

Geometrically accurate, efficient, and flexible quadrature techniques for the tetrahedral finite cell method

Atanas Stavrev^a, Lam H. Nguyen^b, Ruyi Shen^c, Vasco Varduhn^b,
Marek Behr^d, Stefanie Elgeti^d, Dominik Schillinger^{b,*}

^a*Aachen Institute for Advanced Study in Computational Engineering Science (AICES),
RWTH Aachen University, Germany*

^b*Department of Civil, Environmental, and Geo-Engineering, University of Minnesota, Twin Cities, USA*

^c*Carleton College, Northfield/Minnesota, USA*

^d*Chair for Computational Analysis of Technical Systems, RWTH Aachen University, Germany*

Abstract

We illustrate the importance of geometrically accurate volume quadrature for obtaining optimal accuracy with non-boundary-fitted finite element discretizations, when the problem domain is defined by sharp boundaries. We consider the tetrahedral finite cell method (TetFCM) and replace its recursive subdivision based integration approach with geometrically accurate quadrature rules that emanate from higher-order geometric parametrizations of cut tetrahedral elements. The element-wise parametrization procedure relies on the identification of the intersection topology and a series of higher-order mappings based on Lagrange polynomials. We demonstrate with several 3D examples that geometrically faithful local parametrization ensures optimal accuracy, while significantly reducing the number of quadrature points with respect to recursive subdivision. On the other hand, we highlight the strength of subdivision quadrature in the context of a patient-specific workflow for the simulation-based performance analysis of coupled bone/implant configurations. In particular, we show that accuracy, flexibility and computational efficiency of the TetFCM critically depends on flexibly applying the two different quadrature variants for fuzzy imaging data and sharp boundary representations, respectively.

Keywords: Tetrahedral finite cell method, non-boundary-fitted discretization, higher-order geometric parametrization, numerical integration

*Corresponding author;

Dominik Schillinger, Department of Civil Engineering, University of Minnesota, 500 Pillsbury Drive S.E., Minneapolis, MN 55455, USA; Phone: +1 612 624 0063; Fax: +1 612 626 7750; E-mail: dominik@umn.edu

Preprint submitted to Computer Methods in Applied Mechanics and Engineering

July 25, 2016

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 3 |
| 2 | Brief review of the tetrahedral finite cell method | 4 |
| 2.1 | Discretization with non-boundary-fitted tetrahedral elements | 4 |
| 2.2 | Quadrature of cut tetrahedra based on recursive subdivision | 6 |
| 3 | Importance of geometrically accurate quadrature in embedded domain analysis | 7 |
| 3.1 | A model problem | 8 |
| 3.2 | Review of theoretical results for isoparametric finite elements | 9 |
| 3.3 | Interrelation between quadrature and solution accuracy in the finite cell method | 10 |
| 4 | Quadrature rules based on local parametrization of cut tetrahedra | 11 |
| 4.1 | Identification and classification of valid intersection topologies | 11 |
| 4.2 | Beyond valid intersection topologies | 13 |
| 4.3 | Parametrization based on nodal Lagrange polynomials | 13 |
| 4.4 | Sequence of higher-order mappings | 15 |
| 4.5 | Higher-order geometrically accurate quadrature rules | 18 |
| 5 | Numerical examples in three dimensions | 19 |
| 5.1 | Plate with a cylindrical hole | 20 |
| 5.2 | Thick spherical shell under internal pressure | 21 |
| 5.3 | Fractured femur bone with a nail implant | 24 |
| 5.3.1 | Fuzzy imaging data vs. sharply defined B-rep models | 25 |
| 5.3.2 | Phase-field modeling of brittle fracture | 26 |
| 5.3.3 | Bone/implant coupling based on Nitsche’s method | 28 |
| 5.3.4 | Supporting accuracy and flexibility of the TetFCM by adequate quadrature | 28 |
| 6 | Summary and conclusions | 31 |

1. Introduction

Embedded domain finite element methods approximate the solution of boundary value problems using non-boundary-fitted meshes [1]. Their primary goal is to increase the geometric flexibility of the finite element method and to alleviate mesh related obstacles that often appear for geometrically complex domains. For example, embedded domain concepts have been successfully employed to deal with trimmed CAD surfaces [2–5] and image based geometries [6, 7], to prevent mesh updating and mesh distortion effects [8–11], or to handle fluid–structure interaction problems involving large displacements and contact [12–15].

Embedded domain methods require two fundamental components beyond standard finite element technology. First, they need to be able to enforce Dirichlet boundary conditions at embedded surfaces, for which variational methods such as Lagrange multipliers [16–18] or Nitsche-type techniques [19–21] have been successfully employed. Second, they need to be able to evaluate surface and volume integrals in cut elements. In this context, the importance of geometrically faithful quadrature has been recently highlighted in a series of papers. An early account can be found in the thesis of STAVREV [22] who demonstrated the direct relation between geometric accuracy, numerical quadrature of cut elements and reduced rates of convergence in the Cartesian finite cell method. KAMENSKY et al. [13] recently presented the role of geometry in embedded domain-type methods from a broader perspective, coining the term *immersogeometric analysis* for methods that precisely capture immersed geometry in a non-boundary-fitted background mesh. Several authors recently contributed techniques to generate geometrically accurate quadrature points for cut elements [18, 23–28]. We highlight the framework presented by FRIES and OMERVIĆ [24], which we will make frequent use of in our implementation.

The finite cell method introduced by PARVIZIAN, DÜSTER and RANK [29, 30] combines non-boundary-fitted Cartesian meshes with higher-order approximation of the solution fields and adaptive quadrature of intersected elements based on recursive subdivision. A concise summary of the Cartesian finite cell method can be found for example in the recent review in [31]. Readers interested in an instructive and easy-to-handle starting point for implementing the finite cell method are referred to the open source code FCMLab¹ [32]. The tetrahedral finite cell method (TetFCM) is an emerging variant that employs non-boundary-fitted meshes based on tetrahedra with direct access to local mesh adaptivity [33, 34] as an alternative to other local refinement schemes in the Cartesian finite cell method [35–38]. A considerable advantage of the finite cell method is the flexibility of its recursive subdivision based quadrature approach, which is easy to implement and can operate with almost any type of geometric model. It is based on the decomposition of each intersected element into adaptive sub-cells that can be efficiently organized in hierarchical tree data structures, aggregating quadrature points around the intersecting surface. Given a large enough depth of the tree, the geometry in intersected elements is resolved with sufficient fidelity, so that higher-order accuracy of the solution fields can be maintained. The finite cell method therefore falls under the umbrella of immersogeometric methods [33, 34]. However, the evaluation of the large number of quadrature points in intersected elements involves a prohibitively large computational cost, in particular for high-fidelity analysis of three-dimensional problems.

¹<http://fcmlab.cie.bgu.tum.de>

In this paper, we first outline a framework for automatically parametrizing the geometry of tetrahedral elements that are arbitrarily intersected by a smooth surface parametrized by non-uniform rational B-splines (NURBS) [39]. The element-wise parametrization procedure is based on the identification of cut elements in the non-boundary-fitted mesh and their classification in terms of a few topological cut cases that decompose each tetrahedron into a combination of sub-elements. Core components of our framework are root-finding algorithms and a series of higher-order mappings motivated by the NURBS-enhanced finite element concept [40–42]. The former determine element intersections with a smooth spline surface, the latter blend sub-element surfaces on the intersecting spline surface. The robustness and efficiency of this procedure can be improved by a variety of techniques shown in [24].

We use the element-wise geometric parametrizations to derive geometrically accurate volume quadrature rules as an alternative to the recursive-subdivision based quadrature approach in the tetrahedral finite cell method. The main part of the paper attempts a careful assessment of each quadrature variant with respect to accuracy and convergence of the solution, the number of quadrature points and the associated computational cost, and the compatibility with different geometric models. To this end, we employ several benchmark problems defined by sharp spline surfaces to compare the performance of the two quadrature variants in the TetFCM. We also present a coupled bone/implant simulation of a fractured human femur bone that is held together by a proximal femoral nail implant. This example involves both geometric models based on fuzzy medical imaging data and sharp CAD boundary representations. Our results emphasize that, depending on the specific geometric model, both quadrature variants play an important role to preserve the flexibility of the TetFCM, while increasing its computational efficiency in high-fidelity 3D computations.

The paper is organized as follows: In Section 2, we provide a concise summary of the TetFCM technology, with particular focus on the quadrature of intersected elements based on recursive subdivision. Section 3 illustrates the importance of geometrically accurate volume quadrature for obtaining optimal accuracy with non-boundary-fitted meshes. Section 4 outlines our parametrization framework, discussing its basic components. Section 5 presents numerical benchmarks and the coupled bone/implant analysis, computed with our TetFCM framework that uses a combination of the two quadrature variants. Section 6 puts the numerical results into perspective and draws conclusions about the competitiveness of quadrature techniques in the non-boundary-fitted setting.

2. Brief review of the tetrahedral finite cell method

We start with a summary of the tetrahedral finite cell method in the context of linear elasticity. For details, we refer the interested reader to the recent TetFCM contributions in [33, 34].

2.1. Discretization with non-boundary-fitted tetrahedral elements

The starting point is the variational form of a problem, defined on the domain Ω with Dirichlet and Neumann boundaries Γ_D and Γ_N , respectively. For linear elasticity, we use the principle of virtual work

$$\delta W(\mathbf{u}, \delta \mathbf{u}) = \int_{\Omega} \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon} \, d\Omega - \int_{\Omega} \delta \mathbf{u} \cdot \mathbf{b} \, d\Omega - \int_{\Gamma_N} \delta \mathbf{u} \cdot \mathbf{t} \, d\Gamma_N = 0 \quad (1)$$

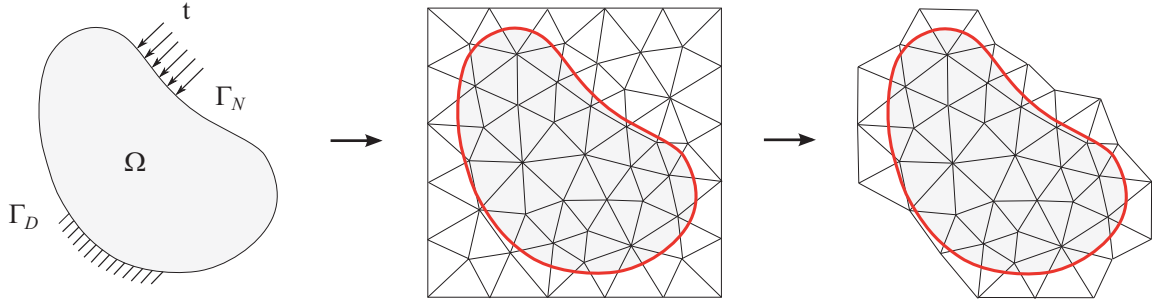


Figure 1: Boundary value problem defined on Ω and its discretization with a non-boundary-fitted triangular mesh, leading to elements intersected by the embedded boundary (in red).

where \mathbf{u} and $\delta\mathbf{u}$ are the true and virtual displacements, $\boldsymbol{\sigma}$ and $\delta\boldsymbol{\varepsilon} = 1/2(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$ denote the Cauchy stress and virtual strain tensors, and \mathbf{b} and \mathbf{t} are body forces and boundary traction, respectively [43, 44].

In contrast to the standard finite element method, the TetFCM allows the discretization of the variational form (1) with basis functions that can arbitrarily overlap the domain boundary Γ . This leads to a non-boundary-fitted finite element mesh, whose elements can be arbitrarily intersected by the domain boundary. Figure 1 illustrates the concept for the two-dimensional analogue with triangular elements. Releasing the constraint of boundary-fitted elements constitutes a significant simplification for meshing geometrically complex domains. From a practical point of view, it is convenient to first define an embedding domain of simple geometry that can be meshed easily and subsequently remove all elements without support in the problem domain.

The same principle applies in three dimensions for tetrahedral elements. Figure 2 illustrates the generation of a non-boundary-fitted tetrahedral mesh. In a first step, we identify a suitable embedding domain and employ a standard octree [45] to specify regions of local refinement. The octree can be easily transferred into a measure of local mesh density by writing out the position of the center of each octree leaf and the associated edge length that specifies the local element width h in its vicinity. In a second step, we feed this cloud of points and the associated h -values into the open-source mesh generator Netgen [46] (or any other standard tetrahedral meshing tool), which generates a corresponding adaptive non-boundary-fitted tetrahedral mesh. Advanced tetrahedral mesh generators make use of built-in efficient mesh smoothing and regularization algorithms [47]

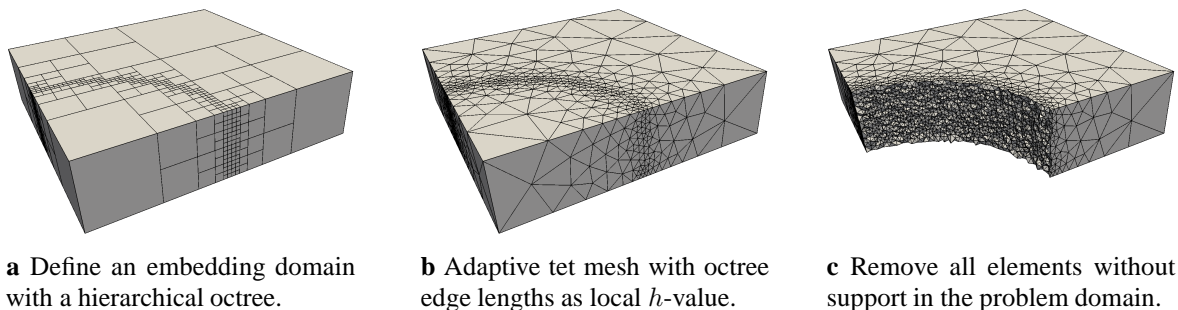


Figure 2: The three steps of generating an adaptive non-boundary-fitted tet mesh for a plate with a circular hole.

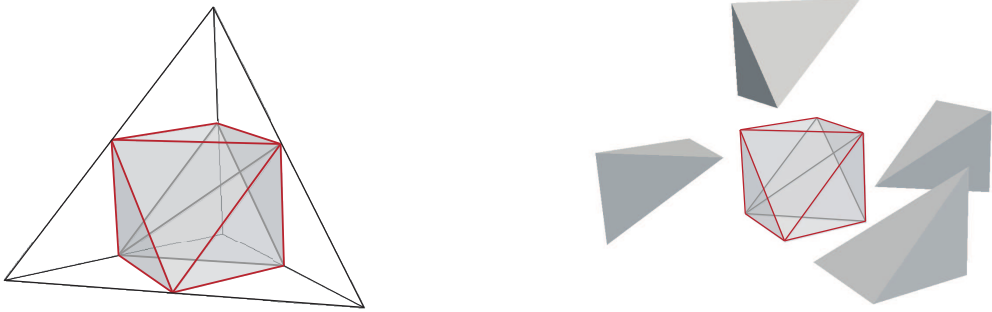
that ensure high-quality undistorted elements with well-formed angles. In the third step, we remove all elements that have no support in the problem domain. The resulting tetrahedral mesh is independent of the geometric boundaries of the immersed object and its local features.

To enforce Dirichlet boundary conditions at embedded surfaces, the TetFCM uses variationally consistent Nitsche methods, which do not introduce additional unknowns and preserve a positive definite stiffness matrix. For linear elasticity, the Nitsche method extends the principle of virtual work (1) as follows

$$\begin{aligned} \delta W_{\mathbf{K}}(\mathbf{u}, \delta \mathbf{u}) = & \int_{\Omega} \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon} \, d\Omega - \gamma \int_{\Gamma_D} \mathbf{u}_h \cdot (\delta \boldsymbol{\sigma} \cdot \mathbf{n}^+) \, d\Gamma \\ & - \int_{\Gamma_D} (\boldsymbol{\sigma}_h \cdot \mathbf{n}^+) \cdot \delta \mathbf{u} \, d\Gamma + \beta \int_{\Gamma_D} \mathbf{u} \cdot \delta \mathbf{u} \, d\Gamma \end{aligned} \quad (2)$$

$$\delta W_f(\delta \mathbf{u}) = \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{u} \, d\Omega - \gamma \int_{\Gamma_D} \hat{\mathbf{u}} \cdot (\delta \boldsymbol{\sigma} \cdot \mathbf{n}^+) \, d\Gamma + \int_{\Gamma_N} \mathbf{t} \cdot \delta \mathbf{u} \, d\Gamma + \beta \int_{\Gamma_D} \hat{\mathbf{u}} \cdot \delta \mathbf{u} \, d\Gamma \quad (3)$$

where $\delta W_{\mathbf{K}} = \delta W_f$. Function $\hat{\mathbf{u}}$ denotes the prescribed displacements along the Dirichlet boundary Γ_D and \mathbf{n}^+ is the outward unit normal vector on Γ_D . With $\gamma = 1$, we obtain the standard symmetric form of Nitsche's method, which we use in this paper. It requires a scalar stabilization parameter β that can be determined empirically or from a generalized eigenvalue problem [20, 48]. We note that with $\gamma = -1$, we obtain a parameter-free non-symmetric Nitsche method [49, 50], which was recently shown to work accurately and robustly for non-boundary-fitted discretizations of linear elastic problems without stabilization (i.e., $\beta = 0$) [21].



a Separate the four corner sub-cells first.

b Split the remaining octahedron into four sub-cells.

Figure 3: Building block of the recursive subdivision approach: a tetrahedron is split into 8 sub-cells.

2.2. Quadrature of cut tetrahedra based on recursive subdivision

Tetrahedral elements intersected by the embedded boundary require special numerical integration methods, because the volume integrals in (2) and (3) are only defined over portions of the element domain. The tetrahedral finite cell method uses a quadrature technique based on recursive octree subdivision of all cut elements. Its basic building block is the split of a tetrahedron into eight tetrahedral sub-cells as shown in Fig. 3. This split can be applied recursively for each cut sub-cell

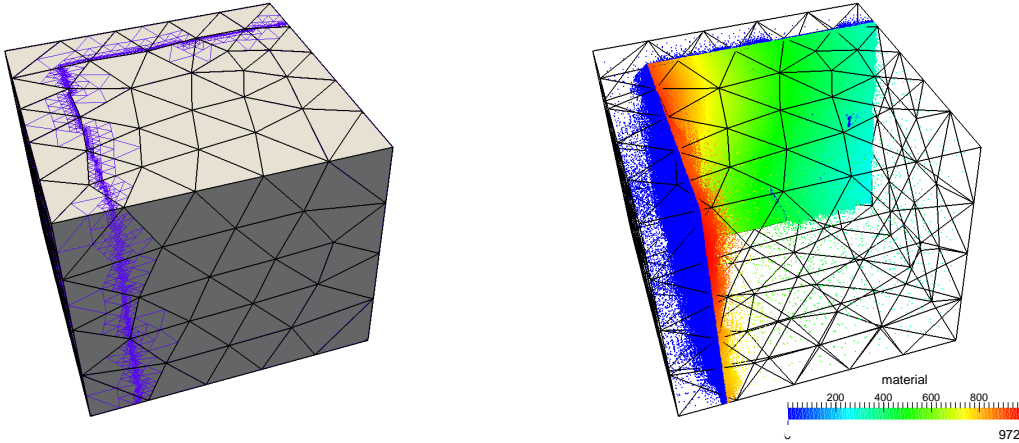


Figure 4: Example of recursive subdivision quadrature: Tetrahedral mesh (black lines) of the embedding domain, resolution of the cube geometry with sub-cells (blue lines), and quadrature points (green to red - inside, blue - outside).

until a predefined maximum level of sub-cells is reached. In each sub-cell, we apply a standard 5-point monomial rule for quadratic basis functions and an 11-point quadrature rule for cubic basis functions [51], so that quadrature points are aggregated at the immersed boundary. The weights of the quadrature points in each sub-cell are scaled with the volume of the sub-cell. The concept of recursive subdivision is illustrated for a cube that is discretized by a non-boundary-fitted mesh. Figure 4 illustrates the tetrahedral elements (black lines), the adaptive sub-cell structure (blue lines) and the quadrature points. We note that all quadrature points outside the problem domain are not taken into account during integration. Figure 4 illustrates the significant increase in point evaluations produced by recursive subdivision. The TetFCM in this form therefore shifts the effort from meshing to numerical quadrature of intersected tetrahedral cells.

From an algorithmic viewpoint, we implement recursive subdivision in a “bottom-up” fashion. We first refine the complete tetrahedral element by generating all possible leaves at the maximum tree depth. We then start to build up the octree by combining sets of uncut sub-cells into one sub-cell of higher level. This pruning procedure is repeated recursively until we reach the root at the top, that is, the original finite element. Our intersection test relies on an inside/outside test that determines whether quadrature points of a sub-cell are located on different sides of the geometric boundary. A major advantage of this approach is that small cuts are captured reliably, if they have at least one quadrature point inside the problem domain, but left out otherwise. The bottom-up procedure is computationally more expensive than the top-down procedure applied e.g. in [33], but is very well suited for parallelization. We refer to [34] for a detailed description of the algorithm.

3. Importance of geometrically accurate quadrature in embedded domain analysis

Before discussing the technical aspects of parametrization-based quadrature, we illustrate the importance of geometrically accurate volume quadrature for obtaining optimal solution accuracy on non-boundary-fitted finite element meshes. To this end, we examine the convergence behavior of a simple numerical example for two different quadrature variants and interpret our observations by pointing out analogies to standard theoretical results for isoparametric finite elements.

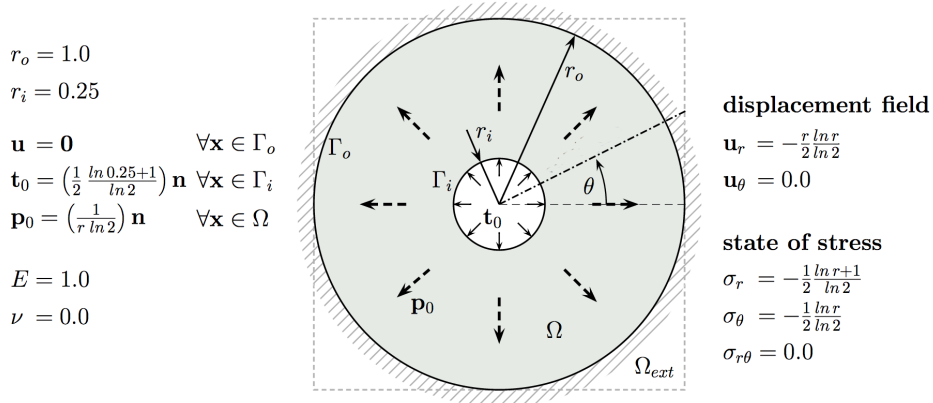


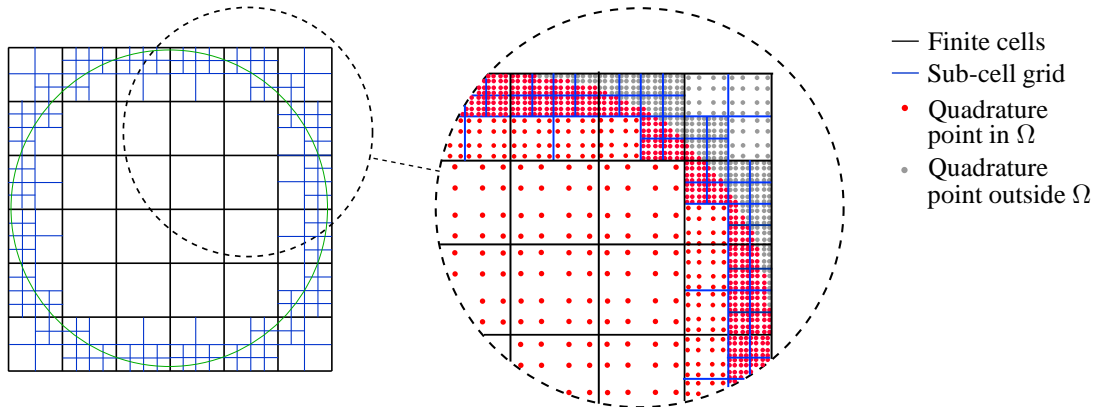
Figure 5: Annular ring under internal pressure (from [31]).

3.1. A model problem

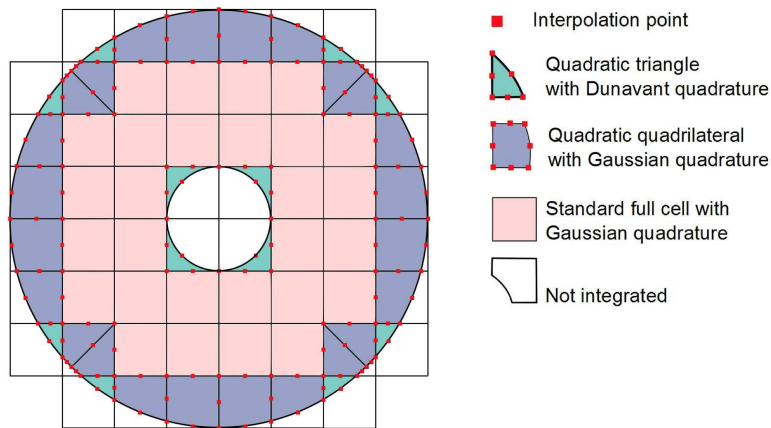
We consider a confined ring in plain strain elasticity, subjected to internal pressure, which is shown in the sketch of Fig. 5. A detailed description of the problem and its analytical solution is given in [7]. We discretize the problem domain with Cartesian meshes whose elements do not conform to the circular boundaries. We emphasize that the geometric aspects addressed in the following do not depend on a specific type of basis functions. In the present case, we employ cubic B-splines, but the same observations can be made for nodal quadrilateral or triangular elements. The Dirichlet boundary conditions on the outer circle are imposed with Nitsche’s method, where the stabilization parameter β is estimated from a local eigenvalue problem [21, 48, 52].

Following the thesis of STAVREV [22], we compare recursive-subdivision and element-wise parametrization based quadrature for numerically integrating intersected elements. The two quadrature variants are illustrated in Figs. 6a and 6b, respectively. The recursive-subdivision based approach subdivides each cut element into a hierarchical sub-cell structure and subsequently removes all quadrature points located outside of the problem domain. The approach for quadrilaterals shown in Fig. 6a is equivalent to what has been described for tetrahedra in Section 2.2 above. The alternative quadrature approach of Fig. 6b is based on the element-wise parametrization of the portion of each element domain that is located within the annular ring. Since we employ cubic B-splines for the approximation of the solution in this example, we use cubic Lagrange polynomials defined over quadrilaterals and triangles for the approximation of the circular boundary in each intersected element. The coefficients of the polynomial interpolation simply emanate from the nodal coordinates of quadrilateral and triangular reference elements.

Figures 7a and 7b illustrate the convergence in the L^2 norm and the H^1 semi-norm, respectively, as the mesh is uniformly refined. We observe that the computations based on the recursive-subdivision approach achieve optimal higher-order rates of convergence in the pre-asymptotic range. However, convergence abruptly stops after a critical error level has been reached, which depends on the depth of the hierarchical sub-cell structure. In the present case, we use four recursive levels of adaptive sub-cells. In contrast, the computations based on element-wise parametrization of cut elements are not affected by the quadrature scheme, achieving optimal higher-order rates of convergence also in the asymptotic range.



a Subdivision quadrature: recursively refined sub-cells aggregate quadrature points at the geometric boundary.



b Element-wise parametrization with cubic quadrilateral and triangular reference elements.

Figure 6: The two quadrature variants employed in the annular ring benchmark.

3.2. Review of theoretical results for isoparametric finite elements

Before performing computations, we review a theoretical result for boundary-fitted finite elements given in the classical book by STRANG and FIX [53]. In chapter 4.4, “Approximation of domain and boundary conditions”, **Strang and Fix examine the influence of the geometry approximation on the accuracy of the finite element solution for standard isoparametric discretizations.** In what they denote as “case 2”, they consider a boundary value problem with Neumann boundary conditions only, defined over an exact problem domain Ω and discretized by isoparametric elements. Case 2 resembles our embedded domain situation, where Dirichlet boundary conditions based on Nitsche’s method have a variational format similar to Neumann boundary conditions.

The main points can be summarized as follows: In general, a finite element mesh does not represent the problem domain Ω exactly, but represents an approximation Ω^h of the exact domain Ω . In the case of Neumann boundary conditions only, there is no question of imposing conditions on the wrong boundary, since basis functions are unrestricted at the boundary. Therefore, the effect of using an approximate domain enters exclusively via integrating over Ω^h instead of Ω .

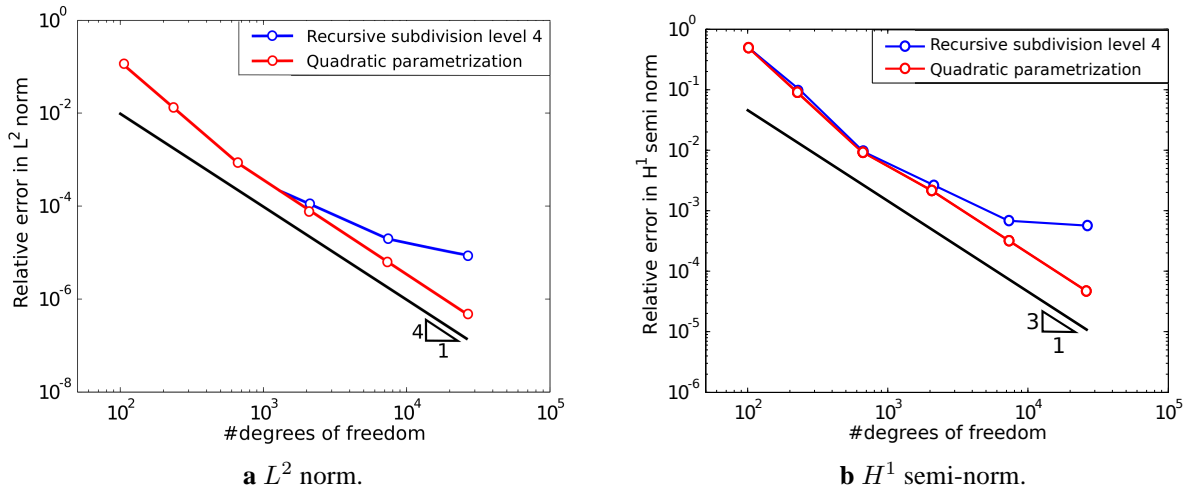


Figure 7: Convergence in relative error norms vs. the number of degrees of freedom for the annular ring problem.

In this sense, the impact of integrating over an approximate domain geometry and the impact of integrating with an inaccurate quadrature rule can be directly related and the corresponding errors commute. In [53], Strang and Fix provide an error estimate in terms of the (squared) error in strain energy E^2 due to integration over Ω^h instead of Ω , which reads

$$E^2 \leq C_1 h^{2p} + C_2 g^{2q+1} \quad (4)$$

The first term in Eq. 4 represents the usual strain energy error due to the difference between the exact solution u and the finite element approximation u^h . It involves an error constant C_1 , the characteristic mesh size h and the polynomial degree p of the finite element approximation. The second term is purely geometrical and represents the effect of approximating the geometry of the problem domain. It involves another error constant C_2 , the characteristic mesh size g and the degree q of the polynomial geometry approximation. For isoparametric elements, the characteristic lengths h and g and the polynomial degrees p and q coalesce. It is then easy to see that the geometric part of Eq. 4 converges at a faster rate than the first term and will thus never affect the performance of standard isoparametric methods.

3.3. Interrelation between quadrature and solution accuracy in the finite cell method

Equation 4 directly generalizes to the embedded domain case, as Strang and Fix did not assume at any point during their derivation that the finite element basis functions or the underlying elements need to be fitted to the domain Ω . In contrast to the isoparametric case discussed in [53], the finite cell method uses a geometry approximation that is not tied to the finite element mesh. In the embedded domain case, g and q of Eq. 4 are therefore independent from h and p .

The key to understanding the interrelation between quadrature and solution accuracy in the finite cell method is to consider the influence of the two quadrature variants on g and q . When the recursive-subdivision based approach is used, the approximate domain Ω^h is represented by a cloud of quadrature points that is generated from the sub-cell structure (see Fig. 6a). While

the interpretation of g as the edge length of the finest sub-cell is obvious, q remains unknown, as the cloud of quadrature points is not a polynomial approximation. However, it is intuitively clear when considering the “saw-tooth” pattern of the sub-cell structure that the equivalent to q must be low order. The geometric fidelity can always be increased by adding more sub-cell levels, that is, decreasing g . However, if the finite cell method employs basis functions of higher-order polynomial degree p , the first term in Eq. 4 will converge significantly faster under mesh refinement with constant number of recursive sub-cell levels than the second part that is dominated by a low order q . It is therefore unavoidable that at some error level, the low order convergence of the geometry will start to dominate the overall error, leading to an abrupt change in convergence rate as shown in Figs. 7a and 7b. In addition, increasing the depth of the sub-cell structure is limited by the exponential growth in the number of quadrature points, which makes this approach prohibitively expensive even for moderate recursive sub-cell refinement. The recursive subdivision approach can therefore be considered unable to achieve higher-order convergence rates in the asymptotic limit. With this being said, we emphasize that the recursive subdivision approach can still lead to full accuracy in the pre-asymptotic range, if we start with a large enough number of sub-cell levels. In particular for low order instantiations of the finite cell method, the recursive subdivision approach can be a competitive choice (see e.g. [33]).

When an element-wise parametrization is used, the approximate domain Ω^h is represented by higher-order accurate polynomials (see Fig. 6b). As we define geometric parametrizations for each intersected element, the corresponding characteristic length g is of the same order as h . It is therefore straightforward to see that if we choose polynomials of degree $q = p$ for parametrizing the geometry in cut elements we are in the same situation as in the isoparametric case. The second term of Eq. 4 converges slightly faster than the first term and will thus never affect the convergence rate of the finite cell method. This is confirmed by Figs. 7a and 7b, which illustrates asymptotic higher-order convergence rates for the local parametrization variant.

4. Quadrature rules based on local parametrization of cut tetrahedra

The discussion in Section 3 motivates the use of local parametrization-based quadrature rules in the tetrahedral finite cell method. In the following, we describe our framework that generalizes the element-wise parametrization concept for tetrahedral elements intersected by trimmed NURBS surfaces. Our implementation consists of three basic steps: (1) find intersected elements and classify them into topologically valid and invalid cases; (2) parametrize the cut element domain with Lagrange polynomials if possible; (3) generate geometrically accurate quadrature rules from each local parametrization. There are several contributions in the recent literature that describe different variants for parametrizing cut elements [18, 22, 24–27]. In particular, we would like to point interested readers to the work of FRIES and OMERović [24], which contains a comprehensive discussion of basic concepts and technical details for all common 2D and 3D element types. We also note that the basic concept of element-wise parametrization is closely related to blending techniques employed in higher-order finite element methods [54, 55].

4.1. Identification and classification of valid intersection topologies

We start with a geometric boundary representation that consists of a trimmed NURBS surface immersed into a non-boundary-fitted tetrahedral mesh. In the first step, we identify all elements

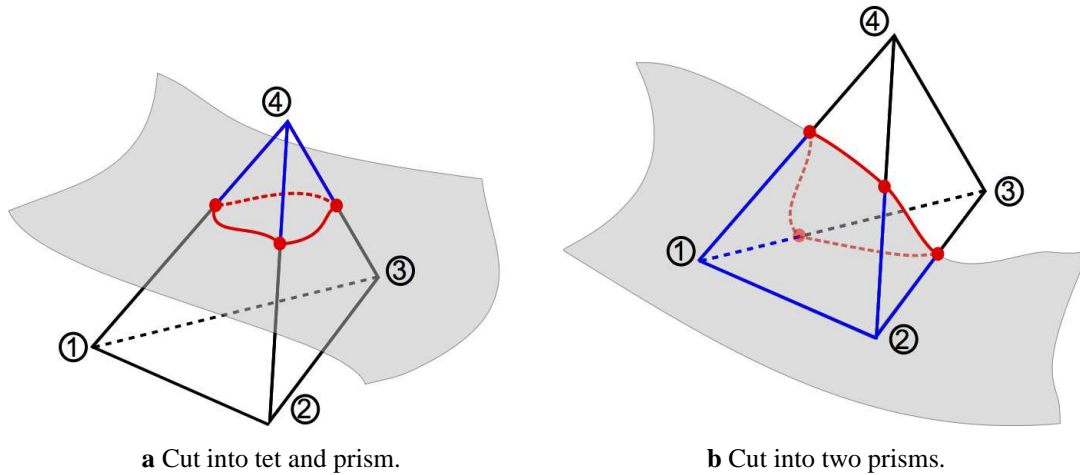


Figure 8: Intersection topologies that allow for parametrization with basic combinations of Lagrange polynomials.

that are intersected by the immersed boundary. To this end, we perform inside/outside tests for all vertex points and several test points distributed over each triangular face of the tetrahedral mesh. For the latter, we use the locations of the 6-point triangular quadrature rule, which we found sufficient for practical spline surfaces. We note that for surfaces with pronounced curvature in coarse unfitted discretizations, one should increase the number of points. The inside/outside test is based on a ray tracing framework for trimmed NURBS surfaces described in [56].

If we find that different points are located on different sides of the surface, we mark the element as cut, otherwise as completely inside or outside. For tetrahedral elements, there are only two intersection topologies that allow for parametrization with basic combinations of Lagrange polynomials. As illustrated in Fig. 8, a tetrahedron can be either split into a sub-tetrahedron and a sub-prism or into two sub-prisms [24, 57]. We determine if a cut element belongs to either of the two valid intersection topologies by carrying out the following tests for each of the four triangular faces:

- If all three vertex points of a face lie on one side of the NURBS surface, all other test points on that face must lie on the same side.
- If vertex points of a face lie on different sides, two edges must be cut with only one cut per edge. If a cut goes through a vertex, the opposite edge or no other edge must be cut.
- If we find a total of three cut edges in the element, we have the tet-prism case. If we find a total of four cut edges in the element, we have the prism-prism case.

For determining intersections with element edges, we employ the same ray tracing framework [56]. The global coordinates of intersection points are stored for use in the next step. We note that there exist fast alternatives to ray tracing that are computationally more efficient. An example is a GPU-accelerated point membership classification based on a finely sampled voxelization of the CAD object and a rendering technique developed by Krishnamurthy et al. [58]. Recent tests in the context of the finite cell method have shown that this method is able to determine an inside/outside identification of half a billion voxels in less than 30 seconds on a GPU of a standard laptop [59].

4.2. Beyond valid intersection topologies

There are a number of situations, when a valid intersection topology as defined above cannot be established. Typical examples are cut elements with edges that are cut more than once or with edges that are not cut at all, although the corresponding face is cut. These situations are quickly encountered even for simple geometries and coarse meshes (see, e.g., [24] for illustrations). We react to these situations by consecutively applying two different remedies. In a first step, we employ subdivision into sub-elements. In many cases, this will restore valid intersection topologies for each sub-element, for which the parametrization procedure can be then applied separately. For all sub-elements, whose intersection topology is still invalid, subdivision can be recursively repeated until a maximum depth is reached or only valid intersection topologies are found. In a second step, if recursive sub-division could not restore a complete set of valid intersection topologies, we resort to the standard recursive subdivision based quadrature of the finite cell method (see Section 2.2). We note that we will also apply the second option, if the geometric model has features, for example sharp edges along trimming curves, that cannot be resolved by higher-order parametrization. We anticipate that some sort of local mesh generation could be applied to fit sub-elements to sharp features (see, e.g., [26]), which is outside the scope of this work.

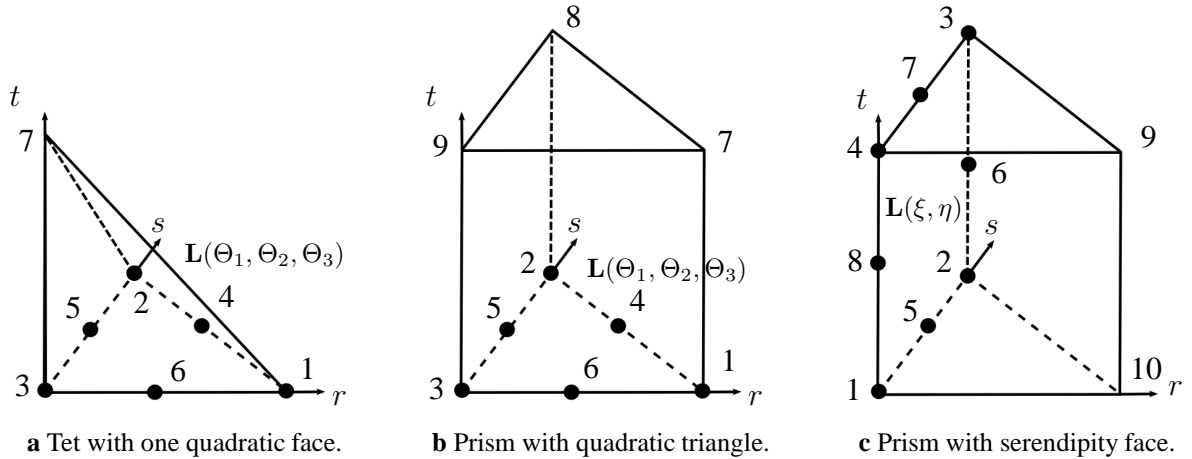


Figure 9: Reference elements for quadratic parametrization employed in this paper.

4.3. Parametrization based on nodal Lagrange polynomials

If the cut element can be characterized by one of two intersection topologies shown in Fig. 8, we can parametrize the relevant part of the element domain with combinations of Lagrange polynomials. These polynomials can be defined in a straightforward way by standard tetrahedral and prism elements with one higher-order face. In the scope of this paper, we employ the following three types of element parametrizations illustrated in Fig. 9: (a) a tetrahedron with a quadratic triangular face, (b) a prism with a quadratic triangular face, and (c) a prism with a quadrilateral face based on the quadratic serendipity element. This concept can be easily extended to higher-order polynomials, if required, by using more nodes on each curved surface.

In the following, we provide some basic technical details for the three cases applied in this work. For triangular faces, we use the basis functions L_i of the 6-noded Lagrange triangle in natural coordinates $\{\theta_1, \theta_2, \theta_3\} = [0, 1]$, which read [60]

$$\begin{aligned} L_1 &= \theta_1(2\theta_1 - 1); & L_2 &= \theta_2(2\theta_2 - 1); & L_3 &= \theta_3(2\theta_3 - 1) \\ L_4 &= 4\theta_1\theta_2; & L_5 &= 4\theta_2\theta_3; & L_6 &= 4\theta_1\theta_3 \end{aligned} \quad (5)$$

The first and second lines represent vertex and edge modes, respectively.

The basis functions L_i of the 8-noded quadrilateral serendipity element in parametric coordinates $\{\xi, \eta\} = [0, 1]$ read [60]

$$\begin{aligned} L_1 &= \frac{1}{4}(1 + \eta)(1 - \xi)(1 + \eta + \xi); & L_2 &= \frac{1}{4}(1 - \eta)(1 - \xi)(1 - \eta + \xi) \\ L_3 &= \frac{1}{4}(1 - \eta)(1 + \xi)(1 - \eta - \xi); & L_4 &= \frac{1}{4}(1 + \eta)(1 + \xi)(1 + \eta - \xi) \\ L_5 &= \frac{1}{2}(1 - \eta^2)(1 - \xi); & L_6 &= \frac{1}{2}(1 - \xi^2)(1 + \eta) \\ L_7 &= \frac{1}{2}(1 - \eta^2)(1 + \xi); & L_8 &= \frac{1}{2}(1 - \xi^2)(1 - \eta) \end{aligned} \quad (6)$$

where the first two lines represent the vertex modes and the last two lines the edge modes. We choose the serendipity over the 9-noded quadrilateral element, since it does not require finding geometric parameters for the center node, but still achieves quadratic convergence.

The geometric mappings of the parametrized volumetric sub-elements are then obtained by blending linear basis functions with the quadratic surface mappings (5) and (6) at the higher-order face. The sub-element geometric maps are parametrized by the coordinates $[\hat{x}_i, \hat{y}_i, \hat{z}_i]^T$ of nodal points. For the sub-tetrahedron with one quadratic triangular face, we obtain the map

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} (r, s, t) = (1 - t) \sum_{i=1}^6 L_i(\theta_1, \theta_2, \theta_3) \cdot \begin{bmatrix} \hat{x}_i \\ \hat{y}_i \\ \hat{z}_i \end{bmatrix} + t \begin{bmatrix} \hat{x}_7 \\ \hat{y}_7 \\ \hat{z}_7 \end{bmatrix} \quad (7)$$

For the sub-prism with one quadratic triangular face, we obtain the map

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} (r, s, t) = (1 - t) \sum_{i=1}^6 L_i(\theta_1, \theta_2, \theta_3) \cdot \begin{bmatrix} \hat{x}_i \\ \hat{y}_i \\ \hat{z}_i \end{bmatrix} + tr \begin{bmatrix} \hat{x}_7 \\ \hat{y}_7 \\ \hat{z}_7 \end{bmatrix} + ts \begin{bmatrix} \hat{x}_8 \\ \hat{y}_8 \\ \hat{z}_8 \end{bmatrix} + t(1 - r - s) \begin{bmatrix} \hat{x}_9 \\ \hat{y}_9 \\ \hat{z}_9 \end{bmatrix} \quad (8)$$

In both (7) and (8), the natural coordinates of the quadratic triangle are $\theta_1 = \frac{r}{1-t}$, $\theta_2 = \frac{s}{1-t}$ and $\theta_3 = \frac{1-r-s}{1-t}$. For the sub-prism with one quadratic quadrilateral face, we obtain the map

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} (r, s, t) = (1 - r) \sum_{i=1}^8 L_i(\xi, \eta) \cdot \begin{bmatrix} \hat{x}_i \\ \hat{y}_i \\ \hat{z}_i \end{bmatrix} + tr \begin{bmatrix} \hat{x}_9 \\ \hat{y}_9 \\ \hat{z}_9 \end{bmatrix} + (1 - t)r \begin{bmatrix} \hat{x}_{10} \\ \hat{y}_{10} \\ \hat{z}_{10} \end{bmatrix} \quad (9)$$

where the parametric coordinates of the serendipity basis functions are $\xi = \frac{2s}{1-r} - 1$ and $\eta = 2t - 1$.

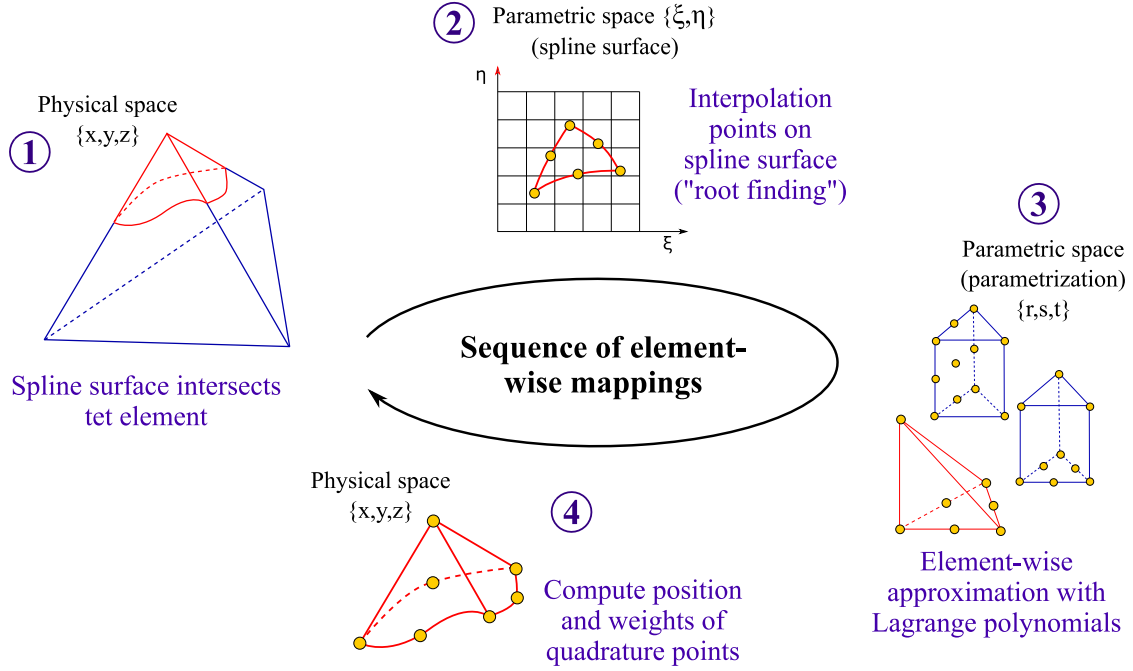


Figure 10: Sequence of mappings that enable the geometrically accurate parametrization of intersected elements.

4.4. Sequence of higher-order mappings

We can now construct a sequence of higher-order mappings that take us from the identification of the intersection topology to the local geometric parametrization of the physical element domain. Figure 10 illustrates the complete process. Once a valid intersection topology for a cut element is found, we use the ray tracing framework [56] to find interpolation points on the element faces that provide geometric parameters for the edge modes of the surface mappings (5) and (6). To this end, we construct a search line on each cut triangular face that connects the mid-point of the uncut edge with the opposite vertex. The ray tracing framework delivers the coordinates of the interpolation nodes in parametric surface coordinates $\{\xi, \eta\}$ of the spline space (yellow dots in Fig. 10). For higher-order parametrizations where several interpolation nodes have to be chosen, optimal search paths for different intersection topologies and element types are given in [24].

We can then recover the corresponding coordinates in physical space $\{x, y, z\}$ by using the NURBS surface map, for which we know B-spline basis functions, control points and weights [39]. The corresponding anisotropic sub-elements carry the parameter space of the geometric parametrization in parametric coordinates $\{r, s, t\}$. Using the physical coordinates $[\hat{x}_i, \hat{y}_i, \hat{z}_i]^T$ of the interpolation nodes as parametric input in expressions (7), (8), and (9), we can parametrize the local geometry in each intersected tetrahedral element, mapping the sub-elements to the physical space. For valid intersection topologies, the framework shown in Fig. 10 is a robust way to establish connections between the different geometric entities. There are a few interesting aspects that we would like to address in the following three remarks.

Remark 1: As we use tetrahedral elements with strictly planar faces, the relation between global

coordinates $\{x, y, z\}$ and barycentric element coordinates $\{\zeta_1, \zeta_2, \zeta_3, \zeta_4\}$, in which the tetrahedral basis functions are defined, can be exactly expressed as [34, 61]

$$\begin{bmatrix} 1 \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix} \quad (10)$$

where the coordinates in the matrix are the vertex points of the original cut element. It is easy to compute the inverse relation of (10) analytically, from which also partial derivatives of each physical coordinate with respect to each barycentric coordinate can be established (see for example [61]). These partial derivatives can be used to map derivatives with respect to barycentric coordinates to derivatives with respect to physical coordinates. Relation (10) also holds for higher-order elements (subparametric mapping).

Remark 2: While the Jacobian matrix for mapping the basis functions is controlled by (10), the integration over the parametrization domain requires the computation of a second “geometric” Jacobian matrix, whose determinant is required for mapping differential elements from the parametric space $\{r, s, t\}$ of the sub-elements to global coordinates $\{x, y, z\}$. This “geometric” Jacobian can be computed in the usual way by re-expressing all basis functions in expressions (7), (8), and (9) in terms of $\{r, s, t\}$ and taking corresponding partial derivatives.

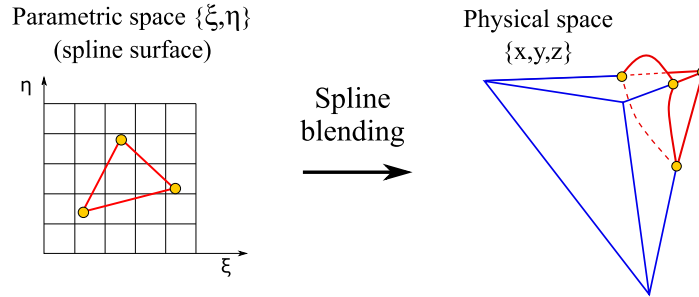
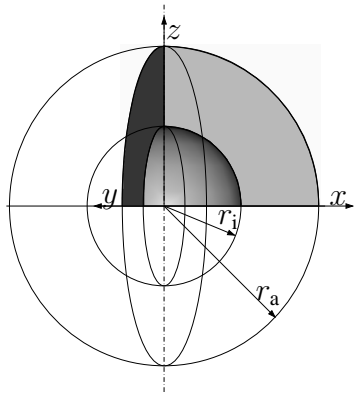
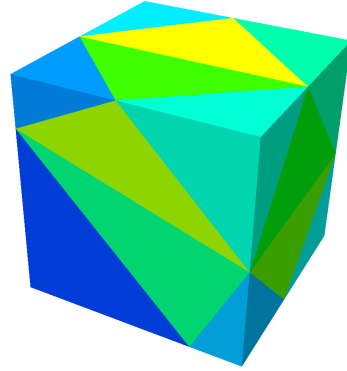


Figure 11: If a linear interpolation between the intersection vertices is assumed in the spline space, the parametrizations in general do not maintain planar faces of the tetrahedral elements.

Remark 3: An important point of our framework is the higher-order polynomial interpolation of the curved spline surface in each intersected element (see Fig. 10). As the interpolation nodes are located on the planar faces of the tetrahedral element, this guarantees that the polynomial parametrizations conform to the planar faces of the intersected tetrahedral element. In this context, we would like to emphasize that applying an exact spline blending method, e.g., in the sense of the NURBS-enhanced finite element method [18, 42, 62], assumes a linear interpolation between the intersection vertices in the parametric space of the spline surface. It is clear that when mapped forward to the physical space the resulting curves do not lie in the planar faces of the tetrahedral element. In particular, they arbitrarily bulge out of the planar element faces as illustrated in Fig. 11.

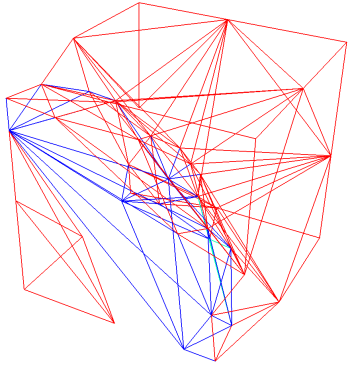


a Geometry.

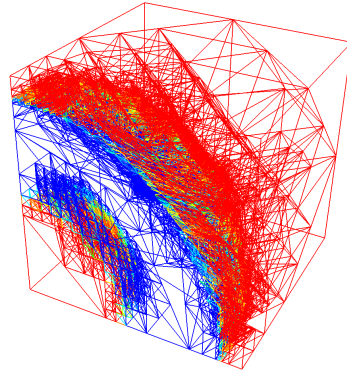


b Tet mesh of the embedding cube.

Figure 12: Octant of a thick spherical shell.

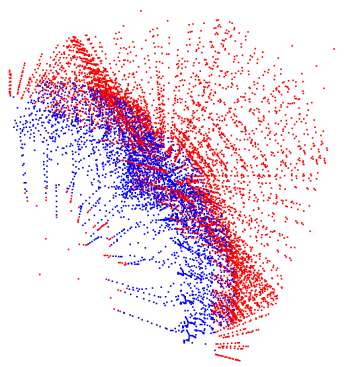


a Quadratic parametrization (curved edges approximated by straight lines).

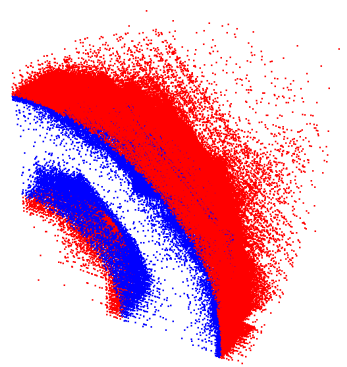


b Recursive subdivision using four levels of sub-cell refinement.

Figure 13: Sub-element structure.



a Quadratic parametrization.



b Recursive subdivision.

Figure 14: Quadrature points corresponding to the sub-element structures shown in Fig. 13.

To close the gap between the original affine non-boundary-fitted tetrahedron, on which finite element basis functions are defined, and the spline based geometric parametrization, one could map the finite element basis functions on the curved geometry via the same exact spline blending functions. However, this requires that common faces of neighboring elements are mapped in exactly the same way to maintain a properly defined geometry without gaps and overlaps. Therefore, re-introducing curved element faces into the non-boundary-fitted mesh leads to several restrictions. For example, in discretizations with both standard boundary-fitted elements and embedded boundaries, the spline mapping in general also deforms the boundary-fitted faces in elements that see both types of boundaries. Another restriction is that switching between several quadrature variants in neighboring elements, e.g. spline blending and subdivision based quadrature, is not possible anymore, because these mappings in general impose different geometric mappings that lead to incompatible common faces.

4.5. Higher-order geometrically accurate quadrature rules

The sequence of mappings illustrated in Fig. 10 provides the basis for generating higher-order quadrature rules that also represent the geometry of a cut element with higher-order accuracy. For tetrahedral sub-elements, we use a 5-point quadrature rule for the integration of quadratic basis functions and an 11-point quadrature rule for the integration of cubic basis functions [51]. For prism-type sub-elements, we employ tensor-product extensions of 6- and 12-point triangular monomial rules, multiplied by corresponding Gauss-Legendre rules in a tensor-product sense, for integrating over quadratic and cubic basis functions, respectively. These rules are not exact, as the “geometric” Jacobian is a rational function. We note that any other monomial rule can be used as well that yields the desired accuracy (see for example [61, 63–65]). Before using a local parametrization based quadrature rule we check the “geometric” Jacobian of the parametrization mapping to ensure that it is well defined and larger than zero. If it is smaller than zero, which could potentially happen in case of intersecting spline surfaces with very high curvature [24], we initiate the subdivision procedure described in Section 4.2.

As a first test case, we use our quadrature framework for non-boundary-fitted tetrahedral elements for the integration of the volume of a thick spherical shell shown in Fig. 12a. The inner and outer radii are chosen as $R_i=50$ and $R_a=100$, respectively. The domain is embedded in a cube, which is discretized by a sequence of tetrahedral meshes. The coarsest mesh is shown in Fig. 12b. We also compare the performance of our framework based on element-wise quadratic parametrization with the recursive subdivision approach that has been used within the finite cell method so far. Figures 13a and 13b plot the sub-elements generated by the element-wise parametrization and by recursive subdivision of level 4, respectively. We note that we choose this particular level of sub-cell refinement, as it achieves approximately the same accuracy as the parametrization approach for mesh densities of practical engineering interest (see Section 5). Figures 14a and 14b plot the corresponding quadrature points generated by the two quadrature variants. Blue sub-elements and points are within the domain, red points are outside and thus neglected in the quadrature process.

We observe from these plots that the recursive subdivision approach requires significantly more quadrature points than the parametrization approach to achieve the same level of accuracy. This is confirmed by Figs. 15a and 15b. They plot the relative error in the volume integral and the quadrature cost in terms of the number of points that are required to achieve this specific error

level with respect to increasing mesh size. For the third mesh, the two quadrature variants shown in Figs. 14a and 14b achieve the same accuracy, but the recursive subdivision approach at level 4 requires approximately 8 times more point evaluations. For the fourth tetrahedral mesh, the accuracy of the local parametrization approach is already out of reach for recursive subdivision at level 5, which requires 15 times the number of quadrature points and is the maximum number of levels computable on a standard laptop.

Remark 4: There is a cost involved for generating parametrization-based quadrature rules. In particular, the root-finding step can be computationally expensive, when implemented in a naive fashion. However, the final realistic computational effort for geometry-related operations is difficult to estimate, as the implementation of corresponding algorithms in our research code could be significantly improved, based on an abundance of related work in fields such as computer graphics, computational visualization, or animation [66]. In addition, nonlinear or multiphysics simulations, where computational cost is decisive for the feasibility of an analysis, typically require a large number of iterations with the same quadrature points. From an analysis viewpoint, the reparametrization cost quickly becomes insignificant, no matter how expensive, and the number of quadrature points remains the only significant measure. In Figure 15 and all other figures in the remainder of this paper, we therefore use the number of quadrature points as the only measure for computational cost of a quadrature method.

5. Numerical examples in three dimensions

We are now in a position to test the performance of element-wise parametrization based volume quadrature of intersected elements in the tetrahedral finite cell method. To this end, we will consider two benchmark problems and an application oriented example of a coupled bone/implant simulation. The latter involves geometric models based on fuzzy imaging data and sharp CAD

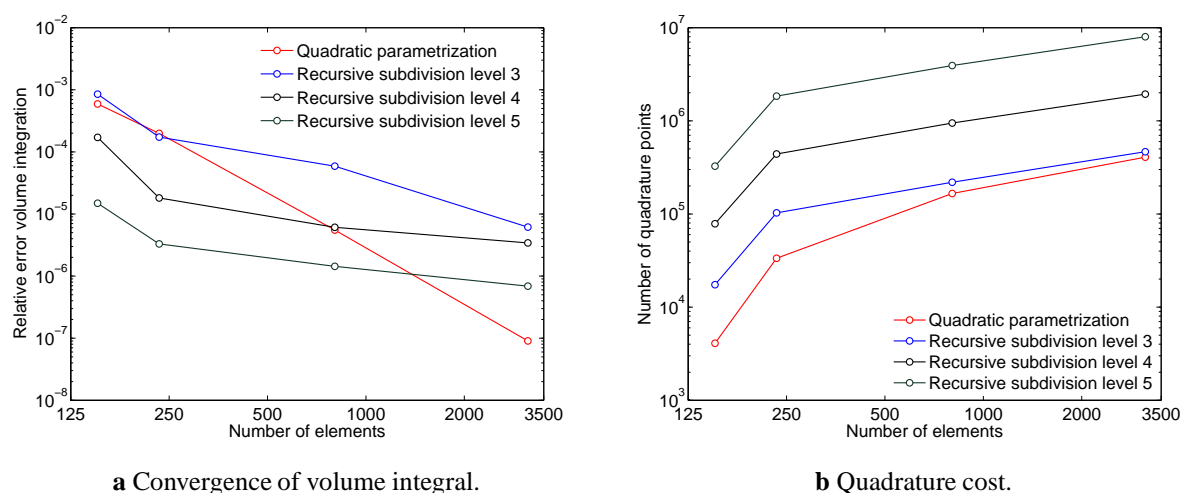


Figure 15: Comparison of element-wise parametrization and recursive-subdivision based quadrature for numerically integrating an octant of a thick spherical shell over a non-boundary-fitted tetrahedral mesh.

boundary representations, providing an opportunity to highlight the advantages of recursive subdivision and parametrization based quadrature variants. In the scope of this work, we focus on local parametrizations based on quadratic polynomials, using the framework discussed in the previous section. We emphasize that second-order geometric accuracy is the most relevant case from a practical engineering viewpoint, as the finite cell method has been shown to be particularly attractive at moderate polynomial degrees 2 and 3, representing a compromise between higher-order convergence and a reasonably conditioned system [34]. We note that our framework can be extended in a straightforward way to geometric parametrizations higher than second order [24].

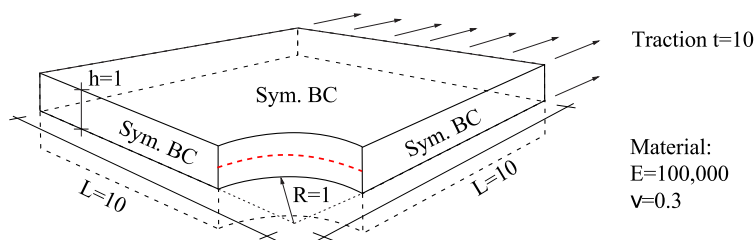


Figure 16: Three-dimensional thick plate with a circular hole.

5.1. Plate with a cylindrical hole

As a first benchmark, we consider a thick plate with a circular hole under uniform tension shown in Fig. 16. We reduce the system to one octant of the original domain and apply symmetry boundary conditions in a weak sense by Nitsche’s method. We generate a sequence of four uniformly refined non-boundary-fitted tetrahedral meshes that discretize the embedding box irrespective of the hole. Quadrature of intersected elements is performed with recursive subdivision at different levels of sub-cell refinement and with element-wise parametrization. Figures 17a and 17b show the corresponding quadrature points for the coarsest non-boundary-fitted mesh.

Figure 18 plots the strain energy error versus the number of degrees of freedom for the four different meshes. We compute the relative error in strain energy norm defined as [43, 44, 67]

$$e_r = \sqrt{\frac{|U_{num} - U_{ref}|}{U_{ref}}} \quad (11)$$

where U_{num} represents the numerical strain energy obtained for a specific discretization, and U_{ref} is a reference strain energy computed with an overkill discretization [34]. We observe that the finite cell method with local parametrization achieves optimal rates of convergence. For the same accuracy level, the subdivision approach requires at least 3 levels of recursive sub-cell refinement.

In Fig. 19, we plot the circumferential stress, evaluated along the circular boundary of the hole ($\theta = 0 \dots \pi/2$), for different quadrature variants in the second mesh of our series. The corresponding location is illustrated in Fig. 16 as a dashed red line. We observe that the geometric error of the subdivision approach with just one level of sub-cell refinement leads to a significant error in the boundary stresses. The error can be mitigated by adding additional sub-cells, but we need at least four levels to match the accuracy of the element-wise parametrization approach.

Geometrically accurate quadrature rules based on parametrization guarantee an accurate stress solution directly at the embedded boundary from the onset. The results of Fig. 19 highlight again the importance of geometric accuracy in embedded domain analysis, re-enforcing the opening discussion of Section 3 that has been based on initial 2D results. We emphasize again that all results of Fig. 19 have been generated from the same mesh with the same quadratic finite element basis functions, with the only difference being the quadrature rule in intersected elements.

5.2. Thick spherical shell under internal pressure

As a second example, we examine the spherical thick shell under internal pressure, whose geometry corresponds to the example shown in Fig. 12a. Due to symmetry, we consider only one eighth of the original problem. Following the derivation shown in [68, 69], there exists an analytical solution in spherical coordinates $\{r, \phi, \theta\}$ of the form

$$\sigma_r = -\frac{p}{\left(\frac{R_a}{R_i}\right)^3 - 1} \left[\left(\frac{R_a}{r}\right)^3 - 1 \right] \quad (12)$$

$$\sigma_\phi = \sigma_\theta = \frac{p}{\left(\frac{R_a}{R_i}\right)^3 - 1} \left[\frac{1}{2} \left(\frac{R_a}{r}\right)^3 + 1 \right] \quad (13)$$

$$u_r = \frac{r}{E} [(1 - \nu)\sigma_\theta - \nu\sigma_r] \quad (14)$$

The rest of the displacement components are zero due to symmetry. We choose an inner radius $R_i=50$, an outer radius $R_a=100$, Young's modulus $E=10,000$ and an internal pressure $p=50$.

We discretize an embedding box in such a way that both doubly curved surfaces are intersected by the tetrahedral mesh. The coarsest discretization of a series of four uniformly refined meshes is

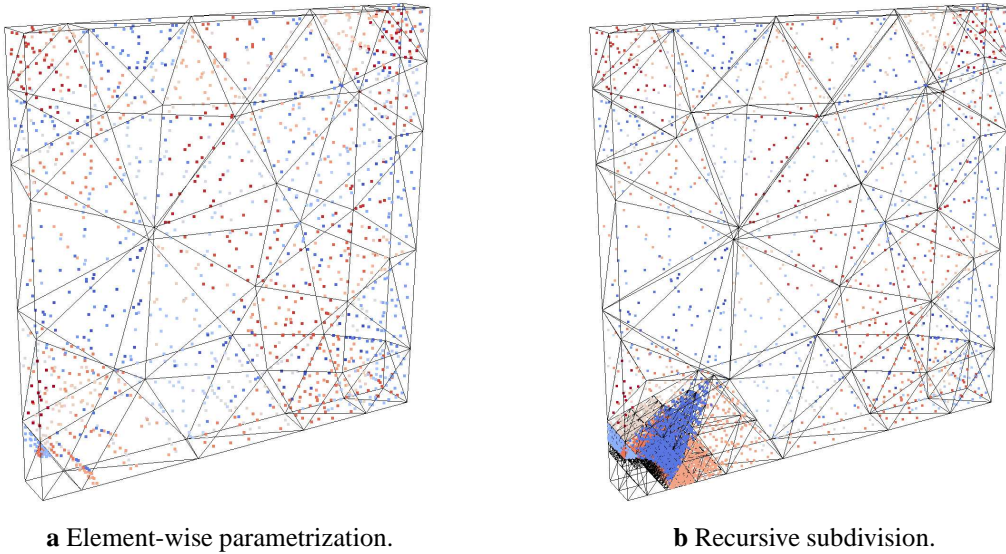


Figure 17: Coarsest mesh for the plate problem with integration points derived from the two quadrature variants.

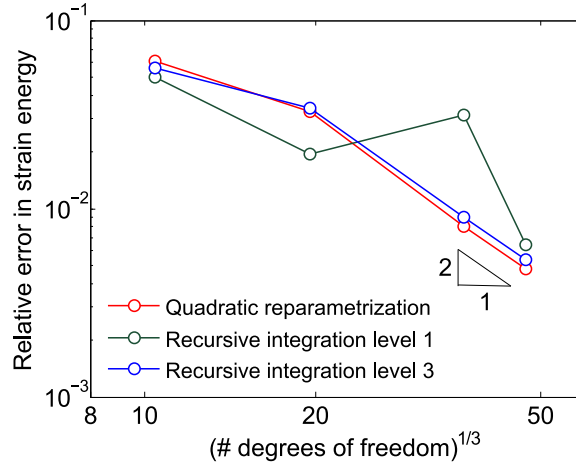


Figure 18: Convergence of the relative error in strain energy for the plate problem.

shown in Fig. 12b and the corresponding quadrature points generated from element-wise quadratic parametrization and recursive subdivision are plotted in Figs. 14a and 14b, respectively. We employ quadratic basis functions and impose symmetry boundary conditions weakly using Nitsche’s method as formulated in (2) and (3).

Figures 20a to 20c plot the von Mises stresses in a section of the thick shell computed with the third mesh from our series and different quadrature variants. The three plots show that in the bulk of the domain the stress results are equivalent irrespective of the quadrature method used. However, the stress accuracy at the surface varies greatly between the methods. We observe in Fig. 20a that using recursive subdivision with only one level of adaptive refinement leads to large oscillations in the surface stress. We can improve the stress accuracy by adding more levels (see Fig. 20b), since this increases the geometric fidelity at the embedded boundary. Figure 20c demonstrates that quadrature points based on quadratic parametrization remove the oscillations entirely, leading to an accurate stress solution at the surface.

Figure 21a compares the convergence in strain energy obtained with quadratic non-boundary-fitted tetrahedral elements and different quadrature variants as the mesh is uniformly refined. We once more observe the characteristic convergence behavior that we have already observed in Section 3. Recursive subdivision based quadrature with only one level of sub-cell refinement leads to sub-optimal rates of convergence due to its low-order geometry approximation. Increasing the number of recursive refinement levels ensures optimal rates in the pre-asymptotic range, but leads to a sudden decrease asymptotically, as soon as the low-order geometric convergence of the subdivision approach reaches its accuracy limit. Quadratic parametrization and associated quadrature points represent the geometry with a second-order rate, which enables optimal rates of convergence in the energy error at all accuracy levels.

Figure 21b repeats the same study for cubic tetrahedra, showing strain energy errors in the pre-asymptotic range. We observe that the subdivision approach at both one and three levels of recursive sub-cell refinement does not lead to full accuracy. **The parametrization approach based on quadratic Lagrange polynomials shows good accuracy, although the geometry representation is one order lower than the cubic approximation of the solution. We observe that for the targeted**

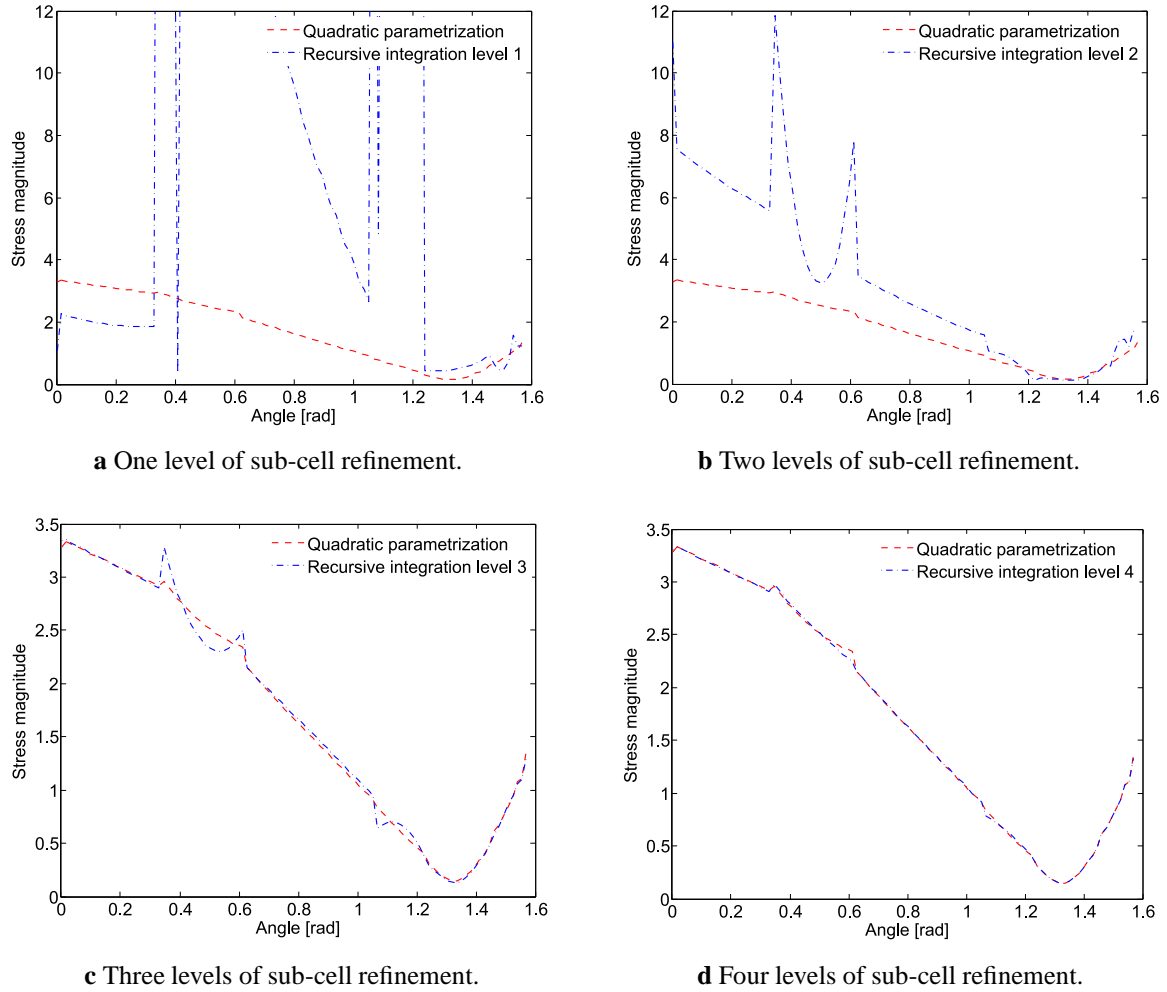


Figure 19: Accuracy of the circumferential stress along the circular boundary for different levels of sub-cell refinement. Note that we always use the same mesh and quadratic finite element basis functions.

error level can be reached before the lower quadratic approximation of the geometry, on which quadrature points are based, can affect the strain energy convergence, governed the cubic approximation of the solution fields. This indicates that for engineering accuracy, quadratic element-wise parametrization seems to be a viable approach also for tetrahedra with cubic or higher-order basis functions.

Figures 22a and 22b demonstrate the advantages of element-wise parametrization from a computational cost point of view. They compare the strain energy accuracy computed on cubic tetrahedral meshes with respect to the number of quadrature points at different levels of sub-cell refinement. Accuracy is measured by the ratio between the “best possible” energy error on a given mesh provided there is no geometry error and the actual strain energy error obtained with a given quadrature variant. We use element-wise quadratic parametrization to approximate the “best possible” strain energy error, as this is the most accurate way of integrating intersected elements currently available to us. We observe that for achieving a ratio close to one, the recursive subdivision ap-

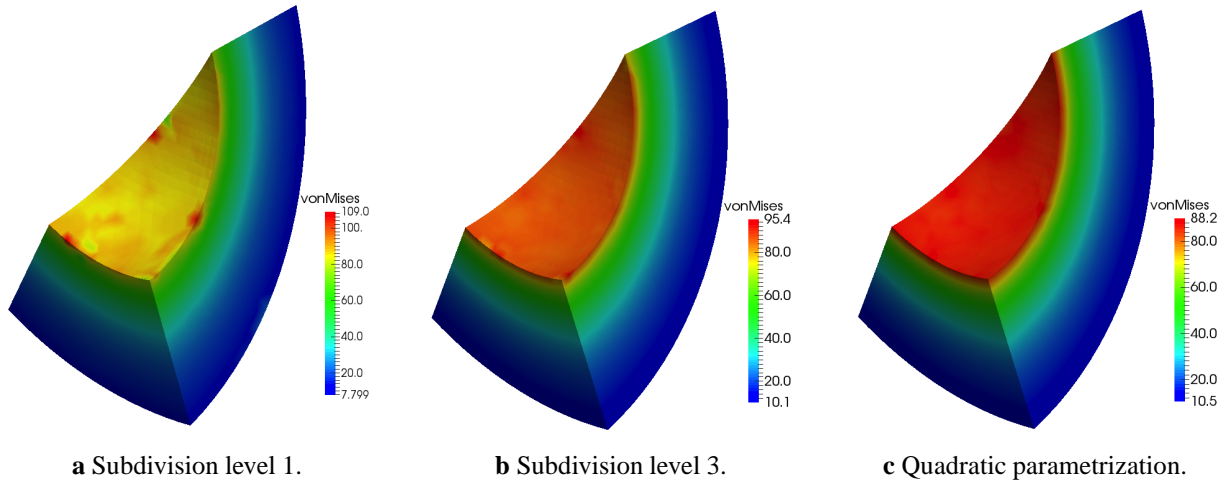


Figure 20: Accuracy of the von Mises stress at the inner circular boundary, computed with the tetrahedral finite cell method on the same mesh based on recursive subdivision at different sub-cell levels and quadratic parametrization.

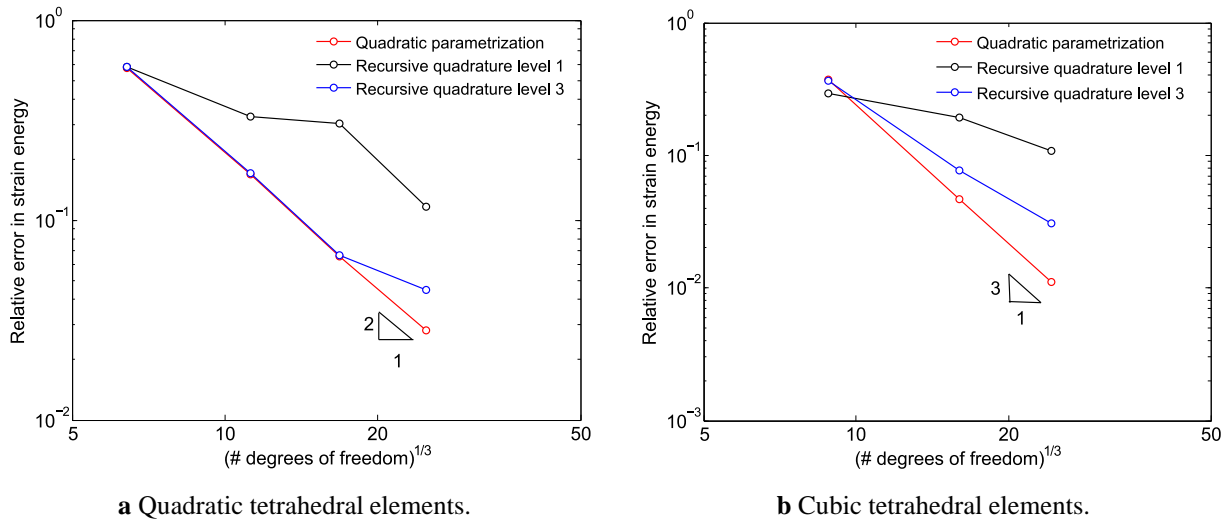
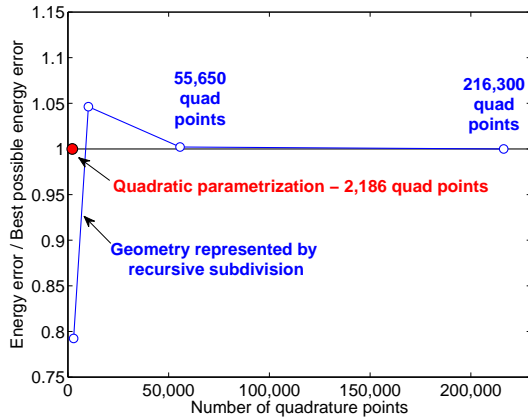


Figure 21: Convergence of the relative error in strain energy for the thick spherical shell problem..

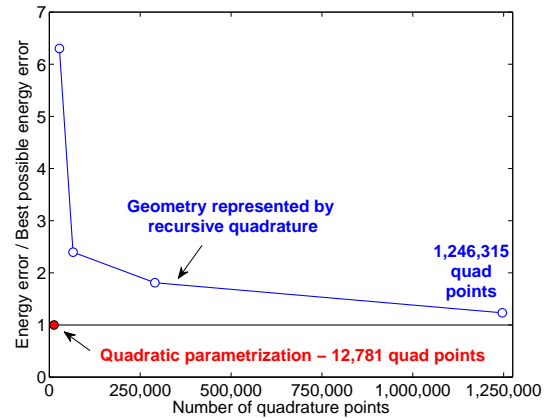
proach requires in general between one and two orders of magnitude more point evaluations than quadrature based on element-wise parametrization. Figures 22a and 22b repeat the same study for the initial coarse mesh and the mesh after two uniform mesh refinement steps. We see that the difference in point evaluations at the same accuracy level increases with mesh refinement, since the higher approximation accuracy in the finer mesh also requires more levels of recursive refinement to achieve the corresponding geometric fidelity.

5.3. Fractured femur bone with a nail implant

The previous examples demonstrated the advantages in terms of improved accuracy and computational efficiency that element-wise parametrization of cut elements brings about for the tetrahedral finite cell method on domains with sharply defined boundaries. The following example



a Coarsest mesh (28 tet elements).



b Finer mesh (802 tet elements).

Figure 22: Ratio of actual vs. best possible energy error with increasing levels of sub-cell refinement. The first blue point on the left corresponds to one level of sub-cell refinement, which is then increased in steps of one.

illustrates that the success of a particular quadrature scheme depends very much on the type of geometric model under consideration. In particular, we will demonstrate that depending on the specific geometric model, both subdivision based quadrature and element-wise parametrization play an important role to preserve the flexibility of the TetFCM.

5.3.1. Fuzzy imaging data vs. sharply defined B-rep models

We present a coupled bone/implant simulation of a fractured human femur bone, which involves both diffuse medical imaging data and a CAD model based on sharply defined boundary data (B-rep) [66]. From an application point of view, the problem to be analyzed is part of a patient-specific workflow for simulation-based performance analysis of coupled bone/implant configurations. It supports physicians in reliably identifying the most suitable implant type and its optimal position in a fractured bone of a specific patient. Predicting implant performance constitutes the basis for an effective and reliable treatment of bone fractures.

Patient-specific stress and fracture simulations need to be based on imaging data of an individual bone. Figure 23a shows an X-ray of a broken bone that illustrates the fuzziness of the underlying imaging based geometric information. Figure 23b plots data obtained from quantitative computed tomography (qCT) scans after the layer-wise images have been transferred into a 3D rasterized voxel structure. Each voxel contains a color value that can be associated with the bone's mineral density. Based on experimental observations, an isotropic heterogeneous linear elastic material can be assumed [70–72]. The distribution of the corresponding Young's modulus

¹Adapted from “Fixing Hip Fractures” by S. Mears, http://www.hopkinsmedicine.org/gec/series/fixing_hip_fractures

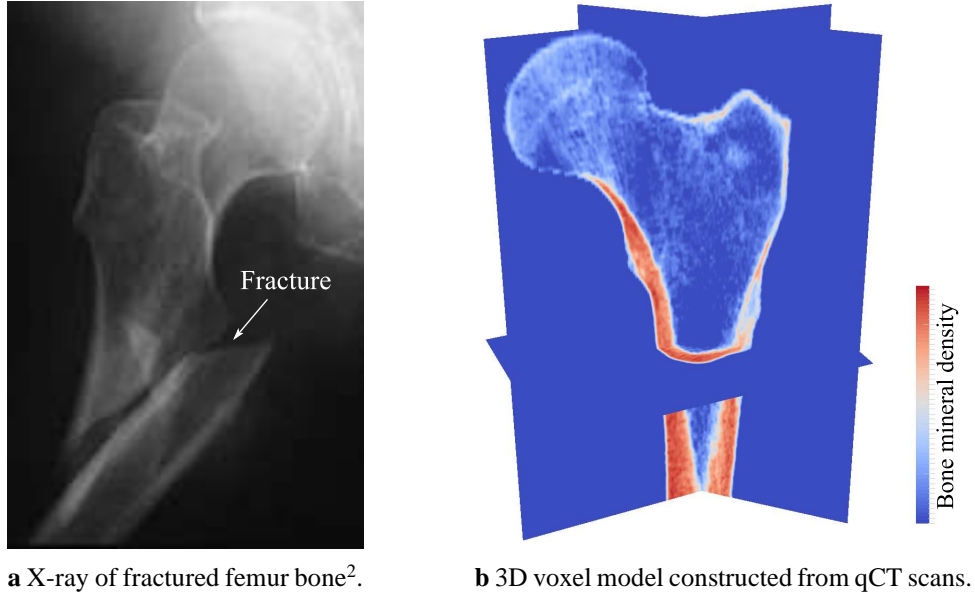


Figure 23: Diffuse patient-specific imaging data of human femur bones.

can be inferred at each image point from the following empirical relations [72, 73]

$$\rho_{ash} = (1.22 \rho_{eqm} + 0.0523) \text{ [g/cm}^3\text{]} \quad (15)$$

$$E_{cort} = 10200 \times \rho_{ash}^{2.01} \text{ [MPa]} \quad (16)$$

$$E_{trab} = 5307 \times \rho_{ash} + 469 \text{ [MPa]} \quad (17)$$

where ρ_{eqm} and ρ_{ash} denote the mineral density and the ash density, respectively [73]. Depending on the local density, we use relation (16) for the cortical region (stiffer material in the outer part) or relation (17) for the trabecular region (spongy material in the inner part). In addition, we assume a homogeneous Poisson's ratio $\nu = 0.3$.

In contrast to the bone, the implant is a well-designed and optimized technical device, manufactured from titanium with highly homogeneous material parameters, Young's modulus $E = 116$ GPa and Poisson's ratio $\nu = 0.3$, and a geometry that is known exactly from the CAD model. Figures 24a and 24b show an X-ray of a nail implant under operating conditions and a plot of the technical device itself along with corresponding B-rep NURBS surfaces and control points of the CAD model. The CAD model uses the trimming paradigm to represent a perfectly bonded joint.

5.3.2. Phase-field modeling of brittle fracture

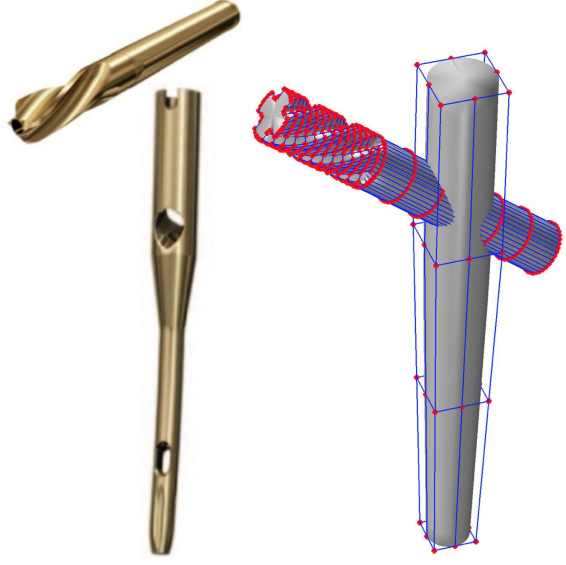
Our representation of a crack is based on a phase-field model for brittle fracture [74–77], which is represented in variational form for quasistatic conditions by the following multifield equations

$$\int \left(\frac{4l_0 \mathcal{H}_0(x)}{\mathcal{G}_c} + 1 \right) c q \, d\Omega + \int 4l_0^2 \nabla c \nabla q \, d\Omega = \int q \, d\Omega \quad (18)$$

³Adapted from “Fixing Hip Fractures” by S. Mears, http://www.hopkinsmedicine.org/gec/series/fixing_hip_fractures



a X-ray of a femur bone with a nail implant³.



b Technical device itself, related CAD model.

Figure 24: A nail implant device defined by sharp boundary surfaces.

$$\int (\boldsymbol{\sigma}^+ + \boldsymbol{\sigma}^-) : \nabla \dot{\boldsymbol{w}} \, d\Omega = \int \boldsymbol{b} \cdot \boldsymbol{w} \, d\Omega + \int \boldsymbol{t} \cdot \boldsymbol{w} \, d\partial\Omega \quad (19)$$

The pairs $\{\boldsymbol{u}, \boldsymbol{w}\}$ and $\{c, q\}$ represent the displacement and phase-field solutions and corresponding test functions. Material parameters involved are the energy release rate \mathcal{G}_c , the characteristic length scale l_0 , and the Lamé parameters of linear elasticity λ and μ . The tensile and compressive parts of the stress tensor read

$$\boldsymbol{\sigma}^+ := c^2 (\lambda \langle \text{tr}(\boldsymbol{\varepsilon}) \rangle^+ \boldsymbol{I} + 2\mu \boldsymbol{\varepsilon}^+) \quad (20)$$

$$\boldsymbol{\sigma}^- := \lambda \langle \text{tr}(\boldsymbol{\varepsilon}) \rangle^- \boldsymbol{I} + 2\mu \boldsymbol{\varepsilon}^- \quad (21)$$

which are based on a corresponding additive split of the strain tensor. \mathcal{H}_0 is a history function that tracks the maximum strain energy induced by the tensile part of the strain tensor. The phase-field part (18) requires homogeneous Neumann boundary conditions, the elasticity part (19) the usual traction and displacement constraints.

The basic idea of the phase-field fracture model (18) and (19) is to represent cracks by a continuous scalar field c that has a value of one away from the crack and is zero at the crack location. The phase-field serves as a multiplication factor to tensile energy components in (20) such that it locally penalizes the capability of the material to carry tensile stress at the crack location. The diffusiveness of the crack approximation is controlled by the length-scale parameter l_0 . From a numerical point of view, the diffusive approximation of the crack by a continuous phase-field eliminates the need for explicit discontinuities in the mesh.

5.3.3. Bone/implant coupling based on Nitsche's method

In the scope of the present work, we assume perfect bonding at the interface between the bone and the implant, neglecting any nonlinear or cohesive effects at the interface. The bone and the implant domains can therefore be coupled weakly by introducing an interface term [21, 78] into the variational form of Nitsche's method (2) that extends the bilinear form as follows

$$\begin{aligned} \delta W_{\mathbf{K}}(\mathbf{u}, \delta \mathbf{u}) = & \sum_{\mathcal{K}} \int_{\mathcal{K}} \boldsymbol{\sigma}_h : \delta \boldsymbol{\varepsilon} \, d\Omega - \gamma^* \int_{\Gamma^*} \llbracket \mathbf{u} \rrbracket : \{ \delta \boldsymbol{\sigma} \} \, d\Gamma - \int_{\Gamma^*} \{ \boldsymbol{\sigma} \} : \llbracket \delta \mathbf{u} \rrbracket \, d\Gamma \\ & - \gamma \int_{\Gamma_D} \mathbf{u} \cdot (\delta \boldsymbol{\sigma} \cdot \mathbf{n}^+) \, d\Gamma - \int_{\Gamma_D} (\boldsymbol{\sigma} \cdot \mathbf{n}^+) \cdot \delta \mathbf{u} \, d\Gamma \\ & + \beta^* \int_{\Gamma^*} \llbracket \mathbf{u} \rrbracket \cdot \llbracket \delta \mathbf{u} \rrbracket \, d\Gamma + \beta \int_{\Gamma^*} \mathbf{u} \cdot \delta \mathbf{u} \, d\Gamma \end{aligned} \quad (22)$$

where quantities denoted with a star are defined on the bone/implant interface Γ^* . The jump and average operators in (22) are defined as

$$\llbracket \mathbf{u} \rrbracket = \mathbf{u}^+ \otimes \mathbf{n}^+ + \mathbf{u}^- \otimes \mathbf{n}^- \quad (23)$$

$$\{ \boldsymbol{\sigma} \} = \frac{1}{2} (\boldsymbol{\sigma}^+ + \boldsymbol{\sigma}^-) \quad (24)$$

With $\gamma^* = 1$, we obtain the standard symmetric form of Nitsche's method, which we use in this paper. It requires a stabilization parameter β^* that can be determined from a generalized eigenvalue problem [3, 52, 79]. In the present study, we choose the parameter empirically to be as small as possible. We note that for $\gamma^* = -1$, we obtain a parameter-free non-symmetric Nitsche method [49, 50], which has been shown to work accurately and robustly for non-boundary-fitted discretizations of linear elastic problems without stabilization (i.e., $\beta^* = 0$) [21].

5.3.4. Supporting accuracy and flexibility of the TetFCM by adequate quadrature

We discretize the coupled bone/implant problem with two independent non-boundary-fitted tetrahedral meshes, which are plotted in Figs. 25a and 25b for the bone and the implant, respectively. The geometric basis for the patient-specific bone analysis is the voxel model shown in Fig. 23b, which provides the qCT-based spatial distribution of Young's modulus from (16) and (17). For a fuzzy voxel model, the concept of intersected elements does not directly apply, as there exists no sharply defined boundary of the problem domain. We therefore leverage the subdivision approach to take into account the inhomogeneous stiffness distribution at the quadrature level. We first locate all tetrahedral elements that are completely located outside of the bone domain. This requires that the color value of all voxels located within a specific element are below a predefined stiffness threshold. All elements outside of the bone domain are then removed from the mesh.

Second, we subdivide all remaining tetrahedral elements into sub-cells. The sub-cell resolution is chosen such that the density of the resulting quadrature points corresponds to the voxel resolution. As a consequence, each voxel is associated with at least one quadrature point, so that the voxel information is fully taken into account during the formation of the stiffness matrix. We emphasize that a finer resolution of quadrature points should be avoided, as it could resolve sharp interfaces and re-entrant corners between single voxels, which are an artifact of the geometric

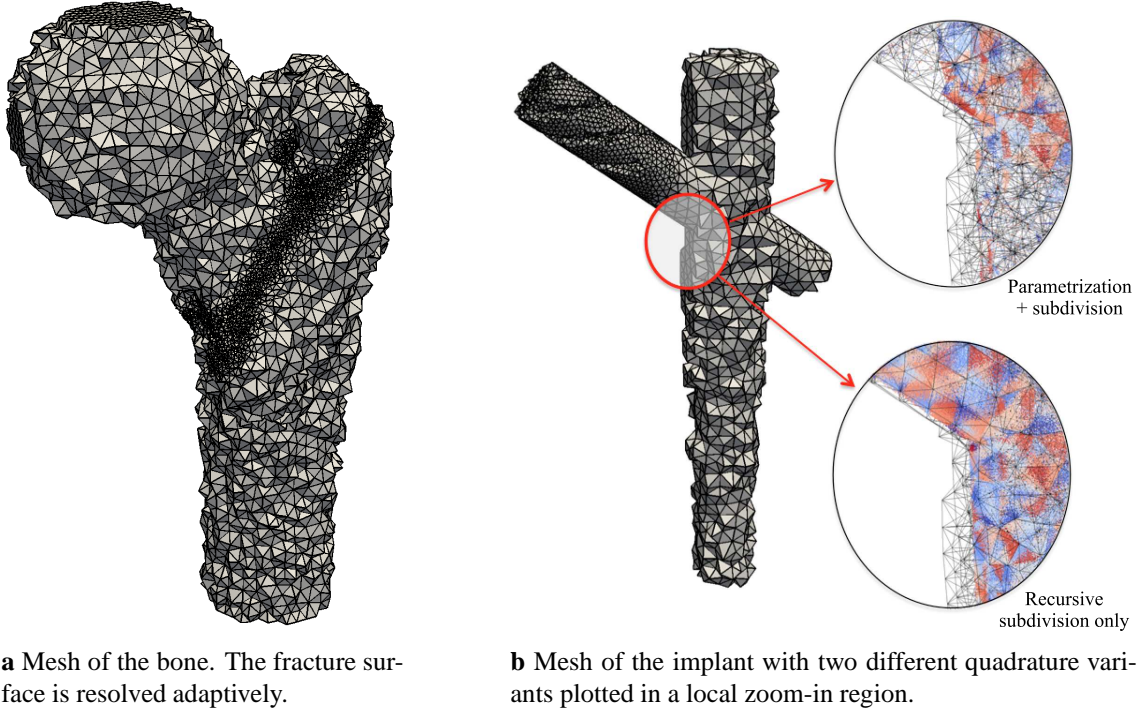


Figure 25: Non-boundary-fitted tetrahedral meshes for the bone and the implant.

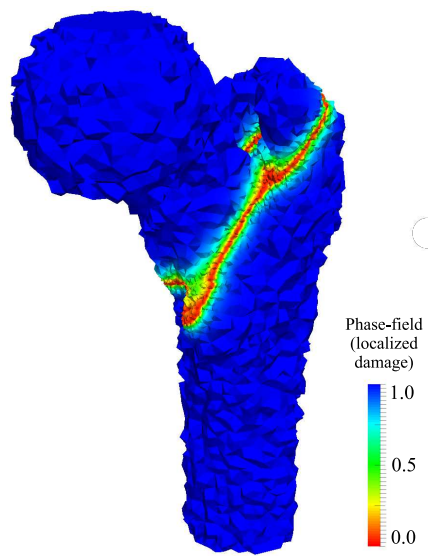
model. In voxel based analysis, the abundance of points of subdivision based quadrature is a clear benefit, as it is the essential mechanism that enables the incorporation of the influence of all voxels in a coarse finite element discretization that is significantly below the voxel resolution. In this sense, the subdivision based quadrature approach can be interpreted as a simple type of homogenization procedure [80].

Having set up the image based model of the bone, we induce an existing fully developed fracture by solving the phase-field equation (18) with a given static \mathcal{H}_0 of the form

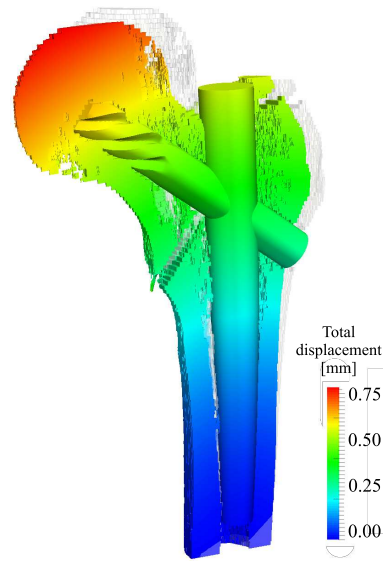
$$\mathcal{H}_0(x) = \begin{cases} B \frac{\mathcal{G}_c}{4l_0} \left(1 - \frac{d}{l_0}\right) & \text{if } d(x) \leq l_0 \\ 0.0 & \text{if } d(x) > l_0 \end{cases} \quad (25)$$

where $B = 1,000$ [81]. In our simulation, \mathcal{H}_0 represents a typical fracture plane between the greater and the lesser trochanters, where approximately 40% of all hip fractures happen. We note that (25) eliminates the material property \mathcal{G}_c , reducing the number of open parameters to the characteristic length scale l_0 , which we choose as $l_0 = h/2$ with respect to the mesh size h at the crack location.

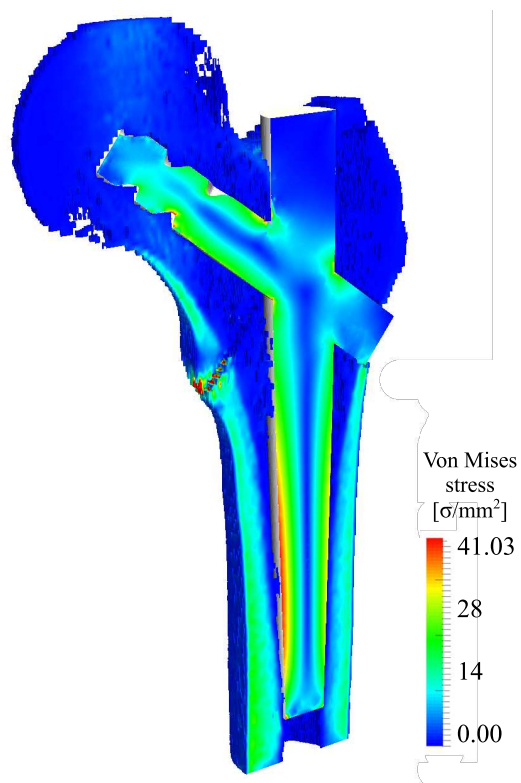
We adaptively resolve the fracture region by a finer mesh (see Fig. 25a), leveraging straightforward mesh refinement in three dimensions as one of the central advantages of the TetFCM [34]. The corresponding phase-field solution is plotted in Fig. 26a. The fracture topology in a typical application scenario is given in terms of fuzzy images (see e.g. Fig. 23a), which impedes the extraction of an exact sharp fracture surface. Therefore, the diffuse fracture representation conceptually fits well in this simulation framework.



a Phase-field solution plotted on the non-boundary-fitted mesh.



b Displacements on the deformed bone/implant configuration.



c Von Mises stresses in the bone and the implant, plotted at a central cutting plane.

Figure 26: Coupled bone/implant analysis based on the TetFCM with flexibly applied quadrature variants, a phase-field approach to fracture, and perfect bond Nitsche coupling.

Being based on a sharp B-rep model, the implant fully benefits from the advantages of the quadrature approach based on element-wise parametrization developed in this paper. As we are given the exact smooth spline surface of the implant, we can derive quadratic parametrizations in a large part of the intersected elements, which serve as a higher-order accurate geometric map for corresponding quadrature points. The presence of sharp edges along trimming curves and at the outer edges of the two implant shafts lead to intersection cases that are not considered valid in the scope of the present paper. This requires the application of the remedies that we discussed in Section 4.2. In the present case, we apply recursive subdivision with three levels of adaptive refinement. In Fig. 25b, the quadrature points generated by element-wise parametrization and recursive subdivision are plotted for a zoom-in close to the joint of the two shafts. We observe that element-wise parametrization significantly reduces the number of quadrature points away from the joint. In the area around the joint the density of quadrature points remains unchanged, as both variants currently use subdivision quadrature due to the presence of sharp edges.

In the final step of the analysis, we tie the bone and implant meshes together, using the coupling (22). Taking advantage of the exact CAD surface representation, we generate surface quadrature points for integrating coupling terms directly based on the Bézier elements of the spline parametrization [59]. We note that although we assume perfect bonding, we are fully taking into account the varying local bone stiffness along the coupling interface. In the limiting case, we are coupling the full-stiffness titanium shaft of the implant to a zero stiffness voxel of the bone mesh, which will effectively replicate a free surface on the implant. The imposition of homogeneous Neumann boundary conditions in the phase-field equation (18) at the bone surface does not pose a problem in the TetFCM, since this can be achieved without surface quadrature. For the elasticity equation (19), we apply a load of 1000 N on the bone head distributed over a spherically shaped loading area [6]. Displacement boundary conditions at the bones lower end are weakly enforced with Nitsches method.

Figures 26b and 26c plot the coupled bone/implant displacements on the deformed configuration and the von Mises stress over a surface that cuts the two non-boundary-fitted meshes in half. The plots demonstrate the successful coupling between the bone and the implant that is essential for bridging the intertrochanteric fracture. We observe that in accordance with our tests of Fig. 20, element-wise parametrization based quadrature enables an oscillation free stress representation. Due to the fuzziness of the image based geometric basis of the bone, the accuracy of stresses at a specific point of the bone may vary, although the overall stress distribution from a global viewpoint is accurate (see e.g. [6]). The disadvantage of subdivision based quadrature in terms of point-wise accuracy on a surface therefore carries far less weight in image based analysis.

6. Summary and conclusions

The goal of this paper was to highlight the role of two different quadrature variants for intersected elements in the context of the tetrahedral finite cell method (TetFCM). On the one hand stands subdivision quadrature that is based on the recursive refinement of quadrature sub-cells in cut elements and has been a fundamental component of the finite cell method since its beginnings. On the other hand stands the element-wise parametrization of cut elements by higher-order Lagrange polynomials. It accommodates higher-order geometric fidelity that has been recently

shown to be a key ingredient for preserving optimal accuracy in embedded domain-type methods. Recalling a basic proof given by Strang and Fix on geometric accuracy in isoparametric finite elements, we outlined explanations for the break-down of convergence rates in the asymptotic range due to the geometric error introduced by recursive subdivision quadrature. We also confirmed that element-wise parametrization with Lagrange polynomials that match the order of the basis functions for the approximation of the solution do not produce such a geometric error, thus preserving optimal convergence rates in the asymptotic limit.

In the technical part of the paper, we described a framework for parametrizing cut tetrahedral elements with quadratic Lagrange polynomials, open to straightforward extension to arbitrary order. Assuming a sharp boundary based on a spline representation, our framework is based on characterizing the intersection topology for each cut tetrahedral element and subsequently constructing a sequence of higher-order mappings. The latter includes root finding to determine intersection points of element edges and further rays with the spline surface. These points serve as interpolation points for quadratic Lagrange polynomials that are defined on tetrahedral and prismatic reference elements. These reference elements can be used to map standard tetrahedral and Gaussian quadrature rules to geometrically accurate points and weights in intersected element domains.

We presented several benchmark problems that demonstrated the significant advantages of element-wise parametrization over subdivision quadrature in terms of accuracy and computational cost in problems that are defined by a sharp boundary. The TetFCM with subdivision quadrature is (in theory) able to achieve full accuracy, but at a prohibitively large number of quadrature points. In contrast, the TetFCM with element-wise parametrization is able to achieve full accuracy, while keeping the number of quadrature points at a level that is only slightly increased with respect to boundary-fitted discretizations of comparable size. We also showed the critical importance of an accurate geometry resolution for the accurate approximation of solution fields directly at the embedded boundary, which is particularly important when the embedded boundary represents a coupling interface to another domain.

In the final part of the paper, we applied a combination of both quadrature variants in the context of a patient-specific workflow for the simulation-based performance analysis of coupled bone/implant configurations. This example involves two different geometric models based on fuzzy medical imaging data for the bone and sharp CAD boundary representations for the implant. We highlighted the specific strengths of each quadrature variant depending on which geometric model is considered. In the bone, the large number of subdivision based quadrature points proved essential to take into account the inhomogeneous stiffness distribution at each voxel in the sense of a homogenization procedure. In the implant, element-wise parametrization enabled an accurate oscillation free stress resolution at the surface of the implant, while significantly reducing the number of point evaluations. We conclude that there is no optimal quadrature method for the TetFCM, but its accuracy, flexibility, and computational efficiency in complex simulation scenarios relies on flexibly applying different quadrature variants.

Acknowledgments. The first author gratefully acknowledges support from the German Research Foundation (DFG) program GSC 111 (AICES Graduate School). The Minnesota group gratefully acknowledges support from the National Science Foundation (CISE-1565997) and the In-

ternship Opportunities Program (IOP) run by the Department of Civil, Environmental, and Geo-Engineering at the University of Minnesota. The authors also acknowledge the Minnesota Supercomputing Institute (MSI) of the University of Minnesota for providing computing resources that have contributed to the research results reported within this paper (<https://www.msi.umn.edu/>).

References

- [1] E. Burman, S. Claus, P. Hansbo, M.-G. Larson, and A. Massing. CutFEM: discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104(7):472–501, 2015.
- [2] D. Schillinger, L. Dede’, M.A. Scott, J.A. Evans, M.J. Borden, E. Rank, and T.J.R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-250:116–150, 2012.
- [3] M. Ruess, D. Schillinger, A.I. Özcan, and E. Rank. Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries. *Computer Methods in Applied Mechanics and Engineering*, 269:46–71, 2014.
- [4] A.P. Nagy and D.J. Benson. On the numerical integration of trimmed isogeometric elements. *Computer Methods in Applied Mechanics and Engineering*, 284:165–185, 2015.
- [5] M. Breitenberger, A. Apostolatos, B. Philipp, R. Wüchner, and K.-U. Bletzinger. Analysis in computer aided design: Nonlinear isogeometric b-rep analysis of shell structures. *Computer Methods in Applied Mechanics and Engineering*, 284:401–457, 2015.
- [6] M. Ruess, D. Tal, N. Trabelsi, Z. Yosibash, and E. Rank. The finite cell method for bone simulations: Verification and validation. *Biomechanics and Modeling in Mechanobiology*, 11(3):425–437, 2012.
- [7] D. Schillinger, M. Ruess, N. Zander, Y. Bazilevs, A. Düster, and E. Rank. Small and large deformation analysis with the p - and B-spline versions of the Finite Cell Method. *Computational Mechanics*, 50(4):445–478, 2012.
- [8] J. Parvizian, A. Düster, and E. Rank. Topology optimization using the finite cell method. *Optimization and Engineering*, 13:57–78, 2012.
- [9] J. Benk, H.-J. Bungartz, M. Mehl, and M. Ulbrich. Immersed boundary methods for fluid-structure interaction and shape optimization within an FEM-based PDE toolbox. In *Advanced Computing*, pages 25–56. Springer, 2013.
- [10] F. Cirak and K. Bandara. Multiresolution shape optimisation with subdivision surfaces. In *Isogeometric Analysis and Applications 2014*, pages 127–156. Springer, 2015.
- [11] K. Bandara, T. Rüber, and F. Cirak. Shape optimisation with multiresolution subdivision surfaces and immersed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 300:510–539, 2016.
- [12] M.-C. Hsu, D. Kamensky, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes. Fluid–structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation. *Computational Mechanics*, 54(4):1055–1071, 2014.
- [13] D. Kamensky, M.-C. Hsu, D. Schillinger, J.A. Evans, A. Aggarwal, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes. An immersogeometric variational framework for fluid–structure interaction: application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015.
- [14] D. Kamensky, J.A. Evans, and M.-C. Hsu. Stability and conservation properties of collocated constraints in immersogeometric fluid–thin structure interaction analysis. *Communications in Computational Physics*, 18:1147–1180, 2015.
- [15] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, M. Wu, J. Mineroff, A. Reali, Y. Bazilevs, and 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*, 55:1211–1225, 2015.
- [16] A. Gerstenberger and W.A. Wall. An embedded Dirichlet formulation for 3D continua. *International Journal for Numerical Methods in Engineering*, 82:537–563, 2010.
- [17] E. Burman and P. Hansbo. Fictitious domain finite element methods using cut elements: a stabilized Lagrange multiplier method. *Computer Methods in Applied Mechanics and Engineering*, 62(4):2680–2686, 2010.

- [18] O. Marco, R. Sevilla, Y. Zhang, J.J. Ródenas, and M. Tur. Exact 3D boundary representation in finite element analysis based on Cartesian grids independent of the geometry. *International Journal for Numerical Methods in Engineering*, 103(6):445–468, 2015.
- [19] A. Hansbo and P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193:3523–3540, 2004.
- [20] M. Ruess, D. Schillinger, Y. Bazilevs, V. Varduhn, and E. Rank. Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method. *International Journal for Numerical Methods in Engineering*, 95(10):811–846, 2013.
- [21] D. Schillinger, I. Harari, M.-C. Hsu, D. Kamensky, K.F.S. Stoter, Y. Yu, and Z. Ying. The non-symmetric Nitsche method for the parameter-free imposition of weak boundary and coupling conditions in immersed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 309:625–652, 2016.
- [22] A. Stavrev. *The role of higher-order geometry approximation and accurate quadrature in NURBS based immersed boundary methods*. Master Thesis, Technische Universität München, 2012.
- [23] G. Legrain. A NURBS enhanced extended finite element approach for unfitted CAD analysis. *Computational Mechanics*, 1:34, 2013.
- [24] T.-P. Fries and S. Omerovic. Higher-order accurate integration of implicit geometries. *International Journal for Numerical Methods in Engineering*, 106(1):323–371, 2016.
- [25] L. Kudela, N. Zander, T. Bog, S. Kollmannsberger, and E. Rank. Efficient and accurate numerical quadrature for immersed boundary methods. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):1–22, 2015.
- [26] L. Kudela, N. Zander, S. Kollmannsberger, and E. Rank. Smart octrees: Accurately integrating discontinuous functions in 3D. *Computer Methods in Applied Mechanics and Engineering*, page doi:10.1016/j.cma.2016.04.006, 2016.
- [27] C. Lehrenfeld. High order unfitted finite element methods on level set domains using isoparametric mappings. *Computer Methods in Applied Mechanics and Engineering*, 300:716–733, 2016.
- [28] R. Bouclier, J.-C. Passieux, and M. Salaün. Local enrichment of NURBS patches using a non-intrusive coupling strategy: Geometric details, local refinement, inclusion, fracture. *Computer Methods in Applied Mechanics and Engineering*, 300:1–26, 2016.
- [29] J. Parvizian, A. Düster, and E. Rank. Finite cell method: h - and p - extension for embedded domain methods in solid mechanics. *Computational Mechanics*, 41:122–133, 2007.
- [30] A. Düster, J. Parvizian, Z. Yang, and E. Rank. The finite cell method for three-dimensional problems of solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 197:3768–3782, 2008.
- [31] D. Schillinger and M. Ruess. The Finite Cell Method: A review in the context of higher-order structural analysis of CAD and image-based geometric models. *Archives of Computational Methods in Engineering*, 22(3):391–455, 2015.
- [32] N. Zander, T. Bog, M. Elhaddad, R. Espinoza, H. Hu, A.F. Joly, C. Wu, P. Zerbe, A. Dster, S. Kollmannsberger, J. Parvizian, M. Ruess, D. Schillinger, and E. Rank. FCMLab: A finite cell research toolbox for MATLAB. *Advances in Engineering Software*, 74:49–63, 2014.
- [33] F. Xu, D. Schillinger, D. Kamensky, V. Varduhn, C. Wang, and M.-C. Hsu. The tetrahedral finite cell method for fluids: Immersogeometric analysis of turbulent flow around complex geometries. *Computers & Fluids*, doi:10.1016/j.compfluid.2015.08.027, 2015.
- [34] V. Varduhn, M.-C. Hsu, M. Ruess, and D. Schillinger. The tetrahedral finite cell method: Higher-order immersogeometric analysis on adaptive non-boundary-fitted meshes. *International Journal for Numerical Methods in Engineering*, doi:10.1002/nme.5207, 2016.
- [35] D. Schillinger and 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):3358–3380, 2011.
- [36] D. Schillinger, A. Düster, and E. Rank. The hp - d adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering*, 89:1171–1202, 2012.
- [37] C.V. Verhoosel, G.J. van Zwieten, B. van Rietbergen, and R. de Borst. Image-based goal-oriented adaptive isogeometric analysis with application to the micro-mechanical modeling of trabecular bone. *Computer Methods in Applied Mechanics and Engineering*, 284:138–164, 2015.

- [38] N. Zander, T. Bog, S. Kollmannsberger, D. Schillinger, and E. Rank. Multi-Level hp-Adaptivity: High-Order mesh adaptivity without the difficulties of constraining hanging nodes. *Computational Mechanics*, 55:499–517, 2015.
- [39] L. Piegl and W. Tiller. *The NURBS Book*. Springer, 1997.
- [40] R. Sevilla, S. Fernández-Méndez, and A. Huerta. Comparison of high-order curved finite elements. *International Journal for Numerical Methods in Engineering*, 87:719–734, 2011.
- [41] R. Sevilla, S. Fernández-Méndez, and A. Huerta. 3D NURBS-Enhanced Finite Element Method (NEFEM). *International Journal for Numerical Methods in Engineering*, 88:103–125, 2011.
- [42] A. Stavrev, P. Knechtges, S. Elgeti, and A. Huerta. Space-time NURBS-enhanced finite elements for free-surface flows in 2D. *International Journal for Numerical Methods in Fluids*, DOI: 10.1002/fld.4189, 2016.
- [43] T.J.R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000.
- [44] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method – The Basis*, volume 1. Butterworth-Heinemann, 6th edition, 2005.
- [45] H. Samet. *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann Publishers, 2006.
- [46] Netgen Mesh Generator, developed by j. schoeberl, <http://sourceforge.net/projects/netgen-mesher/>, 2015.
- [47] Joachim Schöberl. NETGEN. An advancing front 2D/3D-mesh generator based on abstract rules. *Computing and Visualization in Science*, 1(1):41–52, 1997.
- [48] A. Embar, J. Dolbow, and I. Harari. Imposing Dirichlet boundary conditions with Nitsche’s method and spline-based finite elements. *International Journal for Numerical Methods in Engineering*, 83:877–898, 2010.
- [49] E. Burman. A penalty-free nonsymmetric Nitsche-type method for the weak imposition of boundary conditions. *SIAM Journal on Numerical Analysis*, 50(4):1959–1981, 2012.
- [50] T. Boiveau and E. Burman. A penalty-free Nitsche method for the weak imposition of boundary conditions in compressible and incompressible elasticity. *IMA Journal of Numerical Analysis*, page drv042, 2015.
- [51] P. Keast. Moderate-degree tetrahedral quadrature formulas. *Computer Methods in Applied Mechanics and Engineering*, 55:339–348, 1986.
- [52] W. Jiang, C. Annavarapu, J.E. Dolbow, and I. Harari. A robust Nitsche’s formulation for interface problems with spline-based finite elements. *International Journal for Numerical Methods in Engineering*, 104(7):676–696, 2015.
- [53] G. Strang and G.J. Fix. *An Analysis of the Finite Element Method*. Prentice-Hall, 1973.
- [54] G. Királyfalvi and B.A. Szabó. Quasi-regional mapping for the p-version of the finite element method. *Finite Elements in Analysis and Design*, 27(1):85–97, 1997.
- [55] S. Dey, M.S. Shephard, and J.E. Flaherty. Geometry representation issues associated with p-version finite element computations. *Computer Methods in Applied Mechanics and Engineering*, 150:39–55, 1997.
- [56] W. Martin, E. Cohen, R. Fish, and P. Shirley. Practical ray tracing of trimmed NURBS surfaces. *Journal of Graphics Tools*, 5(1):27–52, 2000.
- [57] A.B. Mor and T. Kanade. Modifying soft tissue models: Progressive cutting with minimal new element creation. In *Proc. of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2000*, pages 598–607. Springer, 2000.
- [58] A. Krishnamurthy, R. Khardekar, and S. McMains. Optimized GPU evaluation of arbitrary degree NURBS curves and surfaces. *Computer-Aided Design*, 41(12):971–980, 2009.
- [59] M.-C. Hsu, C. Wang, F. Xu, A.J. Herrema, and A. Krishnamurthy. Direct immersogeometric fluid flow analysis using B-rep CAD models. *Computer Aided Geometric Design*, doi:10.1016/j.cagd.2016.02.007, 2016.
- [60] C. Felippa. Introduction to finite element methods. *Course notes, available online at <http://www.colorado.edu/engineering/CAS/courses.d/IFEM.d/Home.html>*.
- [61] C. Felippa. Advanced finite element methods. *Course notes, available online at <http://www.colorado.edu/engineering/CAS/courses.d/AFEM.d/Home.html>*.
- [62] R. Sevilla, S. Fernández-Méndez, and A. Huerta. NURBS-enhanced finite element method (NEFEM). *Archives of Computational Methods in Engineering*, 18(4):441–484, 2011.
- [63] A. Stroud. *Approximate Calculation of Multiple Integrals*. Prentice Hall, 1971.
- [64] R. Cools. Monomial cubature rules since “Stroud”: A compilation - Part 2. *Journal of Computational and*

- Applied Mathematics*, 112(1):21–27, 1999.
- [65] D. Schillinger, S.J. Hossain, and T.J.R. Hughes. Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 277:1–45, 2014.
 - [66] M.K. Agoston. *Computer graphics and geometric modeling*, volume 2. Springer, 2005.
 - [67] B. Szabó and I. Babuška. *Finite Element Analysis*. Wiley, 1991.
 - [68] R. Hill. *The mathematical theory of plasticity*. Oxford University Press, 1950.
 - [69] J. Lubliner. *Plasticity Theory*. Macmillan Publishing Company, 1990.
 - [70] Z. Yosibash, R. Padan, L. Joskowicz, and C. Milgrom. A CT-based high-order finite element analysis of the human proximal femur compared to in-vitro experiments. *ASME Journal of Biomechanical Engineering*, 129:297, 2007.
 - [71] Z. Yosibash, N. Trabelsi, and C. Milgrom. Reliable simulations of the human proximal femur by high-order finite element analysis validated by experimental observations. *Journal of Biomechanics*, 40(16):3688–3699, 2007.
 - [72] N. Trabelsi, Z. Yosibash, and C. Milgrom. Validation of subject-specific automated p-FE analysis of the proximal femur. *Journal of Biomechanics*, 42(3):234–241, 2009.
 - [73] J.H. Keyak and Y. Falkinstein. Comparison of in situ and in vitro CT scan-based finite element model predictions of proximal femoral fracture load. *Medical Engineering & Physics*, 25(9):781–787, 2003.
 - [74] G.A. Francfort and J.-J. Marigo. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids*, 46(8):1319–1342, 1998.
 - [75] B. Bourdin, G.A. Francfort, and J.-J. Marigo. The variational approach to fracture. *Journal of Elasticity*, 91(1-3):5–148, 2008.
 - [76] C. Miehe, M. Hofacker, and F. Welschinger. A phase field model for rate-independent crack propagation: Robust algorithmic implementation based on operator splits. *Computer Methods in Applied Mechanics and Engineering*, 199(45):2765–2778, 2010.
 - [77] D. Schillinger, M.J. Borden, and H.K. Stolarski. Isogeometric collocation for phase-field fracture models. *Computer Methods in Applied Mechanics and Engineering*, 284:583–610, 2015.
 - [78] D.N. Arnold, F. Brezzi, B. Cockburn, and D. Marini. Discontinuous Galerkin methods for elliptic problems. In *Discontinuous Galerkin Methods*, pages 89–101. Springer, 2000.
 - [79] C. Annavarapu, M. Hautefeuille, and J.E. Dolbow. A robust Nitsche’s formulation for interface problems. *Computer Methods in Applied Mechanics and Engineering*, 225:44–54, 2012.
 - [80] A. Düster, H.-G. Sehlhorst, and E. Rank. Numerical homogenization of heterogeneous and cellular materials utilizing the finite cell method. *Computational Mechanics*, 50:413–431, 2012.
 - [81] M.J. Borden, C.V. Verhoosel, M.A. Scott, Hughes T.J.R., and C.M. Landis. A phase-field description of dynamic brittle fracture. *Computer Methods in Applied Mechanics and Engineering*, 217–220:77–95, 2012.