TECHNISCHE MECHANIK
an open access journal

journal homepage: www.ovgu.de/techmech

Tech. Mech., Vol. 41, Is. 1, (2021), 1–11

Received: 08.10.2020
Accepted: 11.01.2021
Available online: 12.01.2021

B-Spline meshing for high-order finite element analysis of multi-physics problems

Marek Werner[1]* and Kerstin Weinberg[1]

[1] Universität Siegen, Lehrstuhl für Festkörpermechanik, Department Maschinenbau, Paul-Bonatz-Str. 9-11, 57076 Siegen, Germany

Abstract: Multi-physics problems often involve differential equations of higher-order, which cannot be solved with standard finite element methods. B-splines as finite element basis functions provide the required continuity and smoothness. However, the mesh generation for arbitrarily shaped domains is non-intuitively and traditional techniques often lead to distorted elements. Here, isoparametric B-spline based meshes for curves, surfaces, and volumes are designed. The error of the homeomorphic transformation into curved boundaries is estimated. For selected two and three-dimensional shapes, the knot vectors and the control points are calculated. Exemplarily, a finite element analysis of a helical structure subjected to a chemo-mechanical deformation with phase decomposition illustrates the proposed strategy.

Keywords: multi-field computations, isogeometric analysis, high-order continuity, geometry approximation, phase decomposition

1 Introduction

Challenging engineering problems often involve weakly or strongly coupled fields, where mechanical parts have to account for changes in temperature, electric fields, phase separation, chemical potentials, and solid-fluid interaction. Numerical simulations of such problems often include partial differential equations (PDEs) of higher-order, e.g., mechanical deformations in combination with Cahn-Hilliard diffusion (Stein et al., 2017), Kuramoto-Sivashinsky flow interfaces (Anders et al., 2012a), or high-order topology-optimization and phase-field fracture (Borden et al., 2014; Dede et al., 2012; Hesch et al., 2016). From a numerical point of view, the solution of such PDEs can be achieved in different ways. The simplest way is the finite difference method with all their pros and cons, particularly their restriction to rectangular domains. This geometrical limitation can be overcome by the finite element method (FEM) whereby standard techniques require mixed formulations of the high-order terms which come with additional effort and, in part, unclear boundary constraints, cf. (Anders et al., 2012b). An efficient way to fulfill all continuity requirements of the FEM is the choice of sophisticated basis functions, e.g., non-rational basis spline functions (B-splines) or non-uniform rational B-spline functions (NURBS). These functions have become popular in the context of the Isogeometric Analysis (IGA), cf. (Cottrell et al., 2009). However, the focus of the IGA is on the geometry description, which should comply with shapes generated by computer-aided design. NURBS typically describe these models' geometry and may go along with a local C^0-continuity, which is sufficient to solve classical second-order PDEs. In opposite to this concept, we refer here to the weak formulation of high-order PDEs and apply splines as finite element basis functions. These basis functions offer adequate accuracy, robustness, and, most importantly, versatility in terms of higher-order continuity and smoothness.

For finite element computation, we consider a system or a body with domain Ω and local material points at $\boldsymbol{x} \in \Omega$, which evolve with time t. For (each component of) the unknown fields, we make an approximation of the form

$$u^h(\boldsymbol{x}, t) = \sum_{A \in \Omega} u_A(t) N_A(\boldsymbol{x}), \tag{1}$$

where u_A are the local values, and N_A characterize the respective basis functions. The set A denotes the degrees of freedom in the solution space $\mathcal{V}^h$. The required continuity of the solution $u^h \in \mathcal{V}^h \subset \mathcal{H}^{k+1}(\Omega)$ can only be fulfilled by means of basis functions N_A, which are piecewise smooth and globally at least C^k-continuous. Lagrangian finite element basis functions provide C^0-continuity only, an enhancement by other polynomials is extremely elaborate, cf. (Zienkiewicz and Taylor, 2000), and so spline functions are a natural choice as FEM basis functions of higher-order. They are defined recursively for a given degree (order) p and provide adjustable continuity up to C^{p-1}.

In opposite to the classical node-based FEM, the geometrical mesh of B-spline basis functions differs from the control parameter mesh. The latter is obtained by piecewise linear interpolation of the control points $\boldsymbol{P}_A \in \mathbb{R}^d$, d is the coupled problem's dimension. The geometrical mesh consists of patches and knot spans. This duality makes the mesh generation for arbitrarily shaped domains non-intuitively. In practice, the FEM mesh is often generated on a reference rectangle in two dimensions or a reference cuboid in three dimensions and then mapped to the real shape. This results in computational domains with patchy refinements and distorted elements, especially for circular domains or along holes. Figure 1 illustrates such a backward engineering, where the wanted eighth of a ball — a four-faced shape — is derived from a uniformly meshed six-faced cube. Clearly, this procedure gives wedge-shaped internal elements. Because the exploitation of symmetry is of limited use in multi-physics simulations, we prefer to mesh the whole ball instead.

* E-mail address: marek.werner@uni-siegen.de
doi: 10.24352/UB.OVGU-2021-001
2021 | All rights reserved.

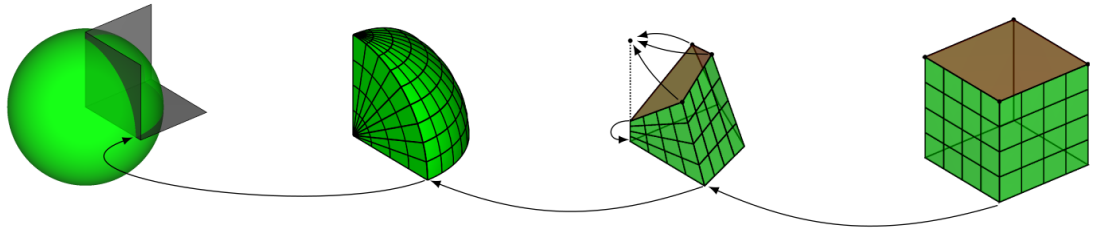

Fig. 1: Backward engineering of a sphere section out of a tensor-product meshed cube: two faces of the cube are reduced to two lines, one is curved. This procedure gives wedge-shaped internal elements.

All numerical simulation techniques come along with a certain error of the geometry approximation. Like stability and contact problems, some simulations are very sensitive against geometric imperfections, and different strategies have been developed to tackle these problems. For example, the NURBS based IGA provides a better geometry description (Predrag and Marinkovic, 2013). Also, surface reconstruction coupled with optimization algorithms for scattered 3D data points have been developed (Makhlouf et al., 2019), but often specific assumptions are made like closed lines (Zhao et al., 2011) coupled with periodic shape functions. The step to obtain 3D volumes from watertight trimmed surface patches is not trivial (Sederberg et al., 2008). Alternatively, geometry approximation may improve when knots are free variables subjected to an optimization problem (Gálvez and Iglesias, 2013; Dung and Tjahjowidodo, 2017); examples involve genetic algorithms (Gálvez et al., 2012) or level set optimization (Noël et al., 2020). However, such schemes are numerically very expensive, require specialized software, and are only feasible for a few specific problems. Moreover, in multi-field problems with moving boundaries, local deformations can change with time over the entire domain, which indeed differs from the initial domain, as obtained, e.g., by the optimization strategy proposed in (Xu et al., 2011). Adaptive strategies may be then applied to define the new knot positions, but they are sensitive to the variation of the curvature and have a high computational cost (Li et al., 2005). Here instead, we aim for a uniform initial mesh generation.

In this technical paper, we suggest a strategy to manually generate 'good' C^k-continuous meshes for basic geometrical forms, specifically for a circular domain, a cylinder, and a sphere. The meshes are good in the sense that distorted elements are avoided, the meshing is as uniform as possible and minimizes the geometrical discretization error in that way. For the finite element approximation, we employ B-splines, which are a special case of NURBS, as basis functions of the whole domain.

The paper is organized as follows: In the following Section 2, we introduce the B-spline basis functions shortly. Here we also outline the common technique of curve approximation. Section 3 is devoted to our proposed strategy for the isoparametric FEM meshing of curved domains; the geometrical discretization error is estimated, and selected approximations are described in detail. In Section 4, we present a numerical example, namely a deforming helical curve under chemo-mechanical loading. In the Appendix, the control point positions for the presented meshes are listed.

2 Basics

We start considering a one-dimensional unit-length domain and introduce a local coordinate $\xi \in [0, 1]$. The set of non-decreasing discrete values $\{\xi_0, \xi_1, \ldots, \xi_{N-1}\} := \Xi$ is called a knot vector; ξ_i is the ith knot, and $\xi_0 = 0$, $\xi_{N-1} = 1$. A knot may repeat with multiplicity m. The unique subset $\Xi^u = \{\xi_0^u, \xi_1^u, ..., \xi_{N_u-1}^u\} \subseteq \Xi$ splits the domain into intervals (elements) and represents the element corners, whereby the knots do not have to be equidistant.

Between two knots ξ_i and ξ_{i+1}, a piecewise-constant shape function of order zero is defined as an indicator function.

$$\mathcal{N}_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \le \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in \{0, \ldots, N-1\} \tag{2}$$

Univariate B-splines of order $p > 0$ are defined recursively by the Cox-de Boor formula (de Boor, 1972; Cox, 1972)

$$\mathcal{N}_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} \mathcal{N}_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} \mathcal{N}_{i+1,p-1}(\xi) \qquad \forall i \in \{0, \ldots, n-1\} \tag{3}$$

and terminated with condition (2); a ratio of the form $0/0$ is defined as zero. The number of knots N, the number of basis functions n, and the polynomial order p are related by $N = n + 1 + p$. The number of unique knots N_u can only be determined for specific cases.
B-spline functions are linearly independent and establish a partition of unity,

$$\sum_{i=0}^{n-1} \mathcal{N}_{i,p}(\xi) = 1 \qquad \forall \xi \in [0, 1]\,. \tag{4}$$

The support of each B-spline $\mathcal{N}_{i,p}$ is compact and contained in the interval $\left[\xi_i, \xi_{i+p+1}\right]$. Additionally, each spline is a non-negative function, i.e., $\mathcal{N}_{i,p}(\xi) \ge 0$, for all ξ. From the partition of unity follows $\mathbf{0} = \sum_i \nabla \mathcal{N}_{i,p}$.

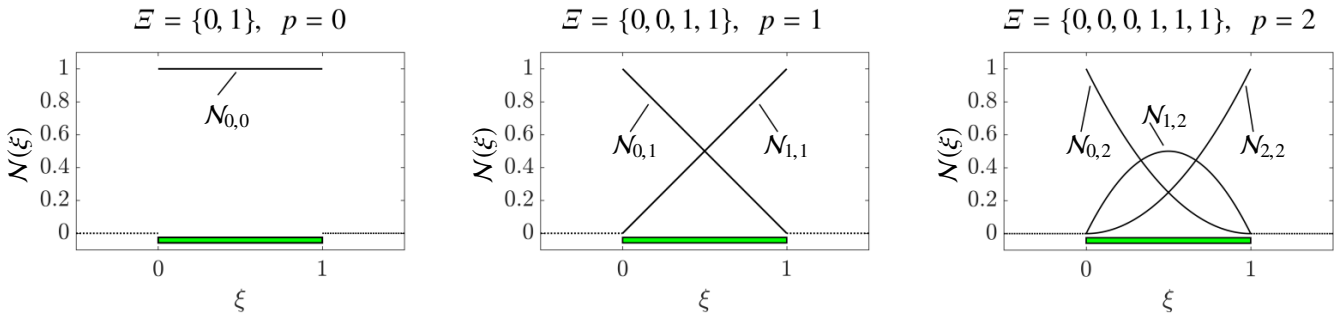

Fig. 2: Order elevation: Indicator function for knot vector $\Xi = \{0, 1\}$ (left). Order elevation from $\Xi = \{0, 1\}$ to $\Xi = \{0, 0, 1, 1\}$ together with linear basis functions $N_{0,1}$ and $N_{1,1}$ (middle). A second elevation from $\Xi = \{0, 0, 1, 1\}$ to $\Xi = \{0, 0, 0, 1, 1, 1\}$ gives the three quadratic basis functions $N_{0,2}$, $N_{1,2}$, and $N_{2,2}$ (right).

2.1 Finite element basis functions

In one dimension, the B-splines directly serve as basis functions, $N_A := N_{i,p}$. For a knot vector Ξ^u without internal multiplicity and order $p = 1$, the B-splines coincide with piecewise linear finite element basis functions, which are C^0-continuous. For higher-order, $p \geq 2$, B-splines' shapes and properties differ significantly from standard finite element basis functions. For example, the quadratic B-Splines $N_{i,2}$ are globally smooth and C^1-continuous. Each quadratic B-Spline spans over four knots.

For two- and three-dimensional domains, multivariate B-spline functions can be derived by tensor products of univariate functions. The shape functions in two dimensions read in the two directions 1 and 2 with corresponding coordinates $\xi^{(1)}$ and $\xi^{(2)}$

$$N_A(\xi^{(1)}, \xi^{(2)}) := N_{i_1,p_1}(\xi^{(1)}) \cdot N_{i_2,p_2}(\xi^{(2)}) \quad \text{with } A = i_1 + i_2 n_1 \,, \tag{5}$$

and in three dimensions

$$N_A(\xi^{(1)}, \xi^{(2)}, \xi^{(3)}) := N_{i_1,p_2}(\xi^{(1)}) \cdot N_{i_2,p_2}(\xi^{(2)}) \cdot N_{i_3,p_3}(\xi^{(3)}) \quad \text{with } A = i_1 + i_2 n_1 + i_3 n_1 n_2 \,, \tag{6}$$

where the functions $N_{i_j,p}$ are defined analogously by the Cox-de Boor recursion formula (3) for the corresponding polynomial order p and n_j control points in the $j \in \{1, 2, 3\}$ direction. The equations illustrate that the global index A of our shape functions N_A is directly associated with the control mesh's indices i_1, i_2, i_3.

2.2 Meshing and curve approximation

A linear combination of univariate B-spline basis functions $N_{i,p}$ approximates a spatial curve $\boldsymbol{C}$ by

$$\boldsymbol{C}^h(\xi) = \sum_{i=0}^{n-1} N_{i,p}(\xi) \boldsymbol{P}_i \,, \tag{7}$$

where the control points $\boldsymbol{P}_i$ do not depend on ξ and typically do not lie on the curve. The finite element approximation (7) is isoparametric in the sense that the same basis functions $N_{i,p}$ are used for both description of geometry and approximation of unknown field vales u^h in (1).

For finite element meshing, let us start with a one-element straight line and two knots $\Xi = \{0, 1\}$ for simplicity. This knot vector is sufficient to define the indicator function $N_{0,0}$ of Eq. (2), see Fig. 2 (left), but not sufficient to define higher-order basis functions. A continuous finite element mesh requires shape functions of at least $p = 1$, which can be obtained by *order elevation*. To do so, the multiplicity of all knots in Ξ is increased by one, which gives $\Xi = \{0, 0, 1, 1\}$. With recursion formula (3), this results in the linear shape functions $N_{0,1}$ and $N_{1,1}$ of Fig. 2 (mid). Another order elevation from $\Xi = \{0, 0, 1, 1\}$ to $\Xi = \{0, 0, 0, 1, 1, 1\}$ raises the order to $p = 2$ and results in the quadratic shape functions $N_{0,2}$, $N_{1,2}$, and $N_{2,2}$, see Fig. 2 (right).

A mesh refinement, i.e., additional finite elements, can now be designed by *knot insertion*. With $\xi_i < \xi_j < \xi_{i+1}$, a new[1] position ξ_j is inserted in the knot vector Ξ and splits the corresponding interval. Each inserted knot adds a shape function of the current order p. In our example, starting with the quadratic shape functions $N_{0,2}, N_{1,2}$ and $N_{2,2}$ and inserting a knot at $\xi_j = 1/3$ gives a new quadratic shape function $N_{3,2}$; see Fig. 5 (mid) and Fig. 4. However, this additional knot also modifies the former shape functions since they are defined recursively via formula (3). The modification is repeated with every knot insertion; see Fig. 5 (right).

By using both the knot insertion and the order elevation technique, the three refinement strategies (h, p, k) are possible: h-refinement maintains the order and constructs a finer mesh, i.e., element size h is reduced; p-refinement maintains the mesh and increases the order, i.e., p is increased; k-refinement refines the mesh and elevates the order (first h is reduced and then p is increased). We emphasize that order elevation and knot insertion do not commute.

[1] Technically, the 'new' position can coincide with an existing knot to generate a kink but we aim for a continuous differentiable geometry description here.

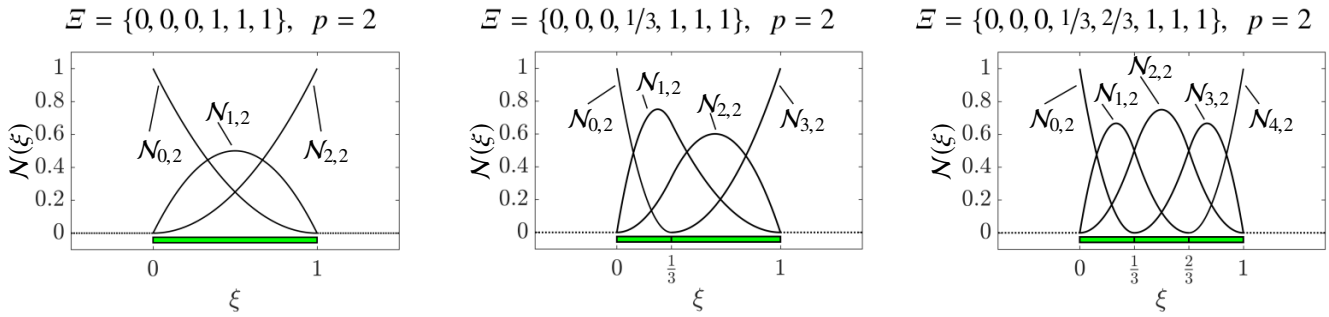

Fig. 3: Knot insertion: Open knot system with $\Xi = \{0, 0, 0, 1, 1, 1\}$ and quadratic basis functions (left). Insertion of a knot at $\xi_3 = 1/3$ (middle) and at $\xi_4 = 2/3$ (right) and corresponding quadratic shape functions $N_{i,p=2}$. The elements are indicated by black markers in the green stripe below the shape functions.

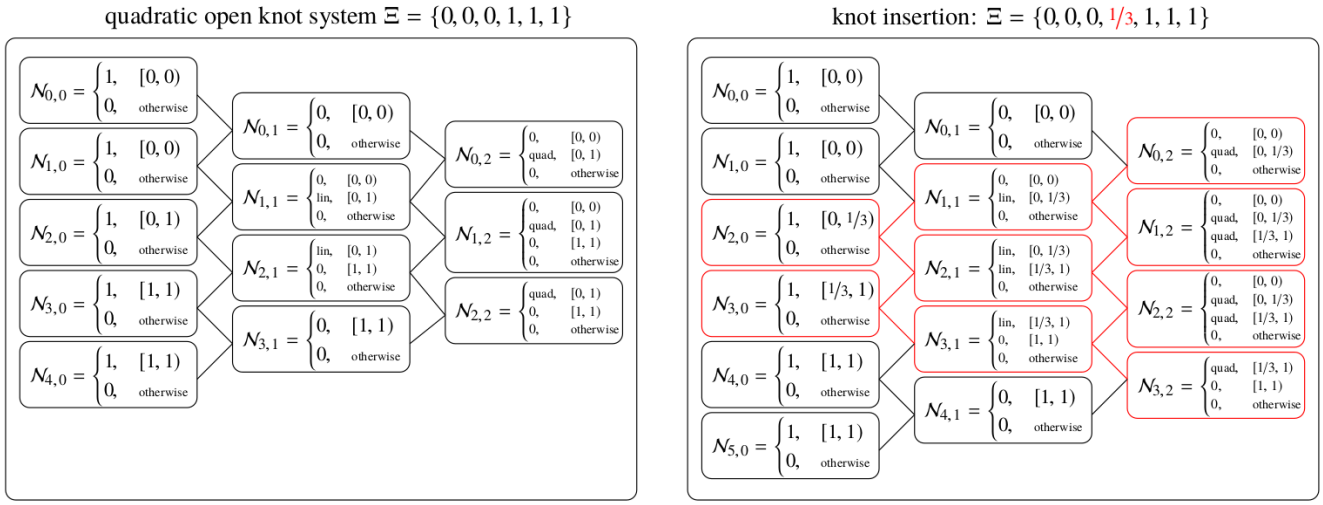

Fig. 4: Modification of basis functions due to knot insertion at $\xi = 1/3$

We can now specify the number of unique knots to be $N_u = n - 1$, or equivalently $N_u = N - 2 - p$. For N_u equidistant unique knots, we obtain a uniformly partitioned domain with elements of length $1/(N_u - 1)$. For repeated knots with multiplicity m, the continuity reduces to C^{p-m} at that knot. If the first and the last knot repeat $m = p + 1$ times, then the system is called an *open knot system*. Such open knot systems are required to define boundary values of u^h.

Traditionally, during knot insertion, the shape of the parameterized geometry is not modified. New control points at the edges are defined as $\boldsymbol{P}_0^{\text{new}} = \boldsymbol{P}_0$ and $\boldsymbol{P}_n^{\text{new}} = \boldsymbol{P}_{n-1}$, and the intermediate control points are calculated by the convex combination

$$\boldsymbol{P}_i^{\text{new}} = \alpha_i \boldsymbol{P}_i + (1 - \alpha_i)\boldsymbol{P}_{i-1} \qquad 1 \le i \le n - 1 \tag{8}$$

where the weight $\alpha_i = (\xi_j - \xi_i)/(\xi_{i+1} - \xi_i)$ is constructed from the distances of the newly inserted knot ξ_j to its nearest neighbors ξ_i and ξ_{i+1}. In that way, the underlying geometry does not change during refinement. That is, for a linear connection $P_0 - P_1$ and order $p = 1$, all further refined control points lie on the straight line or, as depicted in Fig. 6, for a bow described by three control points and $p = 2$, the refined mesh has the same curvature.

Independent of the geometry, a new set of control points arises after the knot insertion process, and so a different set of unknowns is used for the refined mesh. Therefore a spline approximation is neither hierarchical nor refers to previous nodal positions like in Lagrangian finite elements.

3 Isoparametric approximation of a curved geometry

This section will outline a strategy to generate the B-spline mesh for a given analytically described boundary geometry. For simplicity, we suggest to define continuity requirements and the order of approximation at first. Here, we will mostly refer to C^1-continuous quadratic B-spline meshes, but the outlined strategy also works for other demands.

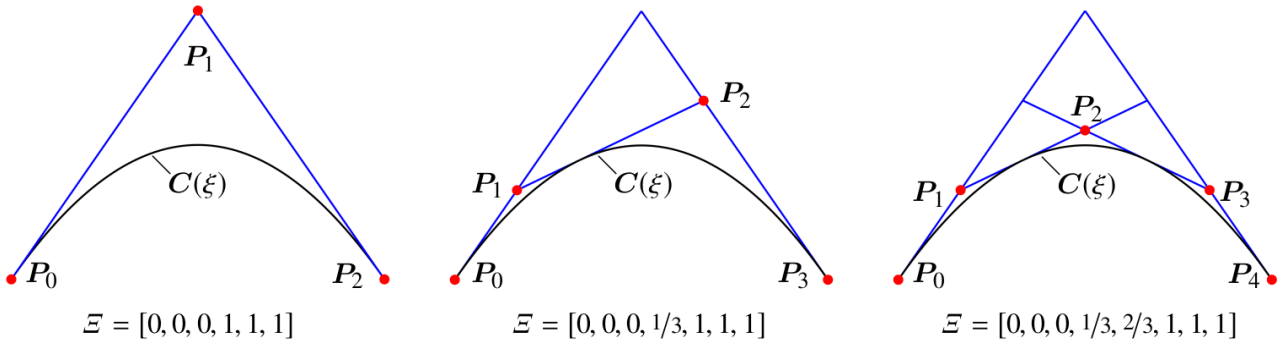

Fig. 5: Traditional curve approximation by quadratic basis functions: the control points $\boldsymbol{P}_i$ are shown as red dots. Insertion of the knots $\xi_3 = 1/3$ (middle) and $\xi_4 = 2/3$ (right); the newly calculated control points $\boldsymbol{P}_i$ do not change the curvature.

3.1 One-dimensional curve approximation

Without loss of generality, we describe a differentiable curve $\boldsymbol{C}$ with cartesian coordinates. The aim is now to approximate $\boldsymbol{C}(x, y)$ by a parametric curve $\boldsymbol{C}(\xi)$ together with a set of control points $\boldsymbol{P}_i$,

$$\boldsymbol{C} = \begin{pmatrix} x \\ y \end{pmatrix} = \sum_{i=0}^{n-1} \boldsymbol{P}_i N_i(\xi) \tag{9}$$

with $\boldsymbol{P}_i = \left(P_i^x, P_i^y\right)^T$ and for $i \in \{0, ..., n-1\}$. We choose the following procedure:

- ▷ Define positions (x_i, y_i) on curve $\boldsymbol{C}$ associated with the unique knot values ξ_i^u partitioning the curve into $n_{\text{el}} = n - 2$ elements.
- ▷ At the boundary for an open knot system, curve and control points coincide:

$$\boldsymbol{P}_0 = \boldsymbol{C}(x_0, y_0) \equiv \boldsymbol{C}(\xi_0^u) \tag{10}$$
$$\boldsymbol{P}_{n-1} = \boldsymbol{C}(x_n, y_n) \equiv \boldsymbol{C}(\xi_n^u) \tag{11}$$

- ▷ At the positions (x_i, y_i), calculate the slope

$$m_i := \partial_x \boldsymbol{C}|_{x_i}\,, \tag{12}$$

  and formulate the tangents' equations $t_i(x) = y_i + m_i(x - x_i)$.
- ▷ Consider neighboring points $i-1$ and i and determine the remaining control point coordinates from the corresponding tangents' intersection:

$$\left.\begin{aligned} t_i(x) &= y_i + m_i(x - x_i) \\ t_{i-1}(x) &= y_{i-1} + m_{i-1}(x - x_{i-1}) \end{aligned}\right\} \xrightarrow{t_i = t_{i-1}} P_i^x = \frac{y_{i-1} - y_i + x_i m_i - x_{i-1} m_{i-1}}{m_i - m_{i-1}} \rightarrow P_i^y = t_i(P_i^x) \tag{13}$$

- ▷ Continue for all $i \in \{1, ..., n-2\}$.

Example: control points of a quarter circle. We chose here $x \in [0, 1]$ to model the arc of a unit circle described by the analytical function $x^2 + y^2 = 1$ or the curve

$$\boldsymbol{C} = \begin{pmatrix} x \\ (1 - x^2)^{\frac{1}{2}} \end{pmatrix}.$$

First, the arc is partitioned into n_{el} elements using a uniform open knot vector Ξ and the corresponding unique knot vector Ξ^u, which read

$$\Xi = \{\xi_0, \xi_1, \xi_2, ..., \xi_N\} = \{0, 0, 0, 1/n_{\text{el}}, 2/n_{\text{el}}, ...n_{\text{el}}-1/n_{\text{el}}, 1, 1, 1\}\,, \qquad n_{\text{el}} \geq 1,$$
$$\Xi^u = \{\xi_0^u, \xi_1^u, ...\xi_n^u\} = \{0, 1/n_{\text{el}}, 2/n_{\text{el}}, ...n_{\text{el}}-1/n_{\text{el}}, 1\}\,.$$

The boundary is fixed at $\xi_0^u = 0$ and $\xi_n^u = 1$ and so

$$\boldsymbol{P}_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \boldsymbol{P}_{n-1} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

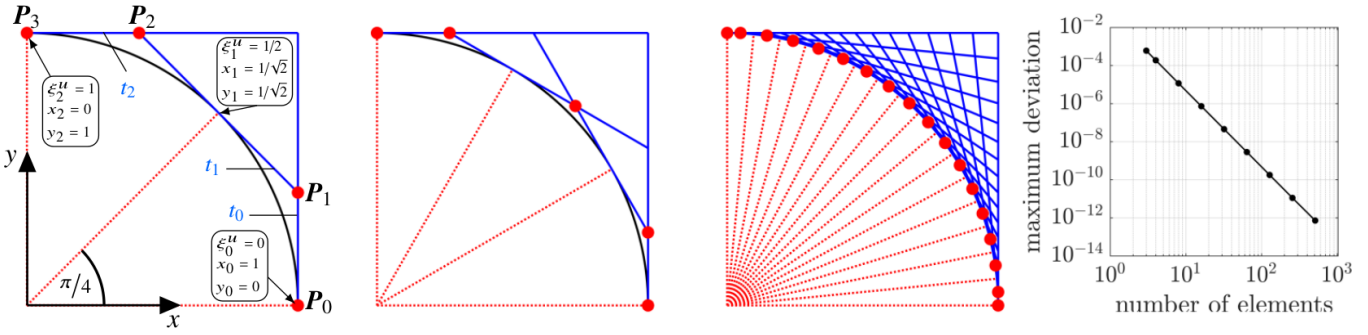

Fig. 6: Approximation of a quarter circle with $n_{\text{el}} = 2, 3$ and 16 elements and $n = 4, 5$ and 18 control points (red dots), and maximum deviation between parametric and analytical curve over the number of elements.

The n_{el} elements shall have the same length, given by the corner positions $x_i = \cos(\varphi_i)$ and $y_i = \sin(\varphi_i)$ with $\varphi_i = i\pi/n$. The slope at point (x_i, y_i) is then

$$m_i = -\frac{x_i}{y_i},$$

which gives together with Eq. (13) the control points

$$P_i^x = \frac{y_{i-1} - y_i - x_i^2/y_i + x_{i-1}^2/y_{i-1}}{-x_i/y_i + x_{i-1}/y_{i-1}}, \qquad P_i^y = y_i - x_i/y_i P_i^x, \qquad \text{and} \quad \boldsymbol{P}_i = \begin{pmatrix} P_i^x \\ P_i^y \end{pmatrix}$$

for all $i \in \{1, ..., n-2\}$.
In Fig. 6, the approximation is illustrated for $n = 4$, $n = 5$ and $n = 18$ control points and $n_{\text{el}} = n - 2$ elements. The control points are shown as red dots. In the right plot of Fig. 6, the maximum difference of the parametric and the analytical curve is plotted for different numbers of elements. From the double-logarithmic scale, it can be seen that for $n_{\text{el}} = 2, \ldots, 512$ elements the maximum radial deviation $\Delta_r = \left\| \boldsymbol{r}_{\text{ana}} - \boldsymbol{r}_{\text{par}} \right\|_2$ is less than $10^{-3}, \ldots, 10^{-12}$, respectively.

Example: control points of cosine curve. Next, we parameterize a trigonometric function within $x \in [0, \pi]$. The analytical function $y = \cos(x)$ or the curve

$$\boldsymbol{C} = \begin{pmatrix} x \\ \cos(x) \end{pmatrix}$$

shall be modeled with n_{el} elements. As before, the boundary control points are defined by the functional values

$$\boldsymbol{P}_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \boldsymbol{P}_{n-1} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The remaining control points are calculated from the intersecting tangents. The slope $m_i = -\sin(x_i)$ can be expressed with the identity $\sin^2(x) + \cos^2(x) = 1$ as $m_i = -(1 - y_i^2)^{1/2}$. It is zero for $y = \pm 1$ which gives for a one element domain the constant tangents $t_0 = 1$ and $t_1 = -1$, which do not intersect. Using Eq. (8), we get an intermediate value for $\boldsymbol{P}_1$, which gives the straight line, as shown in Fig. 7 (left). We need at least one additional tangent to find intersections. So we divide the arc into $n_{\text{el}} = 2$ elements (or $n = 4$ control points) and chose $x_1 = \pi/2$, $y_1 = 0$ to obtain $m_1 = -1$. In this way, the new tangent is $t_1 = -(x - \pi/2)$, and the new tangent t_2 is the old tangent $t_1 = -1$. At the intersection of the neighboring tangents $t_0 = t_1$ and $t_1 = t_2$, we find the new control points

$$\boldsymbol{P}_1 = \begin{pmatrix} \frac{\pi}{2} - 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{P}_2 = \begin{pmatrix} \frac{\pi}{2} + 1 \\ -1 \end{pmatrix},$$

respectively. The result is shown in Fig. 7. This procedure has to be repeated in each interval and for several inserted knots ξ_j, $j = 1, \ldots, N_j$, until the wanted accuracy is reached.

3.2 Two-dimensional surface approximation

Because a curve can just map the boundary, we now consider a two-dimensional domain, originally meshed uniformly with square elements, Fig. 3. At the left boundary (at $\xi = 0$) only the basis function $N_{0,2}$ is equal to unity, $N_{0,2}(\xi = 0) = 1$, and all other basis functions are zero at that position. The same situation is given at the right boundary $\xi = 1$ and $N_{n,2} = 1$. For now, let us focus on the left boundary of a two-dimensional square domain given in the horizontal direction at $\xi^{(1)} = 0$. Then, $\xi^{(2)}$ is the free variable in the vertical direction. The surface

$$\boldsymbol{S}(\xi^{(1)}, \xi^{(2)}) = \sum_A \boldsymbol{P}_A N_A(\xi^{(1)}, \xi^{(2)}) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \boldsymbol{P}_{i_1,i_2} N_{i_1}(\xi^{(1)}) N_{i_2}(\xi^{(2)}) \tag{14}$$

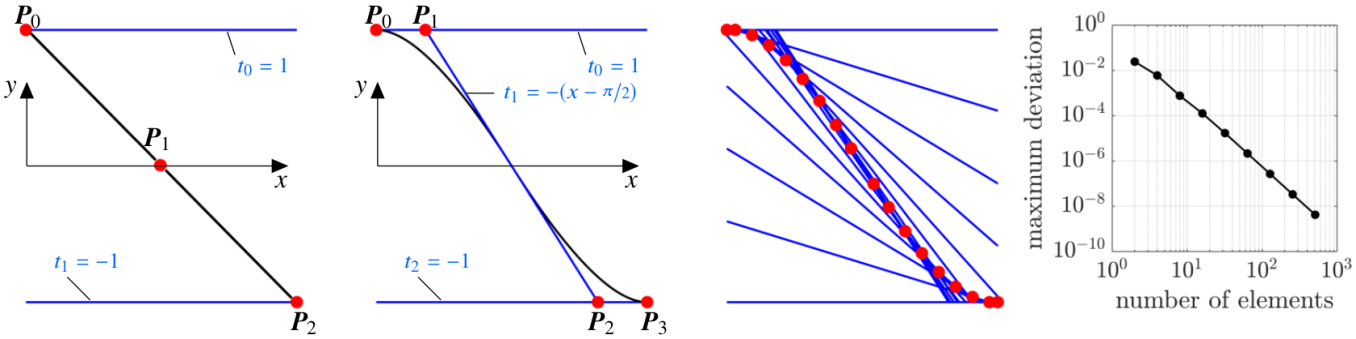

Fig. 7: Approximation of a cosine curve with $n_{\mathrm{el}} = 1, 2$ and 16 elements and $n = 3, 4$ and 18 control points (red dots), and maximum deviation between parametric and analytical curve over the number of elements.

can be reduced at the left boundary $\xi^{(1)} = 0$ to

$$\boldsymbol{S}(\xi^{(1)} = 0, \xi^{(2)}) = \sum_{i_1} \sum_{i_2} \boldsymbol{P}_{i_1,i_2} \underbrace{N_{i_1}(\xi^{(1)} = 0)}_{\equiv 1 \text{ if } i_1 = 0, \text{ else } 0} N_{i_2}(\xi^{(2)}) = \sum_{i_2} \boldsymbol{P}_{0,i_2} N_{i_2}(\xi^{(2)}) = \boldsymbol{C}(\xi^{(2)}) \tag{15}$$

and similar expressions at the other boundaries. The resulting equation (15) is similar to that of a single curve given in Eq. (9). The surface boundary control points $\boldsymbol{P}_{0,i_2}$ can be treated like the control points $\boldsymbol{P}_i$ of the one-dimensional curve. We will do this here for the example of a square mesh with $n_1 = 5$, $n_2 = 5$; see Fig. 8 (left). This square shall be mapped to a round disc.

We identify first the domain's length with the circle's diameter (or zoom the square domain uniformly to obtain the wanted diameter). Then we consider the control points $\boldsymbol{P}_0$, $\boldsymbol{P}_4$, $\boldsymbol{P}_{20}$, and $\boldsymbol{P}_{24}$ as fixed boundary points. The remaining points at the boundary are now moved according to the one-dimensional algorithm of Section 3.1.

Inside the circular disc, different control point arrangements can be thought off. In the simplest case, the nine internal control points are left unchanged. This results in a somewhat distorted mesh, as displayed in Fig. 8 (mid). A more regular mesh is achieved when all internal control points are moved to straight lines, i.e.,

$$P^x_{21} = P^x_{16} = P^x_{11} = P^x_6 = P^x_1, \qquad P^y_{15} = P^y_{16} = P^y_{17} = P^y_{18} = P^y_{19}, \quad \text{and so on.} \tag{16}$$

The corresponding mesh is displayed in Fig. 8 (right), and all control point coordinates are listed in the appendix.

We remark that for quadratic B-splines, the set of control points describing the contour of one element is given by the nine surrounding control points, e.g., the center element in Fig. 8 is determined by $\boldsymbol{P}_6, \boldsymbol{P}_7, \boldsymbol{P}_8, \boldsymbol{P}_{11}, \boldsymbol{P}_{12}, \boldsymbol{P}_{13}, \boldsymbol{P}_{16}, \boldsymbol{P}_{17}$, and $\boldsymbol{P}_{18}$. Therefore, changing the position of the center control point $\boldsymbol{P}_{12}$ influences the structure of all nine elements, while changing $\boldsymbol{P}_0$ affects the left bottom element only.

The effect of different internal control point arrangements is displayed in Fig. 9. Here the boundary is mapped as before, but inner points are moved towards the center. This leads to an annular center element. Such a mesh can be used, for example, to account for an inclusion or to simulate heating from a laser spot inside the body. The third transformation of Fig. 9 shifts the external control points again. Then, the resulting contour is rectangular but with a round internal element.

We remark that mapping a square to a circular domain is a homeomorphic transformation (Gamelin and Greene, 1999). Only such homeomorphic transformations can be obtained by re-arranging control points. A counterexample is a plate with a hole that cannot directly be mapped in this way. Here, the material parameter of the center element above could be adapted accordingly. Alternatively, a section of the plate with a hole could be modeled, replacing the open knot boundary with periodic boundary conditions. An ultimate counterexample, which is not realizable in this way, is a trefoil knot, which is homeomorphic to a torus.

3.3 Three-dimensional geometry approximation

For a three-dimensional body, the mapping is extended to a volume

$$\boldsymbol{V}(\xi^{(1)}, \xi^{(2)}, \xi^{(3)}) = \sum_A \boldsymbol{P}_A N_A(\xi^{(1)}, \xi^{(2)}, \xi^{(3)}) = \sum_{i_1=0}^{n_1-1} \sum_{i_2=0}^{n_2-1} \sum_{i_3=0}^{n_3-1} P_{i_1,i_2,i_3} N_{i_1}(\xi^{(1)}) N_{i_2}(\xi^{(2)}) N_{i_3}(\xi^{(3)}) \tag{17}$$

and, again, a homeomorphic transformation shall be found between a cube and the wanted body. The procedure is given exemplarily for the two solids' meshing: a full cylinder and a full sphere (ball), see Fig. 8. The cylinder can easily be extended from the circular disc. A repetition of the disc's control points in the axial direction gives a solid body mesh. An algorithm for the control point determination of a ball is described in the following.

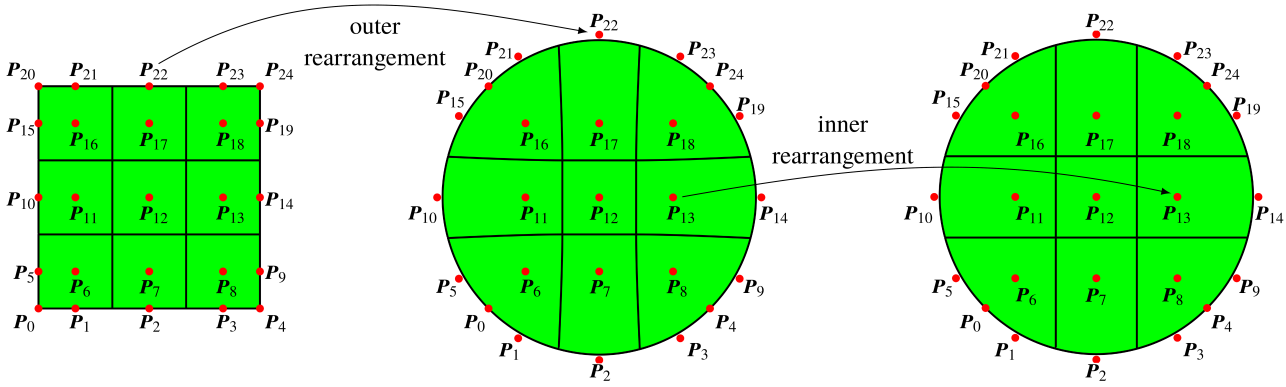

Fig. 8: Mapping of a square mesh to a circular plate with adapted boundary and internal control points.

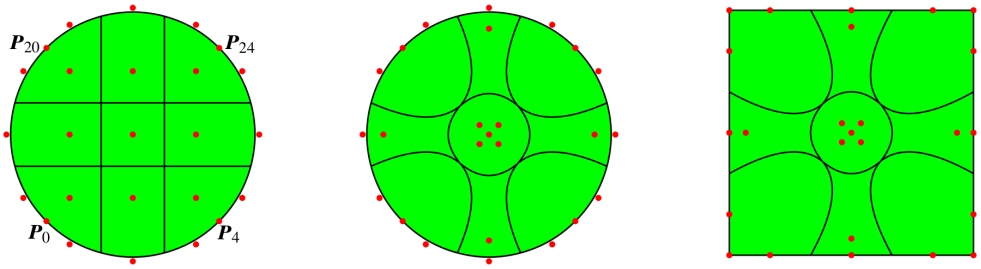

Fig. 9: Different two dimensional spline-based meshes and variation of internal control points.

Control point determination for a 3D ball We describe the control points' positions with a cartesian coordinate system with origin in the center of the structure. First, we set the eight cube corner-points on the unit sphere at coordinates $(\pm 1/\sqrt{3}, \pm 1/\sqrt{3}, \pm 1/\sqrt{3})$. They all have an equal great-circle distance with an inscribed angle of $2 \cdot \arccos(\sqrt{2/3}) \hat{=} 70.52°$. Each great-circle arc, which describes the shortest distance between two neighboring points on a sphere, then connects two corner points of the original cube. The arcs' remaining control points map like shown in Fig. 6 with the minor difference that the procedure needs to be applied to an arc of 70.52°, see Fig. 11 (left).
For further description, we employ the geometrical symmetry to reduce the number of control points that need to be calculated, see Fig. 11; the remaining points follow by rotation or reflection at the symmetry planes. The control points at the surface are determined in analogy to the two tangents intersection point algorithm of the one-dimensional case. On a surface, three tangent planes are needed to intersect. The tangent plane of a surface is given by $(\boldsymbol{x} - \boldsymbol{p}) \cdot \boldsymbol{n} = 0$ where $\boldsymbol{p}$ denotes the support-vector, pointing from the origin to a specific point on the sphere, see the black lines in Fig. 11, and $\boldsymbol{n}$ is the outward normal of the plane. On the unit sphere, we identify $\boldsymbol{p} = \boldsymbol{n}$. To determine a control point position $\boldsymbol{P}$, we need three planes with normals $\boldsymbol{n}_1$, $\boldsymbol{n}_2$, $\boldsymbol{n}_3$ and solve $(\boldsymbol{x} - \boldsymbol{n}_i) \cdot \boldsymbol{n}_i = 0$ for each $i = 1, 2, 3$ so that

$$\boldsymbol{x} = \boldsymbol{N}^{-1} \cdot (\boldsymbol{n}_1^2, \boldsymbol{n}_2^2, \boldsymbol{n}_3^2)^T = \boldsymbol{N}^{-1} \cdot (1, 1, 1)^T \qquad \text{with} \qquad \boldsymbol{N} = [\boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3] \; . \tag{18}$$

The new control point $\boldsymbol{P}$ is positioned at $\boldsymbol{x}$, as indicated in Fig. 11 (right). We remark again that the inserted surface knot positions have to be chosen carefully for a general geometry because parallel tangent planes do not give a solution. For different possible arrangements of the inner control points, we refer to Section 3.2 and Fig. 8. For numerical reasons, we recommend to avoid wedge-shaped internal elements. A possible choice of control points for a sphere is listed in the Appendix.

For a more refined mesh, either element subdivision can be achieved by using the knot insertion technique (8) with the given control points (if the underlying mesh accuracy is sufficient) or the calculation procedure has to be repeated with more elements and, therefore, higher resolution.

4 Example: diffusion induced deformation

Here we present briefly the FEM computation of a helical structure subjected to chemo-mechanical fields. The mechanical response of the structure is determined by the phase decomposition of an initially homogenous binary mixture. Such a decomposition can be observed after solidification, or while battery (de)charging, we refer to (Weinberg et al., 2018; Werner and Weinberg, 2019) for more details.

The structure's chemical state is described by the normalized concentration c and its mechanical response by the deformation gradient $\boldsymbol{F}$. The fields follow two coupled partial differential equations, namely the Cahn-Hilliard equation of diffusion $\dot{c} = \nabla \cdot (M(c) \nabla \delta_c \Psi(c, \boldsymbol{F}))$ with a degenerate mobility term M, and the quasi-static balance of linear momentum $\nabla \cdot (\partial_{\boldsymbol{F}} \Psi(c, \boldsymbol{F})) = \boldsymbol{0}$ in the framework of finite elasticity. Both equations are based on the free energy density Ψ, which has the form

$$\Psi(c, \boldsymbol{F}) = c \ln(c) + (1 - c) \ln(1 - c) + \frac{5}{2} c(1 - c) + \kappa |\nabla c|^2 + J_i \left[\frac{K(c)}{4} (J_e^2 - 1 - 2 \ln(J_e)) + \frac{G(c)}{2} (J^{-\frac{2}{3}} \mathrm{tr}(\mathrm{F}^{\mathrm{T}} \mathrm{F}) - 3) \right] \tag{19}$$

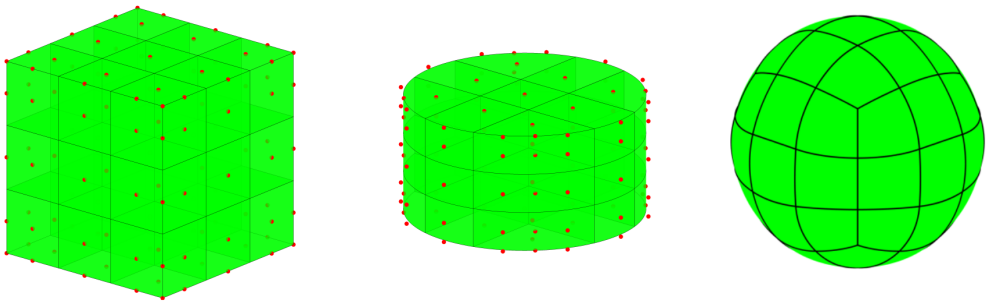

Fig. 10: Mapping of a cubic volume mesh to a cylinder and a ball.

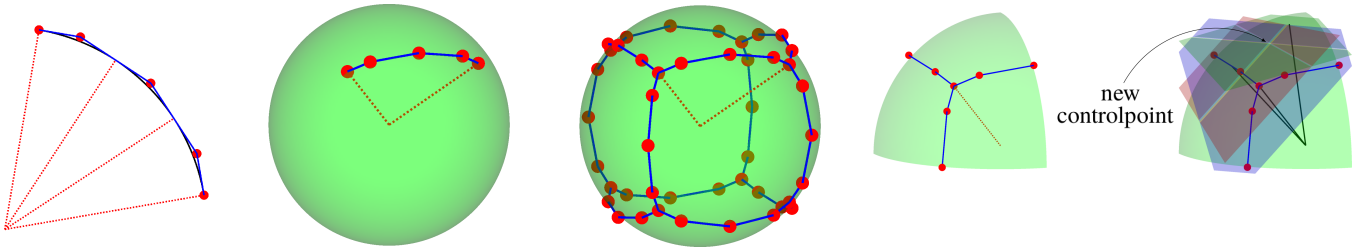

Fig. 11: Determination of control points for a ball: great circle arc (upper line) and intersecting tangent planes (lower line).

where $J = \det(\boldsymbol{F}) = J_e J_i(c)$ is the multiplicatively decomposed determinant of the deformation gradient (Werner and Weinberg, 2019). Elastic bulk and shear modulus K and G are given as a concentration-dependent convex combinations of the two species, and κ is a constant surface coefficient.

The Cahn-Hilliard equation is a fourth-order PDE that requires at least C^1 continuity for finite element approximation. Therefore we meshed the structure with quadratic B-splines. Starting from a uniformly meshed rectangular block of $3\times3\times120$ elements, we derived the helical geometry as outlined in Sect. 3. The helix is initially loaded with $c = 0.4$, the ends are fixed, and it decomposes with time then. In Fig. 12, the initial mesh and the deformed structure are displayed. It can clearly be seen that the deformation is not uniform. As a consequence of the phase decomposition, the phases with a high concentration c show swelling and volumetric expansion, whereas the phases with low concentration relax. This results in a rather triangular cross-section of the initially circular helix. This deformation is magnified by the different stiffnesses of the material given by the concentration-dependent (elastic moduli of the) material.

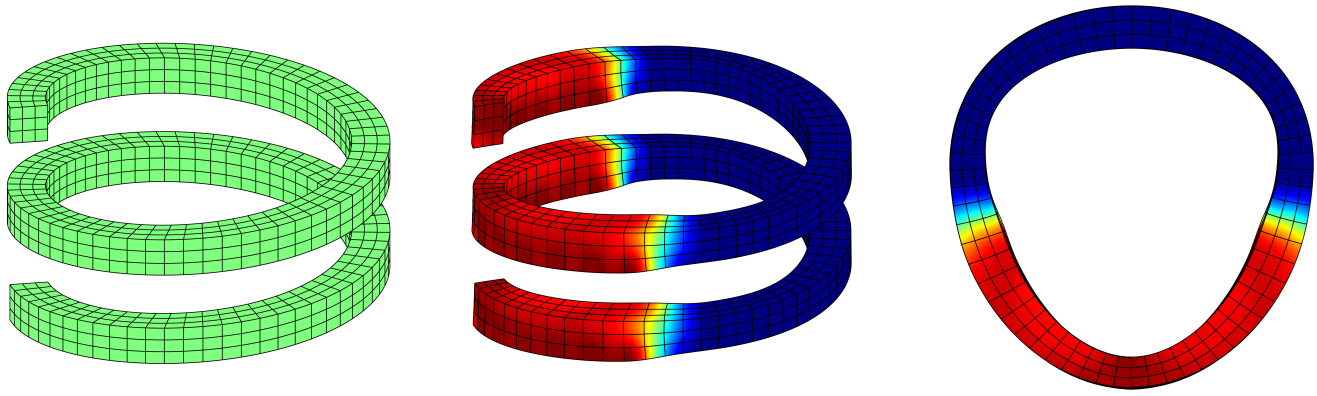

Fig. 12: Mesh of a helical structure and chemo-mechanical response: high concentration phases (red) induce swelling, low concentration phases relax.

5 Summary

Standard FEM ansatz functions do not allow to solve high-order PDEs directly, and a natural way to achieve the required C^p continuity is the use of B-splines of order $p+1$ as finite element basis functions. Such spline-based FEM comes at the expense of a non-intuitive design of the geometrical mesh. Because multi-physics problems, especially with reaction-diffusion terms, require a mesh without distorted elements, we present a strategy to obtain B-spline-based meshes for curves, surfaces, and volumes. For curved objects, we determine the control points and estimate the error of geometry approximation.

The isoparametric B-spline based meshes are based on homeomorphic transformations which allow us to deduce elementary

geometrical shapes from rectangular plate or cuboid solid tensor-product meshes. We present the knot vectors and the control points for a circular plate, a cylinder, and a sphere for practical use. To show our strategy's versatility, we adapt the presented algorithm to mesh a helix structure and show the chemo-mechanically induced deformation due to phase decomposition.

Acknowledgment and Funding Information

The authors gratefully acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) under the project WE-2525/8-1.

Appendix

In Table A.1, the control points for a round disc are given. The model is meshed with 9 square finite elements.

Knots and control points for a 3D-ball are presented in Table A.2. The volume mesh uses brick elements.

Tab. A.1: Control points $\boldsymbol{P}_A = (P_A^x, P_A^y)^T$ of a 3×3 element round disc, where $A = i_1 + i_2 n_1$ and $\Xi_1 = [0, 0, 0, 1/3, 2/3, 1, 1, 1] \equiv \Xi_2$.

A	P_A^x	P_A^y	A	P_A^x	P_A^y	A	P_A^x	P_A^y	A	P_A^x	P_A^y	A	P_A^x	P_A^y
0	-0.7071	-0.7071	5	-0.8966	-0.5176	10	-1.0353	0	15	-0.8966	0.5176	20	-0.7071	0.7071
1	-0.5176	-0.8966	6	-0.5176	-0.5176	11	-0.5176	0	16	-0.5176	0.5176	21	-0.5176	0.8966
2	0	-1.0353	7	0	-0.5176	12	0	0	17	0	0.5176	22	0	1.0353
3	0.5176	-0.8966	8	0.5176	-0.5176	13	0.5176	0	18	0.5176	0.5176	23	0.5176	0.8966
4	0.7071	-0.7071	9	0.8966	-0.5176	14	1.0353	0	19	0.8966	0.5176	24	0.7071	0.7071

Tab. A.2: Control points $\boldsymbol{P}_A = (P_A^x, P_A^y, P_A^z)^T$ of a $3 \times 3 \times 3$ element ball, where $A = i_1 + i_2 n_1 + i_3 n_1 n_2$, knotvector in first direction $\Xi_1 = [0, 0, 0, 1/3, 2/3, 1, 1, 1]$ and knotvector in the remaining directions $\Xi_2 = \Xi_3 = \Xi_1$.

A	P_A^x	P_A^y	P_A^z	A	P_A^x	P_A^y	P_A^z	A	P_A^x	P_A^y	P_A^z	A	P_A^x	P_A^y	P_A^z
0	-0.5774	-0.5774	-0.5774	32	0	-0.4369	-0.4369	64	1.0443	0	0	96	-0.4369	0.8584	0.4369
1	-0.4369	-0.6476	-0.6476	33	0.4369	-0.4369	-0.4369	65	-0.9294	0.5399	0	97	0	0.9294	0.5399
2	0	-0.7347	-0.7347	34	0.8584	-0.4367	-0.4369	66	-0.4369	0.4369	0	98	0.4369	0.8584	0.4369
3	0.4369	-0.6476	-0.6476	35	-0.9294	0	-0.5399	67	0	0.4369	0	99	0.6476	0.6476	0.4369
4	0.5774	-0.5774	-0.5774	36	-0.4369	0	-0.4369	68	0.4369	0.4369	0	100	-0.5774	-0.5774	0.5774
5	-0.6476	-0.4369	-0.6476	37	0	0	-0.4369	69	0.9294	0.5399	0	101	-0.4369	-0.6476	0.6476
6	-0.4369	-0.4367	-0.8584	38	0.4369	0	-0.4369	70	-0.7347	0.7347	0	102	0	-0.7347	0.7347
7	0	-0.5399	-0.9294	39	0.9294	0	-0.5399	71	-0.5399	0.9294	0	103	0.4369	-0.6476	0.6476
8	0.4369	-0.4367	-0.8584	40	-0.8584	0.4367	-0.4369	72	0	1.0443	0	104	0.5774	-0.5774	0.5774
9	0.6476	-0.4369	-0.6476	41	-0.4369	0.4369	-0.4369	73	0.5399	0.9294	0	105	-0.6476	-0.4369	0.6476
10	-0.7347	0	-0.7347	42	0	0.4369	-0.4369	74	0.7347	0.7347	0	106	-0.4369	-0.4367	0.8584
11	-0.5399	0	-0.9294	43	0.4369	0.4369	-0.4369	75	-0.6476	-0.6476	0.4369	107	0	-0.5399	0.9294
12	0	0	-1.0443	44	0.8584	0.4367	-0.4369	76	-0.4369	-0.8584	0.4369	108	0.4369	-0.4367	0.8584
13	0.5399	0	-0.9294	45	-0.6476	0.6476	-0.4369	77	0	-0.9294	0.5399	109	0.6476	-0.4369	0.6476
14	0.7347	0	-0.7347	46	-0.4369	0.8584	-0.4369	78	0.4369	-0.8584	0.4369	110	-0.7347	0	0.7347
15	-0.6476	0.4369	-0.6476	47	0	0.9294	-0.5399	79	0.6476	-0.6476	0.4369	111	-0.5399	0	0.9294
16	-0.4369	0.4367	-0.8584	48	0.4369	0.8584	-0.4369	80	-0.8584	-0.4367	0.4369	112	0	0	1.0443
17	0	0.5399	-0.9294	49	0.6476	0.6476	-0.4369	81	-0.4369	-0.4369	0.4369	113	0.5399	0	0.9294
18	0.4369	0.4367	-0.8584	50	-0.7347	-0.7347	0	82	0	-0.4369	0.4369	114	0.7347	0	0.7347
19	0.6476	0.4369	-0.6476	51	-0.5399	-0.9294	0	83	0.4369	-0.4369	0.4369	115	-0.6476	0.4369	0.6476
20	-0.5774	0.5774	-0.5774	52	0	-1.0443	0	84	0.8584	-0.4367	0.4369	116	-0.4369	0.4367	0.8584
21	-0.4369	0.6476	-0.6476	53	0.5399	-0.9294	0	85	-0.9294	0	0.5399	117	0	0.5399	0.9294
22	0	0.7347	-0.7347	54	0.7347	-0.7347	0	86	-0.4369	0	0.4369	118	0.4369	0.4367	0.8584
23	0.4369	0.6476	-0.6476	55	-0.9294	-0.5399	0	87	0	0	0.4369	119	0.6476	0.4369	0.6476
24	0.5774	0.5774	-0.5774	56	-0.4369	-0.4369	0	88	0.4369	0	0.4369	120	-0.5774	0.5774	0.5774
25	-0.6476	-0.6476	-0.4369	57	0	-0.4369	0	89	0.9294	0	0.5399	121	-0.4369	0.6476	0.6476
26	-0.4369	-0.8584	-0.4369	58	0.4369	-0.4369	0	90	-0.8584	0.4367	0.4369	122	0	0.7347	0.7347
27	0	-0.9294	-0.5399	59	0.9294	-0.5399	0	91	-0.4369	0.4369	0.4369	123	0.4369	0.6476	0.6476
28	0.4369	-0.8584	-0.4369	60	-1.0443	0	0	92	0	0.4369	0.4369	124	0.5774	0.5774	0.5774
29	0.6476	-0.6476	-0.4369	61	-0.4369	0	0	93	0.4369	0.4369	0.4369				
30	-0.8584	-0.4367	-0.4369	62	0	0	0	94	0.8584	0.4367	0.4369				
31	-0.4369	-0.4369	-0.4369	63	0.4369	0	0	95	-0.6476	0.6476	0.4369				

References

Denis Anders, Maik Dittmann, and Kerstin Weinberg. A higher-order finite element approach to the Kuramoto-Sivashinsky equation. *ZAMM - Z. Angew. Math. Mech.*, 92(8):599–607, 2012a.

Denis Anders, Christian Hesch, and Kerstin Weinberg. Computational modeling of phase separation and coarsening in solder alloys. *International Journal of Solids and Structures*, 49(13):1557 – 1572, 2012b. ISSN 0020-7683. doi: https://doi.org/10.1016/j.ijsolstr.2012.03.018.

M.J. Borden, T.J.R. Hughes, C.M. Landis, and C.V. Verhoosel. A higher-order phase-field model for brittle fracture : formulation and analysis within the isogeometric analysis framework. *Computer Methods in Applied Mechanics and Engineering*, 273: 100–118, 2014. ISSN 0045-7825. doi: 10.1016/j.cma.2014.01.016.

J. Austin Cottrell, Thomas J. R. Hughes, and Yuri Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley Publishing, 1st edition, 2009. ISBN 0470748737.

M. G. Cox. The Numerical Evaluation of B-Splines. *IMA Journal of Applied Mathematics*, 10(2):134–149, 10 1972. ISSN 0272-4960. doi: 10.1093/imamat/10.2.134.

Carl de Boor. On calculating with b-splines. *Journal of Approximation Theory*, 6(1):50 – 62, 1972. ISSN 0021-9045. doi: https://doi.org/10.1016/0021-9045(72)90080-9.

Luca Dede, Michael Borden, and Thomas Hughes. Isogeometric analysis for topology optimization with a phase field model. *Archives of Computational Methods in Engineering*, 19, 09 2012. doi: 10.1007/s11831-012-9075-z.

Van Than Dung and Tegoeh Tjahjowidodo. A direct method to solve optimal knots of b-spline curves: An application for non-uniform b-spline curves fitting. *PLOS ONE*, 12(3):1–24, 03 2017. doi: 10.1371/journal.pone.0173857.

Akemi Gálvez and Andrés Iglesias. Firefly algorithm for explicit b-spline curve fitting to data points. *Mathematical Problems in Engineering*, 2013:528215, Nov 2013. ISSN 1024-123X. doi: 10.1155/2013/528215.

Akemi Gálvez, Andrés Iglesias, and Jaime Puig-Pey. Iterative two-step genetic-algorithm-based method for efficient polynomial b-spline surface reconstruction. *Information Sciences*, 182(1):56 – 76, 2012. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2010.09.031. Nature-Inspired Collective Intelligence in Theory and Practice.

T.W. Gamelin and R.E. Greene. *Introduction to Topology*. Dover books on mathematics. Dover Publications, 1999. ISBN 9780486406800. URL https://books.google.de/books?id=thAHAGyV2MQC.

C. Hesch, S. Schuß, M. Dittmann, M. Franke, and K. Weinberg. Isogeometric analysis and hierarchical refinement for higher-order phase-field models. *Computer Methods in Applied Mechanics and Engineering*, 303:185 – 207, 2016. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2016.01.022.

Weishi Li, Shuhong Xu, Gang Zhao, and Li Ping Goh. Adaptive knot placement in b-spline curve approximation. *Computer-Aided Design*, 37(8):791 – 797, 2005. ISSN 0010-4485. doi: https://doi.org/10.1016/j.cad.2004.09.008. URL http://www.sciencedirect.com/science/article/pii/S0010448504002477. CAD '04 Special Issue: Modelling and Geometry Representations for CAD.

A. Makhlouf, B. Louhichi, D. Deneux, and M. Ali Mahjoub. Reconstruction of the cad model using tps surface. In *2019 23rd International Conference Information Visualisation (IV)*, pages 417–424, 2019.

L. Noël, M. Schmidt, C. Messe, J. A. Evans, and K. Maute. Adaptive level set topology optimization using hierarchical b-splines. *Structural and Multidisciplinary Optimization*, Jul 2020. ISSN 1615-1488. doi: 10.1007/s00158-020-02584-6.

Milić Predrag and Dragan Marinkovic. Isogeometric structural analysis based on nurbs shape functions. *Facta Universitatis - series Mechanical Engineering*, 11:193–202, 12 2013.

Thomas W. Sederberg, G. Thomas Finnigan, Xin Li, Hongwei Lin, and Heather Ipson. Watertight trimmed nurbs. *ACM Trans. Graph.*, 27(3):1–8, August 2008. ISSN 0730-0301. doi: 10.1145/1360612.1360678.

Peter Stein, Ying Zhao, and Bai-Xiang Xu. Electrochemical reactions in li-ion battery electrodes and their interaction with mechanical stresses: Size effects, phase segregation, and crack propagation. *Meeting Abstracts*, MA2017-01(1):133, 2017.

Kerstin Weinberg, Marek Werner, and Denis Anders. A chemo-mechanical model of diffusion in reactive systems. *Entropy*, 20 (2), 2018. ISSN 1099-4300. doi: 10.3390/e20020140.

Marek Werner and Kerstin Weinberg. *Coupled Thermal and Electrochemical Diffusion in Solid State Battery Systems*, pages 519–535. Springer International Publishing, Cham, 2019. ISBN 978-3-030-13307-8. doi: 10.1007/978-3-030-13307-8.

Gang Xu, Bernard Mourrain, Régis Duvigneau, and André Galligo. Parameterization of computational domain in isogeometric analysis: Methods and comparison. *Computer Methods in Applied Mechanics and Engineering*, 200(23):2021 – 2031, 2011. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2011.03.005. URL http://www.sciencedirect.com/science/article/pii/S0045782511001101.

Xiuyang Zhao, Caiming Zhang, Bo Yang, and Pingping Li. Adaptive knot placement using a gmm-based continuous optimization algorithm in b-spline curve approximation. *Computer-Aided Design*, 43(6):598 – 604, 2011. ISSN 0010-4485. doi: https://doi.org/10.1016/j.cad.2011.01.015.

O.C. Zienkiewicz and R. L. Taylor. *The finite element method*, volume 1-3. Butterworth-Heinemann, 5th edition, 2000.