

# Isogeometric analysis: an overview and computer implementation aspects

Vinh Phu Nguyen<sup>a,1,\*</sup>, Cosmin Anitescu<sup>c,2</sup>, Stéphane P.A. Bordas<sup>a,b,3</sup>, Timon Rabczuk<sup>c,4</sup>

<sup>a</sup>*School of Engineering, Institute of Mechanics and Advanced Materials, Cardiff University, Queen's Buildings, The Parade, Cardiff CF24 3AA*

<sup>b</sup>*Faculté des Sciences, de la Technologie et de la Communication, University of Luxembourg, 6, rue Richard Coudenhove-Kalergi, 1359, Luxembourg City, Luxembourg*

<sup>c</sup>*Institute of Structural Mechanics, Bauhaus-Universität Weimar, Marienstraße 15 99423 Weimar*

---

## Abstract

Isogeometric analysis (IGA) represents a recently developed technology in computational mechanics that offers the possibility of integrating methods for analysis and Computer Aided Design (CAD) into a single, unified process. The implications to practical engineering design scenarios are profound, since the time taken from design to analysis is greatly reduced, leading to dramatic gains in efficiency. In this manuscript, through a self-contained Matlab<sup>®</sup> implementation, we present an introduction to IGA applied to simple analysis problems and the related computer implementation aspects. Furthermore, implementation of the extended IGA which incorporates enrichment functions through the partition of unity method (PUM) is also presented, where several examples for both two-dimensional and three-dimensional fracture are illustrated. We also describe the use of IGA in the context of strong-form (collocation) formulations, which has been an area of research interest due to the potential for significant efficiency gains offered by these methods. The code which accompanies the present paper can be applied to one, two and three-dimensional problems for linear elasticity, linear elastic fracture mechanics, structural mechanics (beams/plates/shells including large displacements and rotations) and Poisson problems with or without enrichment. The Bézier extraction concept that allows the FE analysis to be performed efficiently on T-spline geometries is also incorporated. The article includes a summary of recent trends and developments within the field of IGA.

*Keywords:* isogeometric analysis, linear elasticity, Matlab<sup>®</sup>, NURBS, finite elements, partition of unity, enrichment, 3D cracks, boundary conditions, CAD, large displacements and rotations, shells, isogeometric collocation

---

\*Corresponding author

<sup>1</sup>nguyenpv@cardiff.ac.uk

<sup>2</sup>canitesc@syr.edu

<sup>3</sup>stephane.bordas@alum.northwestern.edu

<sup>4</sup>timon.rabczuk@uni-weimar.de

---

## 1. Introduction

### 1.1. Underlying concepts of isogeometric analysis

The predominant technology that is used by CAD to represent complex geometries is the Non-Uniform Rational B-spline (NURBS). This allows certain geometries that can only be approximated by polynomial functions to be represented exactly. Examples of this include conic sections and objects such as cylinders, spheres, etc. which are essential in design. There is a vast array of literature focused on NURBS (e.g. [98], [106]) and as a result of several decades of research, many efficient computer algorithms exist for their fast evaluation and refinement. The key concept outlined by Hughes et al. [57] was to employ NURBS not only as a geometry discretisation technology, but also as a discretisation tool for analysis, attributing such methods to the field of ‘Isogeometric Analysis’ (IGA). Since this seminal paper, a monograph dedicated entirely to IGA has been published [31] and applications can now be found in several fields including structural mechanics, solid mechanics, fluid mechanics and contact mechanics.

We give in this section an overview of some of these recent developments while outlining the benefits and present shortcomings of IGA. It should be emphasized that the idea of using CAD technologies in finite elements dates back at least to [62, 63] where B-splines were used as shape functions in FEM. In addition, similar methods which adopt subdivision surfaces have been used to model shells [27]. We also review some of the recent attempts at simplifying the CAD-FEA integration by separating boundary and domain discretizations.

### 1.2. Applications

In contact formulations using conventional geometry discretizations, the presence of faceted surfaces can lead to jumps and oscillations in traction responses unless very fine meshes are used. The benefits of using NURBS over such an approach are evident, since smooth contact surfaces are obtained, leading to more physically accurate contact stresses. Recent work in this area includes [33, 77, 82, 136–138].

IGA has also shown advantages over traditional approaches in the context of optimization problems, [79, 100, 101, 145] where the tight coupling with CAD models offers an extremely attractive approach for industrial applications. Another attractive class of methods include those that require only a boundary discretization, creating a truly direct

coupling with CAD. Isogeometric boundary element methods for elastostatic analysis were presented in [118, 126], demonstrating that mesh generation can be completely circumvented by using CAD discretizations for analysis.

Shell and plate problems are another field where IGA has demonstrated compelling benefits over conventional approaches [11, 13, 14, 16, 41, 67, 140]. The smoothness of the NURBS basis functions allows for a straightforward construction of plate/shell elements. Particularly for thin shells, rotation-free formulations can be easily constructed [66, 67]. Note that for multi-patch NURBS surfaces, rotation-free IGA elements require special treatment at patch boundaries where the basis functions are found to be  $C^0$  continuous. Furthermore, isogeometric plate/shell elements exhibit much less pronounced shear-locking compared to standard FE plate/shell elements. Elements with smooth boundaries such as circular and cylindrical elements were successfully constructed using the IGA concept [76, 78].

The smoothness of NURBS basis functions is attractive for analysis of fluids [7, 53, 96] and for fluid-structure interaction problems [9, 10]. In addition, due to the ease of constructing high order continuous basis functions, IGA has been used with great success in solving PDEs that incorporate fourth order (or higher) derivatives of the field variable such as the Hill-Cahnard equation [52], explicit gradient damage models [142] and gradient elasticity [50]. The high order NURBS basis has also found potential applications in the Kohn-Sham equation for electronic structure modeling of semiconducting materials [81].

NURBS provide advantageous properties for structural vibration problems [30, 59, 139, 146] where  $k$ -refinement (unique to IGA) has been shown to provide more robust and accurate frequency spectra than typical higher-order FE  $p$ -methods. Particularly, the optical branches of frequency spectra, which have been identified as contributors to Gibbs phenomena in wave propagation problems (and the cause of rapid degradation of higher modes in the  $p$ -version of FEM), are eliminated. However when lumped mass matrices were used, the accuracy is limited to second order for any basis order. High order isogeometric lumped mass matrices are not yet available. The mathematical properties of IGA were studied in detail by [47].

The isogeometric concept has also spread to the field of meshfree methods such as [54, 69, 124] in which spline-based meshfree methods were presented and [107] introduced a hybrid IGA and local maximum entropy meshfree method in which the boundary surfaces are presented by B-splines and the interior volume is discretized by meshfree points. B-splines are used in the material point method to alleviate the cell-crossing issue [129]. Industrial applications of IGA have been presented in [55, 118] along with applications in experimental mechanics [45] where NURBS-based DIC

(Digital Image Correlation) was shown to outperform standard FE DIC.

B-splines have been utilized in ANCF (absolute nodal coordinate formulation) finite element analysis of large deformation problems of constrained multibody systems [72, 87, 110, 150].

### *1.3. Shortcomings of NURBS and alternative geometry discretizations*

NURBS are ubiquitous in CAD but are known to exhibit major shortcomings from a computational geometry standpoint. Perhaps the greatest difficulty encountered is the inability of NURBS to produce watertight geometries, often complicating mesh generation. From an analysis perspective, the tensor product structure of NURBS proves to be inefficient, caused by the global nature of refinement operations. In turn, this leads to inefficient error estimation and adaptivity algorithms. One solution which has gathered momentum from both the computational geometry and analysis communities is the use of T-splines [121] which overcome the limitations of NURBS while retaining the familiar structure of NURBS algorithms. T-splines correct the deficiencies of NURBS by creating a single patch, watertight geometry which can be locally refined and coarsened. Buffa et al. [20] note that linear independence of the T-spline basis functions is not guaranteed on generic T-meshes leading to the definition of analysis-suitable T-splines [117], a mildly restricted subset of T-splines which meet the demands of both design and analysis. Utilisation of T-splines in an IGA framework has been illustrated in [8, 39], and by adopting a Bézier extraction process, Scott et al. [116] showed that T-splines can be incorporated efficiently into existing FE codes.

Alternatives to T-splines include polycube splines [147], PHT-splines [35] and LR-splines [37]. PHT-splines (polynomial spline over hierarchical T-meshes) have been extended to rational splines and applied in [94, 95] to problems in elasticity for continua and thin structures. Adaptive refinement with PHT-splines is particularly simple. Although T-splines allow for local adaptive refinement, the complexity of knot insertion under adaptive refinement is complex, particularly in 3D. However, we note that research is currently being pursued on hierarchical T-spline refinement algorithms that address this issue.

Another direction of IGA research includes hierarchical B-splines [19, 71, 112, 143], splines forest [119], unstructured Powell-Sabin splines [128], triangular B-splines [61] and Coons-Gordon interpolation [99]. The hierarchical B-splines finite cell method [112] furnishes a seamless CAD-FEA integration for very complex geometries. We refer also to [70] for IGA combined with finite element based local refinement capabilities. Different subdivision surface techniques (Catmull-Clark, Loop) have also been utilized for solid and shell modeling [21, 148].

In computer aided geometric design, patching multiple NURBS parameterizations to form complex topologies is far from trivial if certain continuity requirements are to be maintained. Trimming techniques provide a promising alternative for representing complex NURBS domains. In [68], a trimmed surface based analysis framework has been proposed where NURBS-enhanced FEM [122, 123] was applied to define a suitable integration domain within parameter space. In a recent contribution [115], the authors presented an alternative method to handle trimmed NURBS geometries.

#### *1.4. Discontinuities and fracture*

IGA has been applied to cohesive fracture [141], outlining a framework for modeling debonding along material interfaces using NURBS and propagating cohesive cracks using T-splines. The method relies upon the ability to specify the continuity of NURBS and T-splines through a process known as knot insertion. As a variation of the eXtended Finite Element Method (XFEM) [84], IGA was applied to Linear Elastic Fracture Mechanics (LEFM) using the partition of unity method (PUM) to capture two dimensional strong discontinuities and crack tip singularities efficiently [34, 51]. The method is usually referred to as XIGA (eXtended IGA). In [133] an explicit isogeometric enrichment technique was proposed for modeling material interfaces and cracks exactly. Note that this method is contrary to PUM-based enrichment methods which define cracks implicitly.

A phase field model for dynamic fracture was presented in [18] using adaptive T-spline refinement to provide an effective method for simulating fracture in three dimensions. In [92] high order B-splines were adopted to efficiently model delamination of composite specimens and in [89], an isogeometric framework for two and three dimensional delamination analysis of composite laminates was presented where the authors showed that using IGA can significantly reduce the usually time consuming pre-processing step in generating FE meshes (solid elements and cohesive interface elements) for delamination computations. Similar work with T-splines based cohesive elements for interface debonding is reported in [36]. A continuum description of fracture using explicit gradient damage models was also studied using NURBS [142] and IGA was also adopted in poromechanical fracture problems [60].

#### *1.5. Alternatives to IGA*

Other techniques which integrate CAD and analysis include the use of subdivision surfaces to model shells [27], NURBS-enhanced finite elements [122, 123] and NURBS for BEM shape optimization [25]. Immersed boundary methods

[111] (and references therein), the finite cell method [102, 112] and the structured XFEM [73, 86] are yet other alternatives which aim to combine analysis and design technologies. In general, it is found that for CAD and analysis technologies to work seamlessly together, the underlying discretization must either be directly compatible or easily converted between the two.

IGA has offered significant advances towards the goal of a unified design and analysis framework, but much research is still needed before this goal is realized. There are several indications of the future promise of IGA for industrial design but ultimately, the litmus test of success for IGA will be whether the approach is widely adopted by industry.

### *1.6. Computational aspects*

Some major computational aspects of IGA which have been studied so far include (i) locking issues, (ii) sensitivity to mesh distortion, (iii) impact of high continuity of NURBS on direct solvers, (iv) collocation methods, (v) competing demands of analysis and computational geometry discretizations, (vi) construction of trivariate solids from given bivariate surface representations and (vii) optimal quadrature rules.

- (i) Although the smoothness of NURBS basis functions reduces to some extent the locking phenomena for constrained problems such as incompressible media, thin-walled structures, NURBS-based FEs are not locking free [6, 22–24, 40, 42, 43]. Existing locking removal techniques in standard FEMs were successfully adapted to IGA such as the Discrete Strain Gap method [40], the F/B-bar method [42, 43], the enhanced assumed strain method [22] and mixed formulations [135].
- (ii) The effect of mesh distortion on the performance of IGA for solid mechanics was discussed in [74] in which it was found that higher-order NURBS functions are able to somewhat alleviate the impact of the distortions.
- (iii) The high order continuity offered by NURBS has a negative impact on the performance of direct solvers as pointed out in [29]. The authors found that for a fixed number of unknowns and basis degree, a higher degree of continuity drastically increases the CPU time and RAM needed to solve the problem when using a direct solver.
- (iv) In an attempt to compete with low order FEs with one-point quadrature that are extensively used in industrial applications, isogeometric collocation methods were developed [3, 4, 113].

- (v) Due to the fact that meshes in an isogeometric framework are defined by the parametrization of the object of interest, the quality of the geometry parametrisation plays an important role in ensuring mesh quality. This issue has, however, been addressed by only a few researchers [28, 114, 132, 149, 154]. In particular, in [28], the authors proposed the concept of “analysis-aware geometry modeling”.
- (vi) In CAD, solids are defined as boundary surfaces in which the interior is not explicitly modeled. In FEA, a solid representation is necessary and therefore, the transition from CAD solids to FEA solids demands a step in which the CAD representations are converted to solid FEA representations. Initial developments have been reported in [1, 46, 80, 151–153]. Note that in this regard, the isogeometric boundary element method (IGABEM) can be considered a truly isogeometric method [97, 118, 126, 127] since BEM analysis requires only the definition of a boundary discretization, completely defined by CAD.
- (vii) Gaussian quadrature is not optimal for IGA. Research is currently focused on optimal integration techniques such as that in [5, 58] in which (nearly) optimal quadrature rules have been presented.

### *1.7. Available implementations*

Some implementation aspects of IGA were reported in [31] and more recently, an open source IGA Matlab<sup>®</sup> code was described in [144] with a restriction to 2D scalar PDEs. An excellent open source IGA code written in Matlab<sup>®</sup> is given in [48]. Incorporating IGA within an object-oriented C++ FE code was discussed in [109]. Implementation details for enriched formulations within an IGA framework are reported in [15] using commercial FE software. An IGA BEM code written in Matlab<sup>®</sup> was presented in [127]. Isogeometric analysis was also incorporated into FEAP [134, 135]. A high performance IGA code was given in [32] which is based on PETSc, the Portable, Extensible Toolkit for Scientific Computation. The Python programming language has also been adopted to implement IGA e.g., [103, 104]. GPU implementation of IGA is also available [64].

### *1.8. Contributions and outline*

In this paper, we present in detail an isogeometric analysis method applied to two- and three-dimensional scalar and vector equations, structural mechanics problems and traction-free crack problems. The discussion is confined to NURBS for the sake of simplicity. Although no new fundamental findings are presented, the contributions of the paper are

- an overview of IGA, applications and recent developments are presented;
- some implementation details for solid mechanics and rotation-free plate and shell elements are provided;
- implementation for 2D and 3D XIGA for traction-free cracks;
- implementation of collocation methods in the context of IGA;
- visualization techniques for (X)IGA are discussed;
- techniques to impose Dirichlet boundary conditions;
- different techniques to model discontinuities in the context of IGA

To the best of our knowledge, none of the publicly available implementations mentioned in the previous section include XIGA, Kirchhoff shells or collocation methods. The closest project of similar scope is GeoPDEs which includes solvers for other PDEs, such as Stokes' equations and Maxwell equations, and seems to emphasize generality. By contrast, we are focusing on solving some specific problems in a self-contained way that is perhaps easier to understand by a newcomer to IGA.

The paper is structured as follows: Section 2 outlines B-spline and NURBS technology used to construct surfaces and solids; the use of NURBS for discretization within a finite element framework is treated in Section 3. A detailed description of our IGA Matlab code is given in Section 4 and Application of IGA to structural mechanics problems is presented in Section 5. Numerical examples including 2D/3D fracture mechanics and large deformation shell problems are given in Section 6.

### 1.9. Notation

We use lowercase indices to indicate a local index and uppercase indices to indicate a global index. We denote  $d_p$  and  $d_s$  as the number of parametric directions and spatial directions respectively. **Boldfont** is used to indicate matrices and vectors where the number of components is implied.

## 2. A brief introduction to B-splines/NURBS

To construct a B-spline, a *knot vector* must be specified and is defined as an ordered set of increasing parameter values  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$ ,  $\xi_i \leq \xi_{i+1}$  where  $\xi_i$  is the  $i$ th knot,  $n$  is the number of basis functions and  $p$  is the



polynomial order. The knot vector divides the parametric space into intervals usually referred to as *knot spans* with the number of coincident knots for a particular knot value referred to as a knot with a certain *multiplicity*  $k$ . That is, a knot has a multiplicity  $k$  if it is repeated  $k$  times in the knot vector. Most commonly, *open* knot vectors are used where the first and last knots have a multiplicity  $k = p + 1$ . Appendix A outlines alternative knot vector notations which remove redundant information, but it can be assumed that the above knot vector notation is used throughout the present work.

### 2.1. B-spline basis functions

Given a knot vector  $\Xi$ , the associated set of B-spline basis functions  $\{N_{i,p}\}_{i=1}^n$  are defined recursively by the Cox-de-Boor formula, starting with the zeroth order basis function ( $p = 0$ )

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

and for a polynomial order  $p \geq 1$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \quad (2)$$

in which fractions of the form  $0/0$  are defined as zero.

Algorithms for efficient evaluation of B-Splines and their derivatives of arbitrary order can be found in [98].

### 2.2. B-spline surfaces and volumes

Given two knot vectors for each parametric direction  $\Xi^1 = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$  and  $\Xi^2 = \{\eta_1, \eta_2, \dots, \eta_{m+q+1}\}$ , and a *control net* of points  $\mathbf{P}_{i,j} \in \mathbb{R}^{d_s}$ , a tensor-product B-spline surface is defined as

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m N_{i,p}(\xi) M_{j,q}(\eta) \mathbf{P}_{i,j}, \quad (3)$$

where  $N_{i,p}(\xi)$  and  $M_{j,q}(\eta)$  are the univariate B-spline basis functions of order  $p$  and  $q$  corresponding to knot vectors  $\Xi^1$  and  $\Xi^2$ , respectively. Defining a global index as

$$A = n(j - 1) + i, \quad (4)$$

Eq. (3) can be rewritten in a more compact form as

$$\mathbf{S}(\boldsymbol{\xi}) = \sum_{A=1}^{n \times m} \mathbf{P}_A N_A^{p,q}(\boldsymbol{\xi}), \quad (5)$$

in which  $N_A^{p,q}$  is a bivariate B-spline basis function defined as  $N_A^{p,q}(\boldsymbol{\xi}) = N_{i,p}(\xi)M_{j,q}(\eta)$ .

The extension to B-spline volumes is straightforward, where a trivariate basis is formed through a tensor product of B-spline basis functions as

$$\mathbf{V}(\xi, \eta, \zeta) = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^l N_{i,p}(\xi)M_{j,q}(\eta)L_{k,r}(\zeta)\mathbf{P}_{i,j,k}, \quad (6)$$

or, by defining a global index  $A$  through

$$A = (n \times m)(k - 1) + n(j - 1) + i, \quad (7)$$

a simplified form of Eq. (6) can be written as

$$\mathbf{V}(\boldsymbol{\xi}) = \sum_{A=1}^{n \times m \times l} \mathbf{P}_A N_A^{p,q,r}(\boldsymbol{\xi}). \quad (8)$$

### 2.3. NURBS

B-splines are convenient for free-form modeling, but they lack the ability to exactly represent some simple shapes such as circles and ellipsoids. This is why today, the *de facto* standard technology in CAD is a generalization of B-splines referred to as NURBS (Non-Uniform Rational B-Splines). NURBS are formed through rational functions of B-splines, forming a super-set of B-splines. They inherit all the favorable properties of B-splines and are favored over their counterpart due to their ability to form exact representations of conic sections such as spheres, ellipsoids,

paraboloids and hyperboloids. In addition, there exist efficient algorithms for their evaluation and refinement.

### 2.3.1. NURBS basis functions

NURBS basis functions are defined as

$$R_{i,p}(\xi) = \frac{N_{i,p}(\xi)w_i}{W(\xi)} = \frac{N_{i,p}(\xi)w_i}{\sum_{i=1}^n N_{i,p}(\xi)w_i} \quad (9)$$

where  $\{N_{i,p}\}_{i=1}^n$  is the set of B-spline basis functions of order  $p$  and  $\{w_i\}_{i=1}^n, w_i > 0$  is the set of NURBS weights. Selecting appropriate weights permits the description of many different types of curves including polynomials and circular arcs. For the special case in which all weights are equal, the NURBS basis reduces to the B-spline basis. NURBS weights for certain simple geometries are given in [98], but in general, weights are user-defined through CAD packages such as Rhino<sup>5</sup>.

### 2.3.2. NURBS curves, surfaces and volumes

In a similar fashion to B-spline curves, the NURBS curve associated with a set of control points and weights  $\{\mathbf{P}_A, w_A\}_{A=1}^n$  and basis functions  $\{R_{A,p}\}_{A=1}^n$  is defined as

$$\mathbf{C}(\xi) = \sum_{A=1}^n \mathbf{P}_A R_{A,p}(\xi). \quad (10)$$

NURBS surfaces are constructed from a linear combination of bivariate NURBS basis functions, control points  $\mathbf{P}_{i,j} \in \mathbb{R}^{d_s}$  and weights  $w_{i,j} > 0$  as

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m \mathbf{P}_{i,j} R_{i,j}^{p,q}(\xi, \eta), \quad (11)$$

where the bivariate NURBS basis functions are defined as

$$R_{i,j}^{p,q}(\xi, \eta) = \frac{N_i(\xi)M_j(\eta)w_{i,j}}{\sum_{i=1}^n \sum_{j=1}^m N_i(\xi)M_j(\eta)w_{i,j}}. \quad (12)$$

---

<sup>5</sup>[www.rhino3d.com](http://www.rhino3d.com)

Alternatively, using the mapping defined by Eq. (4), Eq. (11) can be written more succinctly as

$$\mathbf{S}(\boldsymbol{\xi}) = \sum_{A=1}^{n \times m} \mathbf{P}_A R_A^{p,q}(\boldsymbol{\xi}). \quad (13)$$

NURBS volumes are constructed from control points  $\mathbf{P}_{i,j,k} \in \mathbb{R}^{d_s}$ , weights  $w_{i,j,k} > 0$  as

$$\mathbf{V}(\xi, \eta, \zeta) = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^l \mathbf{P}_{i,j,k} R_{i,j,k}^{p,q,r}(\xi, \eta, \zeta) \quad (14)$$

where the trivariate NURBS basis functions  $R_{i,j,k}^{p,q,r}$  are given by

$$R_{i,j,k}^{p,q,r}(\xi, \eta, \zeta) = \frac{N_i(\xi) M_j(\eta) P_k(\zeta) w_{i,j,k}}{\sum_{\hat{i}=1}^n \sum_{\hat{j}=1}^m \sum_{\hat{k}=1}^l N_{\hat{i}}(\xi) M_{\hat{j}}(\eta) P_{\hat{k}}(\zeta) w_{\hat{i},\hat{j},\hat{k}}}. \quad (15)$$

Using the mapping given by Eq. (7), Eq. (14) can also be written as

$$\mathbf{V}(\boldsymbol{\xi}) = \sum_{A=1}^{n \times m \times l} \mathbf{P}_A R_A^{p,q,r}(\boldsymbol{\xi}) \quad (16)$$

*Remark 2.1.* As mentioned in Section 1.6, CAD representations are usually composed of surface models or boundary-representations. Trivariate discretizations defined by (16) are not normally explicitly given, and therefore some pre-processing is required before domain based numerical methods such as the FEM can be applied. Except for the case of simple cases such as extruded-surface models and swept models, this task is far from trivial. This is currently an open research topic in IGA.

### 3. NURBS as a basis for analysis: isogeometric finite element formulation

Our attention now focuses on the use of B-splines and NURBS as a discretization tool for analysis, outlining the core concepts of isogeometric analysis. In this section the important spaces and mappings are defined, followed by the isogeometric FEM formulation in which we use NURBS as a basis for analysis.

### 3.1. Relevant spaces

Familiarity must be gained with the spaces that are commonplace in isogeometric analysis and the relationships that exist between each. Those that are considered presently in the context of B-splines and NURBS include: parametric, physical and parent space.

#### 3.1.1. Parametric space

Parametric space (sometimes referred to as the ‘pre-image’ of the NURBS mapping) is formed by considering only the non-zero intervals between knot values. For the knot vectors considered previously, the parametric space is illustrated in Fig. 1 which can subsequently be reduced to a unit square through appropriate normalization. All parametric spaces can be reduced to a unit interval ( $d_p = 1$ ), square ( $d_p = 2$ ) or cube ( $d_p = 3$ ) in this manner. We define the parametric space as  $\hat{\Omega} \subset \mathbb{R}^{d_p}$  with a associated set of parametric coordinates  $\boldsymbol{\xi} = (\xi, \eta, \zeta) = (\xi^1, \xi^2, \xi^3) \in \hat{\Omega}$  ( $d_p = 3$ ). If normalization is performed,  $\hat{\Omega} = [0, 1]^{d_p}$ .

Fig. 1 also reveals that regions bounded by knot lines with non-zero parametric area lead to a natural definition of element domains. More formally, a set  $\mathcal{S} \subset \Xi$  of unique knot values can be defined as

$$\mathcal{S} = \{\xi_1, \xi_2, \dots, \xi_{n_s}\} \quad \xi_i \neq \xi_{i+1} \text{ for } 1 \leq i \leq n_s - 1 \quad (17)$$

where  $n_s$  is the number of unique knot values. This is generalized to  $\mathcal{S}^i \subset \Xi^i$  which represents the unique knot values for each parametric direction  $i = 1, 2, \dots, d_p$ . Elements can now be defined in the general multivariate case as

$$\begin{aligned} \hat{\Omega}^e &= [\xi_i, \xi_{i+1}] \otimes [\eta_j, \eta_{j+1}] \otimes [\zeta_k, \zeta_{k+1}] \quad 1 \leq i \leq n_s^1 - 1, \\ & \quad 1 \leq j \leq n_s^2 - 1, \\ & \quad 1 \leq k \leq n_s^3 - 1, \\ & \quad \xi_i \in \mathcal{S}^1, \eta_j \in \mathcal{S}^2, \zeta_k \in \mathcal{S}^3 \end{aligned} \quad (18)$$

where  $n_s^1, n_s^2$  and  $n_s^3$  represent the number unique knots in the  $\xi, \eta$  and  $\zeta$  parametric directions respectively. This leads

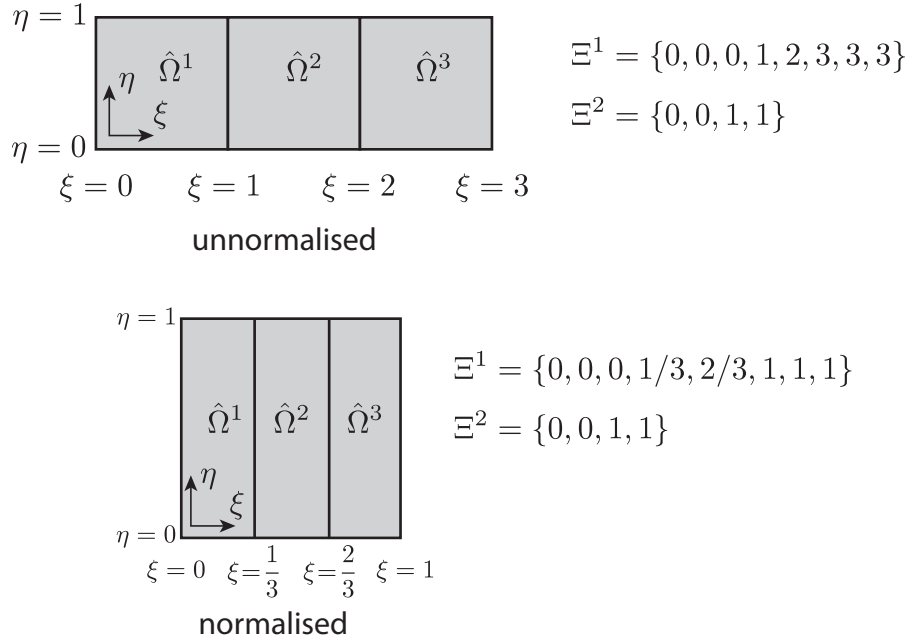


Figure 1: Parametric space defined by non-zero knot intervals. All parametric spaces defined for a B-spline or NURBS patch can be normalized to a unit interval, unit square, unit cube in 1D, 2D, 3D respectively. Knot lines provide a natural definition of element boundaries.

to a natural numbering scheme for elements over a patch as

$$e = k(n_s^2 - 1)(n_s^1 - 1) + j(n_s^1 - 1) + i. \quad (19)$$

Eq. (18) and (19) can be simplified accordingly for  $d_p = 1, 2$ .

### 3.1.2. Physical space

The B-spline and NURBS mappings of Eqs. (5) and (11) transform coordinates in parameter space to physical space  $\Omega \subset \mathbb{R}^{d_s}$ . For three-dimensional domains, we associate a coordinate system  $\mathbf{x} = (x, y, z) = (x^1, x^2, x^3)$  for physical space, which appropriate modifications for one- and two-dimensional problems. Fig. 2 illustrates a NURBS mapping for the parametric space shown in Fig. 1 for an arbitrary set of control points and weights. The control grid (which defines the connectivity between control points) is also shown. The non-interpolatory nature of control points in the interior of the domain is evident, and represents a notable difference over conventional Lagrangian meshes.

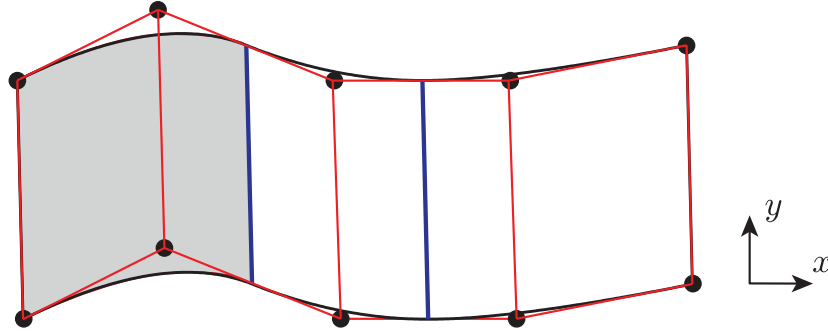


Figure 2: A 2D NURBS surface defined for knot vectors  $\Xi^1 = \{0, 0, 0, 1, 2, 3, 3, 3\}$ ,  $\Xi^2 = \{0, 0, 1, 1\}$ . The control mesh is shown in red with control points denoted by black circles. Knot lines shown in blue indicate element boundaries.

### 3.1.3. Parent space

The previous three spaces are inherent to B-splines and NURBS, but for analysis to be performed we require the definition of an additional space, commonly referred to as parent space  $\tilde{\Omega} = [-1, 1]^{d_p}$ . This is required for the use of numerical integration routines which are often defined over the interval  $[-1, 1]$ . Parent space coordinates are denoted as  $\tilde{\boldsymbol{\xi}} = (\tilde{\xi}, \tilde{\eta}, \tilde{\zeta}) = (\tilde{\xi}^1, \tilde{\xi}^2, \tilde{\xi}^3)$  with corresponding simplifications for  $d_p = 1, 2$ .

### 3.2. Mappings (change of variables)

The use of NURBS basis functions for discretization introduces the concept of parametric space which is absent in conventional FE implementations. The consequence of this additional space is that an additional mapping must be performed to operate in parent element coordinates. As shown in Fig. 3, two mappings are considered for IGA with NURBS: a mapping  $\tilde{\phi}^e : \tilde{\Omega} \rightarrow \hat{\Omega}^e$  and  $\mathbf{S} : \hat{\Omega} \rightarrow \Omega$ . The mapping  $\mathbf{x}^e : \tilde{\Omega} \rightarrow \Omega^e$  is given by the composition  $\mathbf{S} \circ \tilde{\phi}^e$ .

## 4. MIGFEM- A Matlab IGA (X)FEM code

In this section we describe shortly the open source IGA Matlab (X)FEM program which can be downloaded from <https://sourceforge.net/projects/cmcodes/>. The code supports one, two and three dimensional linear elasticity problems. Extended IGA for crack and material interface modeling is also implemented. Geometrically nonlinear solid and structural mechanics models are available. The features of the code include:

- Global  $h,p$  and  $k$ -refinement is provided for one, two and three dimensional meshes.

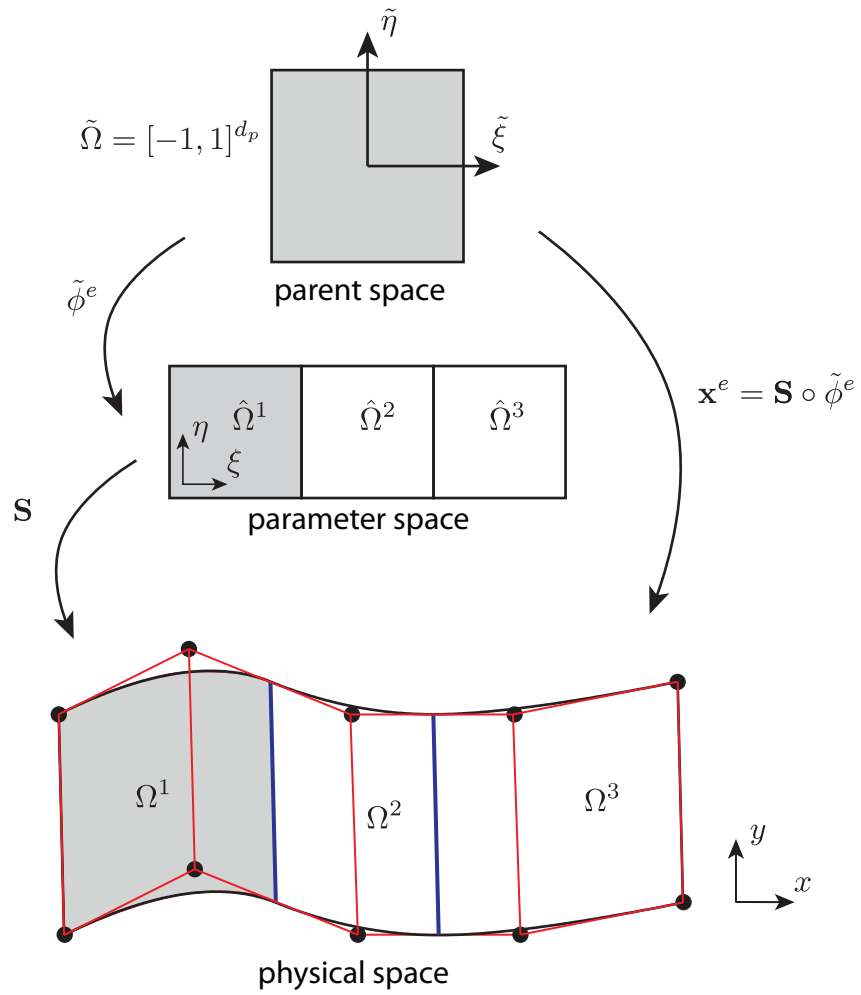


Figure 3: Diagrammatic interpretation of mappings from parent space through parametric space to physical space.



- Extended IGA for 2D/3D stationary traction-free cracks and material interfaces.
- Visualization of displacements and stresses in Paraview.
- Inhomogeneous Dirichlet boundary conditions are treated with the least squares method.
- Compatible multi-patch isogeometric formulation for two dimensional problems.
- Support for T-splines via the Bézier extraction operators.
- Structural elements including beams, plates and thin shells.
- Implementation of collocation methods for IGA.

#### 4.1. Data structure

MIGFEM follows the Matlab FEM code described in [26]. The main data structures include (1) *element* (store the element connectivity), (2) *controlPts* (store control point coordinates), (3) *weights* (store the weights) (4) *K* (stiffness matrix) and (5) *f* (external force vector). Contrary to FEM in which the element connectivity and nodal coordinates are inputs which have been created by a meshing program, in IGA, the input consists of CAD data including knots, control points, order of basis functions. Therefore, one has to construct the *element* matrix based on the knots and the basis orders.

Construction of the *element* matrix is illustrated by a 2D example shown in Fig. 4. Given the knot vectors  $uKnot$  and  $vKnot$  together with the orders of the basis  $p$  and  $q$ , one can compute the number of control points along the  $\xi$  and  $\zeta$  directions, denoted by  $n$  and  $m$ . Then we define a two dimensional matrix of dimension  $m \times n$  called *node\_pattern* given by for the illustrated example shown in Fig. 4 which has 4 control points along  $\xi$  and 4 control points along  $\eta$  directions

$$node\_pattern = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}, \quad (20)$$

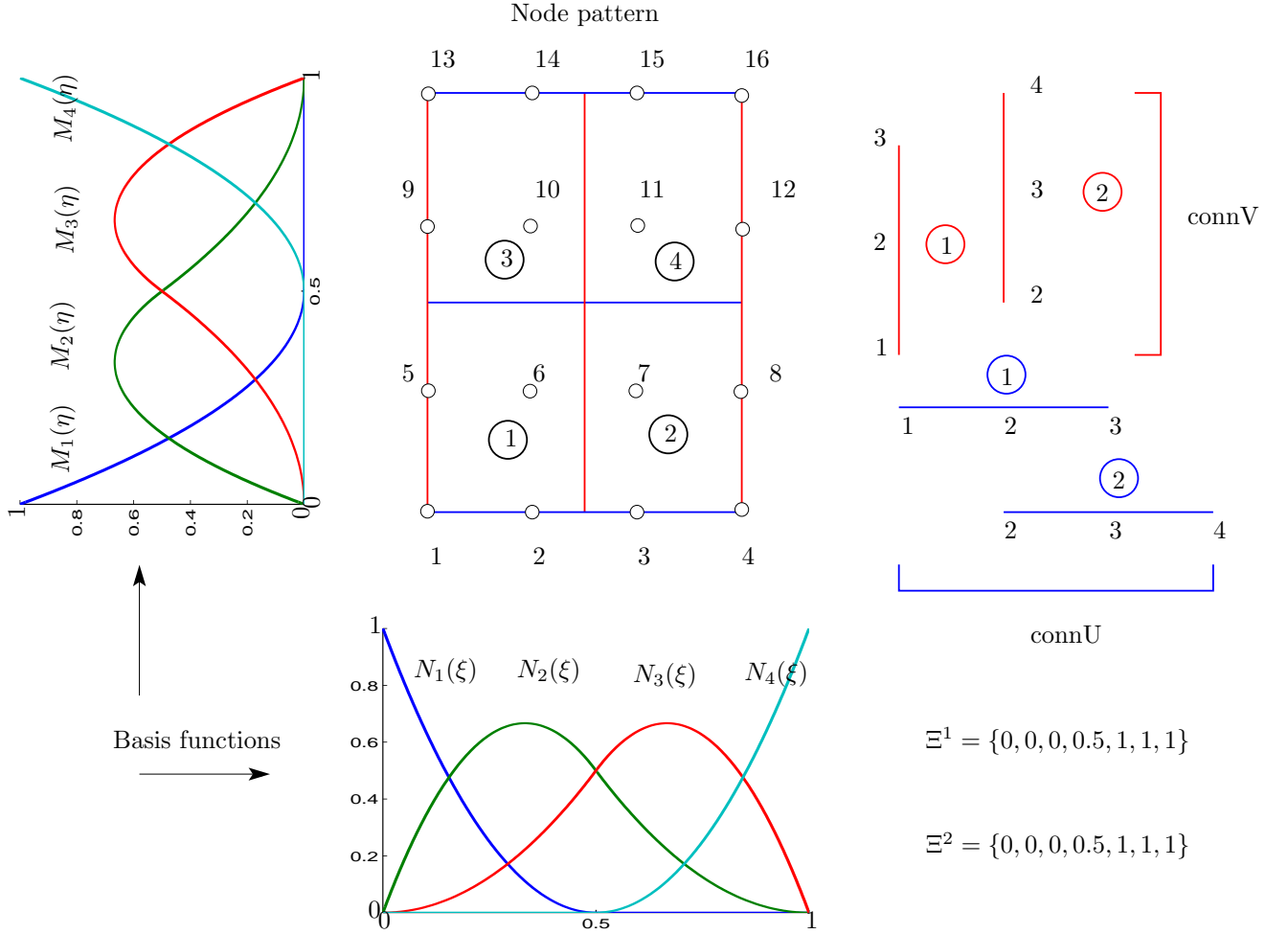


Figure 4: Two dimensional isogeometric analysis: mesh generation for a bi-quadratic NURBS surface ( $2 \times 2$  elements). The circles denote the control points.

which is simply an application of Eq. (4) for defining global indexes.

The number of elements along the two directions  $n_s^1 - 1, n_s^2 - 1$  are  $noElemsU = \text{length}(\text{unique}(uKnot)) - 1$  and  $noElemsV = \text{length}(\text{unique}(vKnot)) - 1$  (in the Matlab language). The connectivity matrix for the  $\xi$  direction, denoted by  $connU$  which is a  $noElemsU \times (p + 1)$  matrix and the connectivity matrix for the  $\eta$  direction, denoted by

$connV$  which is a  $noElemsV \times (q + 1)$  matrix are given by

$$connU = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix}, \quad connV = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix}, \quad (21)$$

for the example under consideration. Matrix  $connU$  stores the indices of the non-zero basis functions at each knot span for the  $\xi$  direction. Having this information and the global indexes of the NURBS basis given in Eq. (20), we are able to define the connectivity matrix for the whole mesh, called  $element$  which is a  $noElemsU * noElemsV \times (p + 1)(q + 1)$  matrix. For the example being considered, this matrix reads

$$element = \begin{bmatrix} 1 & 2 & 3 & 5 & 6 & 7 & 9 & 10 & 11 \\ 2 & 3 & 4 & 6 & 7 & 8 & 10 & 11 & 12 \\ 5 & 6 & 7 & 9 & 10 & 11 & 13 & 14 & 15 \\ 6 & 7 & 8 & 10 & 11 & 12 & 14 & 15 & 16 \end{bmatrix}. \quad (22)$$

Note that the matrix  $element$  is the transpose of the IEN matrix. We decided to use  $element$  instead of IEN to be compatible with the FEM code [26] on which MIGEM is built.

Finally in order to compute the mapping from the parent domain  $\tilde{\Omega}$  to the parametric space, we define the knot intervals in two directions as

$$rangeU = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 1 \end{bmatrix}, \quad rangeV = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 1 \end{bmatrix}. \quad (23)$$

In order to retrieve the parametric coordinates of a specific element, the matrix  $index$ , that is a  $noElemsU * noElemsV \times 2$  matrix, is used. For a given element  $e$ , its parametric coordinates are determined by  $rangeU(index(e,1),:); rangeV(index(e,2),:)$ .

The above discussion, together with the illustration given in Fig. 4 is implemented in Matlab in the file **generateIGA2DMesh.m**, located in folder **meshing**. In that folder, one can find similar M files for generating 1D and 3D meshes.

#### 4.2. Shape function routines

The shape function routines (evaluate NURBS basis functions and derivatives with respect to parametric coordinates) are implemented using MEX files to improve the performance. They are located in folder **C\_files**. The files **NURBS1DBasisDers.c**, **NURBS2DBasisDers.c**, **NURBS3DBasisDers.c** are used to compute the NURBS basis function and their first derivatives in 1D, 2D and 3D at a given point. For collocation methods, the second derivatives are also computed. The corresponding files are located under the **collocation** folder.

#### 4.3. Assembly process

The assembly of an IGA-FEM code is given in each script file, where it can be seen that the procedure is almost identical to that used in the conventional FEM. The minor differences lie in (1) the need of the elements to be located in the parameter space and (2) the second map (from the parent domain to the parametric domain) in the numerical integration of the stiffness matrix. For XIGA, the assembly process needs to take into account the enrichment functions. These are given as the separate subroutine **assembly.m** in the **xiga** folder.

#### 4.4. Post-processing

We present here a simple technique to visualize the IGA results that reuse available visualization techniques for finite elements. To simplify the exposé, only 2D cases are considered here. In the first step, a mesh consisting of four-noded quadrilateral (Q4) elements is generated, see Fig. 5. We call this mesh the visualization mesh (whose connectivity matrix is stored in *elementV* and nodal coordinates are stored in *node*) whose nodes are images of the knots  $\xi_i, \eta_j$  in the physical space. In the second step, quantities of interest e.g., stresses are computed at the nodes of the Q4 mesh. This mesh together with the nodal values can then be exported to a visualization program such as Paraview, see [56], for visualization. It should be emphasized that due to the high order continuity of the NURBS basis, there is no need to perform nodal averaging as required in standard  $C^0$  finite element analysis to obtain smooth fields. The file **plotStress1.m** located in folder **post-processing** contains the code for building the Q4 visualization mesh, computing the stresses at the nodes of this mesh and exporting the result to Paraview. A brief tutorial for using Paraview is included in the **README** file in the top-level directory.

For three-dimensional problems, the same procedure is used where a mesh of tri-linear brick elements is created and the values of interest are computed at the nodes of this mesh (see file **plotStress3d.m**). The results are then exported

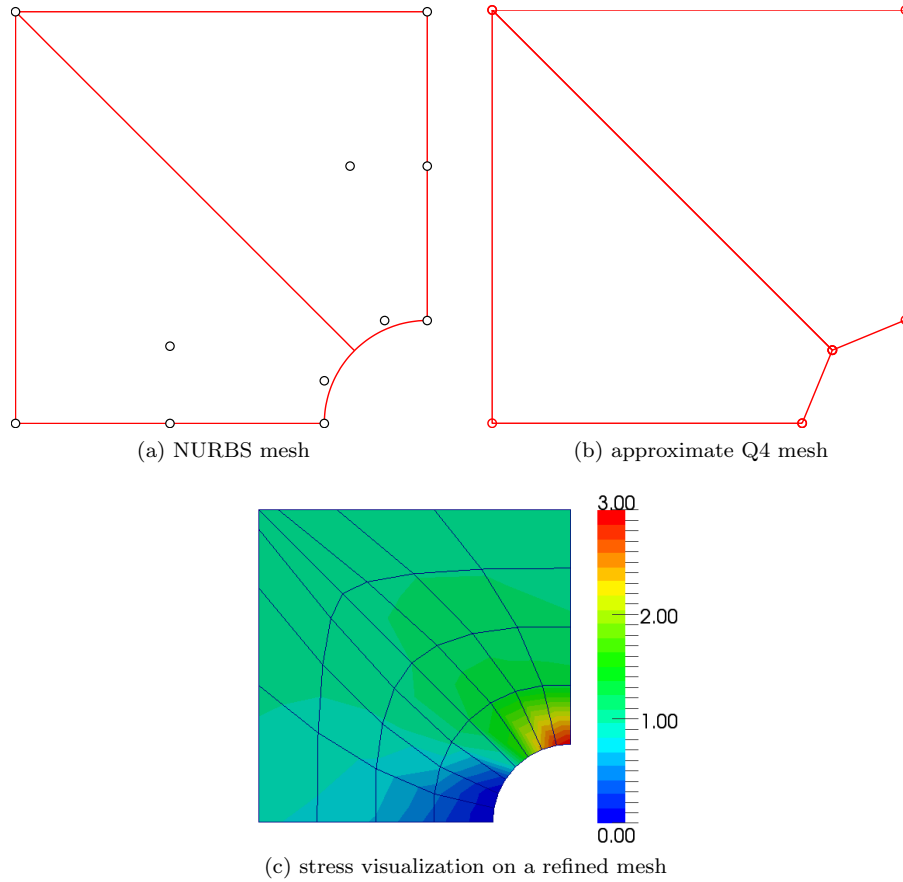


Figure 5: Exact NURBS mesh (top left) and approximate Q4 mesh (top right) for visualization purpose. The nodes in the Q4 mesh are the intersections of the  $\xi$  and  $\eta$  knot lines. The bottom figure shows a contour plot of a stress field in Paraview. It should be emphasized that the mesh in (b) does not provide a sufficiently smooth contour plot for it to be directly usable. The result given in (c) was obtained with a refined NURBS mesh (hence a refined Q4 mesh).

to Paraview under a structured grid format (\*.vts files), see the file **mshToVTK.m**).

#### 4.5. $h, p, k$ -refinement

For the refinement of NURBS, we reuse the NURBS Toolbox described in [48]. We construct a NURBS surface as shown in Fig.6a. Using a uniform  $h$ -refinement that divides a knot span into two we obtain the mesh given in Fig.6b. Finally, Fig.6d gives the mesh which is obtained by the process in which  $h$ -refinement is employed first and then  $p$ -refinement is performed. After using the NURBS toolbox, the NURBS object is then converted to MIGFEM data structures using the function **convert2DNurbs** located in folder **nurbs-util**.

#### 4.6. Input file for MIGFEM

The input files define the polynomial degree, knots and control points for an IGA discretization. This file replaces the standard FE mesh file. Note that, for backward compatibility with older versions of the MIGFEM code, some input files do not use the NURBS toolbox to create the NURBS object. For those input files, it is however impossible to perform order elevation and hence  $k$ -refinement. It is, therefore, recommended to create the NURBS objects using the NURBS toolbox [48] which, besides the aforementioned refinement functionality, also supports many useful operations such as extrusion, rotation etc.

Incorporating NURBS into an existing FE code cannot be considered a trivial task due to the presence of various spaces (index, parameter) that are not present in conventional FE codes, Bézier extraction [17, 116] provides a FE data structure that allows for a straightforward implementation of NURBS/T-splines into any FE codes. Appendix B briefly presents this concept and its implementation in MIGFEM.

#### 4.7. Collocation methods

Collocation methods are based on the strong form of the PDE being solved. The main idea of these methods is that the governing equations, which include higher-order derivatives, should be satisfied at a discrete set of points. This gives rise to a linear system where the coefficients of the basis functions are the unknowns. The computational cost of this procedure is much lower than in the Galerkin method, as no integrals need to be evaluated. Moreover, the assembly process can trivially be parallelized, as each row of the linear system corresponds to one collocation point and

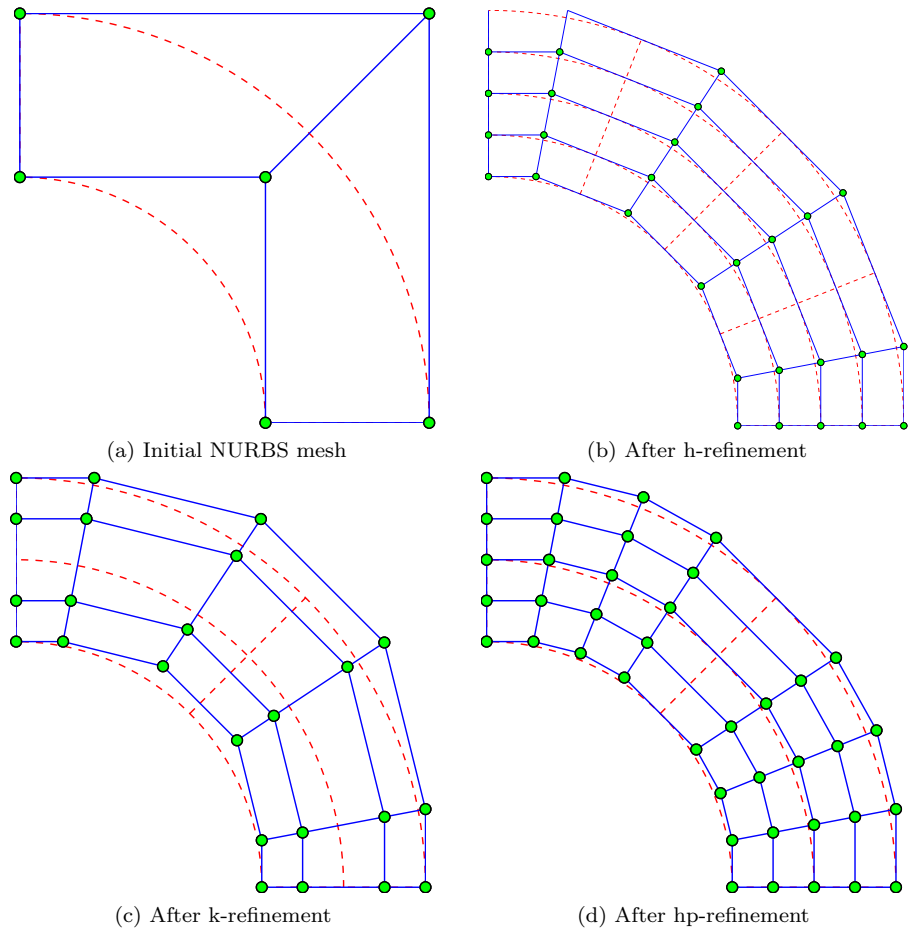


Figure 6: Illustration of the utilization of the NURBS toolbox in building NURBS object (a). From the initial mesh, different meshes can be obtained using either  $h$ -refinement,  $p$ -refinement or combination thereof. As can be seen from (c) and (d),  $k$ -refinement (c) is more efficient than  $hp$ -refinement (d). The function `plotMesh.m` in folder `meshing` is used to plot NURBS mesh and control polygon.

can be computed independently of the others. The accuracy of the method depends on the location of the collocation points. In the context of IGA, these points have been traditionally chosen at the so-called Greville abscissae, defined by knot averages, or Demko abscissae, defined as the extrema of Chebyshev splines [4]. It has been observed that in some cases, the convergence rate of IGA collocation at Greville or Demko abscissae is suboptimal compared to weak form methods, in particular for odd polynomial degrees. Recently a collocation scheme based on superconvergence theory was proposed in [2], which has been observed to give more accurate results at the expense of more basis function evaluations.

The folder **collocation** contains implementation examples of IGA collocation for Poisson’s equation and linear elasticity. Both Greville abscissae collocation and superconvergent collocation are implemented. There is ongoing research in developing efficient iterative solvers for the linear system resulting from collocation [12, 38], imposition of Neumann boundary conditions [75], and plate collocation formulations [65].

#### 4.8. XIGA implementation

There are certainly different ways to implement XIGA. We present a way that reuse most of the tools present in an existing XFEM code. We use the approximate Q4 mesh used for visualization as discussed in Section 4.4 for selection of enriched control points. The level set values defining the crack at the vertices of this mesh are then computed. Based on these level sets, elements cut by the crack and elements containing the crack tip can be determined [130]. In a FEM context, its four nodes are then enriched using the Heaviside function. In an isogeometric framework, however, the control points (basis functions) associated to this element are enriched. The implementation of this process is very similar to that of our XFEM code [93] with only one small modifications, and thus the proposed technique is considered simpler than that adopted in [51]. We emphasize that the crack geometry is defined in the physical space to keep the usual XFEM notation.

*Remark 4.1.* Note that for simple geometries as those tackled in this paper, the use of level sets is not necessary. More generally, describing open surfaces (cracks) with level sets requires two functions that, for crack growth simulations, must be reinitialized for stability (this decreases accuracy) and reorthogonalized every few time steps. This is particularly cumbersome and along with the difficulties in dealing with intersecting and branching cracks, could explain the recent trend of research efforts in the area of phase field models of fracture, see e.g., [18, 83], and the thick level set method [85].



## Crack visualization

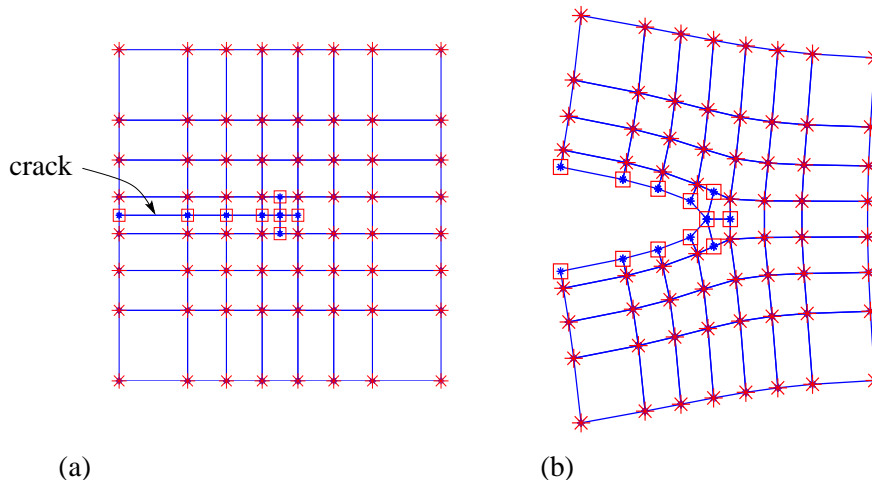


Figure 7: Crack visualization in XIGA: (a) build a mesh that is compatible to the crack by introducing double nodes along the crack (square nodes) and (b) assign the displacement jumps to these new nodes. Note that the square nodes in the tip element are used only for compatibility purposes. Exact mode I displacements are imposed on the bottom, right and top edge using the Lagrange multiplier method while Neumann BCs from the exact stress field are enforced on the left edge. We refer to [93] for a detailed description of this standard problem.

Fig. 7 illustrates the idea for crack visualization with the script `post-processing/crackedMeshNURBS.m` providing implementation details. The contour plots of the displacement and stress field of a mode I cracked sample are given in Fig. 8. Note that the stresses at points on the crack are simply set to zero (traction-free cracks) and the stresses of the new nodes of the tip element are interpolated from the values of the four nodes of this Q4 element.

*Remark 4.2.* As is the case for XFEM, integration over elements cut by the cracks usually requires subdivision of the elements into integration sub cells. We refer to [88] for a recent discussion on this issue. In addition to this popular technique, a simple integration rule is also provided – elements crossed by the crack and tip-enriched elements are numerically integrated using a regular Gauss-Legendre quadrature with a large number of Gauss points as done in [44].

## 5. Structural mechanics

Thanks to the high order continuity provided by NURBS/T-splines, the implementation of rotation-free thin beam/plate/shell elements becomes direct and simple. In this section, we are going to present the implementation of a rotation-free IGA Kirchhoff plate formulation (rotation free shell elements can be found in folder `structural-mechanics`). The

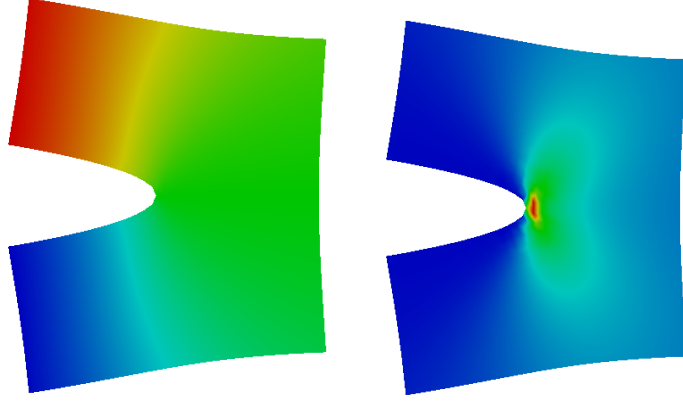


Figure 8: Contour plots on a cracked mesh: (a) vertical displacement and (b) normal stress in the vertical direction.

plate geometry and the deflection are both approximated by NURBS. At control points there is only one unknown—the deflection or transverse displacement. For simplicity, only isotropic elastic plates are considered. We refer to [105] for a treatment of plate theories.

The element stiffness matrix is defined as

$$\mathbf{K}_e = \int_{\Omega_e} \mathbf{B}_e^T \mathbf{D} \mathbf{B}_e d\Omega, \quad (24)$$

where the constitutive matrix  $\mathbf{D}$  reads

$$\mathbf{D} = \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & 0.5(1-\nu) \end{bmatrix}, \quad (25)$$

where  $E, \nu$  are the Young's modulus and Poisson's ratio, respectively;  $h$  denotes the plate thickness and the element displacement-curvature matrix  $\mathbf{B}_e$  that contains the second derivatives of the shape functions is given by

$$\mathbf{B}_e = \begin{bmatrix} R_{1,xx} & R_{2,xx} & \cdots & R_{n,xx} \\ R_{1,yy} & R_{2,yy} & \cdots & R_{n,yy} \\ 2R_{1,xy} & 2R_{2,xy} & \cdots & 2R_{n,xy} \end{bmatrix}, \quad (26)$$

where  $n$  denotes the number of basis functions of element  $e$  and  $R_{A,xx} \equiv d^2 R_A/dx^2$ .

### 5.1. Boundary conditions

For clamped BCs one needs to fix the rotations. The nodal unknowns are, however, only the transverse displacements  $w$ . To fix the rotation of a boundary, we simply fix two rows of control points at the boundary [67] because these control points define the tangent of the surface at the boundary, see Fig. 9.

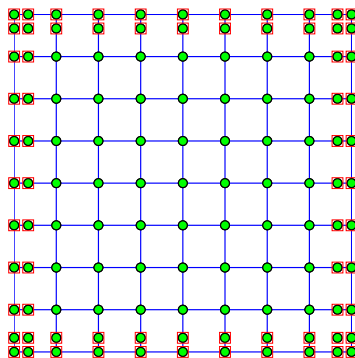


Figure 9: Enforcing BCs for a fully clamped plate: simply fixing the deflections of two rows of control points around the clamped boundary. Note that the set of CPs next to the boundary CPs are not artificially added to impose the rotations. They are simply the CPs defining the geometry of the plate.

### 5.2. Symmetry boundary conditions

Fig. 10 illustrates the use of symmetry boundary conditions when only 1/4 of the plate is modeled. Along the symmetry lines, the rotation should be zero which can be enforced by constraining the deflection ( $w$ ) of two rows of control points along these lines together.

## 6. Verification examples

In this section, numerical examples in linear elasticity and linear elastic fracture mechanics in 2D and 3D are presented with the purpose to serve as a set of verification examples for MIGFEM. They include an infinite plate with a circular hole under constant in-plane tension, the pinched cylinder, an edge cracked plate in tension, a three-dimensional mode I fracture problem and a large deformation thin shell problem. Unless otherwise stated, standard direct imposition of Dirichlet BCs is used. Units are standard International System (SI) units.

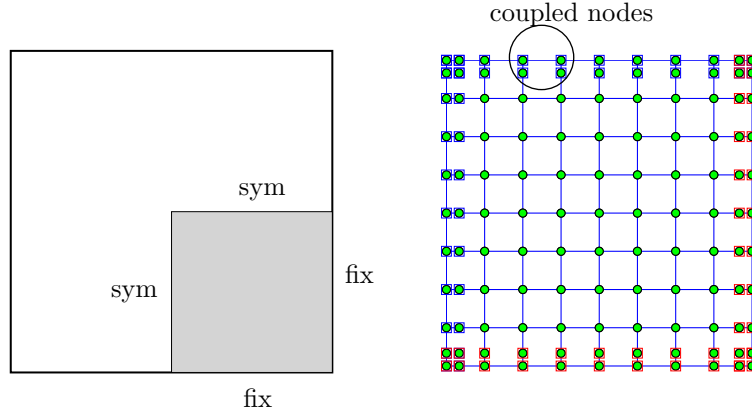


Figure 10: A fully clamped square plate: 1/4 model is analyzed using appropriate symmetry BCs. Along the symmetry lines, the rotation is fixed which can be achieved by enforcing the deflection of two rows of control points that define the tangent of the plate to have the same value.

## 6.1. Two and three dimensional solid mechanics

### 6.1.1. Infinite plate with a circular hole

The problem considered is that of an infinite plate with a circular hole in the center under constant in-plane tension at infinity as shown in Fig. 11 where, due to symmetry, only a quarter of the plate is modeled. The plate dimension is taken to be  $L \times L$  and the circular hole has a radius  $R$ . The exact stress field in the plate is given by

$$\sigma_{xx}(r, \theta) = 1 - \frac{R^2}{r^2} \left( \frac{3}{2} \cos 2\theta + \cos 4\theta \right) + \frac{3 R^4}{2 r^4} \cos 4\theta \quad (27a)$$

$$\sigma_{yy}(r, \theta) = -\frac{R^2}{r^2} \left( \frac{1}{2} \cos 2\theta - \cos 4\theta \right) - \frac{3 R^4}{2 r^4} \cos 4\theta \quad (27b)$$

$$\sigma_{xy}(r, \theta) = -\frac{R^2}{r^2} \left( \frac{1}{2} \sin 2\theta + \sin 4\theta \right) + \frac{3 R^4}{2 r^4} \sin 4\theta, \quad (27c)$$

where  $r, \theta$  are the usual polar coordinates centered at the center of the hole.

The material properties are specified as  $E = 10^3$ , Poisson's ratio  $\nu = 0.3$  and the geometry is such that  $L = 4$ ,  $R = 1$ . A plane stress condition is assumed. The problem is solved with quadratic NURBS meshes such as those shown in Fig. 12. The control points and weights for the coarsest mesh can be found in [57] or file `plateHoleCkData.m`.

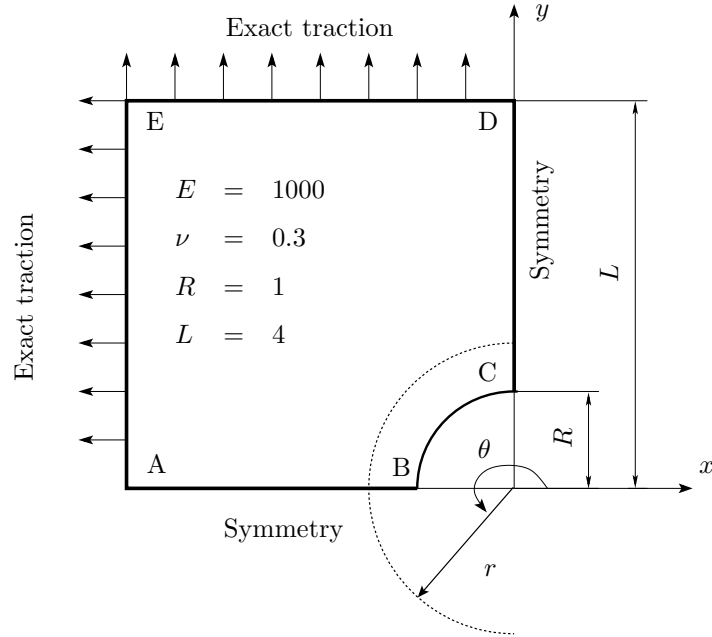


Figure 11: Infinite plate with a circular hole under constant in-plane tension: quarter model. Boundary conditions include:  $u_y = 0$  (AB),  $u_x = 0$  (CD),  $\bar{\mathbf{t}}^T = (-\sigma_{xx}, -\sigma_{xy})$  (AE),  $\bar{\mathbf{t}}^T = (\sigma_{xy}, \sigma_{yy})$  (ED).

Fig. 13, generated in Paraview, illustrates the contour plot of numerical  $\sigma_{xx}$ . Note that the stress concentration at point  $(R, 3\pi/2)$  is well captured and a smooth stress field is obtained throughout.

*Remark 6.1.* Using the visualization technique described in Section 4.4 for this problem, a note should be made on the evaluation of the stress field at the top left corner where there are two coincident control points. This causes a singular Jacobian matrix. Therefore at this corner, the stresses at a point slightly shifted from the original position are used.

### 6.1.2. Pinched cylinder

In order to demonstrate the performance of the 3D IGA implementation, we consider the pinched cylinder problem as shown in Fig. 14. Note that we discretize the shell with solid NURBS elements. Due to symmetry, only 1/8 of the model is analyzed. A tri-quadratic NURBS mesh ( $p = q = r = 2$ ) was used for the computation. Details can be found in the file **igaPinchedCylinder.m**. Fig. 15 shows the mesh and the contour plot of the displacement in the point load direction. Post-processing is done in Paraview and we refer to Section 4.4 for details. We recognize that the problem under consideration is a shell like structure that would be more accurately modelled using appropriate shell elements,

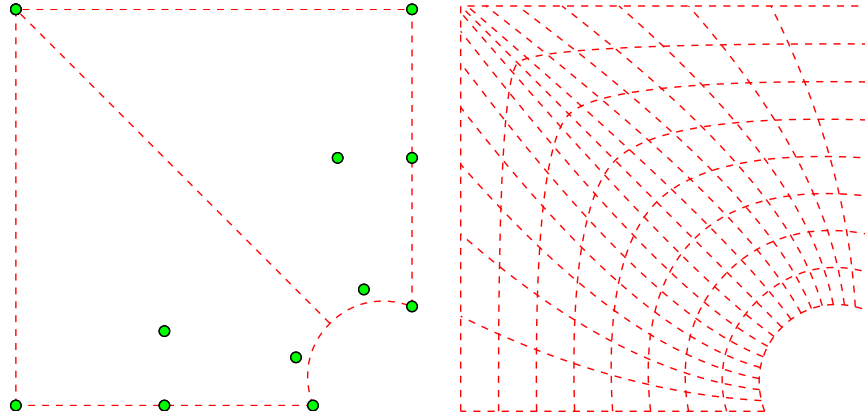


Figure 12: Plate with a hole: coarse mesh of 2 bi-quadratic elements (left) and refined mesh (right).

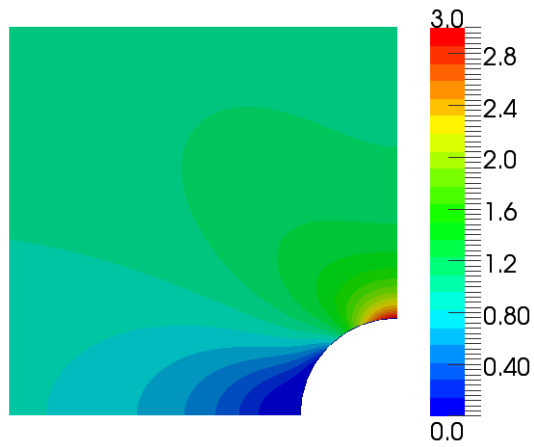


Figure 13: Plate with a hole: distribution of numerical  $\sigma_{xx}$  obtained with a  $32 \times 16$  quadratic mesh having 4488 dofs.

but the example is merely intended to illustrate the ability of the method to analyze 3D geometries.

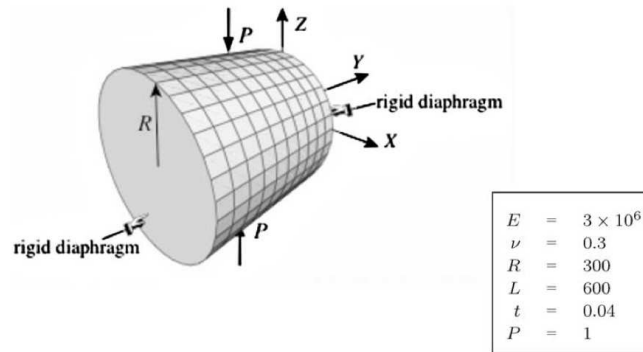


Figure 14: Pinched cylinder. Problem description and data [49].

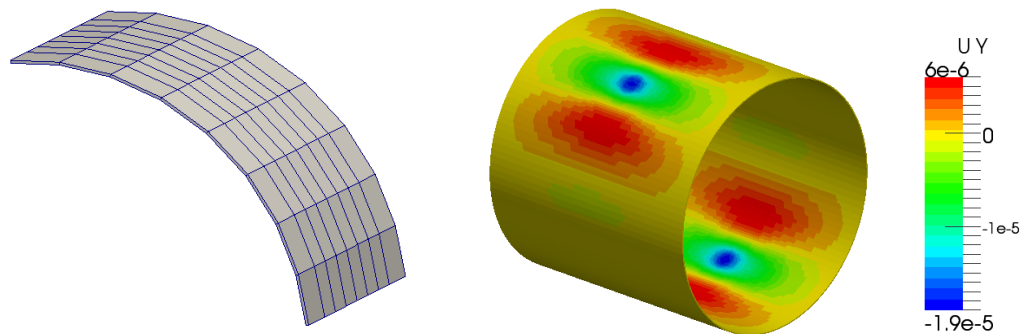


Figure 15: Pinched cylinder: (a) mesh of one octant of the cylinder and (b) contour plot of the displacement in the direction of the point load.

## 6.2. Two and three dimensional fracture mechanics

### 6.2.1. Edge cracked plate in tension

A plate of dimension  $b \times 2h$  is loaded by a tensile stress  $\sigma = 1.0$  along the top edge and bottom edge as shown in Fig. 16. In the computation, the displacement along the  $y$ -axis is fixed at the bottom edge and the bottom left corner is fixed in both  $x$  and  $y$  directions. The material parameters are  $E = 10^3$  and  $\nu = 0.3$ . A plane strain condition is assumed. The reference mode I stress intensity factor (SIF) for this problem is given in [131] and is calculated as

$$K_I = F \left( \frac{a}{b} \right) \sigma \sqrt{\pi a}, \quad (28)$$

where  $a$  is the crack length,  $b$  is the plate width, and  $F(a/b)$  is an empirical function. For  $a/b \leq 0.6$ , function  $F$  is given by

$$F\left(\frac{a}{b}\right) = 1.12 - 0.23\left(\frac{a}{b}\right) + 10.55\left(\frac{a}{b}\right)^2 - 21.72\left(\frac{a}{b}\right)^3 + 30.39\left(\frac{a}{b}\right)^4. \quad (29)$$

In the present implementation, this problem is solved using both XFEM and an extended isogeometric formulation. The SIF is computed using an interaction integral. We refer to [84] for details.

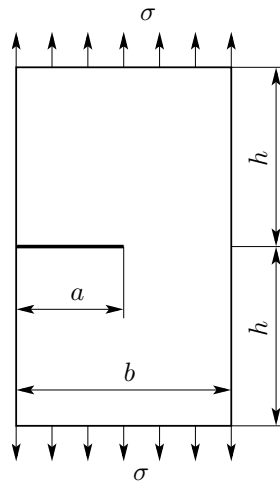


Figure 16: Edge cracked plate in tension: geometry and loading.

We first verify the implementation of the XIGA code by comparing the XIGA result with the XFEM result for the case of  $a = 0.45$ ,  $b = 1$  and  $h = 1$ . The XFEM and XIGA meshes are given in Fig. 17. Both meshes have the same uniform distribution of nodes/control points. For the XFEM mesh, bilinear Q4 elements are used. For the XIGA mesh, cubic ( $p = q = 3$ ) B-spline basis functions are adopted. Fig. 18 shows the contour plots of the vertical displacement obtained with XFEM and XIGA.

We now consider the computation of the mode I SIF for a crack of length  $a = 0.3$ . The reference SIF for this problem is  $K_I^{\text{ref}} = 1.6118$ . Both a linear and a cubic B-spline basis are used for three different meshes. The results are given in Table 1. It should be emphasized that in the computation of the interaction integral, we use bilinear Lagrange shape functions i.e., shape functions of Q4 elements to compute the derivatives of the weight function. This guarantees that the weight function takes a value of unity on an open set containing the crack tip and vanishes on an outer contour



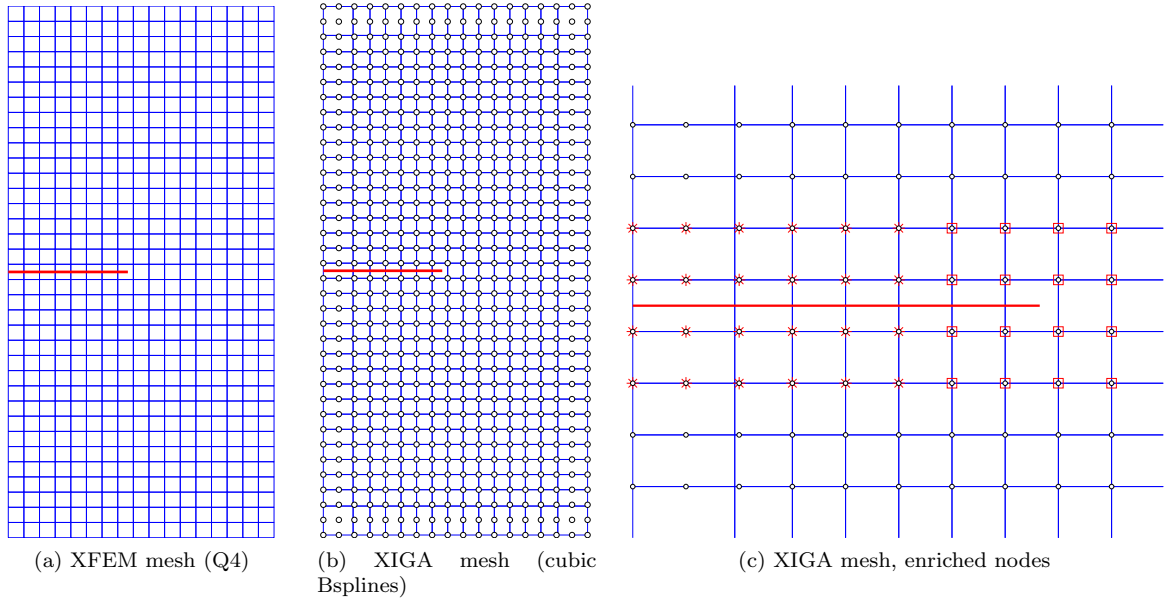


Figure 17: Edge cracked plate: XFEM and XIGA meshes. Both have the same number of displacement dofs of 1296. The thick line denotes the crack. Square nodes denote tip enriched nodes whereas star nodes represent Heaviside enriched nodes.

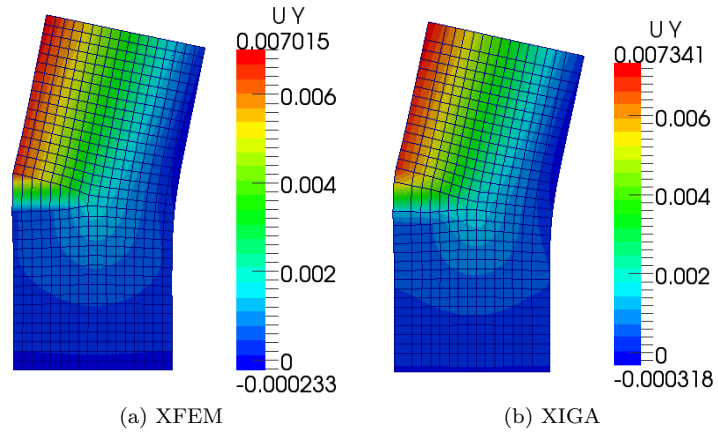


Figure 18: Edge cracked plate:  $u_y$  contour plots on deformed configuration (enlargement factor of 30 used).

as shown in Fig. 19. B-splines functions are not interpolatory and therefore cannot be used to approximate the weight functions.

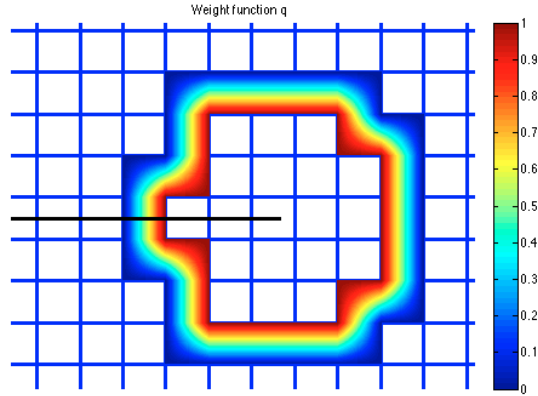


Figure 19: Distribution of weight function used in the computation of the interaction integral. Four-noded quadrilateral elements with bilinear Lagrange shape functions are used to interpolate the weight function.

mesh	disp. dofs	$K_I$ (linear)	Error (%)	$K_I$ (cubic)	Error (%)
$9 \times 18$	324	1.4997	6.96	1.5560	3.46
$18 \times 36$	1296	1.5823	1.83	1.6150	0.20
$36 \times 72$	5184	1.5968	0.93	1.6117	0.01

Table 1: Edge cracked plate: SIFs results. The reference SIF is  $K_I^{\text{ref}} = 1.6118$ . Note that linear NURBS are equivalent to the conventional bilinear finite elements.

### 6.2.2. Three-dimensional mode I fracture problem

This example aims to show the capability of MIGFEM for solving three-dimensional (3D) fracture problems. For 3D cracks, the polar coordinates in the branch functions are defined in terms of the level sets as [125]

$$r = \sqrt{\varphi(\xi, \eta, \zeta)^2 + \psi(\xi, \eta, \zeta)^2}, \quad \theta = \text{atan} \left( \frac{\varphi(\xi, \eta, \zeta)}{\psi(\xi, \eta, \zeta)} \right), \quad (30)$$

where the level set field  $\Phi = (\varphi, \psi)$  is interpolated as

$$\Phi(\xi, \eta, \zeta) = \sum_I R_I(\xi, \eta, \zeta) \Phi_I. \quad (31)$$

We refer to [125] for details concerning the derivatives of the branch functions with respect to the parametric coordinates  $(\xi, \eta, \zeta)$ .

The mode I 3D fracture problem we are solving is given in Fig. 20. The exact displacement field is given by

$$\begin{aligned} u_x(r, \theta) &= \frac{2(1+\nu)}{\sqrt{2\pi}} \frac{K_I}{E} \sqrt{r} \cos \frac{\theta}{2} \left( 2 - 2\nu - \cos^2 \frac{\theta}{2} \right) \\ u_y(r, \theta) &= 0 \\ u_z(r, \theta) &= \frac{2(1+\nu)}{\sqrt{2\pi}} \frac{K_I}{E} \sqrt{r} \sin \frac{\theta}{2} \left( 2 - 2\nu - \cos^2 \frac{\theta}{2} \right), \end{aligned} \tag{32}$$

where  $K_I = \sigma\sqrt{\pi a}$  is the stress intensity factor,  $\nu$  is Poisson's ratio and  $E$  is Young's modulus. In our example,  $a = 100$  mm;  $E = 10^7$  N/mm<sup>2</sup>,  $\nu = 0.3$ ,  $\sigma = 10^4$  N/mm<sup>2</sup>. On the bottom, right and top surfaces, essential BCs taken from Eq. (32) are imposed using the penalty method. A penalty parameter of  $1e10$  was used. We note that this problem can be more effectively solved with two-dimensional elements. This example however aims at presenting how 3D extended IGA can be implemented. Furthermore, it also illustrates how Dirichlet BCs are enforced on surfaces rather than the usual case of line boundaries. To this end, a two-dimensional NURBS mesh for a given surface is generated from the set of control points that define this surface (see the file **surfaceMesh.m**).

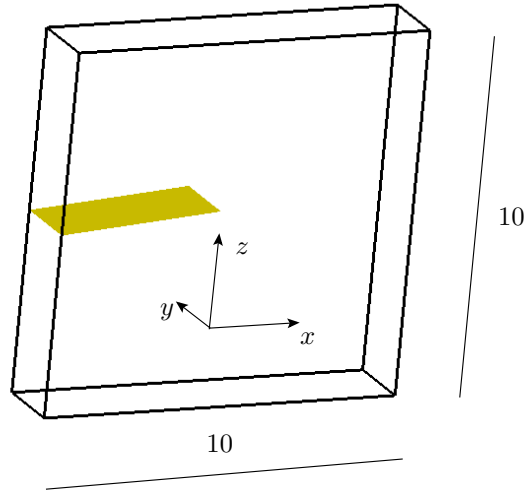


Figure 20: Three-dimensional mode I fracture problem: infinite plate with a center planar crack. The plate thickness is 2, the crack length is 5 and the crack width is 2. The crack is located in the mid-plane of the plate.

The problem is first solved using a linear B-spline basis. A mesh of  $9 \times 9 \times 1$  elements is used. The mesh, enriched nodes and comparison of the numerical deformed configuration against the exact profile are given in Fig. 21. Next, a mesh of  $7 \times 7 \times 2$  elements is used where, in the through-thickness direction, there are two linear elements ( $q = 1$ ) and for the two other directions, a quadratic basis ( $p = r = 2$ ) is used. The result is given in Fig. 22 and we note a good qualitative agreement between Figs. 21 and 22. We did not perform a SIF computation for this problem as 3D SIF computation is not yet implemented at present.

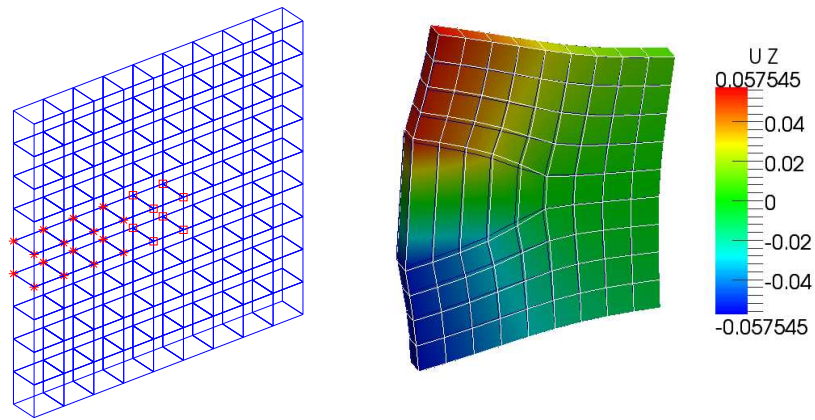


Figure 21: Three-dimensional mode I fracture problem: mesh of linear B-spline elements and enriched control points (left); numerical deformed configuration (magnification factor of 40) superimposed on the exact deformed configuration (right).

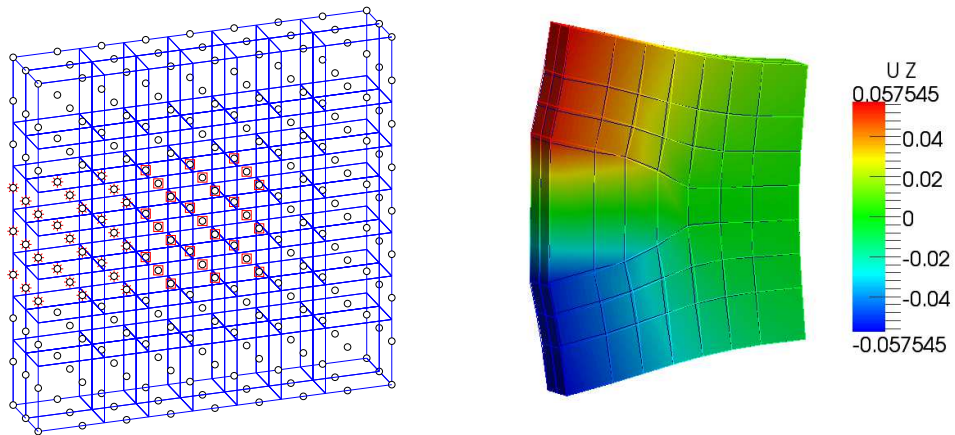


Figure 22: Three-dimensional mode I fracture problem: mesh of quadratic B-spline elements and enriched control points (left); numerical deformed configuration (magnification factor of 40) superimposed on the exact deformed configuration (right).

## 7. Conclusion

We presented a Matlab<sup>®</sup> implementation for one, two and three-dimensional isogeometric finite element analysis for solid and structural mechanics. This paper is addressed to students or researchers who wish to learn the concepts of IGA in a clear and concise manner and is especially suited to those with solid mechanics applications in mind. NURBS are used throughout, where the underlying construction of the basis functions is detailed along with associated refinement algorithms essential for numerical analysis. Differences with conventional FE implementations are made clear with the use of Matlab<sup>®</sup> source code to illustrate isogeometric FE concepts explicitly. In addition, the implementation of an isogeometric XFEM formulation for both two-dimensional and three-dimensional problems is described allowing for simple linear elastic fracture analysis to be performed directly from CAD data. The benefits of isogeometric analysis are made evident. Although not presented, the code supports multi-patch analysis in which compatibility between connecting patches is not required using a Nitsche’s method [90, 91, 108]. Geometrical nonlinearities for solid elements under the Total Lagrange framework are provided as well. In addition, PUM enrichment for holes and inclusions is provided along with implementation of the least squares method for imposing essential boundary conditions. Mass matrices are implemented for almost every elements and popular time integration schemes such as Newmark and central difference explicit are available so that transient analysis on CAD objects can be performed. The code is available for download from <https://sourceforge.net/projects/cmcodes/> for Windows, Linux and OS X machines.

The preliminary concepts and implementation details of isogeometric analysis have been described, but many challenges remain in the field. These include the creation of suitable volume discretizations from given CAD boundary representations, efficient integration schemes and suitable error estimators.

## Acknowledgements

The authors would like to acknowledge the partial financial support of the Framework Programme 7 Initial Training Network Funding under grant number 289361 “Integrating Numerical Simulation and Geometric Design Technology”. Stéphane Bordas also thanks partial funding for his time provided by 1) the EPSRC under grant EP/G042705/1 Increased Reliability for Industrially Relevant Automatic Crack Growth Simulation with the eXtended Finite Element Method and 2) the European Research Council Starting Independent Research Grant (ERC Stg grant agreement No. 279578) entitled “Towards real time multiscale simulation of cutting in non-linear materials with applications to surgical

simulation and computer guided surgery”. The first author would like to express his gratitude towards Professor L.J. Sluys at Delft University of Technology, The Netherlands for his support during the PhD period and the Framework Programme 7 Initial Training Network Funding. The authors acknowledge Dr. Robert Simpson whose comments have improved the paper.

## Appendix A. Knot vector conventions

The definition of an open knot vector as  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$  where the first and last knot values are repeated  $p + 1$  times is sometimes simplified to remove redundant information. It is found that if a knot vector is open, the first and last knot values play no role in the definition of the curve and can be removed entirely. Knot vectors are then defined as  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p-1}\}$  where the first and last knot vectors are repeated  $p$  times. This notation is used frequently through the CAGD community.

Furthermore, B-spline algorithms can be written entirely in terms of knot interval vectors [120], defined as

$$\Delta\Xi = \{\Delta\xi_1, \Delta\xi_2, \dots, \Delta\xi_{n+p-2}\} \quad \text{where} \quad \Delta\xi_i = \xi_{i+1} - \xi_i \quad (\text{A.1})$$

using the previous simplified knot vector notation. T-splines are based entirely on knot interval vector notation.

## Appendix B. Bézier extraction

Bézier extraction [17, 116] provides a tool to facilitate the incorporation of NURBS and even T-splines in to any FE codes. The idea is based on the fact that any B-splines basis can be written as a linear combination of Bernstein polynomials. Written for element  $e$ , the shape functions read

$$\mathbf{N}^e(\boldsymbol{\xi}) = \mathbf{C}^e \mathbf{B}(\tilde{\boldsymbol{\xi}}) \quad (\text{B.1})$$

where  $\mathbf{C}^e$  denotes the elemental Bézier extraction operator and  $\mathbf{B}$  are the Bernstein polynomials which are defined on the parent element  $\tilde{\Omega}$ . Index space, knot vectors are all embedded in the Bézier extrators which are computed in a pre-processing step. Therefore, existing FE solvers can use NURBS/T-splines straightforwardly. We refer to folder **bezier-extraction** for examples.

## References

- [1] M. Aigner, C. Heinrich, B. Jüttler, E. Pilgerstorfer, B. Simeon, A. Vuong, Swept volume parameterization for isogeometric analysis, in: E. Hancock, R. Martin, M.A. Sabin (Eds.), *Mathematics of Surfaces XIII*, volume 5654 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2009, pp. 19–44.
- [2] C. Anitescu, Y. Jia, Y.J. Zhang, T. Rabczuk, An isogeometric collocation method using superconvergent points, *Comput. Methods Appl. Mech. Eng.* 284 (2015) 1073 – 1097.
- [3] F. Auricchio, L. Beirão da Veiga, T.J.R. Hughes, A. Reali, G. Sangalli, Isogeometric collocation methods, *Math. Model. Methods Appl. Sci.* 20 (2010) 2075–2107.
- [4] F. Auricchio, L. Beirão da Veiga, T.J.R. Hughes, A. Reali, G. Sangalli, Isogeometric collocation for elastostatics and explicit dynamics, *Comput. Methods Appl. Mech. Eng.* 249–252 (2012) 2 – 14.
- [5] F. Auricchio, F. Calabro, T.J.R. Hughes, A. Reali, G. Sangalli, A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis, *Comput. Methods Appl. Mech. Eng.* 249–252 (2012) 15 – 27.
- [6] F. Auricchio, L. da Veiga Beirão, A. Buffa, C. Lovadina, A. Reali, G. Sangalli, A fully locking-free isogeometric approach for plane linear elasticity problems: A stream function formulation, *Comput. Methods Appl. Mech. Eng.* 197 (2007) 160–172.
- [7] Y. Bazilevs, I. Akkerman, Large eddy simulation of turbulent Taylor-Couette flow using isogeometric analysis and the residual-based variational multiscale method, *J. Comput. Phys.* 229 (2010) 3402–3414.
- [8] Y. Bazilevs, V. Calo, J. Cottrell, J. Evans, T. Hughes, S. Lipton, M. Scott, T. Sederberg, Isogeometric analysis using T-splines, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 229–263.
- [9] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, Y. Zhang, Isogeometric fluid-structure interaction: theory, algorithms, and computations, *Comput. Mech.* 43 (2008) 3–37.

- [10] Y. Bazilevs, J. Gohean, T. Hughes, R. Moser, Y. Zhang, Patient-specific isogeometric fluid-structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device, *Comput. Methods Appl. Mech. Eng.* 198 (2009) 3534–3550.
- [11] L. Beirão da Veiga, A. Buffa, C. Lovadina, M. Martinelli, G. Sangalli, An isogeometric method for the Reissner-Mindlin plate bending problem, *Comput. Methods Appl. Mech. Eng.* 209–212 (2012) 45–53.
- [12] L. Beirão da Veiga, D. Cho, L.F. Pavarino, S. Scacchi, Overlapping schwarz preconditioners for isogeometric collocation methods, *Comput. Methods Appl. Mech. Eng.* 278 (2014) 239 – 253.
- [13] D. Benson, Y. Bazilevs, M. Hsu, T. Hughes, Isogeometric shell analysis: The Reissner–Mindlin shell, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 276–289.
- [14] D. Benson, Y. Bazilevs, M. Hsu, T. Hughes, A large deformation, rotation-free, isogeometric shell, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1367–1378.
- [15] D.J. Benson, Y. Bazilevs, E. De Luycker, M. Hsu, M. Scott, T.J.R. Hughes, T. Belytschko, A generalized finite element formulation for arbitrary basis functions: From isogeometric analysis to XFEM, *Int. J. for Numer. Methods Eng.* 83 (2010) 765–785.
- [16] D.J. Benson, S. Hartmann, Y. Bazilevs, M. Hsu, T.J.R. Hughes, Blended isogeometric shells, *Comput. Methods Appl. Mech. Eng.* 255 (2013) 133 – 146.
- [17] M.J. Borden, M.A. Scott, J.A. Evans, T.J.R. Hughes, Isogeometric finite element data structures based on Bézier extraction of NURBS, *Int. J. for Numer. Methods Eng.* 87 (2011) 15–47.
- [18] M.J. Borden, C.V. Verhoosel, M.A. Scott, T.J.R. Hughes, C.M. Landis, A phase-field description of dynamic brittle fracture, *Comput. Methods Appl. Mech. Eng.* 217–220 (2012) 77 – 95.
- [19] P.B. Bornemann, F. Cirak, A subdivision-based implementation of the hierarchical b-spline finite element method, *Comput. Methods Appl. Mech. Eng.* 253 (2013) 584 – 598.
- [20] A. Buffa, D. Cho, G. Sangalli, Linear independence of the T-spline blending functions associated with some particular T-meshes, *Comput. Methods Appl. Mech. Eng.* 1437-1445 (2010) 199.



- [21] D. Burkhart, B. Hamann, G. Umlauf, Iso-geometric Finite Element Analysis Based on Catmull-Clark : subdivision Solids, *Comput. Graph. Forum* 29 (2010) 1575–1584.
- [22] R.P.R. Cardoso, J.M.A. Cesar de Sa, The enhanced assumed strain method for the isogeometric analysis of nearly incompressible deformation of solids, *Int. J. for Numer. Methods Eng.* 92 (2012) 56–78.
- [23] R.P.R. Cardoso, J.M.A. Cesar de Sa, Blending moving least squares techniques with NURBS basis functions for nonlinear isogeometric analysis, *Comput. Mech.* 53 (2014) 1327–1340.
- [24] J.F. Caseiro, R.A.F. Valente, A. Reali, J. Kiendl, F. Auricchio, R.J. Alves de Sousa, On the assumed natural strain method to alleviate locking in solid-shell NURBS-based finite elements, *Comput. Mech.* 53 (2014) 1341–1353.
- [25] E. Cervera, J. Trevelyan, Evolutionary structural optimisation based on boundary representation of NURBS. Part I: 2D algorithms, *Comput. Struct.* 83 (2005) 1902–1916.
- [26] J. Chessa, *Programming the finite element method with Matlab*, 2002. Northwestern University, <http://www.tam.northwestern.edu/jfc795/Matlab/>.
- [27] F. Cirak, M. Ortiz, P. Schröder, Subdivision surfaces: a new paradigm for thin-shell finite-element analysis, *Int. J. for Numer. Methods Eng.* 47 (2000) 2039–2072.
- [28] E. Cohen, T. Martin, R. Kirby, T. Lyche, R. Riesenfeld, Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 334–356.
- [29] N. Collier, D. Pardo, L. Dalcin, M. Paszynski, V.M. Calo, The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers, *Comput. Methods Appl. Mech. Eng.* 213–216 (2012) 353 – 361.
- [30] J. Cottrell, A. Reali, Y. Bazilevs, T. Hughes, Isogeometric analysis of structural vibrations, *Comput. Methods Appl. Mech. Eng.* 195 (2006) 5257–5296.
- [31] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA*, Wiley, 2009.
- [32] L. Dalcin, *PetIGA: A framework for high performance Isogeometric Analysis (using Petsc)*, <https://bitbucket.org/dalcinl/petiga>, 2011.

- [33] L. De Lorenzis, I. Temizer, P. Wriggers, G. Zavarise, A large deformation frictional contact formulation using NURBS-bases isogeometric analysis, *Int. J. for Numer. Methods Eng.* 87 (2011) 1278–1300.
- [34] E. De Luycker, D.J. Benson, T. Belytschko, Y. Bazilevs, M.C. Hsu, X-FEM in isogeometric analysis for linear fracture mechanics, *Int. J. for Numer. Methods Eng.* 87 (2011) 541–565.
- [35] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, Y. Feng, Polynomial splines over hierarchical T-meshes, *Graph. Model.* 70 (2008) 76–86.
- [36] R. Dimitri, L. De Lorenzis, P. Wriggers, G. Zavarise, NURBS- and T-spline-based isogeometric cohesive zone modeling of interface debonding, *Comput. Mech.* 54 (2014) 1–20.
- [37] T. Dokken, V. Skytt, Locally refined splines, in: *Proceedings of IV European Conference On Computational Mechanics. Solids, Structures and Coupled Problems in Engineering*, Paris, France.
- [38] M. Donatelli, C. Garoni, C. Manni, S. Serra-Capizzano, H. Speleers, Robust and optimal multi-iterative techniques for iga collocation linear systems, *Comput. Methods Appl. Mech. Eng.* 284 (2015) 1120 – 1146.
- [39] M.R. Dörfel, B. Jüttler, B. Simeon, Adaptive isogeometric analysis by local h-refinement with T-splines, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 264–275.
- [40] R. Echter, M. Bischoff, Numerical efficiency, locking and unlocking of NURBS finite elements, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 374 – 382.
- [41] R. Echter, B. Oesterle, M. Bischoff, A hierarchic family of isogeometric shell finite elements, *Comput. Methods Appl. Mech. Eng.* 254 (2013) 170 – 180.
- [42] T. Elguedj, Y. Bazilevs, V. Calo, T. Hughes, B-bar and f-bar projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements, *Comput. Methods Appl. Mech. Eng.* 197 (2008) 2732–2762.
- [43] T. Elguedj, Y. Bazilevs, V.M. Calo, T.J.R. Hughes, F-bar projection method for finite deformation elasticity and plasticity using NURBS based isogeometric analysis, *Int. J. Material Form.* 1 (2008) 1091–1094.

- [44] T. Elguedj, A. Gravouil, A. Combescure, Appropriate extended functions for X-FEM simulation of plastic fracture mechanics, *Comput. Methods Appl. Mech. Eng.* 195 (2006) 501 – 515.
- [45] T. Elguedj, J. Réthoré, A. Buteri, Isogeometric analysis for strain field measurements, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 40–56.
- [46] J. Escobar, J. Casen, E. Rodriguez, R. Montenegro, A new approach to solid modeling with trivariate T-splines based on mesh optimization, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 3210–3222.
- [47] J.A. Evans, Y. Bazilevs, I. Babuška, T. Hughes, n-Widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method, *Comput. Methods Appl. Mech. Eng.* 198 (2009) 1726–1741.
- [48] C. de Falco, A. Reali, R. Vázquez., GeoPDEs: a research tool for Isogeometric Analysis of PDEs, *Adv. Eng. Softw.* 42 (2011) 1020–1034.
- [49] C.A. Felippa, *Advanced Finite Element Methods (lecture notes)*, <http://www.colorado.edu/engineering/CAS/courses/2001>.
- [50] P. Fischer, M. Klassen, J. Mergheim, P. Steinmann, R. Müller, Isogeometric analysis of 2D gradient elasticity, *Comput. Mech.* 47 (2010) 325–334.
- [51] S.S. Ghorashi, N. Valizadeh, S. Mohammadi, Extended isogeometric analysis for simulation of stationary and propagating cracks, *Int. J. for Numer. Methods Eng.* 89 (2012) 1069–1101.
- [52] H. Gómez, V.M. Calo, Y. Bazilevs, T. Hughes, Isogeometric analysis of the Cahn-Hilliard phase-field model, *Comput. Methods Appl. Mech. Eng.* 197 (2008) 4333–4352.
- [53] H. Gomez, T. Hughes, X. Nogueira, V.M. Calo, Isogeometric analysis of the isothermal Navier-Stokes-Korteweg equations, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 1828–1840.
- [54] D. González, E. Cueto, M. Doblaré, Higher-order natural element methods: Towards an isogeometric meshless method, *Int. J. for Numer. Methods Eng.* 74 (2008) 1928–1954.
- [55] D. Großmann, B. Jüttler, H. Schlusnus, J. Barner, A. Vuong, Isogeometric simulation of turbine blades for aircraft engines, *Comput. Aided Geom. Des.* 29 (2012) 519 – 531.

- [56] A. Henderson, ParaView Guide, A Parallel Visualization Application, Kitware Inc., 2007.
- [57] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput. Methods Appl. Mech. Eng.* 194 (2005) 4135–4195.
- [58] T. Hughes, A. Reali, G. Sangalli, Efficient quadrature for NURBS-based isogeometric analysis, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 301–313.
- [59] T.J.R. Hughes, A. Reali, G. Sangalli, Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS, *Comput. Methods Appl. Mech. Eng.* 197 (2008) 4104 – 4124.
- [60] F. Irzal, J.J.C. Remmers, C.V. Verhoosel, R. de Borst, An isogeometric analysis Bézier interface element for mechanical and poromechanical fracture problems, *Int. J. for Numer. Methods Eng.* 97 (2014) 608–628.
- [61] N. Jaxon, X. Qian, Isogeometric analysis on triangulations, *Comput. Des.* 46 (2014) 45 – 57.
- [62] P. Kagan, A. Fischer, Integrated mechanically based CAE system using B-Spline finite elements, *Comput. Des.* 32 (2000) 539 – 552.
- [63] P. Kagan, A. Fischer, P.Z. Bar-Yoseph, New B-Spline Finite Element approach for geometrical design and mechanical analysis, *Int. J. for Numer. Methods Eng.* 41 (1998) 435–458.
- [64] A. Karatarakis, P. Karakitsios, M. Papadrakakis, GPU accelerated computation of the isogeometric analysis stiffness matrix, *Comput. Methods Appl. Mech. Eng.* 269 (2014) 334 – 355.
- [65] J. Kiendl, F. Auricchio, L. Beirão da Veiga, C. Lovadina, A. Reali, Isogeometric collocation methods for the reissner-mindlin plate problem, *Comput. Methods Appl. Mech. Eng.* 284 (2015) 489 – 507.
- [66] J. Kiendl, Y. Bazilevs, M. Hsu, R. Wüchner, K. Bletzinger, The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 2403–2416.
- [67] J. Kiendl, K. Bletzinger, J. Linhard, R. Wüchner, Isogeometric shell analysis with Kirchhoff-Love elements, *Comput. Methods Appl. Mech. Eng.* 198 (2009) 3902–3914.

- [68] H. Kim, Y. Seo, S. Youn, Isogeometric analysis with trimming technique for problems of arbitrary complex topology, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 2796–2812.
- [69] H. Kim, S. Youn, Spline-based meshfree method, *Int. J. for Numer. Methods Eng.* 92 (2012) 802–834.
- [70] S.K. Kleiss, B. Jüttler, W. Zulehner, Enhancing isogeometric analysis by a finite element-based local refinement strategy, *Comput. Methods Appl. Mech. Eng.* 213–216 (2012) 168 – 182.
- [71] G. Kuru, C.V. Verhoosel, K.G. van der Zee, E.H. van Brummelen, Goal-adaptive isogeometric analysis with hierarchical splines, *Comput. Methods Appl. Mech. Eng.* 270 (2014) 270 – 292.
- [72] P. Lan, A.A. Shabana, Integration of B-spline geometry and ANCF finite element analysis, *Nonlinear Dyn.* 61 (2010) 193–206.
- [73] G. Legrain, A NURBS enhanced extended finite element approach for unfitted CAD analysis, *Comput. Mech.* 52 (2013) 913–929.
- [74] S. Lipton, J.A. Evans, Y. Bazilevs, T. Elguedj, T.J.R. Hughes, Robustness of isogeometric structural discretizations under severe mesh distortion, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 357 – 373.
- [75] L.D. Lorenzis, J.A. Evans, T.J.R. Hughes, A. Reali, Isogeometric collocation: Neumann boundary conditions and contact, *Comput. Methods Appl. Mech. Eng.* 284 (2015) 21 – 54.
- [76] J. Lu, Circular element: Isogeometric elements of smooth boundary, *Comput. Methods Appl. Mech. Eng.* 198 (2009) 2391–2402.
- [77] J. Lu, Isogeometric contact analysis: Geometric basis and formulation for frictionless contact, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 726–741.
- [78] J. Lu, X. Zhou, Cylindrical element: Isogeometric model of continuum rod, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 233–241.
- [79] N.D. Manh, A. Evgrafov, A.R. Gersborg, J. Gravesen, Isogeometric shape optimization of vibrating membranes, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1343–1353.

- [80] T. Martin, E. Cohen, R.M. Kirby, Volumetric parameterization and trivariate B-spline fitting using harmonic functions, *Comput. Aided Geom. Des.* 26 (2009) 648–664.
- [81] A. Masud, R. Kannan, B-splines and NURBS based finite element methods for Kohn-Sham equations, *Comput. Methods Appl. Mech. Eng.* 241-244 (2012) 112 – 127.
- [82] M.E. Matzen, T. Cichosz, M. Bischoff, A point to segment contact formulation for isogeometric, NURBS based finite elements, *Comput. Methods Appl. Mech. Eng.* 255 (2013) 27 – 39.
- [83] C. Miehe, F. Welschinger, M. Hofacker, Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field FE implementations, *Int. J. for Numer. Methods Eng.* 83 (2010) 1273–1311.
- [84] N. Moës, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, *Int. J. for Numer. Methods Eng.* 46 (1999) 131–150.
- [85] N. Moës, C. Stolz, P. Bernard, N. Chevaugeon, A level set based model for damage growth: The thick level set approach, *Int. J. for Numer. Methods Eng.* 86 (2011) 358–380.
- [86] M. Mounnassi, S. Belouettar, E. Béchet, S.P.A. Bordas, D. Quoirin, M. Potier-Ferry, Finite element analysis on implicitly defined domains: An accurate representation based on arbitrary parametric surfaces, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 774 – 796.
- [87] A.A. Nada, Use of B-spline surface to model large-deformation continuum plates: procedure and applications, *Nonlinear Dyn.* 72 (2013) 243–263.
- [88] S. Natarajan, D.R. Mahapatra, S.P.A. Bordas, Integrating strong and weak discontinuities without integration subcells and example applications in an XFEM/GFEM framework, *Int. J. for Numer. Methods Eng.* 83 (2010) 269–294.
- [89] V.P. Nguyen, P. Kerfriden, S. Bordas, Two- and three-dimensional isogeometric cohesive elements for composite delamination analysis, *Compos. Part B: Eng.* 60 (2014) 193–212.
- [90] V.P. Nguyen, P. Kerfriden, M. Brino, S.P.A. Bordas, E. Bonisoli, Nitsche’s method for two and three dimensional nurbs patch coupling, *Comput. Mech.* 53 (2014) 1163–1182.

- [91] V.P. Nguyen, P. Kerfriden, S. Claus, S.P.A. Bordas, A Nitsche’s method for mixed dimensional analysis: conforming and non-conforming solid-structure coupling, 2013. <https://hal.archives-ouvertes.fr/hal-00857372>.
- [92] V.P. Nguyen, H. Nguyen-Xuan, High-order B-splines based finite elements for delamination analysis of laminated composites, *Compos. Struct.* 102 (2013) 261–275.
- [93] V.P. Nguyen, T. Rabczuk, S. Bordas, M. Duflot, Meshless methods: A review and computer implementation aspects, *Math. Comput. Simul.* 79 (2008) 763–813.
- [94] N. Nguyen-Thanh, J. Kiendl, H. Nguyen-Xuan, R. Wüchner, K. Bletzinger, Y. Bazilevs, T. Rabczuk, Rotation free isogeometric thin shell analysis using PHT-splines, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 3410–3424.
- [95] N. Nguyen-Thanh, H. Nguyen-Xuan, S. Bordas, T. Rabczuk, Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1892–1908.
- [96] P.N. Nielsen, A.R. Gersborg, J. Gravesen, N.L. Pedersen, Discretizations in isogeometric analysis of Navier-Stokes flow, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 3242–3253.
- [97] M.J. Peake, J. Trevelyan, G. Coates, Extended isogeometric boundary element method (XIBEM) for two-dimensional Helmholtz problems, *Comput. Methods Appl. Mech. Eng.* 259 (2013) 93 – 102.
- [98] L.A. Piegl, W. Tiller, *The NURBS Book*, Springer, 1996.
- [99] C.G. Provatidis, Two-dimensional elastostatic analysis using Coons-Gordon interpolation, *Meccanica* 47 (2012) 951–967.
- [100] X. Qian, Full analytical sensitivities in NURBS based isogeometric shape optimization, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 2059–2071.
- [101] X. Qian, O. Sigmund, Isogeometric shape optimization of photonic crystals via Coons patches, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 2237–2255.

- [102] E. Rank, M. Ruess, S. Kollmannsberger, D. Schillinger, A. Düster, Geometric modeling, isogeometric analysis and the finite cell method, *Comput. Methods Appl. Mech. Eng.* 249–252 (2012) 104 – 115.
- [103] A. Ratnani, Pigasus, Isogeometric Analysis simulations in Python, 2012. <https://hal.inria.fr/hal-00769225>.
- [104] A. Ratnani, E. Sonnendrücker, Isogeometric analysis in reduced magnetohydrodynamics, *Comput. Sci. & Discov.* 5 (2012) 1–21.
- [105] J.N. Reddy, *An Introduction to the Finite Element Method*, McGraw-Hill Education, 3rd edition, 2005.
- [106] D.F. Rogers, *An Introduction to NURBS with Historical Perspective*, Academic Press, 2001.
- [107] A. Rosolen, M. Arroyo, Blending isogeometric analysis and local maximum entropy meshfree approximants, *Comput. Methods Appl. Mech. Eng.* 264 (2013) 95 – 107.
- [108] M. Ruess, D. Schillinger, A.I. Özcan, E. Rank”, Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries, *Comput. Methods Appl. Mech. Eng.* 269 (2014) 46–71.
- [109] D. Rypl, B. Patzák, From the finite element analysis to the isogeometric analysis in an object oriented computing environment, *Adv. Eng. Softw.* 44 (2012) 116–125.
- [110] G.G. Sanborn, A.A. Shabana, On the integration of computer aided design and analysis using the finite element absolute nodal coordinate formulation, *Multibody Syst. Dyn.* 22 (2009) 181–197.
- [111] R.A.K. Sanches, P.B. Bornemann, F. Cirak, Immersed B-spline (i-spline) finite element method for geometrically complex domains, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1432 – 1445.
- [112] D. Schillinger, L. Dedé, M.A. Scott, J.A. Evans, M.J. Borden, E. Rank, T.J.R. Hughes, An Isogeometric Design-through-analysis Methodology based on Adaptive Hierarchical Refinement of NURBS, Immersed Boundary Methods, and T-spline CAD Surfaces, *Comput. Methods Appl. Mech. Eng.* 242-252 (2012) 116–150.
- [113] D. Schillinger, J.A. Evans, A. Reali, M.A. Scott, T.J.R. Hughes, Isogeometric collocation: Cost comparison with galerkin methods and extension to adaptive hierarchical NURBS discretizations, *Comput. Methods Appl. Mech. Eng.* 267 (2013) 170 – 232.



- [114] R. Schmidt, J. Kiendl, K. Bletzinger, R. Wüchner, Realization of an integrated structural design process: analysis-suitable geometric modelling and isogeometric analysis, *Comput. Vis. Sci.* 13 (2010) 315–330.
- [115] R. Schmidt, R. Wüchner, K.U. Bletzinger, Isogeometric analysis of trimmed NURBS geometries, *Comput. Methods Appl. Mech. Eng.* 241–244 (2012) 93 – 111.
- [116] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sederberg, T.J.R. Hughes, Isogeometric finite element data structures based on Bézier extraction of T-splines, *Int. J. for Numer. Methods Eng.* 88 (2011) 126–156.
- [117] M.A. Scott, X. Li, T.W. Sederberg, T.J.R. Hughes, Local refinement of analysis suitable T-splines, *Comput. Methods Appl. Mech. Eng.* 213–216 (2012) 206–222.
- [118] M.A. Scott, R. Simpson, J. Evans, S. Lipton, S.P.A. Bordas, T.J.R. Hughes, T.W. Sederberg, Isogeometric boundary element analysis using unstructured T-splines, *Comput. Methods Appl. Mech. Eng.* 254 (2013) 197 – 221.
- [119] M.A. Scott, D.C. Thomas, E.J. Evans, Isogeometric spline forests, *Comput. Methods Appl. Mech. Eng.* 269 (2014) 222 – 264.
- [120] T.W. Sederberg, Computer Aided Geometric Design: Course Notes, <http://www.tsplines.com/educationportal>.ht 2010.
- [121] T.W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, *ACM Transactions on Graph.* 22 (2003) 477–484.
- [122] R. Sevilla, S. Fernández-Méndez, A. Huerta, NURBS-enhanced finite element method (NEFEM), *Int. J. for Numer. Methods Eng.* 76 (2008) 56–83.
- [123] R. Sevilla, S. Fernández-Méndez, A. Huerta, 3D NURBS-enhanced finite element method (NEFEM), *Int. J. for Numer. Methods Eng.* 88 (2011) 103–125.
- [124] A. Shaw, D. Roy, NURBS-based parametric mesh-free methods, *Comput. Methods Appl. Mech. Eng.* 197 (2008) 1541 – 1567.

- [125] J. Shi, D. Chopp, J. Lua, N. Sukumar, T. Belytschko, Abaqus implementation of extended finite element method using a level set representation of three-dimensional fatigue crack growth and life predictions, *Eng. Fract. Mech.* 77 (2010) 2840–2863.
- [126] R. Simpson, S. Bordas, J. Trevelyan, T. Rabczuk, A two-dimensional isogeometric boundary element method for elastostatic analysis, *Comput. Methods Appl. Mech. Eng.* 209–212 (2012) 87–100.
- [127] R.N. Simpson, S.P.A. Bordas, H. Lian, J. Trevelyan, An isogeometric boundary element method for elastostatic analysis: 2D implementation aspects, *Comput. & Struct.* 118 (2013) 2 – 12.
- [128] H. Speleers, C. Manni, F. Pelosi, M.L. Sampoli, Isogeometric analysis with Powell-Sabin splines for advection-diffusion-reaction problems, *Comput. Methods Appl. Mech. Eng.* 221–222 (2012) 132 – 148.
- [129] M. Steffen, P.C. Wallstedt, J.E. Guilkey, Examination and analysis of implementation choices within the material point method (MPM), *Comput. Model. Eng. Sci.* 31 (2008) 107–127.
- [130] N. Sukumar, D.L. Chopp, N. Moës, T. Belytschko, Modelling holes and inclusions by level sets in the extended finite element method, *Comput. Methods Appl. Mech. Eng.* 190 (2000) 6183–6200.
- [131] H.P. Tada, P.C. Paris, G.R. Irwin, *The Stress Analysis of Cracks Handbook*, Del Research Corporation, St. Louis, MO, 1985.
- [132] T. Takacs, B. Jüttler, Existence of stiffness matrix integrals for singularly parameterized domains in isogeometric analysis, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 3568–3582.
- [133] A. Tambat, G. Subbarayan, Isogeometric enriched field approximations, *Comput. Methods Appl. Mech. Eng.* 245–246 (2012) 1 – 21.
- [134] R.L. Taylor, A finite element analysis program, programmer manual, <http://www.ce.berkeley.edu/projects/feap/>, 2001.
- [135] R.L. Taylor, Isogeometric analysis of nearly incompressible solids, *Int. J. for Numer. Methods Eng.* 87 (2011) 273–288.

- [136] Í. Temizer, Multiscale thermomechanical contact: Computational homogenization with isogeometric analysis, *Int. J. for Numer. Methods Eng.* 97 (2014) 582–607.
- [137] I. Temizer, P. Wriggers, T. Hughes, Contact treatment in isogeometric analysis with NURBS, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1100–1112.
- [138] I. Temizer, P. Wriggers, T. Hughes, Three-Dimensional Mortar-Based frictional contact treatment in isogeometric analysis with NURBS, *Comput. Methods Appl. Mech. Eng.* 209–212 (2012) 115–128.
- [139] C.H. Thai, H. Nguyen-Xuan, N. Nguyen-Thanh, T. Le, T. Nguyen-Thoi, T. Rabczuk, Static, free vibration, and buckling analysis of laminated composite Reissner-Mindlin plates using NURBS-based isogeometric approach, *Int. J. for Numer. Methods Eng.* 91 (2012) 571–603.
- [140] T. Uhm, S. Youn, T-spline finite element method for the analysis of shell structures, *Int. J. for Numer. Methods Eng.* 80 (2009) 507–536.
- [141] C.V. Verhoosel, M.A. Scott, R. de Borst, T.J.R. Hughes, An isogeometric approach to cohesive zone modeling, *Int. J. for Numer. Methods Eng.* 87 (2011) 336–360.
- [142] C.V. Verhoosel, M.A. Scott, T.J.R. Hughes, R. de Borst, An isogeometric analysis approach to gradient damage models, *Int. J. for Numer. Methods Eng.* 86 (2011) 115–134.
- [143] A. Vuong, C. Giannelli, B. Jüttler, B. Simeon, A hierarchical approach to adaptive local refinement in isogeometric analysis, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 3554–3567.
- [144] A. Vuong, C. Heinrich, B. Simeon, ISOGAT: a 2D tutorial MATLAB code for Isogeometric Analysis, *Comput. Aided Geom. Des.* 27 (2010) 644–655.
- [145] W.A. Wall, M.A. Frenzel, C. Cyron, Isogeometric structural shape optimization, *Comput. Methods Appl. Mech. Eng.* 197 (2008) 2976–2988.
- [146] D. Wang, W. Liu, H. Zhang, Novel higher order mass matrices for isogeometric structural vibration analysis, *Comput. Methods Appl. Mech. Eng.* 260 (2013) 92 – 108.
- [147] H. Wang, Y. He, X. Li, X. Gu, H. Qin, Polycube splines, *Comput. Des.* 40 (2008) 721–733.

- [148] A. Wawrzinek, K. Hildebrandt, K. Polthier, Koiter's thin shells on Catmull-Clark limit surfaces, in: Proceedings of the Vision, Modeling, and Visualization Workshop, Berlin, Germany.
- [149] G. Xu, B. Mourrain, R. Duvigneau, Parameterization of computational domain in isogeometric analysis: Methods and comparison, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 2021–2031.
- [150] H. Yamashita, H. Sugiyama, Numerical convergence of finite element solutions of nonrational B-spline element and absolute nodal coordinate formulation, *Nonlinear Dyn.* 67 (2012) 177–189.
- [151] Y. Zhang, Y. Bazilevs, S. Goswami, C. Bajaj, T.J.R. Hughes, Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, *Comput. Methods Appl. Mech. Eng.* 196 (2007) 2943–2959.
- [152] Y. Zhang, W. Wang, T.J.R. Hughes, Conformal solid T-spline construction from boundary T-spline representations, *Comput. Mech.* 51 (2012) 1–9.
- [153] Y. Zhang, W. Wang, T.J.R. Hughes, Solid T-spline construction from boundary representations for genus-zero geometry, *Comput. Methods Appl. Mech. Eng.* 249–252 (2012) 185 – 197.
- [154] X. Zhou, J. Lu, NURBS-based Galerkin method and application to skeletal muscle modeling, in: Proceedings of the 2005 ACM symposium on Solid and physical modeling, SPM '05, ACM, New York, NY, USA, 2005, pp. 71–78.