linear rate δ, and we will set a linear natural decay for σ at a rate γ.

We will also consider that the concentration of tissue PSA p follows diffusion-reaction dynamics, modeled by

$$\frac{\partial p}{\partial t} = \eta \Delta p + \alpha_h (1 - \phi) + \alpha_c \phi - \gamma_p p \tag{37}$$

where α_h and α_c are the rates of production of PSA per unit volume by healthy and cancerous tissue, respectively. We will also assume that tissue PSA decays naturally following first order kinetics at a rate γ_p. From the definition of the tissue PSA p, it follows that the value of serum PSA P_s can be computed as the integral of p over the whole prostatic gland Ω,

$$P_s = \int_\Omega p \, d\Omega \tag{38}$$

The majority of the parameters in Eq. (35), Eq. (36) and Eq. (37) can be found in the literature [30, 73, 87, 89, 90, 91, 92]. In order to keep the simulations tractable, we will take $\lambda \sim h^2/T$, where h is the characteristic length scale of the computational mesh and T is the characteristic time scale, which we have considered to be $T = 1$ year (see [20] for rationale). Furthermore, it is known that nutrient dynamics is one to two orders of magnitude faster than tumor dynamics. Consequently, we have taken a value for nutrient diffusivity between $\epsilon \sim 10\lambda$ and $\epsilon \sim 100\lambda$.

The model for localized prostate cancer growth that we have described in this section develops steep layers representing the interface between the tumor and the host tissue. In order to capture the interface

19

accurately we require very fine meshes. However, within healthy tissue ($\phi = 0$) or the tumor ($\phi = 1$) we could use a much coarser mesh to represent the constant value of the phase field over these regions. Consequently, localized prostate cancer growth is a paradigmatic problem that can benefit from our local adaptive techniques.

8.2. Spatial discretization

The strong form of our cancer growth problem is composed of Eqs. (35)-(37). We will approximate its solution by means of IGA based on a Bubnov-Galerkin approach that leverages a hierarchical spline basis H, as introduced in Section 3. Let us denote by $\mathcal{V}$ the trial solution and the weighting function spaces, which are assumed to be identical. The space $\mathcal{V}$ is a subset of H^1, the Sobolev space of square integrable functions with square integrable first derivatives. We will only work with free-flux, periodic or zero-valued Dirichlet boundary conditions. Let us define the discrete space $\mathcal{V}^h = \mathcal{H}$, which is a subset of $\mathcal{V}$. Hence, the corresponding variational problem to Eqs. (35)-(37) over the finite dimensional space $\mathcal{V}^h$ can be stated as follows: find $\phi^h, \sigma^h, p^h \in \mathcal{V}^h \subset \mathcal{V}$ such that $\forall u^h, v^h, w^h \in \mathcal{V}^h \subset \mathcal{V}$

$$\int_\Omega \frac{\partial \phi^h}{\partial t} u^h d\Omega + \lambda \int_\Omega \nabla \phi^h \cdot \nabla u^h d\Omega - \int_\Omega u^h \left(\chi \sigma^h - A\phi^h - \frac{1}{\tau} \frac{dF}{d\phi} \left(\phi^h \right) \right) d\Omega = 0 \tag{39}$$

$$\int_\Omega \frac{\partial \sigma^h}{\partial t} v^h d\Omega + \epsilon \int_\Omega \nabla \sigma^h \cdot \nabla v^h d\Omega - \int_\Omega v^h \left(s - \delta \phi^h - \gamma \sigma^h \right) d\Omega = 0 \tag{40}$$

$$\int_\Omega \frac{\partial p^h}{\partial t} w^h d\Omega + \eta \int_\Omega \nabla p^h \cdot \nabla w^h d\Omega - \int_\Omega w^h \left(\alpha_h(1 - \phi^h) + \alpha_c \phi^h - \gamma_p p^h \right) d\Omega = 0 \tag{41}$$

Here ϕ^h, σ^h and p^h are defined as

$$\phi^h(\mathbf{x}, t) = \sum_{A=1}^{n_f} \phi_A(t) N_A(\mathbf{x}) \tag{42}$$

$$\sigma^h(\mathbf{x}, t) = \sum_{A=1}^{n_f} \sigma_A(t) N_A(\mathbf{x}) \tag{43}$$

$$p^h(\mathbf{x}, t) = \sum_{A=1}^{n_f} p_A(t) N_A(\mathbf{x}) \tag{44}$$

where the coefficients ϕ_A, σ_A and p_A are the so-called control variables in the context of IGA. The weighting functions u^h, v^h, and w^h are defined analogously.

8.3. Time integration

We have chosen the generalized-α method [93, 94] in order to perform time integration. This technique is second-order accurate and A-stable. The generalized-α method leads to a nonlinear problem at each time step, which we linearize using the Newton-Raphson method. We used the linear solvers available in the Trilinos open-source package for all the simulations in this paper [95]. In particular, we have used the GMRES algorithm [96] with a basic block diagonal preconditioner to solve the linear system during each corrector step. We also implemented a local update for the consistent tangent matrix during each corrector step because only the term corresponding to the double-well potential in Eq. (35) changes during the corrector phases. We set the parameters of the generalized-α method as in [20]. We chose a constant time step $\Delta t = 0.001$ years for all the examples in this paper.

8.4. Automatic adaptivity

We have automated the operations of refinement and coarsening in our code. To do so, we have introduced two basic features: first, an indicator to determine which elements must undergo an adaptive procedure, and second, a time-step offset for each operation.

We refine the mesh every 5 time steps in order to keep track of the moving interface. However, coarsening is performed every 10 time steps to clean up the finer elements left in the rear of the moving interface and that are no longer required. This means that every 10 time steps we refine and then we coarsen the mesh.

To determine whether to perform an adaptive procedure or not we define the gradient-based indicator

$$\rho_e = \frac{1}{\mathrm{vol}\,(\Omega^e)} \int_{\Omega^e} \mid \nabla\phi \mid^2 d\Omega^e, \tag{45}$$

where we compute the squared L^2-norm of the gradient of the phase field over each element Ω^e and normalize it with respect to its corresponding volume $\mathrm{vol}\,(\Omega^e)$. During a refinement step, we compare ρ_e to its average over all elements and proceed to flag for refinement those elements that exceed an arbitrary threshold. Likewise, during a coarsening step, if ρ_e is smaller than a certain preset value dependent on the average ρ_e, we mark the element for coarsening. These procedures can be expressed algorithmically as

$$\rho_e > \frac{C_R}{n_e} \sum_{e=1}^{n_e} \rho_e \implies \text{refinement} \tag{46}$$

$$\rho_e < \frac{C_C}{n_e} \sum_{e=1}^{n_e} \rho_e \implies \text{coarsening} \tag{47}$$

The constants C_R and C_C in Eq. (46) and Eq. (47) enable us to control the intensity of refinement and coarsening, respectively.

8.5. Initial conditions

The initial condition for the phase field ϕ_0 is an elliptic (2D) or ellipsoidal (3D) tumor that we model with a hyperbolic tangent field. The initial conditions for the nutrient σ and the tissue PSA p are approximations based on ϕ_0 given by $\sigma_0 = c_\sigma^0 + c_\sigma^1 \phi_0$ and $p_0 = c_p^0 + c_p^1 \phi_0$, respectively. The values of the constants c_σ^0, c_σ^1, c_p^0, and c_p^1 are computed so that σ_0 and p_0 reproduce the constant value of the nutrient and the tissue PSA within the tumor and the host tissue, respectively.

We use an adaptive L^2-projection procedure in order to tailor an appropriate initial mesh that fits the initial condition for the tumor phase field. Beginning with a Bézier mesh only composed of elements on level 1, we perform n_h rounds of refinement to progressively enlarge HE with elements in higher levels. During each step k of this procedure we L^2-project ϕ_0 over an initial Bézier mesh composed of elements in levels $\alpha = 1, \ldots, k$. Then we use the gradient-based indicator in Eq. (45) to flag the elements to refine. Finally we refine and balance the resulting Bézier mesh now having elements from levels $\alpha = 1, \ldots, k+1$. In the last refinement round the initial Bézier mesh already includes elements across the whole hierarchy, but we perform one last round of refinement to fine-tune the mesh.

8.6. Results

We carried out several simulations of localized prostate cancer growth leveraging the integrated adaptive methodology presented in this paper. We begin with simple 2D and 3D computational domains in order to focus on the performance of the adaptive algorithms. Then we show the application of these algorithms in the field of mathematical oncology with a tissue-scale, patient-specific simulation of prostate cancer growth. All the simulations presented herein were run in a parallel environment.

8.6.1. Prostate cancer growth in a square domain

We begin investigating the behavior of our adaptive methods on a basic 2D scenario for prostate cancer growth: a 2000×2000 µm square with periodic boundaries. The initial tumor is placed in the center of the computational domain and the lengths of its semiaxes are 100 µm and 150 µm. We used a quadratic hierarchical B-spline space with 5 levels, going from 32×32 elements in E^1 to 512×512 elements in E^5. Table 1 provides the values of the model parameters that we have used in this simulation. We have set $C_R = 1$ and $C_C = 0.001$.

Fig. 7 and Fig. 8 show the evolution of the tumor phase field and the supporting adaptive mesh using two different values of our new function support balance parameter, $n_{fsb} = 1$ and $n_{fsb} = 2$, respectively. The phenomenon of prostate cancer growth is perfectly captured in both simulations (cf. [30]). Initially, the elliptic tumor grows spheroidally but this pattern of growth leads to very low levels of the nutrient inside the tumor. Should the tumor continue to develop with this morphology, it would suffer from starvation, hypoxia, and, eventually, necrosis, which would hamper cancer growth. Alternatively, the tumor undergoes a shape

22

Table 1: List of parameters in the model of prostate cancer growth: square and cubic domain simulations.

Parameter	Notation	Value
Diffusivity of the phase field	λ	$1.6 \cdot 10^5$ µm^2/year
Time scale for the phase field	τ	0.01 years
Nutrient-induced tumor growth rate	χ	600 l/(g·year)
Apoptosis rate	A	600 1/year
Nutrient diffusivity	ϵ	$6.0 \cdot 10^6$ µm^2/year
Nutrient supply	s	2.70 g/(l·day)
Nutrient consumption rate	δ	2.75 g/(l·day)
Nutrient natural decay rate	γ	1000 1/year
Tissue PSA diffusivity	η	$1.6 \cdot 10^5$ µm^2/year
Tissue PSA production rate in healthy tissue	α_h	6.25 (ng/ml)/(cm^3·year)
Tissue PSA production rate in cancerous tissue	α_c	$\alpha_c = 15\alpha_h$
Tissue PSA natural decay rate	γ_p	100 1/year

instability towards a fingered pattern of growth. This shift in morphology begins with a sort of bottleneck, as shown in Fig. 7(b) and Fig. 8(b). Then, in the evolving geometries in Fig. 7(c)-(d) and Fig. 8(c)-(d), the distance from the inner areas of the tumor to the surrounding healthy tissue is much shorter that in an spheroidal morphology, thus preventing the tumoral cells from starving.

The design of the hierarchy is not trivial. For this problem of prostate cancer growth, we knew that a quadratic uniform B-spline mesh of 512×512 elements produced an accurate solution. Then, we started to add levels on top of this original mesh, each consisting of a uniform B-spline mesh with half the elements per parametric direction than the last added. This strategy enabled us to study the benefits of using hierarchical spaces with increasing number of levels, but such that the interface would always be supported by the elements in the original 512×512 mesh.

Leveraging a 5-level hierarchy, the resulting finite element meshes depicted in Fig. 7 and Fig. 8 accomplish our expectation: finer elements to support the interface and coarser elements within the tumor and in healthy tissue regions, where the phase field is constant. The main difference between the meshes in Fig. 7 and Fig. 8 is the distribution of the elements in HE across the hierarchy during the simulation. Fig. 9 portrays the subdomains $\hat{\Omega}^\alpha$, $\alpha = 1, \ldots, n_h$, and the elements in HE that belong to each of these subdomains for the balance parameter values $n_{fsb} = 1$ and $n_{fsb} = 2$. In Fig. 9, we can also observe how the elements in HE$\cap$E$^\alpha$, $\alpha = 1, \ldots, n_h$, can be collapsed from a multilevel topology onto the single-level finite element representation provided by the Bézier mesh HE. The balance parameter n_{fsb} affects how adaptivity is performed beyond refinement and coarsening, limiting the overlapping of the basis functions in H across the hierarchy on each

23

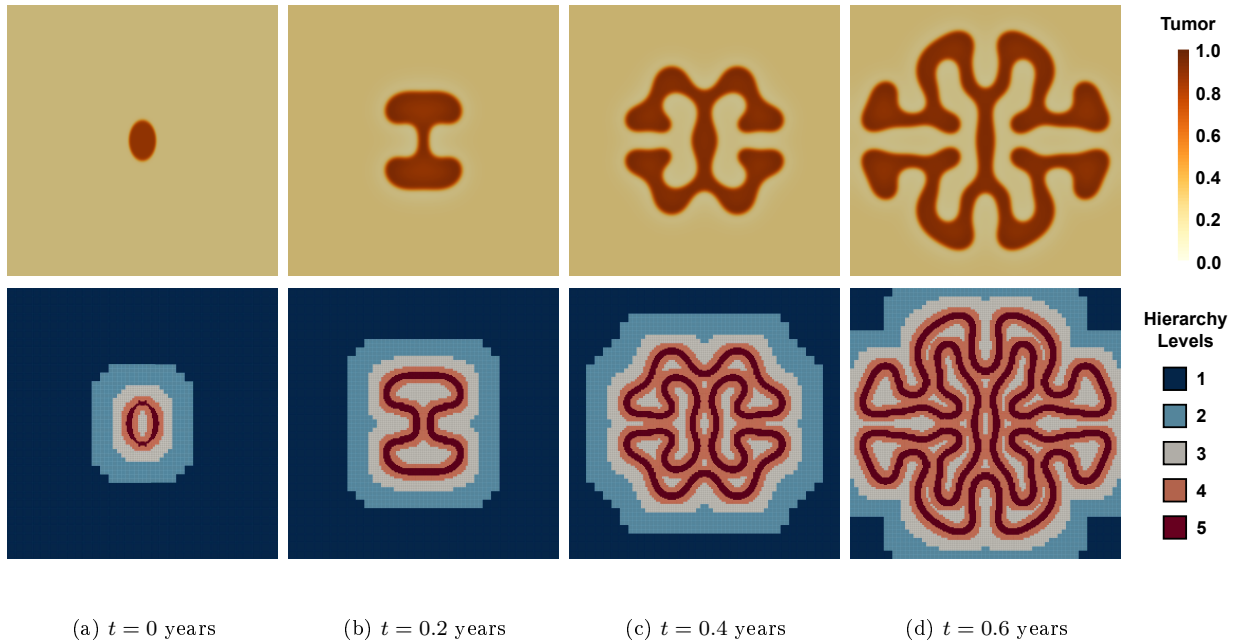(a) $t = 0$ years (b) $t = 0.2$ years (c) $t = 0.4$ years (d) $t = 0.6$ years

Figure 7: Tumor phase field and adaptive mesh for a simulation of 2D prostate cancer growth using a quadratic hierarchical B-spline space with 5 levels and balance parameter $n_{fsb} = 1$. Level 1 is the coarsest level, with 32×32 elements in E^1, and level 5 is the finest, with 512×512 elements in E^5. The size of the tissue domain is 2000×2000 µm.

element. Hence, the differences in the hierarchical meshes depicted in Fig. 7, Fig. 8, and Fig. 9 are a direct consequence of the selected value for the balance parameter.

When $n_{fsb} = 1$, an element in level α can only support hierarchical basis functions in level α or level $\alpha - 1$. Consequently, after refinement or coarsening, balancing enforces a considerable amount of further refinement to enforce the overlapping constraint. Let us illustrate this with an example of the operation of balancing after refinement and coarsening. Consider the elements in an intermediate level α. If a sufficient amount of elements in level α are refined to level $\alpha + 1$, some hierarchical basis functions in level $\alpha - 1$, which originally had some support on those refined elements in level α, may have support in level $\alpha + 1$ now. During balancing, these functions will be flagged for refinement and their entire support will be refined to level α. Let us consider an alternative example. After performing coarsening in level α, we might eliminate enough elements in this level so that some functions in level $\alpha - 1$ enter in the hierarchical basis. However, functions in level $\alpha - 1$ have a larger support and part of it may lie in level $\alpha + 1$ or even deeper. Balancing will flag these new functions in level $\alpha - 1$ and their whole support will be refined to level α. Hence, when $n_{fsb} = 1$ the borders between the subdomains $\hat{\Omega}^\alpha$, are very distant one another in order to prevent the hierarchical basis functions in levels above $\alpha - 1$ and below $\alpha + 1$ from overlapping. As a result, this constraint produces a smooth transition between the levels of the hierarchy, whose elements are disposed in wide rings for each
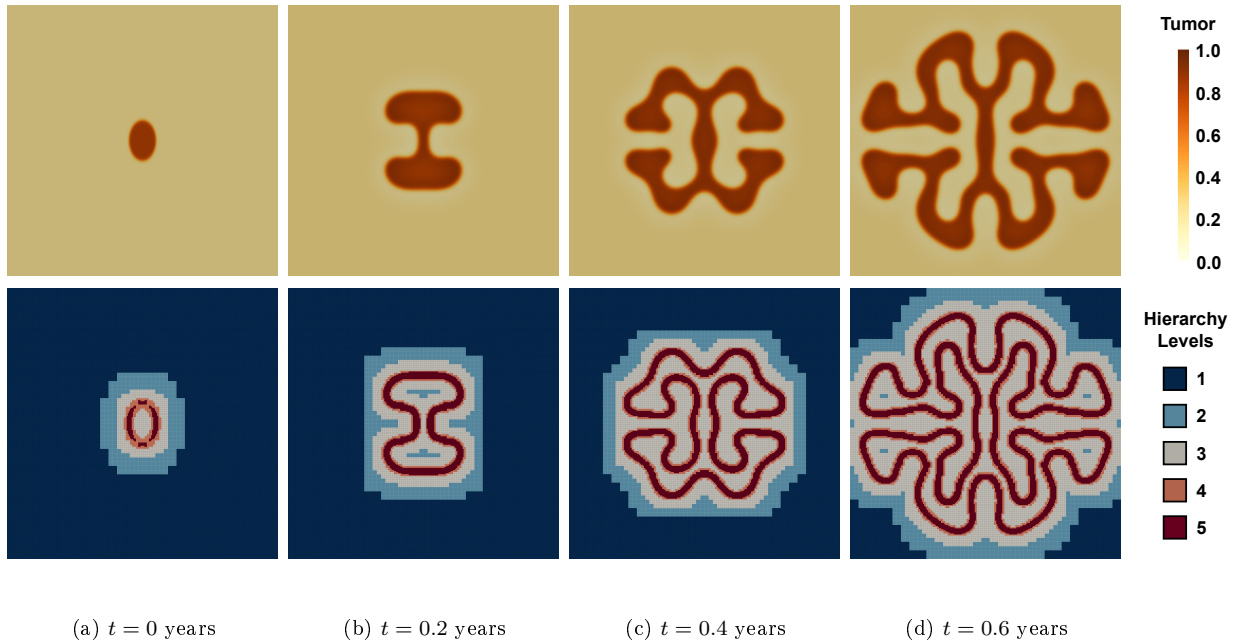
24

(a) $t = 0$ years   (b) $t = 0.2$ years   (c) $t = 0.4$ years   (d) $t = 0.6$ years

Figure 8: Tumor phase field and adaptive mesh for a simulation of 2D prostate cancer growth using a quadratic hierarchical B-spline space with 5 levels and balance parameter $n_{fsb} = 2$. Level 1 is the coarsest level, with 32×32 cells in E^1, and level 5 is the finest, with 512×512 cells in E^5. The size of the tissue domain is 2000×2000 µm.

level, as shown in Fig. 7 and Fig. 9(a)-(b).

The previous examples considered a situation close to the threshold level to perform balancing, that is, a global function in level $\alpha - 1$ that has support in level $\alpha + 1$. However, balancing propagates across the hierarchy and it also prevents the overlapping of basis functions in HE in more distant levels. For example, consider hierarchical basis functions in a shallow level $\alpha - m$, $m >> n_{fsb}$, that after an adaptive operation have support on elements in level α. In this case, balancing refinement is triggered in an element in level α and it will affect functions from level $\alpha - m$ downwards until all the hierarchical basis functions with support in that element meet the balancing constraint.

If we set $n_{fsb} = 2$, the elements in level α can now support hierarchical basis functions in levels α, $\alpha - 1$ and $\alpha - 2$. The overlapping between the functions of different levels of the hierarchy is stronger than for $n_{fsb} = 1$. Hence, the refinement operation caused by balancing is significantly reduced: to trigger refinement to balance the mesh when $n_{fsb} = 2$ it is necessary that an element in level α supports a hierarchical basis function in level $\alpha - 3$ or above. Consequently, the transition between levels in the hierarchical mesh is sharper for $n_{fsb} = 2$, with the elements describing thinner rings, as depicted in Fig. 8 and Fig. 9(c)-(d).

Fig. 9 shows that the coarsest elements in the hierarchy, belonging to level 1, only appear sufficiently far from the interface. In these areas, the gradient is small and the hierarchical basis functions in level 1

25

(a) Bézier mesh HE, $n_{fsb}{=}1$

(b) Bézier meshes E^α at each hierarchical level α, $n_{fsb}{=}1$

(c) Bézier mesh HE, $n_{fsb}{=}2$

(d) Bézier meshes E^α at each hierarchical level α, $n_{fsb}{=}2$
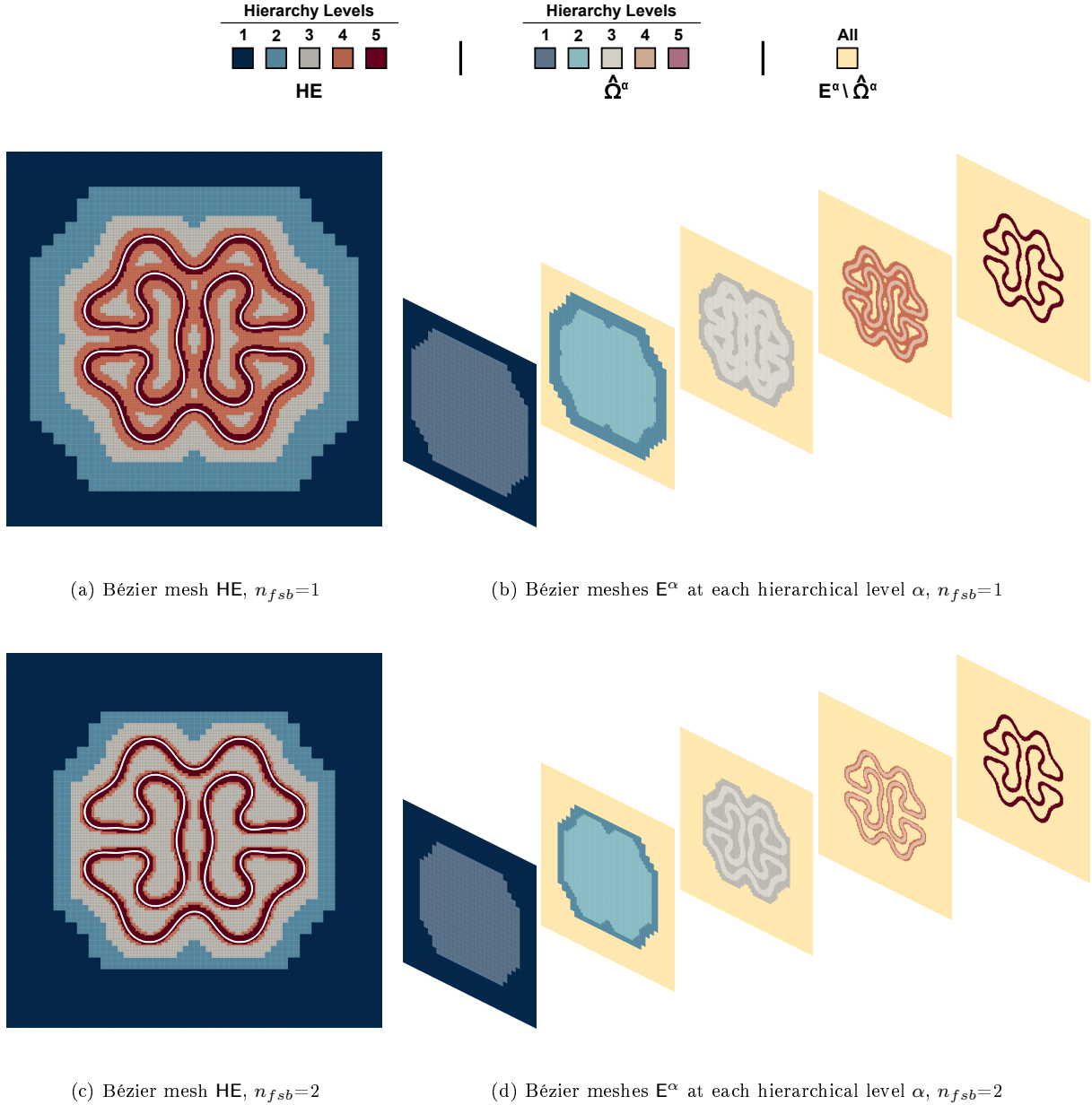
Figure 9: The distribution of elements across the hierarchical levels and the corresponding Bézier meshes for $n_{fsb} = 1$ and $n_{fsb} = 2$ at $t = 0.4$ years.

do not overlap with basis functions in much deeper levels. However, the gradient is also approximately zero in the inner tumor but only elements in level 2 appear here. This is caused by balancing: the vicinity of the interface requires the finest elements and the fingered morphology does not have enough space in the internal areas of the tumor to enable the transition from elements in level 5 to level 1 satisfying the balancing constraint. A similar situation takes place with the elements in level 2 between the branches of the tumor,

26

chiefly when $n_{fsb} = 2$, as depicted in Fig. 9(c)-(d). When $n_{fsb} = 1$, most of those elements in level 2 are refined to level 3 near the interface (see Fig. 9).

The plots in Fig. 10 offer a deeper insight in the differences between a mesh balanced with $n_{fsb} = 1$ and $n_{fsb} = 2$. The parameters that describe the distribution of the number of basis functions per element take higher values when we select $n_{fsb} = 2$ in Fig. 10(a). This feature is caused by the increased overlapping of hierarchical basis functions that is inherent to higher values of the balance parameter. However, the total number of hierarchical basis functions is lower for $n_{fsb} = 2$ than for $n_{fsb} = 1$, as shown in see Fig. 10(b). This is due to the more localized effect of the adaptive operations when balancing is more indulgent. The number of elements is also lower for $n_{fsb} = 2$ than for $n_{fsb} = 1$, as depicted in Fig. 10(c). However, notice that the difference between both choices of balance parameter is mild in Fig. 10(b)-(c) and both led to a remarkable decrease in n_e with respect to the 512×512 element mesh. For instance, at $t = 0.4$ years, the total number of elements is reduced to a 15.1% with $n_{fsb} = 1$ and a 12.2% with $n_{fsb} = 2$.

The minimal number of hierarchical basis functions over a 2D element is exactly $(p + 1)^2$, as in the corresponding single-level B-spline mesh that configures each hierarchical level. We consider the ratio of the number of elements with minimal number of hierarchical basis functions to the total number of elements. Higher values of this ratio reflect minimal overlapping of functions. In the context of our adaptive algorithms, this means that there is a smooth transition between the regions of the mesh corresponding to each hierarchical level, $\hat{\Omega}^\alpha$. Superior accuracy is expected from these meshes because they are composed of broad regions of finer elements near the interface and coarser elements sufficiently far from it (see Fig. 9); not to mention that reduced overlapping improves numerical conditioning. In Fig. 10(c) the number of elements supporting a minimal number of hierarchical basis functions is considerably superior for $n_{fsb} = 1$ than for $n_{fsb} = 2$. Indeed, such difference is of greater magnitude than in terms of the total number of elements. Hence, the ratio of elements with minimal number of hierarchical basis functions to total elements is higher when we choose $n_{fsb} = 1$, as shown in Fig. 10(d).

The cumulative histograms in Fig. 11 show the distribution of the number of hierarchical basis functions per element along the simulations corresponding to Fig. 7 and Fig. 8. Notice the increased dispersion and higher variability for the case $n_{fsb} = 2$, which is caused by the increased overlapping of functions for this choice of balance parameter.

In order to test the accuracy of the solutions computed with a hierarchical basis, in Fig. 12 we provide a comparison between these and a reference solution obtained on a quadratic single-level B-spline mesh of 1024×1024 elements. When we select $n_{fsb} = 1$ the reference and hierarchical solutions are practically indistinguishable, as depicted in Fig. 12(a). If we set $n_{fsb} = 2$, some slight differences between the reference and the hierarchical solution appear, as seen in Fig. 12(b). However, the overall quality of the hierarchical solution is still acceptable. The decrease in the accuracy for $n_{fsb} = 2$ is caused by the increased overlapping of hierarchical basis functions. Consequently, the mesh presents coarser elements closer to the interface,
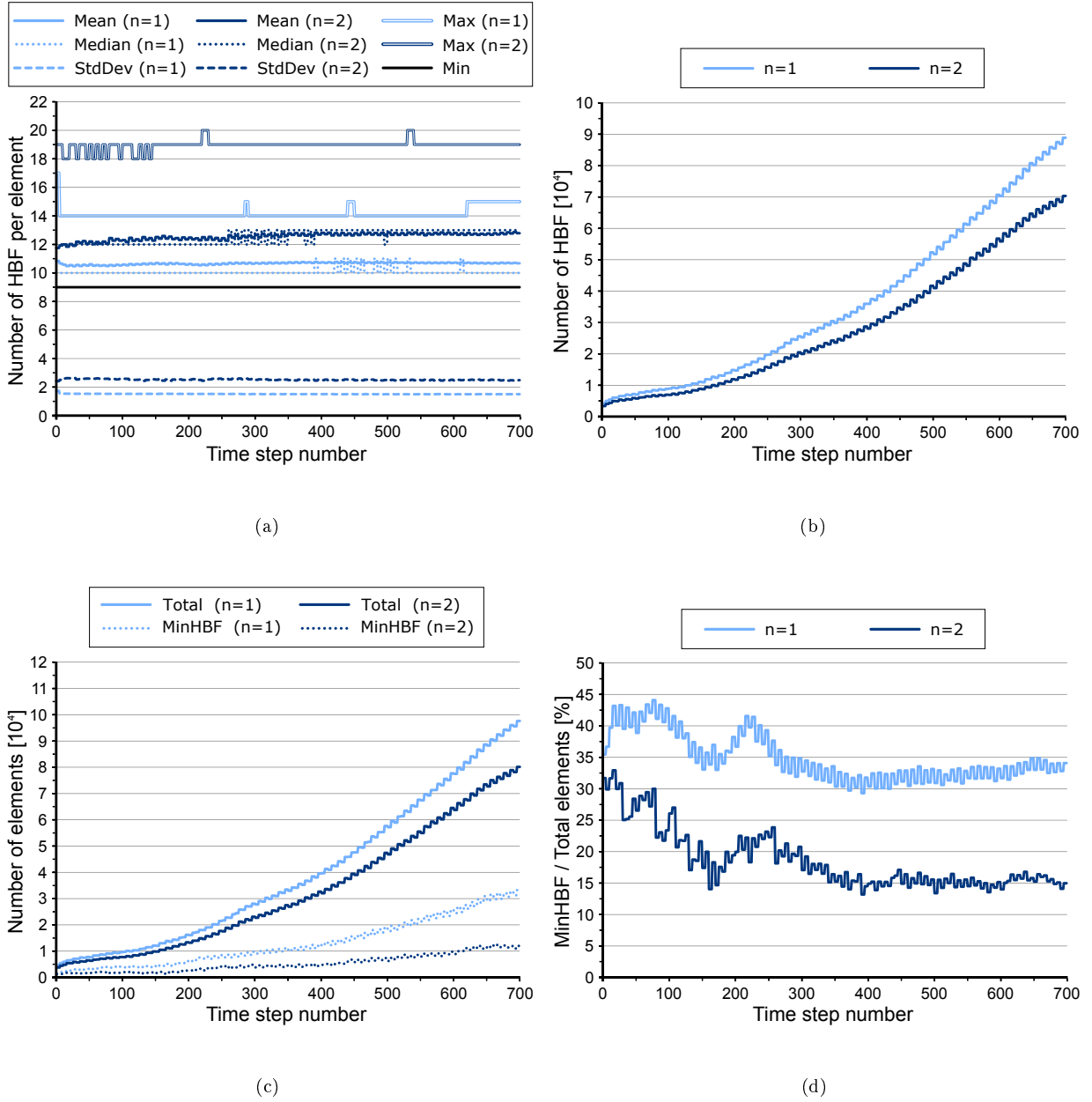
27

(a)

(b)

(c)

(d)

Figure 10: Analysis of the number of hierarchical basis functions (HBF) and elements along the simulations in Fig. 7 and Fig. 8. (a) Time history of the parameters that describe the distribution of the number of HBF, including the mean, the median, the standard deviation (StdDev), the minimum, and the maximum. The minimum number of HBF is exactly $(p+1)^2$. (b) Time history of the total number of HBF. (c) Time history of the total number of elements and the number of elements with minimal number of hierarchical basis functions (MinHBF). (d) Time history of the ratio of the number of MinHBF elements to the total number of elements.
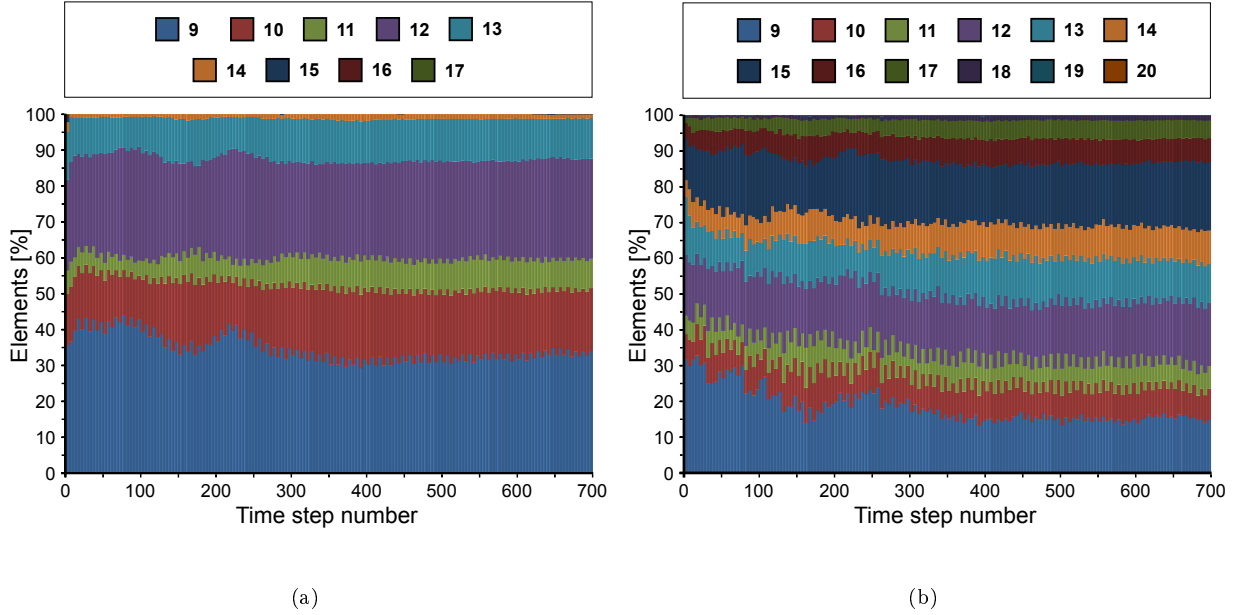
Figure 11: Time history of the distribution of the number of hierarchical basis functions per element for $n_{fsb} = 1$ (a) and $n_{fsb} = 2$ (b).

which reduce the accuracy of the overall hierarchical solution. The balance parameter value $n_{fsb} = 2$ does not bring major savings in the number of elements or functions in the hierarchy (see Fig. 10) and the solution is not more accurate. Hence, we conclude that $n_{fsb} = 1$ suffices to balance the mesh accurately while preserving all the advantages of local adaptivity.

Our adaptive algorithms are fully compatible with global p-refinement, as shown in Fig. 13. Increasing the degree of the polynomial functions that build the hierarchical basis enriches the space of functions that can be represented with that basis. Consequently, the accuracy of the solution also improves. However, global p-refinement also carries some disadvantages, which are mostly manifested for high polynomial degrees, such as $p = 4$ or $p = 5$. First, increasing the polynomial order increments the number of functions on every element in every level of the hierarchy. Second, elevating the polynomial degree leads to an adaptive mesh where refinement and coarsening are less and less local. As the support of each basis function $N \in \mathsf{N}^\alpha$ is $(p+1)^2$ elements in E^α, the refinement operation induced by balancing will affect a larger domain. Hence, the extent of the subdomains $\hat{\Omega}^\alpha$ is larger and those corresponding to coarser levels might become void. As a result, a substantial amount of memory is required to store the tangent matrix and the residual and the simulations run more slowly.

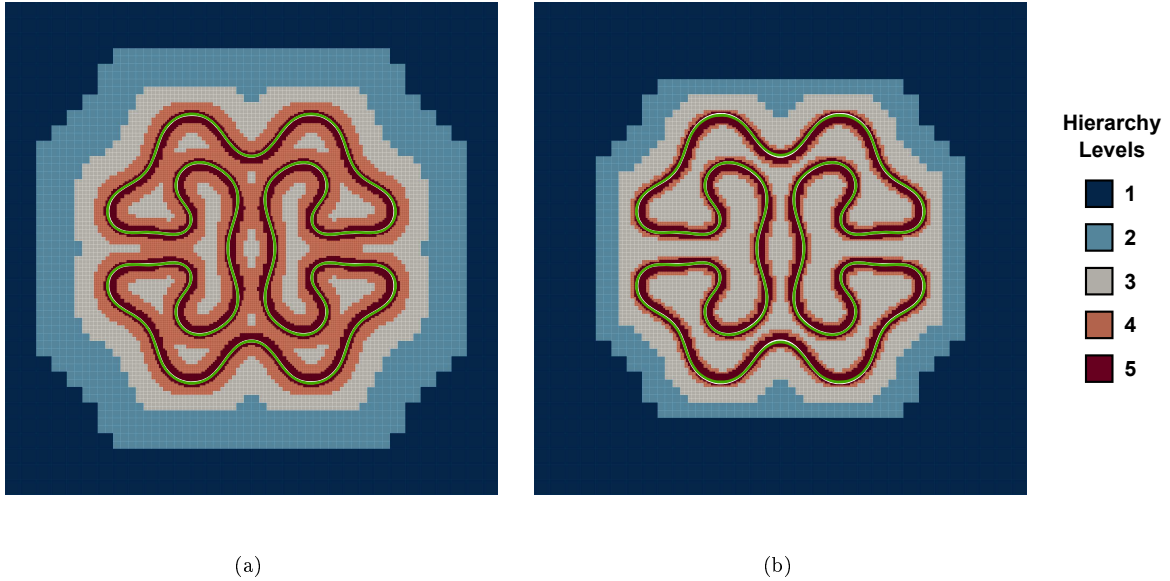<center>(a)   (b)</center>

Figure 12: Comparison between the reference solution (white contour) for the interface ($\phi = 0.5$) at $t = 0.4$ years and a hierarchical solution (green contour) computed with a quadratic 5-level B-spline space for the balance parameter values $n_{fsb} = 1$ and $n_{fsb} = 2$.

8.6.2. Prostate cancer growth in a cubic domain

We tested our adaptive algorithms in a cubic domain with periodic boundaries and side length 2000 µm to study their performance in a simple 3D scenario. The initial tumor is placed in the center of the cube and has two semiaxes of length 100 µm and one with length 150 µm. We have set $C_R = 1$ and $C_C = 0.01$ for this simulation. The model parameters take the values indicated in Table 1.

We designed the hierarchy following the same approach as for the previous 2D examples, but reducing the mesh size to keep simulations tractable. Our cancer growth problem can be solved with sufficient accuracy by means of a single-level B-spline mesh with 256 elements per parametric direction. Hence, we used a quadratic hierarchical B-spline space with 5 levels, ranging from 16 to 256 elements per parametric direction in E^1 and E^5, respectively. According to the analysis of the effect of the function support balance in the previous subsection, we chose $n_{fsb} = 1$ to attain higher accuracy with a reasonable distribution of elements across the levels in HE.

Fig. 14 shows the evolution of the tumor and the adaptive mesh along several time steps. Similar to the 2D scenario, tumor morphology evolves from an initial ellipsoid developing branches to gain better access to the nutrient. The main feature that emerges in this 3D simulation is that some of the finger-like structures turn into curved layers of tumoral tissue. However, if we take a cross section of the tumor we obtain again the characteristic branches that we observed in the 2D scenario. Interestingly, at $t = 0.6$ years the tumor has divided into three independent tumoral regions: a central ellipsoidal mass surrounded by two opposing
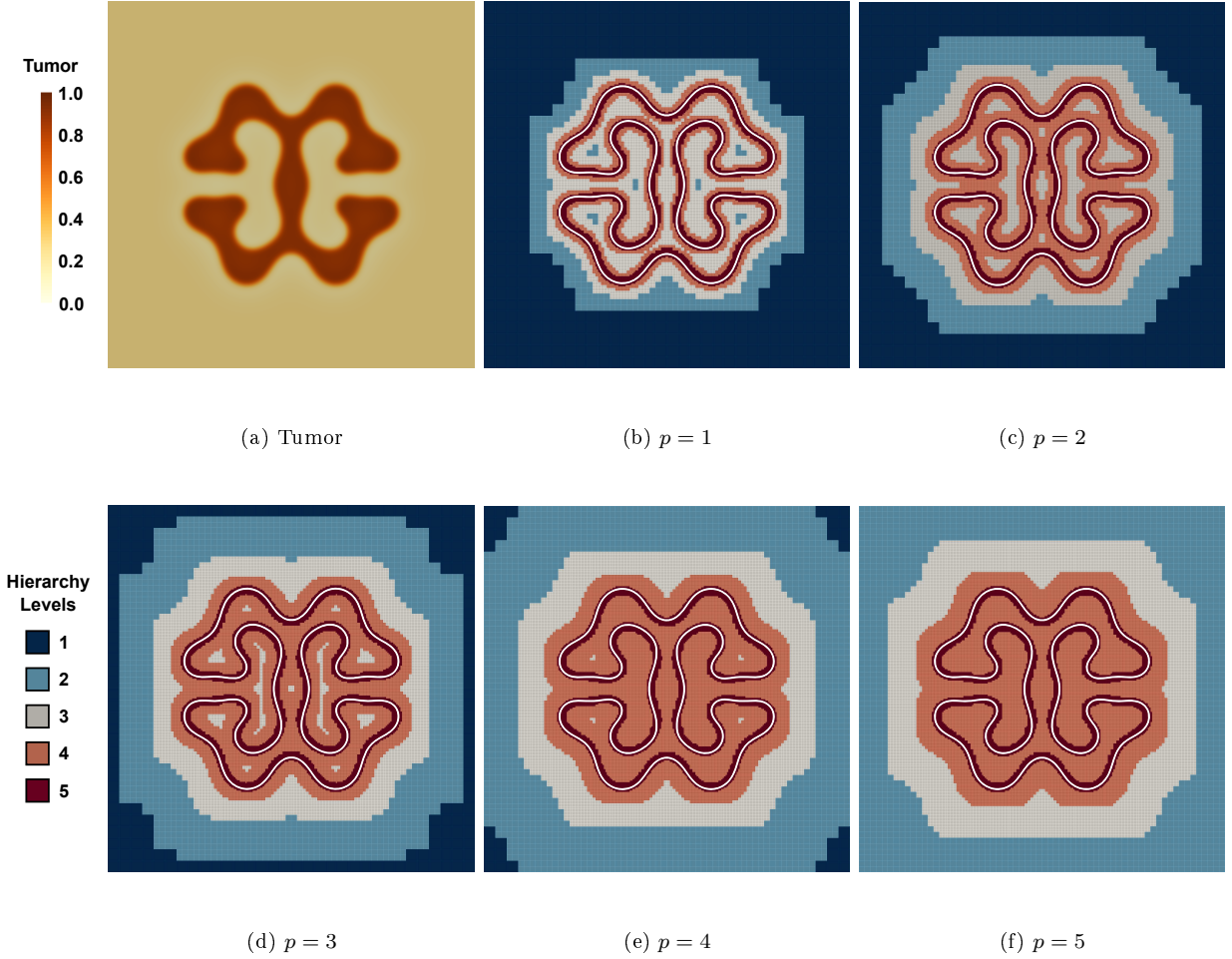
<center>30</center>

(a) Tumor   (b) $p = 1$ (c) $p = 2$

(d) $p = 3$ (e) $p = 4$ (f) $p = 5$

Figure 13: Our adaptive algorithms are compatible with global p-refinement. (a) Tumor phase field at $t = 0.4$ years computed with a quadratic hierarchical B-spline basis with 5 levels as in Fig. 7. (b)-(f) Two-dimensional meshes at $t = 0.4$ years using a 5-level hierarchical B-spline space with increasing polynomial degrees from $p = 1$ to $p = 5$ in both parametric directions. The white contour depicts the level set $\phi = 0.5$, which represents the interface between the tumor and the healthy tissue. We set the balance parameter value to $n_{fsb} = 1$.

concave shell-like structures with a satellite directed outwards. These satellites eventually detach, hence defining two additional cancerous regions.

585   The adaptive mesh follows the evolution of the interface between healthy and tumoral tissue efficiently along the simulation. Our algorithms to implement local adaptivity dramatically lower the total number of elements with respect to an uniform mesh with 256 elements per parametric direction. For example, at $t = 0.4$ years n_e is reduced to 6.9%, and it stays below 15% until $t = 0.56$ years. These rates reveal the promising potential of our algorithms to efficiently resolve local features with IGA in 3D scenarios.

31

(a) $t = 0$ years   (b) $t = 0.2$ years
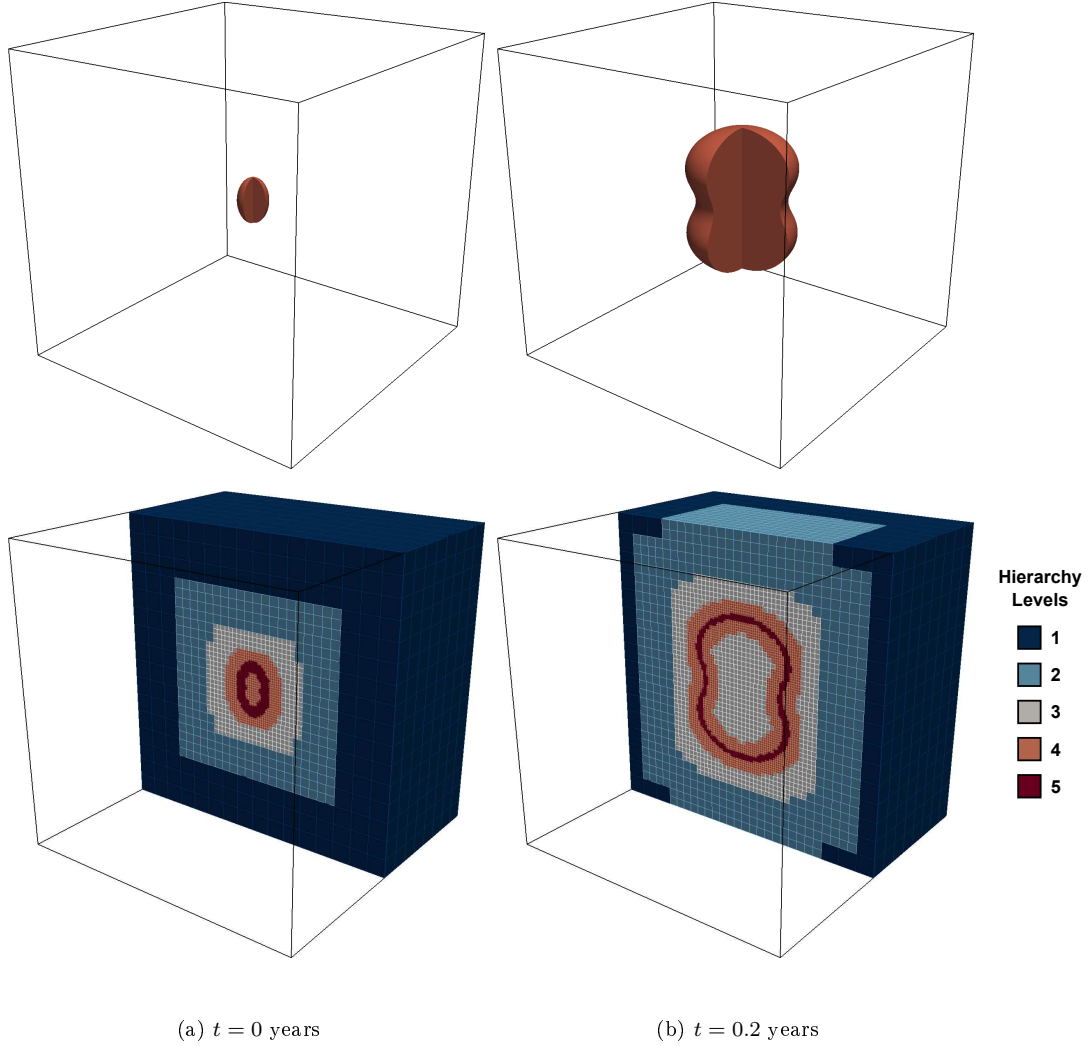
Figure 14: Tumor phase field and adaptive mesh for a simulation of 3D prostate cancer growth using a quadratic hierarchical B-spline space with 5 levels and balance parameter $n_{fsb} = 1$. Level 1 is the coarsest level, with $16 \times 16 \times 16$ elements in E^1, and level 5 is the finest, with $256 \times 256 \times 256$ elements in E^5. The size of the tissue domain is $2000 \times 2000 \times 2000$ µm.

8.6.3. Tissue-scale, patient-specific simulation of prostate cancer growth

Our final example consists of a simulation of prostatic tumor growth over the geometry of a patient's prostate, which we extracted from CT images provided by the Austin Radiological Association. Solid anatomic NURBS models can be constructed following numerous approaches [97, 98, 99]. The geometry of the prostate organ is topologically equivalent to that of a torus. Hence, we leverage a parametric mapping method that deforms a basic torus NURBS model to match the surface of the prostate obtained from the patient's CT data. The original torus and the resulting prostate mesh are both discretized with 32 elements along the toroidal direction, 32 elements along the cross-section circumferential direction, and 8 elements

32
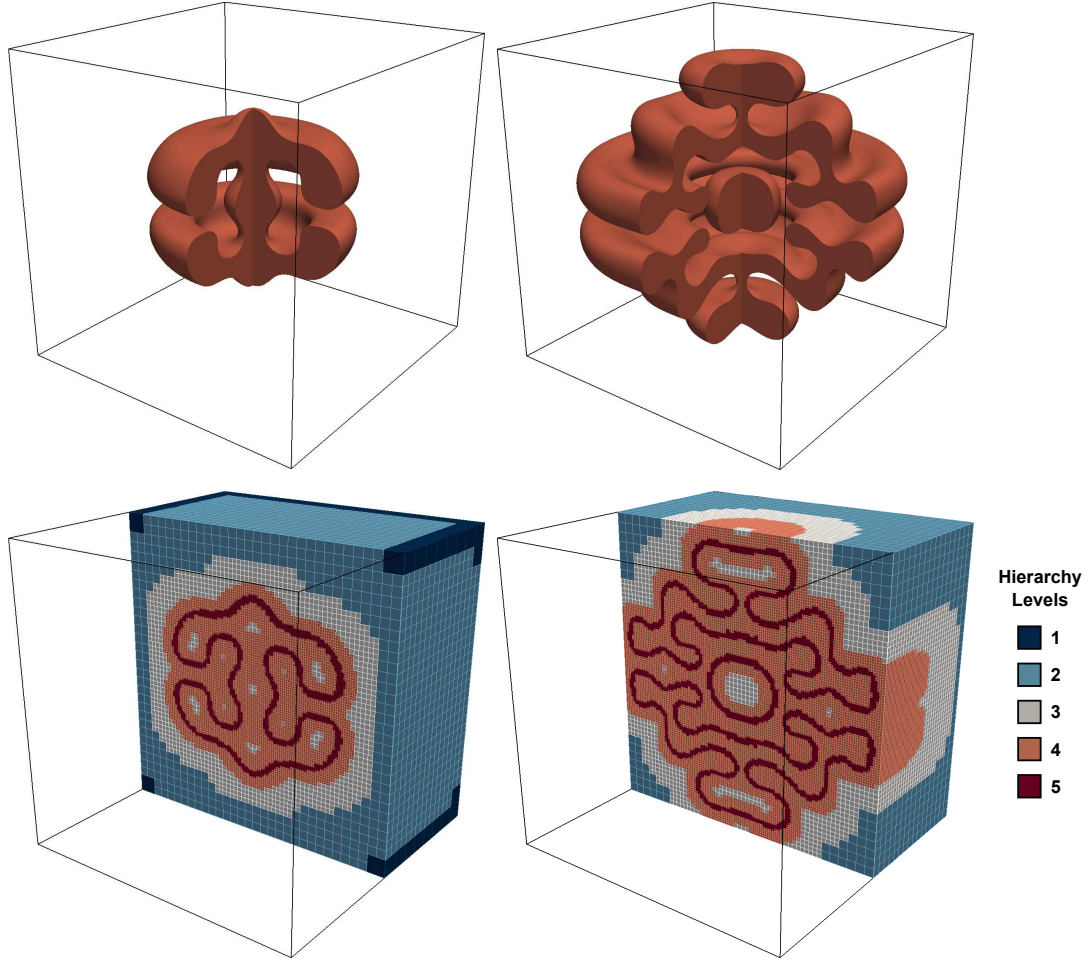
(c) $t = 0.4$ years   (d) $t = 0.6$ years

Figure 14: (Continued) Tumor phase field and adaptive mesh for a simulation of 3D prostate cancer growth using a quadratic hierarchical B-spline space with 5 levels and balance parameter $n_{fsb} = 1$. Level 1 is the coarsest level, with $16 \times 16 \times 16$ elements in E^1, and level 5 is the finest, with $256 \times 256 \times 256$ elements in E^5. The size of the tissue domain is $2000 \times 2000 \times 2000$ µm.

along the cross-section radial direction.

We have approximated the initial tumor geometry to an ellipsoidal mass placed in the left posterior region of the prostate. We recursively refine the extracted prostate mesh in order to adapt to this initial tumor by following the approach described in Section 8.5. The values of the model parameters for this simulation are summarized in Table 2.

Patient-specific, tissue scale prostate cancer growth can be solved using a single-level NURBS with 256 elements in the toroidal direction, 256 elements in the cross-section circumferential direction, and 64 elements

33

Table 2: List of parameters in the model of prostate cancer growth: patient-specific, tissue-scale simulation.

Parameter	Notation	Value
Diffusivity of the phase field	λ	175 mm^2/year
Time scale for the phase field	τ	0.01 years
Nutrient-induced tumor growth rate	χ	550 l/(g·year)
Apoptosis rate	A	600 1/year
Nutrient diffusivity	ϵ	$5.47 \cdot 10^3$ mm^2/year
Nutrient supply	s	2.70 g/(l·day)
Nutrient consumption rate	δ	2.75 g/(l·day)
Nutrient natural decay rate	γ	1000 1/year
Tissue PSA diffusivity	η	175 mm^2/year
Tissue PSA production rate in healthy tissue	α_h	6.25 (ng/ml)/(cm^3·year)
Tissue PSA production rate in cancerous tissue	α_c	$\alpha_c = 15\alpha_h$
Tissue PSA natural decay rate	γ_p	100 1/year

605 in the cross-section radial direction. Therefore, we chose a quadratic hierarchical NURBS space with 4 levels. We place the previous single-level discretization in the highest level and the prostate mesh extracted from CT in level $\alpha = 1$. We set $n_{fsb} = 1$ again to attain higher accuracy with a reasonable distribution of elements across HE. We set $C_R = 0.1$ and $C_C = 0.0005$ for this simulation.

Fig. 15 and Fig. 16 show the evolution of the tumor within the prostate and the adaptive mesh from 610 an anterior and a posterior view, respectively. The initial tumor starts growing as a mass but, eventually, it undergoes a shape instability towards a fingered morphology that grows surrounding the urethra. With this new pattern of development, the tumor reaches more nutrients and escapes hypoxia, starvation, and, eventually, necrosis, which would hamper tumor growth.

The mesh efficiently adapts to the evolution of the tumor in a similar fashion as we described for the 615 square and cubic domains in Section 8.6.1 and Section 8.6.2, respectively. The finest elements from level $\alpha = 4$ are localized to the surroundings of the interface between tumoral and healthy tissue to accurately capture its evolution. The operation of balancing ensures a distribution of elements in coarser levels that guarantees reduced overlapping of basis functions. These features can be observed in the adaptive meshes depicted in Fig. 15 and Fig. 16. Notice also that during the simulation coarser elements from levels $\alpha = 1$ 620 and $\alpha = 2$ cover the majority of the volume of the prostate. As a result, we dramatically reduce the number of elements with respect to the reference single-level NURBS mesh. For instance, at $t = 0.3$ years n_e is reduced to 6.6%, and it stays below 15% until $t = 0.55$ years. Consequently, our adaptive algorithms also reduce the computational resources and time required to run patient-specific, tissue-scale simulations of
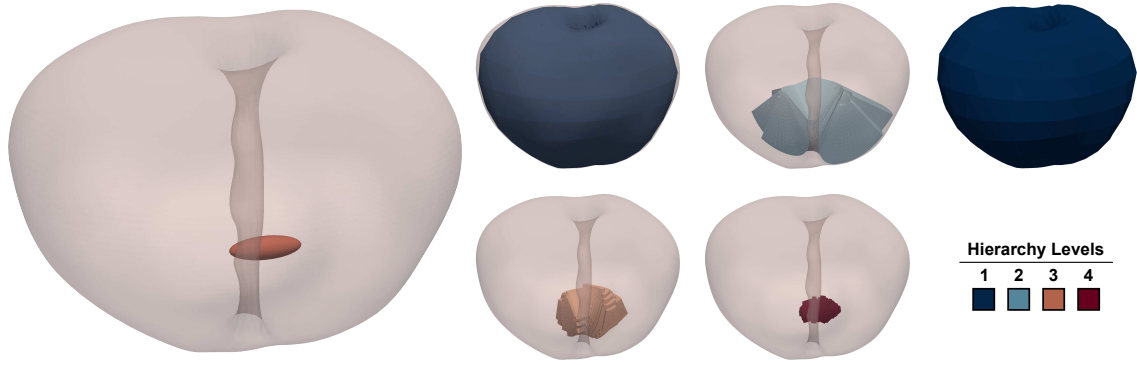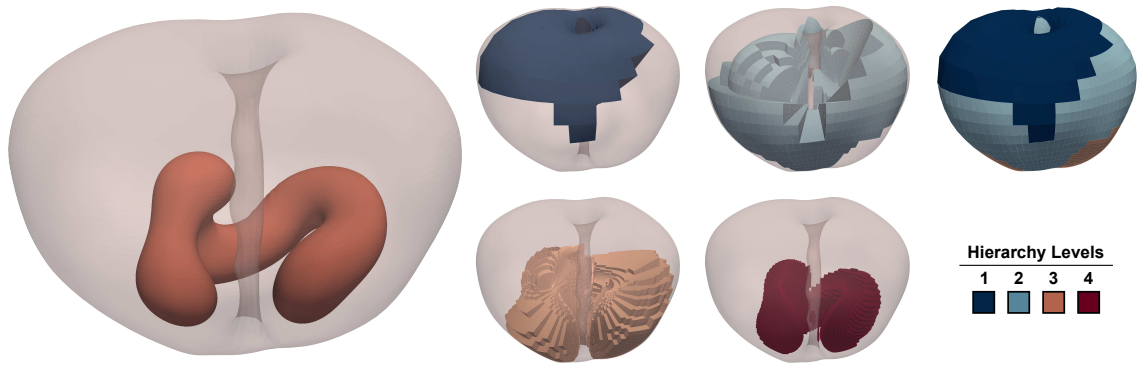
34

(a) Tumor, $t = 0$ years

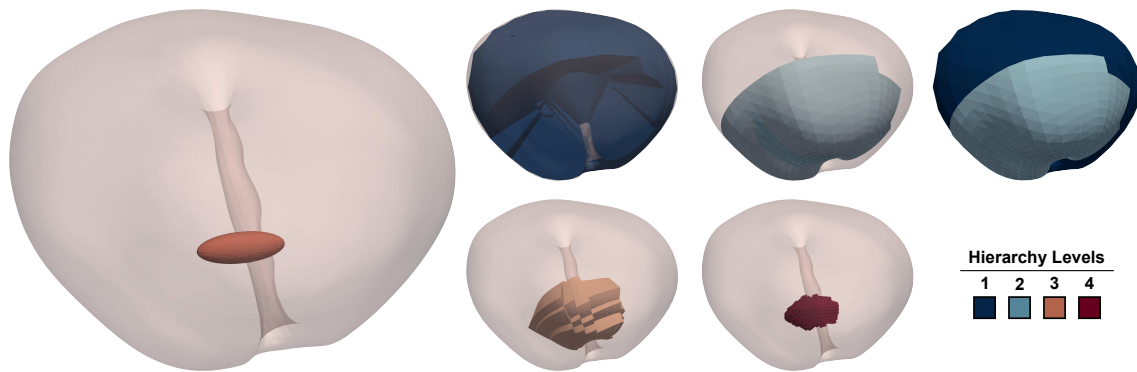(b) HE, $t = 0$ years
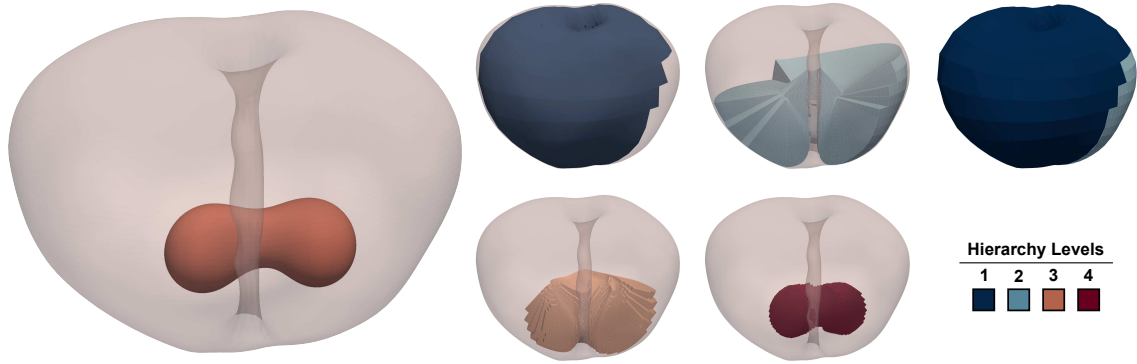
(c) Tumor, $t = 0.3$ years

(d) HE, $t = 0.3$ years
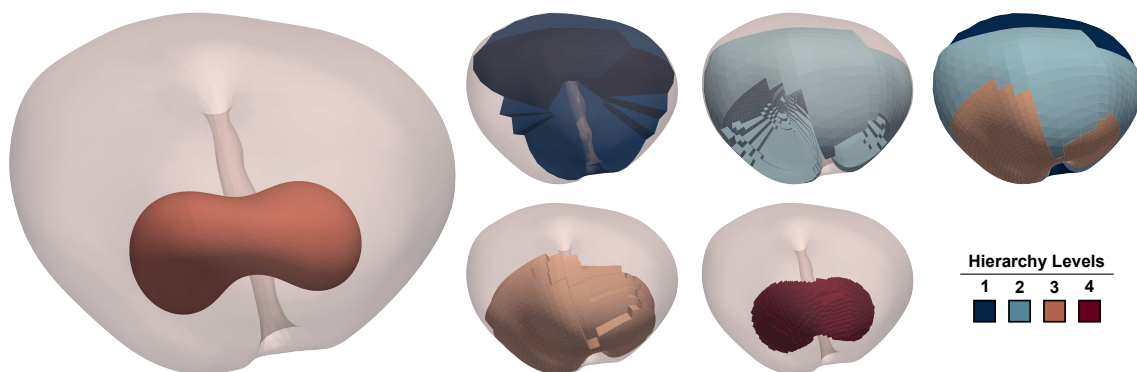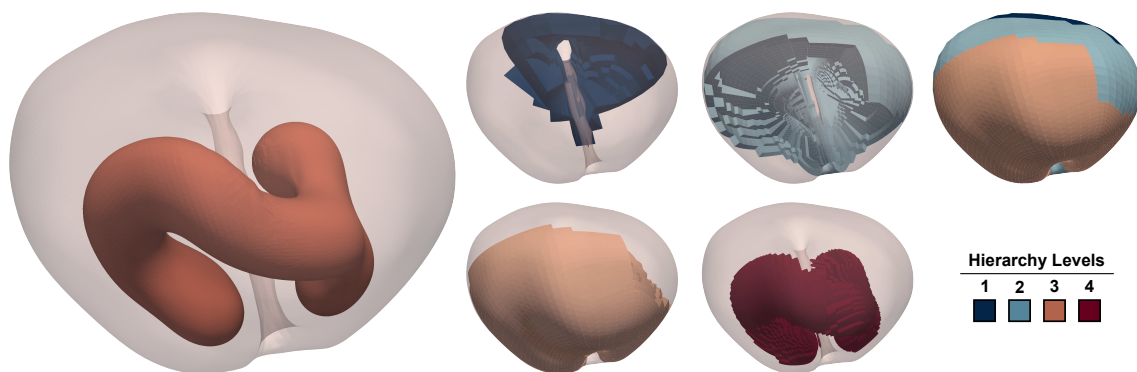
(e) Tumor, $t = 0.6$ years

(f) HE, $t = 0.6$ years

Figure 15: Patient-specific, tissue scale simulation of prostate cancer growth (anterior view). We used a quadratic hierarchical NURBS space with 4 levels and balance parameter $n_{fsb} = 1$. The volume of the prostate is 35.8 cm^3.

(a) Tumor, $t = 0$ years   (b) HE, $t = 0$ years

(c) Tumor, $t = 0.3$ years   (d) HE, $t = 0.3$ years

(e) Tumor, $t = 0.6$ years   (f) HE, $t = 0.6$ years

Figure 16: Patient-specific, tissue scale simulation of prostate cancer growth (posterior view). We used a quadratic hierarchical NURBS space with 4 levels and balance parameter $n_{fsb} = 1$. The volume of the prostate is 35.8 cm^3.

prostate cancer growth.

These advantages are of much interest in view of implementing our model for prostate cancer growth in the clinic. Physicians could use our model to predict the personalized prognosis of a patient and hence determine the best course of clinical management, from scheduling new revisions to starting a radical treatment. Our adaptive algorithms offer an efficient and accurate way to solve the cancer growth problem in a reasonable time. Thus, we believe that our modeling and simulation technology could contribute to improve the medical management of prostate cancer, from diagnosis to treatment and follow-up [30].

9. Summary and conclusions

In this paper, we have presented a set of efficient algorithms to accommodate local h-refinement and h-coarsening of hierarchical B-splines and NURBS in IGA. Our methods are based on Bézier projection, which we have extended to hierarchical spline spaces. We have also introduced the function support balance parameter in order to control the overlapping of functions across the levels of the hierarchy.

The technology presented herein enables the resolution of demanding problems of science and engineering by means of IGA and leveraging hierarchical spline spaces within a finite element framework provided by Bézier extraction. In particular, we have focused in the modeling and simulation of localized prostate cancer growth. We have used a model that we have presented in our previous work [30] in order to demonstrate the analysis potential of our adaptive strategies for hierarchical splines. In all the simulations presented in this paper we obtained remarkable accuracy with very few degrees of freedom in comparison to the uniform mesh that the same simulation would require.

The simulation of patient-specific, tissue-scale prostate cancer growth is a challenging problem. It features a moving interface that indicates the boundary between the tumor and the neighboring healthy tissue. This interface, modeled by means of the phase field method, evolves in the complex anatomy of the patient's prostate driven by nutrients and hormones. Consequently, this is a problem where and adaptive approach introduces a key advantage: it provides a mesh with finer elements to support the interface an coarser elements elsewhere, which leads to dramatic savings of memory and computational time. Hence, we argue that our adaptive algorithms are a cornerstone for the implementation of predictive models of localized prostate cancer growth on a clinical scenario in an efficient and accurate way.

In future work, we plan to test the technology presented herein with more complex models of prostate cancer growth. These involve several tumoral cell species, which means that new interfaces between them must be accurately represented. We also plan to leverage our adaptive strategies in order to test the accuracy of our models using patient-specific anatomical and pathological data extracted from medical images, in particular, from multiparametric magnetic resonance images (mpMRI [100, 101]). Moreover, we will consider p-, k-, and r-adaptivity of hierarchical spline spaces and the extension of our adaptive

37

algorithms to the spline forest setting [43]. We believe that our ongoing lines of research will lead to the development of new design-through-analysis technologies, hence opening the door to the resolution of other demanding problems in complex geometries.

References

[1] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (39-41) (2005) 4135–4195. doi:10.1016/j.cma.2004.10.008.

[2] J. A. Cottrell, T. J. Hughes, Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA, Wiley, 2009.

[3] H. Gomez, L. D. Lorenzis, The variational collocation method, Computer Methods in Applied Mechanics and Engineering 309 (2016) 152–181. doi:10.1016/j.cma.2016.06.003.

[4] 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 (1-5) (2011) 15–47. doi:10.1002/nme.2968.

[5] 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 (2) (2011) 126–156. doi:10.1002/nme.3167.

[6] J. Cottrell, A. Reali, Y. Bazilevs, T. Hughes, Isogeometric analysis of structural vibrations, Computer Methods in Applied Mechanics and Engineering 195 (41-43) (2006) 5257–5296. doi:10.1016/j.cma.2005.09.027.

[7] J. Kiendl, K.-U. Bletzinger, J. Linhard, R. Wüchner, Isogeometric shell analysis with Kirchhoff-Love elements, Computer Methods in Applied Mechanics and Engineering 198 (49-52) (2009) 3902–3914. doi:10.1016/j.cma.2009.08.013.

[8] D. Benson, Y. Bazilevs, M. Hsu, T. Hughes, Isogeometric shell analysis: the Reissner-Mindlin shell, Computer Methods in Applied Mechanics and Engineering 199 (5-8) (2010) 276–289, computational Geometry and Analysis. doi:10.1016/j.cma.2009.05.011.

[9] L. De Lorenzis, I. Temizer, P. Wriggers, G. Zavarise, A large deformation frictional contact formulation using NURBS-based isogeometric analysis, International Journal for Numerical Methods in Engineering 87 (13) (2011) 1278–1300. doi:10.1002/nme.3159.

[10] Y. Bazilevs, V. Calo, J. Cottrell, T. Hughes, A. Reali, G. Scovazzi, Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows, Computer Methods in Applied Mechanics and Engineering 197 (1-4) (2007) 173–201. doi:10.1016/j.cma.2007.07.016.

[11] Y. Bazilevs, C. Michler, V. Calo, T. Hughes, Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes, Computer Methods in Applied Mechanics and Engineering 199 (13-16) (2010) 780–790. doi:10.1016/j.cma.2008.11.020.

[12] Y. Bazilevs, I. Akkerman, Large eddy simulation of turbulent Taylor-Couette flow using isogeometric analysis and the residual-based variational multiscale method, Journal of Computational Physics 229 (9) (2010) 3402–3414. doi:10.1016/j.jcp.2010.01.008.

[13] Y. Bazilevs, V. M. Calo, T. J. R. Hughes, Y. Zhang, Isogeometric fluid-structure interaction: theory, algorithms, and computations, Computational Mechanics 43 (1) (2008) 3–37. doi:10.1007/s00466-008-0315-x.

[14] J. Bueno, C. Bona-Casas, Y. Bazilevs, H. Gomez, Interaction of complex fluids and solids: theory, algorithms and application to phase-change-driven implosion, Computational Mechanics 55 (6) (2015) 1105–1118. doi:10.1007/s00466-014-1098-x.

[15] H. Casquero, C. Bona-Casas, H. Gomez, A NURBS-based immersed methodology for fluid-structure interaction, Computer Methods in Applied Mechanics and Engineering 284 (2015) 943–970. doi:10.1016/j.cma.2014.10.055.

[16] H. Casquero, L. Liu, Y. Zhang, A. Reali, H. Gomez, Isogeometric collocation using analysis-suitable T-splines of arbitrary degree, Computer Methods in Applied Mechanics and Engineering 301 (2016) 164–186. doi:10.1016/j.cma.2015.12.014.

[17] 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 (1) (2011) 115–134. doi:10.1002/nme.3150.

[18] 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 (1-5) (2011) 336–360. doi:10.1002/nme.3061.

[19] M. J. Borden, C. V. Verhoosel, M. A. Scott, T. J. Hughes, C. M. Landis, A phase-field description of dynamic brittle fracture, Computer Methods in Applied Mechanics and Engineering 217-220 (2012) 77–95. doi:10.1016/j.cma.2012.01.008.

[20] H. Gómez, V. M. Calo, Y. Bazilevs, T. J. Hughes, Isogeometric analysis of the Cahn-Hilliard phase-field model, Computer Methods in Applied Mechanics and Engineering 197 (49-50) (2008) 4333–4352. doi:10.1016/j.cma.2008.05.003.

[21] H. Gomez, T. J. Hughes, X. Nogueira, V. M. Calo, Isogeometric analysis of the isothermal Navier-Stokes-Korteweg equations, Computer Methods in Applied Mechanics and Engineering 199 (25-28) (2010) 1828–1840. doi:10.1016/j.cma.2010.02.010.

[22] W. A. Wall, M. A. Frenzel, C. Cyron, Isogeometric structural shape optimization, Computer Methods in Applied Mechanics and Engineering 197 (33-40) (2008) 2976–2988. doi:10.1016/j.cma.2008.01.025.

[23] A. Buffa, G. Sangalli, R. Vázquez, Isogeometric analysis in electromagnetics: B-splines approximation, Computer Methods in Applied Mechanics and Engineering 199 (17-20) (2010) 1143–1152. doi:10.1016/j.cma.2009.12.002.

[24] Y. Bazilevs, V. M. Calo, Y. Zhang, T. J. R. Hughes, Isogeometric fluid–structure interaction analysis with applications to arterial blood flow, Computational Mechanics 38 (4) (2006) 310–322. doi:10.1007/s00466-006-0084-3.

[25] Y. Bazilevs, J. Gohean, T. Hughes, R. Moser, Y. Zhang, Patient-specific isogeometric fluid-structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device, Computer Methods in Applied Mechanics and Engineering 198 (45-46) (2009) 3534–3550. doi:10.1016/j.cma.2009.04.015.

[26] G. Vilanova, I. Colominas, H. Gomez, Capillary networks in tumor angiogenesis: from discrete endothelial cells to phase-field averaged descriptions via isogeometric analysis, International Journal for Numerical Methods in Biomedical Engineering 29 (10) (2013) 1015–1037. doi:10.1002/cnm.2552.

[27] G. Vilanova, I. Colominas, H. Gomez, Coupling of discrete random walks and continuous modeling for three-dimensional

tumor-induced angiogenesis, Computational Mechanics 53 (3) (2014) 449–464. doi:10.1007/s00466-013-0958-0.

[28] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, M. C. H. Wu, J. Mineroff, A. Reali, Y. Bazilevs, M. S. Sacks, Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models, Computational Mechanics 55 (6) (2015) 1211–1225. doi:10.1007/s00466-015-1166-x.

[29] J. Xu, G. Vilanova, H. Gomez, A mathematical model coupling tumor growth and angiogenesis, PLoS ONE 11 (2) (2016) 1–20. doi:10.1371/journal.pone.0149422.

[30] G. Lorenzo, M. A. Scott, K. Tew, T. J. R. Hughes, Y. J. Zhang, L. Liu, G. Vilanova, H. Gomez, Tissue-scale, personalized modeling and simulation of prostate cancer growth, Submitted for publication.

[31] D. Thomas, M. Scott, J. Evans, K. Tew, E. 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, Computer Methods in Applied Mechanics and Engineering 284 (2015) 55–105. doi:10.1016/j.cma.2014.07.014.

[32] L. A. Piegl, W. Tiller, The NURBS Book (Monographs in Visual Communication), Springer, 1996.

[33] D. F. Rogers, An Introduction to NURBS: With Historical Perspective (The Morgan Kaufmann Series in Computer Graphics), Morgan Kaufmann, 2000.

[34] G. Farin, Curves and Surfaces for CAGD, Fifth Edition: A Practical Guide (The Morgan Kaufmann Series in Computer Graphics), Morgan Kaufmann, 2001.

[35] A.-V. 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. doi:10.1016/j.cma.2011.09.004.

[36] 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. doi:10.1016/j.cma.2011.08.002.

[37] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, T. J. 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. doi:10.1016/j.cma.2012.03.017.

[38] 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. doi:10.1016/j.cma.2013.11.026.

[39] C. Giannelli, B. Jüttler, H. Speleers, THB-splines: The truncated basis for hierarchical splines, Computer Aided Geometric Design 29 (7) (2012) 485–498. doi:10.1016/j.cagd.2012.03.025.

[40] 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. L. Schumaker (Eds.), Mathematical Methods for Curves and Surfaces: 8th International Conference, MMCS 2012, Oslo, Norway, June 28 – July 3, 2012, Revised Selected Papers, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 304–323. doi:10.1007/978-3-642-54382-1_18.

[41] C. Giannelli, B. Jüttler, H. Speleers, Strongly stable bases for adaptively refined multilevel spline spaces, Advances in Computational Mathematics 40 (2) (2014) 459–490. doi:10.1007/s10444-013-9315-2.

[42] P. Hennig, S. Müller, M. Kästner, Bézier extraction and adaptive refinement of truncated hierarchical NURBS, Computer Methods in Applied Mechanics and Engineering 305 (2016) 316–339. doi:10.1016/j.cma.2016.03.009.

[43] M. Scott, D. Thomas, E. Evans, Isogeometric spline forests, Computer Methods in Applied Mechanics and Engineering 269 (2014) 222–264. doi:10.1016/j.cma.2013.10.024.

[44] T. W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, ACM Trans. Graph. 22 (3) (2003) 477–484. doi:10.1145/882262.882295.

[45] X. Li, J. Zheng, T. W. Sederberg, T. J. Hughes, M. A. Scott, On linear independence of T-spline blending functions,

780 Computer Aided Geometric Design 29 (1) (2012) 63–76. `doi:10.1016/j.cagd.2011.08.005`.

[46] M. Scott, X. Li, T. Sederberg, T. Hughes, Local refinement of analysis-suitable T-splines, Computer Methods in Applied Mechanics and Engineering 213-216 (2012) 206–222. `doi:10.1016/j.cma.2011.11.022`.

[47] 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. `doi:10.1016/j.cma.2012.02.025`.

785 [48] L. Beirão 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. `doi:10.1142/S0218202513500231`.

[49] X. Li, M. A. Scott, Analysis-suitable T-splines: Characterization, refineability, and approximation, Mathematical Models and Methods in Applied Sciences 24 (06) (2014) 1141–1164. `doi:10.1142/S0218202513500796`.

790 [50] H. Casquero, L. Liu, C. Bona-Casas, Y. Zhang, H. Gomez, A hybrid variational-collocation immersed method for fluid-structure interaction using unstructured T-splines, International Journal for Numerical Methods in Engineering 105 (11) (2016) 855–880. `doi:10.1002/nme.5004`.

[51] E. Evans, M. Scott, X. Li, D. Thomas, Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 284 (2015) 1–20.
795 `doi:10.1016/j.cma.2014.05.019`.

[52] X. Li, J. Deng, F. Chen, Dimensions of spline spaces over 3D hierarchical T-meshes, Journal of Information and Computational Science 3 (487-501) (2006) 110.

[53] X. Li, J. Deng, F. Chen, Surface modeling with polynomial splines over hierarchical T-meshes, The Visual Computer 23 (12) (2007) 1027–1033. `doi:10.1007/s00371-007-0170-3`.

800 [54] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, Y. Feng, Polynomial splines over hierarchical T-meshes, Graphical Models 70 (4) (2008) 76–86. `doi:10.1016/j.gmod.2008.03.001`.

[55] X. Li, J. Deng, F. Chen, Polynomial splines over general T-meshes, The Visual Computer 26 (4) (2010) 277–286. `doi:10.1007/s00371-009-0410-9`.

[56] H. Kang, F. Chen, J. Deng, Modified T-splines, Computer Aided Geometric Design 30 (9) (2013) 827–843. `doi:10.1016/`
805 `j.cagd.2013.09.001`.

[57] T. Dokken, T. Lyche, K. F. Pettersen, Polynomial splines over locally refined box-partitions, Computer Aided Geometric Design 30 (3) (2013) 331–356. `doi:10.1016/j.cagd.2012.12.005`.

[58] A. Bressan, Some properties of LR-splines, Computer Aided Geometric Design 30 (8) (2013) 778–794. `doi:10.1016/j.cagd.2013.06.004`.

810 [59] C. Bracco, D. Berdinsky, D. Cho, M. jae Oh, T. wan Kim, Trigonometric generalized T-splines, Computer Methods in Applied Mechanics and Engineering 268 (2014) 540–556. `doi:10.1016/j.cma.2013.09.015`.

[60] P. C. Walsh, J. F. Worthington, Dr. Patrick Walsh's Guide to Surviving Prostate Cancer, Second Edition, Grand Central Life & Style, 2007.

[61] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, P. Walter, Molecular Biology of the Cell, 5th Edition, Garland
815 Science, 2007.

[62] C. A. Taylor, M. T. Draney, J. P. Ku, D. Parker, B. N. Steele, K. Wang, C. K. Zarins, Predictive medicine: Computational techniques in therapeutic decision-making, Computer Aided Surgery 4 (5) (1999) 231–247. `doi:10.3109/10929089909148176`.

[63] C. Taylor, C. Figueroa, Patient-specific modeling of cardiovascular mechanics, Annual Review of Biomedical Engineering
820 11 (1) (2009) 109–134. `doi:10.1146/annurev.bioeng.10.061807.160521`.

[64] S. Sanga, H. B. Frieboes, X. Zheng, R. Gatenby, E. L. Bearer, V. Cristini, Predictive oncology: a review of multidisciplinary, multiscale in silico modeling linking phenotype, morphology and growth, NeuroImage 37, Supplement 1 (2007)

S120–S134. doi:10.1016/j.neuroimage.2007.05.043.

[65] R. A. Gatenby, P. K. Maini, Mathematical oncology: cancer summed up, Nature 421 (6921) (2003) 321–321. doi:10.1038/421321a.

[66] A. R. Anderson, V. Quaranta, Integrative mathematical oncology, Nature Reviews Cancer 8 (3) (2008) 227–234. doi:10.1038/nrc2329.

[67] A. H. Chauviere, H. Hatzikirou, J. S. Lowengrub, H. B. Frieboes, A. M. Thompson, V. Cristini, Mathematical oncology: how are the mathematical and physical sciences contributing to the war on breast cancer?, Current Breast Cancer Reports 2 (3) (2010) 121–129. doi:10.1007/s12609-010-0020-6.

[68] E. A. B. F. Lima, J. T. Oden, R. C. Almeida, A hybrid ten-species phase-field model of tumor growth, Mathematical Models and Methods in Applied Sciences 24 (13) (2014) 2569–2599. doi:10.1142/S0218202514500304.

[69] A. Hawkins-Daarud, K. G. van der Zee, J. Tinsley Oden, Numerical simulation of a thermodynamically consistent four-species tumor growth model, International Journal for Numerical Methods in Biomedical Engineering 28 (1) (2012) 3–24. doi:10.1002/cnm.1467.

[70] R. D. Travasso, M. Castro, J. C. Oliveira, The phase-field model in tumor growth, Philosophical Magazine 91 (1) (2011) 183–206. doi:10.1080/14786435.2010.501771.

[71] H. B. Frieboes, J. S. Lowengrub, S. Wise, X. Zheng, P. Macklin, E. L. Bearer, V. Cristini, Computer simulation of glioma growth and morphology, NeuroImage 37, Supplement 1 (2007) S59 – S70. doi:10.1016/j.neuroimage.2007.03.008.

[72] S. Wise, J. Lowengrub, H. Frieboes, V. Cristini, Three-dimensional multispecies nonlinear tumor growth – I: Model and numerical method, Journal of Theoretical Biology 253 (3) (2008) 524–543. doi:10.1016/j.jtbi.2008.03.027.

[73] V. Cristini, X. Li, J. S. Lowengrub, S. M. Wise, Nonlinear simulations of solid tumor growth using a mixture model: invasion and branching, Journal of Mathematical Biology 58 (4-5) (2009) 723–763. doi:10.1007/s00285-008-0215-x.

[74] H. B. Frieboes, F. Jin, Y.-L. Chuang, S. M. Wise, J. S. Lowengrub, V. Cristini, Three-dimensional multispecies nonlinear tumor growth – II: Tumor invasion and angiogenesis, Journal of Theoretical Biology 264 (4) (2010) 1254–1278. doi:10.1016/j.jtbi.2010.02.036.

[75] E. L. Bearer, J. S. Lowengrub, H. B. Frieboes, Y.-L. Chuang, F. Jin, S. M. Wise, M. Ferrari, D. B. Agus, V. Cristini, Multiparameter computational modeling of tumor invasion, Cancer Research 69 (10) (2009) 4493–4501. doi:10.1158/0008-5472.CAN-08-3834.

[76] R. S. Qin, H. K. Bhadeshia, Phase field method, Materials Science and Technology 26 (7) (2010) 803–811. doi:10.1179/174328409X453190.

[77] H. Gomez, L. Cueto-Felgueroso, R. Juanes, Three-dimensional simulation of unstable gravity-driven infiltration of water into a porous medium, Journal of Computational Physics 238 (2013) 217–239. doi:10.1016/j.jcp.2012.12.018.

[78] A. Karma, A. E. Lobkovsky, Unsteady crack motion and branching in a phase-field model of brittle fracture, Phys. Rev. Lett. 92 (2004) 245510. doi:10.1103/PhysRevLett.92.245510.

[79] R. Qin, H. Bhadeshia, Phase-field model study of the crystal morphological evolution of hcp metals, Acta Materialia 57 (11) (2009) 3382–3390. doi:10.1016/j.actamat.2009.04.001.

[80] R. T. Farouki, The Bernstein polynomial basis: a centennial retrospective, Computer Aided Geometric Design 29 (6) (2012) 379–419. doi:10.1016/j.cagd.2012.03.001.

[81] R. T. Farouki, C. A. Neff, On the numerical condition of Bernstein-Bézier subdivision processes, Mathematics of Computation 55 (192) (1990) 637–647. doi:10.1090/S0025-5718-1990-1035933-0.

[82] E. Doha, A. Bhrawy, M. Saker, Integrals of Bernstein polynomials: An application for the solution of high even-order differential equations, Applied Mathematics Letters 24 (4) (2011) 559–565. doi:10.1016/j.aml.2010.11.013.

[83] B. Jüttler, The dual basis functions for the Bernstein polynomials, Advances in Computational Mathematics 8 (4) (1998) 345–352. doi:10.1023/A:1018912801267.

[84] M. G. Cox, The numerical evaluation of B-splines, IMA Journal of Applied Mathematics 10 (2) (1972) 134–149. `doi:` `10.1093/imamat/10.2.134.`

[85] C. De Boor, On calculating with B-splines, Journal of Approximation Theory 6 (1) (1972) 50–62. `doi:10.1016/` `0021-9045(72)90080-9.`

[86] T. J. R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis (Dover Civil and Mechanical Engineering), Dover Publications, 2000.

[87] K. Ellem, G. Kay, The nature of conditioning nutrients for human malignant melanoma cultures, Journal of Cell Science 62 (1) (1983) 249–266.
URL `http://jcs.biologists.org/content/62/1/249`

[88] P. Ciarletta, L. Foret, M. Ben Amar, The radial growth phase of malignant melanoma: multi-phase modelling, numerical simulations and linear stability analysis, Journal of The Royal Society Interface 8 (56) (2011) 345–368. `doi:10.1098/` `rsif.2010.0285.`

[89] V. Andasari, A. Gerisch, G. Lolas, A. P. South, M. A. J. Chaplain, Mathematical modeling of cancer cell invasion of tissue: biological insight from mathematical analysis and computational simulation, Journal of Mathematical Biology 63 (1) (2011) 141–171. `doi:10.1007/s00285-010-0369-1.`

[90] G. Sciumè, S. Shelton, W. Gray, C. Miller, F. Hussain, M. Ferrari, P. Decuzzi, B. Schrefler, A multiphase model for three-dimensional tumor growth, New Journal of Physics 15 (1) (2013) 015005. `doi:10.1088/1367-2630/15/1/015005.`

[91] W. Mueller-Klieser, J. Freyer, R. Sutherland, Influence of glucose and oxygen supply conditions on the oxygenation of multicellular spheroids, British journal of cancer 53 (3) (1986) 345–353. `doi:10.1038/bjc.1986.58.`

[92] R. R. Berges, J. Vukanovic, J. I. Epstein, M. CarMichel, L. Cisek, D. E. Johnson, R. W. Veltri, P. C. Walsh, J. T. Isaacs, Implication of cell kinetic changes during the progression of human prostatic cancer, Clinical Cancer Research 1 (5) (1995) 473–480.
URL `http://clincancerres.aacrjournals.org/content/1/5/473`

[93] J. Chung, G. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method, Journal of applied mechanics 60 (2) (1993) 371–375. `doi:10.1115/1.2900803.`

[94] K. E. Jansen, C. H. Whiting, G. M. Hulbert, A generalized-α method for integrating the filtered Navier–Stokes equations with a stabilized finite element method, Computer Methods in Applied Mechanics and Engineering 190 (3–4) (2000) 305–319. `doi:10.1016/S0045-7825(00)00203-6.`

[95] M. Heroux, R. Bartlett, V. Hoekstra, J. Hu, T. Kolda, R. Lehoucq, K. Long, R. Pawlowski, E. Phipps, A. Salinger, et al., An overview of Trilinos: Technical Report SAND2003-2927, Sandia National Laboratories, Albuquerque, NM.

[96] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 856–869. `doi:10.1137/0907058.`

[97] M. S. Floater, K. Hormann, Surface parameterization: a tutorial and survey, in: N. A. Dodgson, M. S. Floater, M. A. Sabin (Eds.), Advances in Multiresolution for Geometric Modelling, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 157–186. `doi:10.1007/3-540-26808-1_9.`

[98] Y. Zhang, G. Xu, C. Bajaj, Quality meshing of implicit solvation models of biomolecular structures, Computer Aided Geometric Design 23 (6) (2006) 510 – 530. `doi:10.1016/j.cagd.2006.01.008.`

[99] Y. Zhang, Y. Bazilevs, S. Goswami, C. L. Bajaj, T. J. Hughes, Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, Computer Methods in Applied Mechanics and Engineering 196 (29-30) (2007) 2943–2959. `doi:` `10.1016/j.cma.2007.02.009.`

[100] B. Turkbey, H. Mani, V. Shah, A. R. Rastinehad, M. Bernardo, T. Pohida, Y. Pang, D. Daar, C. Benjamin, Y. L. McKinney, H. Trivedi, C. Chua, G. Bratslavsky, J. H. Shih, W. M. Linehan, M. J. Merino, P. L. Choyke, P. A. Pinto, Multiparametric 3T prostate magnetic resonance imaging to detect cancer: histopathological correlation using prosta-

tectomy specimens processed in customized magnetic resonance imaging based molds, The Journal of Urology 186 (5) (2011) 1818–1824. `doi:10.1016/j.juro.2011.07.013`.

[101] R. Stoyanova, K. Huang, K. Sandler, H. Cho, S. Carlin, P. B. Zanzonico, J. A. Koutcher, E. Ackerstaff, Mapping tumor hypoxia in vivo using pattern recognition of dynamic contrast-enhanced MRI data, Translational Oncology 5 (6) (2012) 437–447. `doi:10.1593/tlo.12319`.

910