Fortschritt-Berichte VDI

Reihe 18

Mechanik/
Bruchmechanik

Nr. 355

M.Sc. Simeon Hubrich,
Hamburg

The hierarchical finite cell method for nonlinear problems: Moment fitting quadratures, basis function removel, and remeshing

The hierarchical finite cell method for nonlinear problems: Moment fitting quadratures, basis function removal, and remeshing

Vom Promotionsausschuss der
Technischen Universität Hamburg

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von
Simeon Hubrich, M.Sc.

aus
Bremen

2021

Vorsitzender des Prüfungsausschusses
Prof. Dr.-Ing. Thomas Rung

Gutachter
1. Gutachter: Prof. Dr.-Ing. habil. Alexander Düster
2. Gutachter: Prof. Dr. rer. nat. Ernst Rank

Tag der mündlichen Prüfung: 16. Februar 2021

Fortschritt-Berichte VDI

Reihe 18

Mechanik/
Bruchmechanik

M.Sc. Simeon Hubrich,
Hamburg

Nr. 355

The hierarchical finite cell method for nonlinear problems: Moment fitting quadratures, basis function removel, and remeshing

VDI verlag

Hubrich, Simeon
**The hierarchical finite cell method for nonlinear problems:
Moment fitting quadratures, basis function removel, and remeshing**
Fortschr.-Ber. VDI Reihe 18 Nr. 355. Düsseldorf: VDI Verlag 2021.
176 Seiten, 111 Bilder, 7 Tabellen.
ISBN 978-3-18-335518-1, ISSN 0178-9457,
€ 62,00/VDI-Mitgliederpreis € 55,80.

Keywords: Finite cell method – Fictitious domain approach – High-order finite element methods – Numerical integration – Moment fitting quadratures – Basis function removal – Remeshing – Data transfer – Nonlinear problems – Finite strain problems

In this thesis, several approaches are discussed in order to further enhance the performance of the finite cell method (FCM). Thereby, novel moment fitting quadrature schemes are introduced that allow to reduce the effort of the numerical integration process significantly. Further, a basis function removal scheme is proposed to improve the conditioning behavior of the resulting equation system. Finally, an innovative remeshing strategy is presented that overcomes the problem of severely distorted elements for simulations with large deformations.

Bibliographische Information der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Daten sind im Internet unter www.dnb.de abrufbar.

Bibliographic information published by the Deutsche Bibliothek
(German National Library)
The Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliographie (German National Bibliography); detailed bibliographic data is available via Internet at www.dnb.de.

Arbeitsgruppe Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik

Als Manuskript gedruckt. Printed in Germany.

ISSN 0178-9457

ISBN 978-3-18-335518-1

Acknowledgements

The present thesis is the result of my research work during my employment at the Institute for Ship Structural Design and Analysis (M-10) at Hamburg University of Technology (TUHH) in the period from January 2015 to April 2020. The work was funded by the Deutsche Forschungsgemeinschaft in the Priority Programme 1748 (DFG SPP 1748), in which the main objective was the development of modern non-standard discretization methods.

At this point, I would like to take the opportunity to thank everyone who contributed to my work and supported me during this time.

First of all, I would like to express my deep and sincere gratitude to my doctoral supervisor Prof. Dr.-Ing. habil. Alexander Düster. Dear Prof. Düster, many thanks for all the fruitful discussions, great suggestions, and your continuous support, which contributed considerably to the success of this work. While working with you, I was able to benefit a lot from your broad expertise and your long-time experience. Thank you very much!

Next, I would also like to thank Prof. Dr. rer. nat. Ernst Rank for acting as the second supervisor of my thesis – and Prof. Dr.-Ing. Thomas Rung for chairing my examination.

Further, I would like to thank all my colleagues at M-10 and of the SPP 1748 for the very successful and pleasant collaboration in many projects.

Furthermore, a big thank you goes to my family and friends for their support and for all the pleasant diversions from work.

Finally, I would like to express my deepest thanks to my love Anna. Dear Anna, over the years, we shared many happy adventures and have successfully overcome several challenges as a team. Together, we have a beautiful daughter. Thank you for your endless love, encouragement, and support during all these years. Thank you for everything, I love you so much!

To my love Anna
and our beautiful daughter Clari,
I love you!

Contents

Abstract

Over the last decade, nonstandard discretization methods based on the fictitious domain approach have gained increased interest. In these methods, the physical domain is embedded into a fictitious one – resulting in an extended domain of a simple shape. Consequently, structured meshes or Cartesian grids can be employed for the spatial discretization, thus simplifying the mesh generation process significantly. Due to this reason, such methods are a powerful tool for the numerical analysis of complex structures such as foam-like materials. A well-known example for these methods is the *finite cell method* (FCM), which combines the fictitious domain approach with high order finite elements. In the FCM, these elements are denoted as finite cells – thus giving the method its name – in order to distinguish them from boundary-conforming finite elements. However, the simplification in the mesh generation is accompanied by several numerical difficulties, induced by cut finite cells, reducing the efficiency and robustness of the FCM. In this thesis, we focus on the following issues in order to further improve the FCM.

The first topic is related to the **numerical integration of finite cells**. In general adaptive Gaussian quadrature schemes are used – commonly resulting in a large number of integration points, which renders the numerical integration computationally expensive. To overcome this problem, we propose novel quadrature methods based on **moment fitting**. Thereby, a promising approach is introduced that circumvents the necessity of having to solve an equation system. We show that this moment fitting method results in efficient and accurate quadrature rules for linear problems of the FCM, reducing the effort during the numerical integration process significantly. Moreover, in order to improve the performance for nonlinear applications, an adaptive moment fitting approach is presented.

The second topic addresses the **ill-conditioning of the global system**. To improve the conditioning behavior, we propose a new **basis function removal** approach applied to the hierarchic shape functions of the FCM. In this approach, shape functions with a small contribution to the diagonal entries of the global system matrix are removed from the ansatz. To this end, a global criterion based on the discrete gradient operator is introduced to estimate the contribution. Moreover, by maintaining the nodal modes of the hierarchic shape functions, the modified basis preserves the representation of the rigid body modes. Several examples show that the basis functions removal improves the conditioning behavior and, thus, the performance of the FCM significantly.

The last topic is related to the issue of **severely distorted finite cells for applications in finite strain**. To overcome this problem, we introduce a novel **remeshing strategy** that is based on a multiplicative decomposition of the deformation gradient. The essential idea of this strategy is to create a new mesh whenever the analysis fails due to severe distortions of the computational mesh – and then to continue the simulation. Further, a local radial basis function interpolation scheme for the implementation of the data transfer is presented. Considering problems of different complexity, we show that the remeshing strategy allows to improve the robustness behavior of the FCM considerably, especially in combination with the presented basis function removal.

1 Introduction

1.1 Motivation

Novel materials – like composites or metal foams – as well as modern manufacturing processes, such as additive manufacturing, allow to produce structures of high quality. Since these structures provide a high lightweight potential, they have gained increased interest in the automotive industry and in aerospace and maritime applications. Another interesting application field is the design of medical implants, for instance. Here, the additive manufacturing process allows to produce patient-specific implants, e.g. bone or tooth implants that are composed of very complex shapes or geometries. There is also increased interest in these materials and manufacturing processes in other industrial fields, such as the sports industry, for example. Due to the increased demand for more sophisticated products and quality standards as well as low production costs, it has become necessary to apply numerical simulation tools in the development phase. Here, the application of the *finite element method* (FEM) is one of the most widely used standards in order to analyze and optimize such kind of structures in the early stages of the development cycle. Therefore, many commercial FEM software packages exist, such as Abaqus [1], ANSYS [2], LS-DYNA [3], or MSC Software [4], just to name a few. However, since the discretization approach of the standard FEM is based on boundary-conforming elements, the mesh generation of structures with complex geometries often turns out to be difficult or, in some cases, even practically impossible. Cottrell et al. [5] and Hughes et al. [6] mentioned that the effort in the mesh generation makes up the main part of the analysis. Here, the time of the meshing step is estimated at 80% regarding the overall analysis time.

In order to overcome the bottleneck in the mesh generation, nonstandard discretization methods based on the *fictitious domain approach* can be applied instead of the standard FEM. To the best knowledge of the author, the fictitious domain approach was first introduced by Saulev [7, 8]. Further contributions that are based on a similar approach can be found in Neittaanmäki and Tiba [9], Peskin [10], Del Pino and Pironneau [11], Mittal and Iaccarino [12], Glowinski and Kuznetsov [13], Ramière et al. [14, 15], Burman and Hansbo [16, 17], and Elfverson et al. [18]. Several different terms have been found for these related approaches – such as *embedded domain method*, *immersed boundary method*, *CutFEM*, or *CutIGA*.

In the context of this thesis, we apply the *finite cell method* (FCM) which was introduced in [19–22]. The FCM combines the fictitious domain approach with finite elements using shape functions of higher order. Since its introduction, the FCM has been successfully applied in various fields, e.g. applications to elastic and plastic problems in small and large strain [19, 20, 23–33], homogenization of heterogeneous and cellular materials as well as foams [34–40], topology optimization [41, 42], problems including material interfaces [43–47], contact problems [40, 48–54], multi-physic problems [55–62], fracture simulation [63, 64], or simulation of wave propagation [65–68]. In order to demonstrate the efficiency of the FCM, an illustrative example is given in Fig. 1.1 where a foam-like structure is analyzed. Thanks to the fictitious domain approach, the mesh generation can be carried out efficiently by employing a Cartesian grid, see Fig. 1.1b. Consequently, the effort in the mesh generation is reduced significantly. Further, since the elements do not conform to the boundary of the body, we denote them as finite cells – giving the method its name – in order to distinguish them from boundary-conforming finite elements. Furthermore, the high-order shape functions ensure high convergence rates, provided that the solution of the problem is sufficiently smooth [21].

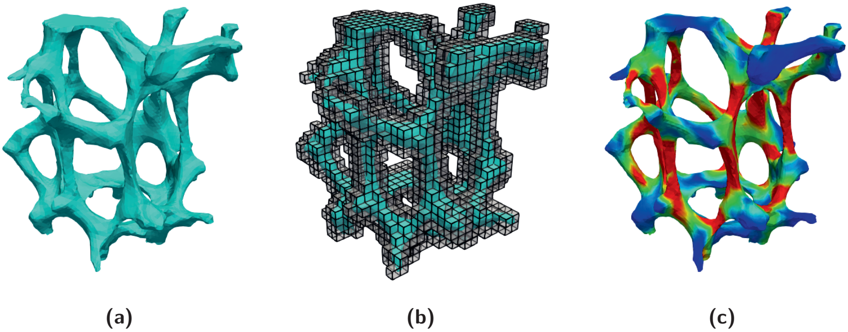


Figure 1.1: FCM analysis of a foam-like structure. **(a)** Geometry. **(b)** Spatial discretization. **(c)** Results of the simulation.

The simplification in the mesh generation of the FCM due to the fictitious domain approach, however, is accompanied by several numerical difficulties. Some of these issues are listed below.

- A major bottleneck of the FCM is related to the **numerical integration of finite cells**. Integrals of cells that are cut by the boundary of the domain are characterized by discontinuous integrands, which is why standard Gaussian quadrature does not perform well anymore. Thus, it is common to employ adaptive Gaussian quadratures that are generally based on subcell meshes in order to resolve for the domain of interest [19–21]. Then, standard Gaussian quadrature rules are applied on each subcell. Although these adaptive integration methods are robust and can be carried out in an automatic manner, they commonly result in many subcells and, thus, in a large number of integration points. This in turn renders the numerical calculation of the integrals of cut cells computationally expensive.

- Another well-known issue when applying the FCM is related to the **ill-conditioning of the global system matrix**. Often, the ill-conditioning behavior is a result of badly cut finite cells which possess a small support or result in nearly linearly dependent shape functions. In order to overcome this problem, several approaches have been proposed, e.g. applying a fictitious material [19–21] or preconditioning techniques [69]. Although these approaches help to improve the solvability of the resulting global equation system, the demand for novel approaches or the extension of existing methods in order to improve the robustness of the FCM are still a matter of ongoing research – especially in the field of nonlinear problems.

- Further, the FCM suffers from **severely distorted finite cells for applications in finite strain**. When considering this type of problems, the fictitious domain usually undergoes much larger deformations than the physical one. Due to this reason, badly cut cells get distorted severely during the analysis [25, 26]. As a result of this behavior, the simulation is usually aborted because the Newton-Raphson procedure does not converge anymore. In most of the case the analysis is aborted due to self-penetration of cut finite cells.

- Another issue is to be seen in applications where the solution of the problem is not smooth enough, for instance due to the occurrence of discontinuities or singularities. A representative of this kind of problem are structures composed of heterogeneous materials. Here, the material interface in cut finite cells represents a weak discontinuity that can not be approximated accurately enough by employing a basis containing smooth shape functions. As a result, increasing the order of the shape functions does not improve the convergence behavior. Consequently, appropriate **enrichment and refinement strategies** are required for the analysis of such kind of problems. To this end, several strategies have been introduced within the context of the FCM [43, 47, 59, 70].

- A further problem of the FCM is related to the **treatment of boundary conditions**. In [20] Düster et al. presented an approach for the incorporation of inhomogeneous Neumann boundary conditions considering three-dimensional applications. Here, the Neumann boundary is parameterized using a triangulated surface mesh. In doing so, the load vector of the cells are computed by applying a Gaussian quadrature on each triangle. Moreover, the Dirichlet boundary conditions are taken into account in a weak sense. To this end, different approaches may be applied – such as the Nitsche method or the penalty method [71–73].

1.2 Goal and scope of this thesis

The goal of this thesis is to further develop the FCM, in particular for nonlinear problems. To this end, we are focusing on three main topics of the aforementioned difficulties.

- In doing so, the first topic is related to the **numerical integration of finite cells**. Within the context of the FCM, it is common to employ adaptive Gaussian quadrature schemes. However, since these quadrature schemes frequently result in a large number of integration points, which renders the numerical integration computationally expensive, we propose novel quadrature methods based on the **moment fitting approach** in this thesis, in order to account for discontinuous integrals. To this end, we present a moment fitting method based on distinct point distribution schemes. Further, we propose an optimization procedure in order to solve the nonlinear moment fitting equation system, resulting in optimized points and weights. Furthermore, we introduce an efficient moment fitting method based on Gauss-Legendre points and Lagrange polynomials, thus circumventing the necessity of having to solve the moment fitting equation system. Moreover, in order to improve the robustness of the moment fitting quadratures for nonlinear problems of the FCM, an adaptive moment fitting approach is proposed. Considering several problems in linear and nonlinear applications of the FCM, we show that the presented moment fitting quadratures perform much more efficiently without loss in accuracy.

- The second topic of this thesis is addressed to the **ill-conditioning of the resulting global system matrix**. Commonly, an approach utilizing a fictitious material is applied in order to improve the conditioning behavior. In the scope of this thesis, we introduce a **basis function removal approach** applied to the hierarchic shape functions. In doing so, shape functions with only small contribution to the overall solution are removed from the ansatz. Thereby, we ensure that rigid body modes are preserved within the modified basis. This is especially of interest for applications in finite strain. By studying benchmark problems in linear and nonlinear problems of the FCM, we demonstrate that the presented basis function removal strategy allows to improve the conditioning behavior of the FCM significantly. Further, we show that the combination of the basis function removal together with the fictitious material approach helps to improve the robustness of FCM applications in finite strain. This is demonstrated by considering several problems of different complexity.

- The third and last topic of this thesis discusses the issue of **severely distorted finite cells for applications in finite strain**, which originate from large deformations of the fictitious domain within cut cells. As a result, the Newton-Raphson method fails, thus terminating the analysis. In order to overcome this problem, we introduce a **remeshing strategy** within the framework of a total Lagrangian formulation. The basic idea of this strategy is that whenever the analysis fails (due to severe distortions of the computational mesh), the simulation is continued based on a new mesh. In doing so, the mesh generation of the deformed structure can be carried out in an efficient manner thanks to the fictitious domain approach. For the transfer of the necessary data from the old mesh to the new one, we present a local radial basis function interpolation scheme. The performance of the proposed remeshing strategy is demonstrated by considering several applications in hyperelasticity. Further, the accuracy is investigated by means of a benchmark problem.

1.3 Outline of this thesis

The outline of this thesis is as follows.

- In **Chapter 2**, basic elements of continuum mechanics are summarized that are needed within this thesis. To this end, first kinematical relations and strain measures are discussed. Next, the equilibrium equations are derived, as well as important stress measures. Then, the constitutive equations regarding elastic and elastoplastic material models in small and large strain are briefly described. Finally, the chapter is concluded by deriving the weak and the linearized weak form of equilibrium.

- **Chapter 3** provides a brief description of the FCM. In doing so, we first motivate the essential idea of the fictitious domain approach and show the formulations of the weak and linearized weak form with respect to the extended domain. Next, we describe the spatial discretization and derive the governing equations of the discretized weak form. The chapter is concluded by a detailed explanation of the Gaussian quadrature method as well as an adaptive Gaussian quadrature scheme based on a spacetree decomposition.

- **Chapter 4** addresses the numerical integration of cut finite cells. To this end, a detailed review of various numerical integration approaches in nonstandard discretization methods is provided at the beginning of the chapter. Next, the basic idea and equations of the moment fitting approach are given. Then, various moment fitting methods are introduced. The performance of the different moment fitting quadratures in terms of accuracy and efficiency is studied by considering examples in numerical integration as well as several applications for linear and nonlinear problems of the FCM.

- In **Chapter 5**, we focus on the ill-conditioning behavior of the FCM. To this end, we start with a detailed review on various approaches that were developed in order to improve the conditioning of nonstandard discretization methods based on the fictitious domain or related approaches. Next, we propose a basis function removal strategy applied to the hierarchical shape functions that removes basis functions which provide a small contribution to the diagonal entries of the global stiffness matrix. The influence on the conditioning behavior as well as the accuracy of the FCM is investigated in detail with regard to linear and nonlinear benchmarks. Further, the effect on the robustness for FCM applications in finite strain is studied by considering several problems of different complexity.

- In **Chapter 6**, we present a remeshing strategy in order to overcome the problem of severely distorted cells for FCM applications in finite strain. Thus, we start off by introducing the necessary kinematical relations. Next, the remeshing procedure is explained. Here, several remeshing criteria are proposed to decide whether the analysis is continued based on a new mesh. For the spatial discretization, we employ a surface triangulation in order to account for the deformed boundary of the structure under consideration. The remeshing procedure is completed by introducing a local radial basis function interpolation scheme in order to transfer the necessary data from the old mesh to new one. The performance of the presented remeshing strategy in terms of accuracy is investigated regarding a benchmark problem. Further, in order to study the effect on the robustness, FCM applications of different complexity are considered.

Finally, the thesis is concluded in **Chapter 7**. Here, we give a brief summary and provide an outlook on further research opportunities in connection with the finite cell method.

2 Basic elements of continuum mechanics

In this chapter, we briefly outline the basic elements of continuum mechanics that are needed for the formulation of the finite cell method considering quasi-static nonlinear problems in solid mechanics. To this end, we start with the kinematics of a deformable body and introduce important strain measures, see Sec. 2.1. Then, in Sec. 2.2, the governing equilibrium equations are derived – in addition to fundamental stress measures. Next, in Sec. 2.3, the constitutive equations are introduced describing the relationship between stress measures and kinematic quantities, e.g. strain measures. Sec. 2.4 provides the strong as well as the weak form of the nonlinear boundary value problem. The linearization of the weak form is given in Sec. 2.5. In doing so, all stated relations are given with respect to the initial (undeformed) and the current (deformed) configuration of a body. As the formulations are described in brief, the interested reader is referred to the textbooks of Wriggers [74], Belytschko et al. [75], Simo and Hughes [76], Bonet and Wood [77], and Neto et al. [78] for a more detailed overview. For this thesis, we chose a notation that is closely related to that used in Wriggers [74] and Neto et al. [78].

2.1 Kinematics

Kinematics provides a description of the motion and the deformation of a continuum without reference to the cause, e.g. external loads. Thereby, in classical continuum mechanics, we distinguish between two fundamental approaches: the *Lagrangian* and the *Eulerian* description of motion. While the Eulerian description is widely used in fluid mechanics, the Lagrangian description is mostly applied in the context of *structural* (or *solid*) mechanics. Since we are focusing on the analysis of quasi-static problems in solid mechanics, in this thesis, kinematic relations and quantities – which are needed within the constitutive models and the weak (or variational) form – are described from the Lagrangian point of view. In doing so, we start with the general description of the motion and the deformation of a continuum, then concluding with the derivation of relevant strain measures.

2.1.1 Motion and deformation

Let us begin with the description of the motion and the deformation of a continuum. In structural mechanics, a continuum is defined by a set of continuously distributed particles, also called material points, forming a homogeneous body Ω with closed boundary $\partial\Omega$, as depicted in Fig. 2.1. The *undeformed* state of such a body at time 0 is called the *initial* configuration Ω in which the location of a material point is defined by the position vector $\boldsymbol{X}$. Now, let us assume that the body experiences a deformation over time. Then, the *deformed* state a of the body at time t is called *current* configuration $\varphi(\Omega)$. Based on the Lagrangian approach – which follows the motion of a particle in space and time – the position of a material point in time is described by

$$\boldsymbol{x} = \boldsymbol{\varphi}\left(\boldsymbol{X}, t\right) \quad . \tag{2.1}$$

Here, $\boldsymbol{\varphi}\left(\boldsymbol{X}, t\right)$ defines a nonlinear bijective mapping where the material position $\boldsymbol{X}$ represents an independent variable. Further, the spatial position $\boldsymbol{x}$ describes a variable depending on both the material position $\boldsymbol{X}$ and the time $t \geq 0$. In other words, the mapping $\boldsymbol{\varphi}\left(\boldsymbol{X}, t\right)$ relates the *material* coordinates $\boldsymbol{X}$ of the *initial* configuration Ω at time 0 with the *spatial* coordinates of the *current* configuration $\varphi\left(\Omega\right)$ at a fixed time instant t. The difference between the position vectors $\boldsymbol{X}$ and $\boldsymbol{x}$ is represented by the displacement $\boldsymbol{u}$

$$\boldsymbol{u} = \boldsymbol{\varphi}\left(\boldsymbol{X}, t\right) - \boldsymbol{X} \quad . \tag{2.2}$$

Consequently, the map can be expressed as a function of initial position of a particle and its displacement as

$$\boldsymbol{\varphi}\left(\boldsymbol{X}, t\right) = \boldsymbol{X} + \boldsymbol{u} \quad . \tag{2.3}$$

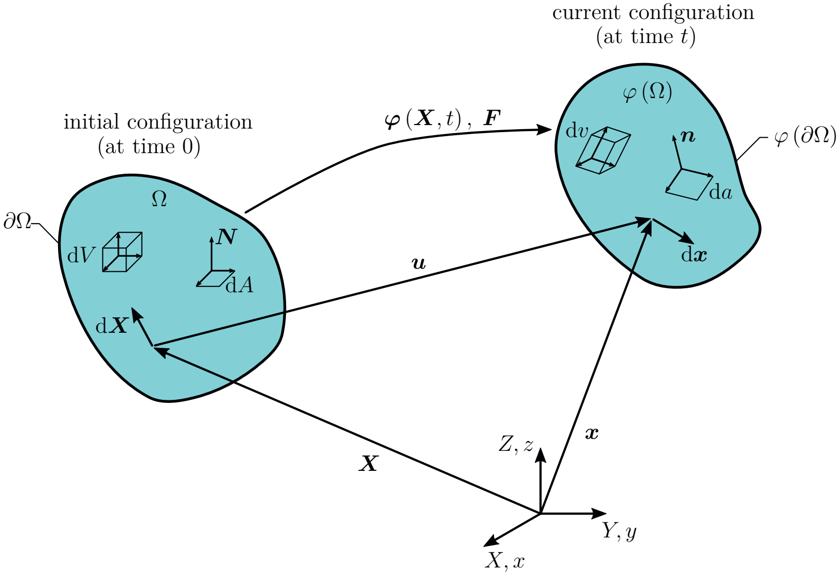


Figure 2.1: The motion and the deformation of a body Ω.

In order to provide a better understanding of the following formulations and relations, we give a brief summary of the *initial* as well as the *current* configurations, respectively.

- The *initial* configuration Ω – also called *reference*, *undeformed*, or *material* configuration – refers to the undeformed and stress-free state of a body at time 0. In this configuration, the position of the material points is defined by the initial (or material) coordinates $\boldsymbol{X}$ which represent an independent variable. This means that $\boldsymbol{X}$ does not change over time. In the following formulations, the usage of capital letters for quantities or operators is associated with the *initial* configuration – e.g. $\operatorname{Grad}\boldsymbol{u}$ defines the gradient of the displacement vector with respect to the *initial* coordinates ($\operatorname{grad}_{\boldsymbol{X}}\boldsymbol{u}$).

- The *current* configuration $\varphi(\Omega)$ – also called *deformed* or *spatial* configuration – refers to the deformed state of a body at the current time t. In this configuration, the position of the material points is defined by the spatial coordinates $\boldsymbol{x}$ which represent a variable depending on both the material coordinates $\boldsymbol{X}$ and the time t. The dependency is described by the nonlinear bijective mapping function $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t)$. In the following formulations, the usage of small letters for quantities or operators is associated with the *current configuration* – e.g. $\operatorname{grad}\boldsymbol{u}$ defines the gradient of the displacement vector with respect to the *current* coordinates ($\operatorname{grad}_{\boldsymbol{x}}\boldsymbol{u}$).

In order to study the deformation (the change in size and shape) of a body when it is mapped from the initial to the current configuration, let us introduce the *deformation gradient* $\boldsymbol{F}$ – which, in continuum mechanics, is an important quantity when it comes to describing local deformation processes. To this end, we consider an infinitesimal vector $\mathrm{d}\boldsymbol{X}$ in the initial configuration, connecting two material points $\boldsymbol{X}$ and $\boldsymbol{X} + \mathrm{d}\boldsymbol{X}$. Thus, the vector $\mathrm{d}\boldsymbol{X}$ can be interpreted as an infinitesimal line segment at point $\boldsymbol{X}$. The related deformed vector $\mathrm{d}\boldsymbol{x}$ in the current configuration is defined by the current positions of these material points $\boldsymbol{\varphi}(\boldsymbol{X}, t)$ and $\boldsymbol{\varphi}(\boldsymbol{X} + \mathrm{d}\boldsymbol{X}, t)$, respectively. The transformation of the initial vector $\mathrm{d}\boldsymbol{X}$ to the current vector $\mathrm{d}\boldsymbol{x}$ is defined by the gradient of the deformation map

$$\mathrm{d}\boldsymbol{x} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}}\,\mathrm{d}\boldsymbol{X} = \operatorname{Grad}\boldsymbol{\varphi}(\boldsymbol{X}, t)\;\mathrm{d}\boldsymbol{X} \quad . \tag{2.4}$$

Thus, with the definition of the deformation gradient $\boldsymbol{F}$

$$\boldsymbol{F} = \operatorname{Grad}\boldsymbol{\varphi}(\boldsymbol{X}, t) \tag{2.5}$$

Eq. (2.4), finally, reads

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{F}\,\mathrm{d}\boldsymbol{X} \quad . \tag{2.6}$$

Consequently, the deformation gradient $\boldsymbol{F}$ represents a linear operator that maps an infinitesimal vector $\mathrm{d}\boldsymbol{X}$ from the initial configuration to its counterpart $\mathrm{d}\boldsymbol{x}$ in the current configuration. Further, to preserve the connection of the body during the deformation process and to avoid self-penetration, we postulate the following condition for the determinant of the deformation gradient

$$J = \det\boldsymbol{F} \geq 0 \quad . \tag{2.7}$$

Moreover, the deformation gradient can be also formulated in terms of the displacements by utilizing the relation stated in Eq. (2.3). Then, the deformation gradient $\boldsymbol{F}$ reads

$$\boldsymbol{F} = \operatorname{Grad}\boldsymbol{\varphi}(\boldsymbol{X}, t) = \operatorname{Grad}\boldsymbol{X} + \operatorname{Grad}\boldsymbol{u} = \mathbf{1} + \boldsymbol{H} \quad , \tag{2.8}$$

where $\boldsymbol{H}$ defines the *displacement gradient* and $\mathbf{1}$ denotes the *second-order identity* tensor.

Now, having introduced the deformation gradient allows us to describe the transformation of further geometric quantities such as surface and volume elements. To this end, let us consider an area element $\mathrm{d}\boldsymbol{A}$ which is located at the surface $\partial\Omega$ of the undeformed body Ω. Further, let us assume that the area $\mathrm{d}\boldsymbol{A}$ is defined by two infinitesimal and linearly independent vectors $\mathrm{d}\boldsymbol{X}$ and $\mathrm{d}\boldsymbol{Y}$ describing tangents on an arbitrary point at surface $\partial\Omega$. Note that vector $\mathrm{d}\boldsymbol{X}$ is not the same as in Fig. 2.1. Thus, the area element in the initial configuration can be described by the cross product of the tangent vectors as

$$\boldsymbol{N}\,\mathrm{d}A = \mathrm{d}\boldsymbol{X} \times \mathrm{d}\boldsymbol{Y} \quad , \tag{2.9}$$

where $\boldsymbol{N}$ is the unit normal to the tangents and $\mathrm{d}A$ defines the area. Now, using the transformation relation in Eq. (2.6), we can map the tangent vectors form the undeformed surface $\partial\Omega$ onto the deformed one $\varphi\,(\partial\Omega)$. Thus, the deformed area element in the current configuration can be described as

$$\boldsymbol{n}\,\mathrm{d}a = \mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{y} = \boldsymbol{F}\,\mathrm{d}\boldsymbol{X} \times \boldsymbol{F}\,\mathrm{d}\boldsymbol{Y} \quad , \tag{2.10}$$

where $\boldsymbol{n}$ represents the unit normal to the deformed tangents and $\mathrm{d}a$ defines the deformed infinitesimal area. Finally, the area elements of the initial and the current configuration are related to each other by the well-known Nanson formula

$$\boldsymbol{n}\,\mathrm{d}a = J\boldsymbol{F}^{-T}\boldsymbol{N}\,\mathrm{d}A \quad . \tag{2.11}$$

Next, let us discuss the change in volume of a body. To this end, we consider an infinitesimal volume element $\mathrm{d}V$ at an arbitrary point within the inside of the undeformed body in the initial configuration. In doing so, the volume element is defined by three infinitesimal and linearly independent vectors $\mathrm{d}\boldsymbol{X}$, $\mathrm{d}\boldsymbol{Y}$, and $\mathrm{d}\boldsymbol{Z}$. Observe that $\mathrm{d}X$ does not represent the vector given in Fig. 2.1. Then, the volume of the undeformed element can be described as

$$\mathrm{d}V = (\mathrm{d}\boldsymbol{X} \times \mathrm{d}\boldsymbol{Y}) \cdot \mathrm{d}\boldsymbol{Z} \quad . \tag{2.12}$$

Using the relation in Eq. (2.6), the deformed volume element $\mathrm{d}v$ in the current configuration reads

$$\mathrm{d}v = (\mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{y}) \cdot \mathrm{d}\boldsymbol{z} = (\boldsymbol{F}\,\mathrm{d}\boldsymbol{X} \times \boldsymbol{F}\,\mathrm{d}\boldsymbol{Y}) \cdot \boldsymbol{F}\,\mathrm{d}\boldsymbol{Z} \quad . \tag{2.13}$$

Thus, from Eq. (2.12) and (2.13), we can deduce the following mapping that relates the undeformed volume element $\mathrm{d}V$ with the deformed one $\mathrm{d}v$

$$\mathrm{d}v = \det\boldsymbol{F}\,\mathrm{d}V = J\,\mathrm{d}V \quad . \tag{2.14}$$

Having described the local deformation process by introducing the deformation gradient $\boldsymbol{F}$, let us conclude this section by mentioning some important transformation rules that are needed to perform *push forward* and *pull back* operations. From a theoretical point of view, there is no difference to describing the basic relations of continuum mechanics with respect to the initial or the current configuration. Thus, formulations in the current configuration can be transformed to the initial one by applying *pull back* operations and vice versa by applying *push forward* operations. Some important transformation rules of the *push forward* operations concerning the gradient and divergence operators are

$$\operatorname{grad}\beta = \boldsymbol{F}^{-T}\operatorname{Grad}\beta \quad , \quad \operatorname{grad}\boldsymbol{\beta} = \operatorname{Grad}\boldsymbol{\beta}\,\boldsymbol{F} \quad , \quad \operatorname{div}\boldsymbol{\beta} = \frac{1}{J}\operatorname{Div}\boldsymbol{\beta} \quad , \tag{2.15}$$

where β represents a scalar field and $\boldsymbol{\beta}$ is a vector field. Further rules characterizing the integral transformations are given as

$$\int_{\varphi(\Omega)} (.)\,\mathrm{d}v = \int_{\Omega} (.) J\,\mathrm{d}V \quad , \quad \int_{\varphi(\partial\Omega)} (.)\boldsymbol{n}\,\mathrm{d}a = \int_{\partial\Omega} (.) J\boldsymbol{F}^{-T}\boldsymbol{N}\,\mathrm{d}A \tag{2.16}$$

Finally, the corresponding transformations of the *pull back* operations are obtained by reformulation of the above expressions.

2.1.2 Strain measures

In the previous section, we introduced the deformation gradient that maps an infinitesimal vector $\mathrm{d}\boldsymbol{X}$ connecting two material points from the initial to the current configuration. If the distance between the material points changes after deformation, we consider the vector as *strained* – otherwise it is *unstrained.* Consequently, it is obvious that the strain is independent in terms of rotation. Due to this fact, it is reasonable to split the deformation gradient multiplicatively into a *rotation* and a *stretch* tensor where the stretch characterizes the change in length of the deformed vector. The multiplicative split is also known as the *polar decomposition* of the deformation gradient

$$\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U} = \boldsymbol{V}\boldsymbol{R} \quad , \tag{2.17}$$

where $\boldsymbol{R}$ denotes the *rotation* tensor, which is orthonormal, and $\boldsymbol{U}$ and $\boldsymbol{V}$ define the *right* and *left stretch* tensor, respectively. Thereby, $\boldsymbol{U}$ describes the stretch of a vector with respect to the initial configuration and $\boldsymbol{V}$ characterizes the stretch with respect to the current configuration. Further, $\boldsymbol{U}$ and $\boldsymbol{V}$ are *symmetric* and *positive definite* tensors which is why they can be represented in terms of their eigenvalues and eigenvectors by applying the spectral decomposition

$$\boldsymbol{U} = \lambda_i\,\boldsymbol{L}_i \otimes \boldsymbol{L}_i \quad \text{and} \quad \boldsymbol{V} = \lambda_i\,\boldsymbol{e}_i \otimes \boldsymbol{e}_i \quad , \tag{2.18}$$

where λ_i are the eigenvalues, also referred to as *principal stretches*, and $\boldsymbol{L}_i$ as well as $\boldsymbol{e}_i$ characterize the *Lagrangian* and the *Eulerian principal* directions, respectively. Moreover, the *right* and *left Cauchy-Green* tensors – which represent fundamental stretch measures in continuum mechanics – are defined based on the stretch tensors $\boldsymbol{U}$ and $\boldsymbol{V}$ as

$$\boldsymbol{C} = \boldsymbol{U}^2 = \boldsymbol{F}^T\boldsymbol{F} \quad \text{and} \quad \boldsymbol{b} = \boldsymbol{V}^2 = \boldsymbol{F}\boldsymbol{F}^T \quad . \tag{2.19}$$

Now, with the definition of the stretch, we can formulate the squared distance of the deformed vector $\mathrm{d}\boldsymbol{x}$ in terms of the stretch tensors as

$$\|\mathrm{d}\boldsymbol{x}\|_2^2 = \mathrm{d}\boldsymbol{x}\cdot\mathrm{d}\boldsymbol{x} = \boldsymbol{F}\mathrm{d}\boldsymbol{X}\cdot\boldsymbol{F}\mathrm{d}\boldsymbol{X} = \boldsymbol{F}^T\boldsymbol{F}\mathrm{d}\boldsymbol{X}\cdot\mathrm{d}\boldsymbol{X} = \boldsymbol{C}\mathrm{d}\boldsymbol{X}\cdot\mathrm{d}\boldsymbol{X} = \boldsymbol{U}^2\,\mathrm{d}\boldsymbol{X}\cdot\mathrm{d}\boldsymbol{X} \quad . \tag{2.20}$$

However, the stretch tensors do not represent a measure to describe the difference between the deformed vector $\mathrm{d}\boldsymbol{x}$ with respect to its undeformed counterpart $\mathrm{d}\boldsymbol{X}$. Consequently, additional measures are needed to describe the straining process. Thereby, one important strain measure in the context of continuum mechanics is the *Green-Lagrange strain* tensor

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{C} - \boldsymbol{1}\right) = \frac{1}{2}\left(\boldsymbol{U}^2 - \boldsymbol{1}\right) = \frac{1}{2}\left(\boldsymbol{H} + \boldsymbol{H}^T + \boldsymbol{H}^T\boldsymbol{H}\right) \tag{2.21}$$

which vanishes for rigid body motions. This behavior defines a fundamental property that strain measures have to satisfy when problems with large deformation, especially large rotations, are considered. Note that the *Green-Lagrange strain* defines a tensor in the initial configuration and, thus, that it can be described in terms of the *principal stretches* and *Lagrangian principal* directions as

$$\boldsymbol{E} = \frac{1}{2}\left(\lambda_i^2 - 1\right)\boldsymbol{L}_i \otimes \boldsymbol{L}_i \quad . \tag{2.22}$$

Now, with the definition of the *Green-Lagrange strain* tensor the squared distance of the deformed vector can be formulated in terms of $\boldsymbol{E}$ as follows

$$||\mathrm{d}\boldsymbol{x}||^2 = (\boldsymbol{1} - 2\boldsymbol{E})\,\mathrm{d}\boldsymbol{X}\cdot\mathrm{d}\boldsymbol{X} \quad . \tag{2.23}$$

Considering problems of *infinitesimal* deformations – where the displacement gradient is adequately small – the higher order term $\boldsymbol{H}^T\boldsymbol{H}$ of the *Green-Lagrange strain* tensor can be neglected, thus resulting in the well-known *infinitesimal strain* tensor

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\boldsymbol{H} + \boldsymbol{H}^T\right) \quad . \tag{2.24}$$

The *infinitesimal strain* tensor $\boldsymbol{\varepsilon}$ is also known as *engineering* or *small strain* tensor.

Let us conclude this section by introducing further important strain measures from the field of continuum mechanics. Following Seth (1964), Hill (1978), and Odgen (1984) a generalization of the *Lagrangian strain* tensors – which are defined with respect to the initial configuration – can be given as

$$\boldsymbol{E}^m = \begin{cases} \frac{1}{m}\left(\boldsymbol{U}^m - \boldsymbol{1}\right) & m \neq 0 \\ \ln \boldsymbol{U} & m = 0 \end{cases} \tag{2.25}$$

where m is a real number. Observe that for any choice of m the related strain tensor vanishes for rigid body motions ($\boldsymbol{F} = \boldsymbol{R}$). Further, note that for $m = 2$ the expression in Eq. (2.25) results in the *Green-Lagrange strain* tensor. Other important strain tensors are the *Biot* ($m = 1$), the *Hencky* ($m = 0$), and the *Almansi strain* tensor ($m = -2$). Moreover, the relation in Eq. (2.25) can be rephrased in terms of its spectral decomposition

$$\boldsymbol{E}^m = f\left(\lambda_i\right)\boldsymbol{L}_i \otimes \boldsymbol{L}_i \tag{2.26}$$

where $f\left(\lambda_i\right)$ reads

$$f\left(\lambda_i\right) = \begin{cases} \frac{1}{m}\left(\lambda_i^m - 1\right) & m \neq 0 \\ \ln \lambda_i & m = 0 \quad . \end{cases} \tag{2.27}$$

A generalized formulation of the *Eulerian strain* tensors – which are defined with respect to the current configuration – can be obtained in an analogous way

$$\boldsymbol{\varepsilon}^m = \begin{cases} \frac{1}{m}\left(\boldsymbol{V}^m - \boldsymbol{1}\right) & m \neq 0 \\ \ln \boldsymbol{V} & m = 0 \quad . \end{cases} \tag{2.28}$$

Given in terms of its spectral decomposition, the *Eulerian strain* tensors read

$$\boldsymbol{\varepsilon}^m = f\left(\lambda_i\right)\boldsymbol{e}_i \otimes \boldsymbol{e}_i \tag{2.29}$$

where

$$f\left(\lambda_i\right) = \begin{cases} \frac{1}{m}\left(\lambda_i^m - 1\right) & m \neq 0 \\ \ln \lambda_i & m = 0 \quad . \end{cases} \tag{2.30}$$

2.2 Equilibrium and stress measures

In this section, we derive the equilibrium equations and introduce important stress measures.

2.2.1 Equilibrium

In the following, let us derive the governing equilibrium equations of a general deformable body that is subjected to prescribed displacements and under the action of body and traction loads, as depicted in Fig. 2.2. In the figure, $\bar{\boldsymbol{u}}$ defines the prescribed displacement which is applied on the boundaries Γ_{D} and $\varphi\left(\Gamma_{\mathrm{D}}\right)$, $\rho\boldsymbol{b}$ and $\rho_0\boldsymbol{b}$ denote the body loads per unit volume acting on the domains Ω and $\varphi\left(\Omega\right)$, and $\bar{\boldsymbol{t}}$ as well as $\hat{\boldsymbol{t}}$ are the traction loads per unit area which act on to the boundaries Γ_{N} and $\varphi\left(\Gamma_{\mathrm{N}}\right)$.

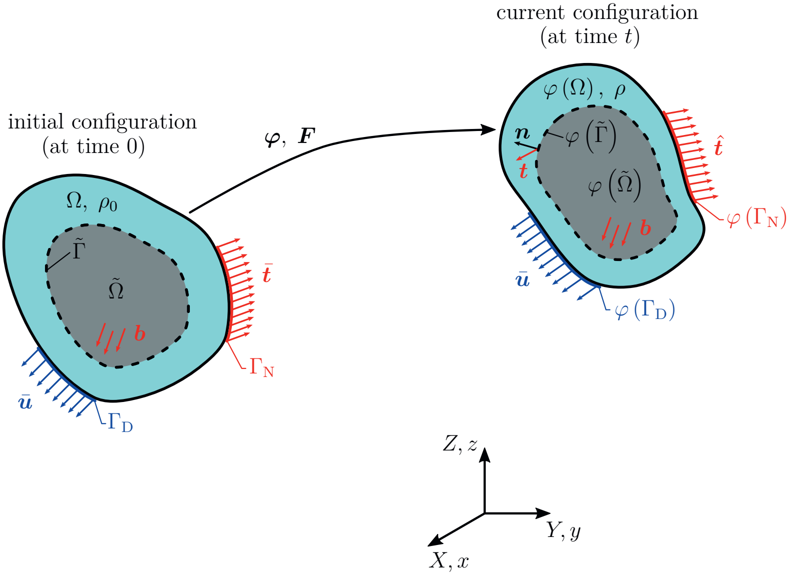


Figure 2.2: A deformable body under the action of traction and body loads.

In order to derive the partial differential equations of the equilibrium, let us consider the finite partial volume $\varphi\left(\tilde{\Omega}\right)$ with boundary $\varphi\left(\tilde{\Gamma}\right)$ of the deformed body in the current configuration. Further, for reasons of simplicity, inertia terms are neglected since we are dealing with quasi-static problems in this thesis. In doing so, the translational equilibrium postulates that the sum of the external loads acting on the finite volume and its boundary has to vanish. This results in the following condition

$$\int_{\varphi(\tilde{\Gamma})} \boldsymbol{t} \, \mathrm{d}a + \int_{\varphi(\tilde{\Omega})} \rho \boldsymbol{b} \, \mathrm{d}v = \boldsymbol{0} \quad , \tag{2.31}$$

where $\boldsymbol{t}$ is the stress vector acting on the boundary $\varphi\left(\tilde{\Gamma}\right)$ and $\boldsymbol{n}$ denotes the normal vector of the boundary. Now, replacing the stress vector $\boldsymbol{t}$ with the relation between the *Cauchy stress* tensor $\boldsymbol{\sigma}$ and the normal vector $\boldsymbol{n}$ by applying Cauchy's theorem, and using the divergence theorem, the surface integral of the first term in Eq. (2.31) can be transformed into a volume integral resulting in the following expression

$$\int_{\varphi(\tilde{\Omega})} \left(\mathrm{div}\boldsymbol{\sigma} + \rho \boldsymbol{b}\right) \mathrm{d}v = \boldsymbol{0} \quad . \tag{2.32}$$

For an any finite volume of the body, the formulation in Eq. (2.32) can only be satisfied if the integrand vanishes

$$\mathrm{div}\boldsymbol{\sigma} + \rho \boldsymbol{b} = \boldsymbol{0} \quad . \tag{2.33}$$

Consequently, the relation in Eq. (2.33) defines the local equilibrium with respect to the current configuration, which has to be fulfilled at any point within the inside of the deformed body $\varphi\left(\boldsymbol{\Omega}\right)$.

The well-known symmetry of the *Cauchy stress* tensor can be shown considering the rotational equilibrium. In doing so, let us again consider the finite partial volume $\varphi\left(\tilde{\Omega}\right)$ in Fig. 2.2. Now, to achieve rotational equilibrium, the total moment of body and traction loads with reference to any point, such as the origin, has to vanish

$$\int_{\varphi(\tilde{\Gamma})} \boldsymbol{x} \times \boldsymbol{t} \, \mathrm{d}a + \int_{\varphi(\tilde{\Omega})} \boldsymbol{x} \times \rho \boldsymbol{b} \, \mathrm{d}v = \boldsymbol{0} \tag{2.34}$$

where we recall that the moment results from the cross product of the force with a position vector $\boldsymbol{x}$, see Fig. 2.1. Once again – using Cauchy's theorem, which relates the stress vector $\boldsymbol{t}$ with the Cauchy stress $\boldsymbol{\sigma}$ and the normal vector $\boldsymbol{n}$, and by applying the divergence theorem – the rotational equilibrium can be formulated as follows

$$\int_{\varphi(\tilde{\Omega})} \boldsymbol{x} \times \left(\boldsymbol{\sigma} \boldsymbol{n}\right) \mathrm{d}v + \int_{\varphi(\tilde{\Omega})} \boldsymbol{x} \times \rho \boldsymbol{b} \, \mathrm{d}v = \boldsymbol{0} \quad . \tag{2.35}$$

After some manipulations of Eq. (2.35) and by taking the translational equilibrium in Eq. (2.33) into account, the rotational equilibrium implies the symmetry of the *Cauchy stress* tensor

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \quad . \tag{2.36}$$

2.2.2 Stress measures

Often, it is preferable to formulate the local equilibrium equation with respect to the initial configuration of the undeformed body Ω. In order to do so, it is necessary to introduce further stress measures that are related to each other by special transformation rules.

Applying Nanson's formula Eq. (2.11) which describes the change in area between the deformed $\varphi(\Omega)$ and the undeformed body Ω, the surface integral of the stress vector $\boldsymbol{t}$ in Eq. (2.31) can be transformed to the initial configuration as

$$\int\limits_{\varphi(\tilde{\Gamma})} \boldsymbol{t}\,\mathrm{d}a = \int\limits_{\varphi(\tilde{\Gamma})} \boldsymbol{\sigma}\boldsymbol{n}\,\mathrm{d}a = \int\limits_{\tilde{\Gamma}} \boldsymbol{\sigma} J \boldsymbol{F}^{-T}\boldsymbol{N}\,\mathrm{d}A = \int\limits_{\tilde{\Gamma}} \boldsymbol{P}\boldsymbol{N}\,\mathrm{d}A \quad . \tag{2.37}$$

Eq. (2.37) introduces the stress tensor $\boldsymbol{P}$ which denotes the *first Piola-Kirchhoff stress* tensor. From Eq. (2.37), the following relation between the *first Piola-Kirchhoff* and the *Cauchy stress* tensor can be established

$$\boldsymbol{P} = J\boldsymbol{\sigma}\boldsymbol{F}^{-T} \quad . \tag{2.38}$$

Further, using the condition of conversation of mass

$$\rho \mathrm{d}v = \rho_0 \mathrm{d}V \tag{2.39}$$

the volume integral in Eq. (2.31) can then be transformed to the initial configuration. Thus, the translational equilibrium with respect to the initial configurations follows as

$$\int\limits_{\tilde{\Gamma}} \boldsymbol{P}\boldsymbol{N}\,\mathrm{d}A + \int\limits_{\tilde{\Omega}} \rho_0 \boldsymbol{b}\,\mathrm{d}V = \boldsymbol{0} \quad . \tag{2.40}$$

Once again applying the divergence theorem, the surface integral can be transformed into a volume integral

$$\int\limits_{\tilde{\Omega}} \left(\operatorname{Div}\boldsymbol{P} + \rho_0\boldsymbol{b}\right)\,\mathrm{d}V = \boldsymbol{0} \quad . \tag{2.41}$$

The expression can only be satisfied if the term of the integral vanishes and, thus, the local equilibrium with respect to the initial configuration reads

$$\operatorname{Div}\boldsymbol{P} + \rho_0\boldsymbol{b} = \boldsymbol{0} \quad . \tag{2.42}$$

Since the *first Piola-Kirchhoff stress* tensor $\boldsymbol{P}$ is nonsymmetric, it is necessary to introduce the *second Piola-Kirchhoff stress* tensor $\boldsymbol{S}$ – which is symmetric and which results from a complete transformation of the *Cauchy stress* tensor

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\boldsymbol{P} = J\boldsymbol{F}^{-1}\boldsymbol{\sigma}\boldsymbol{F}^{-T} \quad . \tag{2.43}$$

Observe that $\boldsymbol{S}$ represents a pure mathematical quantity and, thus, can not be interpreted in a physical manner. Within the constitutive theory, however, $\boldsymbol{S}$ occupies an important role since it forms a work conjugated pair together with the *Green-Lagrange strain* tensor.

Another important stress measure is the so-called *Kirchhoff stress* tensor $\boldsymbol{\tau}$, which results from a *push forward* of the *second Piola-Kirchhoff stress* tensor to the current configuration

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^{T} \quad , \tag{2.44}$$

thus, the relation between the *Kirchhoff* and the *Cauchy stress* tensor is given as

$$\boldsymbol{\tau} = J\boldsymbol{\sigma} \quad . \tag{2.45}$$

2.3 Constitutive equations

The kinematic relations as well as the equilibrium equations introduced in the previous sections are formulated independently in terms of the material behavior of a body. Consequently, further formulations are needed in order to solve boundary value problems in continuum mechanics. The relations describing the material behavior of a deformable body are governed by the constitutive equations. These equations include the necessary formulations between stress measures and kinematic relations, e.g. strain measures. In the scope of this thesis, we consider elastic and elastoplastic material models for problems with small and finite strain. Therefore, this section serves to provide a brief description of the underlying constitutive equations of the different material models.

2.3.1 Linear elasticity

In the simple case of an elastic and isotropic material behavior of a deformable body undergoing small displacements and deformations, the constitutive equations are given as

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\varepsilon} + \lambda \operatorname{tr} \boldsymbol{\varepsilon} \, \mathbf{1} \quad . \tag{2.46}$$

These equations, which are well-known as *Hooke's law*, define the relation between the *Cauchy stress* tensor and the *infinitesimal strain* tensor, which is also known as the *engineering strain* tensor. In *Hooke's law*, $\boldsymbol{\sigma}$ depends linearly on $\boldsymbol{\varepsilon}$. This linear relationship is defined by the Lamé parameters λ and μ, which are also known as the Lamé constants or Lamé coefficients. Note that $\boldsymbol{\sigma}$ can also be computed based on other material parameters, e.g. Young's modulus E and Poisson's ratio ν.

2.3.2 Hyperelasticity

In order to describe the material behavior of an elastic and isotropic body undergoing large displacements and deformations, hyperelastic material models are well-suited. These models are characterized by a strain (or stored) energy density function describing a scalar valued function

$$W\left(\boldsymbol{F}\right) = W\left(\boldsymbol{C}\right) \tag{2.47}$$

that defines a potential [79]. Based on the strain energy function, the constitutive equations can be derived by taking the derivative of $W\left(\boldsymbol{F}\right)$ or $W\left(\boldsymbol{C}\right)$ with respect to the deformation gradient $\boldsymbol{F}$ or the *right Cauchy-Green* tensor $\boldsymbol{C}$, respectively. In doing so, the constitutive equations of the *first Piola-Kirchhoff stress* tensor are given as

$$\boldsymbol{P} = \frac{\partial W}{\partial \boldsymbol{F}} = 2\boldsymbol{F}\frac{\partial W}{\partial \boldsymbol{C}} \tag{2.48}$$

and the constitutive equations with respect to the *second Piola-Kirchhoff stress* read

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\frac{\partial W}{\partial \boldsymbol{F}} = 2\frac{\partial W}{\partial \boldsymbol{C}} \quad . \tag{2.49}$$

Further, with the relation given in Eq. (2.38) and (2.45), the constitutive equations of the *Cauchy* as well as the *Kirchhoff stress* tensor are obtained as

$$\boldsymbol{\sigma} = \frac{1}{J}\frac{\partial W}{\partial \boldsymbol{F}}\boldsymbol{F}^T = \frac{2}{J}\boldsymbol{F}\frac{\partial W}{\partial \boldsymbol{C}}\boldsymbol{F}^T \tag{2.50}$$

and

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{F}} \boldsymbol{F}^T = 2\boldsymbol{F} \frac{\partial W}{\partial \boldsymbol{C}} \boldsymbol{F}^T \quad . \tag{2.51}$$

In the scope of this thesis, we utilize a hyperelastic material model – introduced by Ciarlet in [79] – that is based on a polyconvex strain energy density function

$$W = \frac{\mu}{2} \left(\mathrm{tr}\, \boldsymbol{C} - 3 \right) + \frac{\lambda}{4} \left(J^2 - 1 \right) - \left(\frac{\lambda}{2} + \mu \right) \ln J \tag{2.52}$$

with λ and μ denoting the Lamé parameters. Then, using the definitions in Eq. (2.48), (2.49), (2.50), and (2.51) the constitutive equations of the different stress measures are obtained as

$$\boldsymbol{P} = \frac{\lambda}{2} \left(J^2 - 1 \right) \boldsymbol{F}^{-T} + \mu \left(\boldsymbol{F} - \boldsymbol{F}^{-T} \right) \tag{2.53}$$

$$\boldsymbol{S} = \frac{\lambda}{2} \left(J^2 - 1 \right) \boldsymbol{C}^{-1} + \mu \left(\boldsymbol{1} - \boldsymbol{C}^{-1} \right) \tag{2.54}$$

$$\boldsymbol{\sigma} = \frac{\lambda}{2J} \left(J^2 - 1 \right) \boldsymbol{1} + \frac{\mu}{J} \left(\boldsymbol{b} - \boldsymbol{1} \right) \tag{2.55}$$

$$\boldsymbol{\tau} = \frac{\lambda}{2} \left(J^2 - 1 \right) \boldsymbol{1} + \mu \left(\boldsymbol{b} - \boldsymbol{1} \right) \quad . \tag{2.56}$$

2.3.3 Small strain elastoplasticity

Next, we briefly describe the governing equations of elastoplasticity regarding small strains. For a more detailed overview of the material model, the reader is referred to [74, 76, 78, 80].

In this thesis, we assume a model that is based on J_2 flow theory of plasticity with non-linear isotropic hardening. The formulation of the material model starts with an additive decomposition of the *infinitesimal strain* tensor $\boldsymbol{\varepsilon}$ into an elastic $\boldsymbol{\varepsilon}_e$ and a plastic part $\boldsymbol{\varepsilon}_p$ as

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_e + \boldsymbol{\varepsilon}_p \quad . \tag{2.57}$$

In doing so, the constitutive relation between the *Cauchy stress* tensor $\boldsymbol{\sigma}$ and the strain measure is given by an isotropic and linear elastic material model as

$$\boldsymbol{\sigma} = \boldsymbol{D}_e : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_p) = \boldsymbol{D}_e : \boldsymbol{\varepsilon}_e \quad , \tag{2.58}$$

where $\boldsymbol{D}_e$ defines the fourth-order *elasticity* tensor. Next, in order to account for admissible stress states of $\boldsymbol{\sigma}$, the von Mises yield criterion is introduced

$$\Phi \left(\boldsymbol{\sigma}, \bar{\alpha} \right) = \| \mathrm{dev}\, \boldsymbol{\sigma} \| - \sqrt{\frac{2}{3}} K \left(\bar{\alpha} \right) \leq 0 \quad . \tag{2.59}$$

Consequently, the elastic domain is defined by $\Phi \left(\boldsymbol{\sigma}, \bar{\alpha} \right) < 0$ and the plastic domain is given by the isosurface of the yield function $\Phi \left(\boldsymbol{\sigma}, \bar{\alpha} \right) = 0$. Further, $\Phi \left(\boldsymbol{\sigma}, \bar{\alpha} \right) > 0$ denote inadmissible stress states. In Eq. (2.59), $\| \mathrm{dev}\, \boldsymbol{\sigma} \|$ represents the Euclidean norm of the deviatoric part of the *Cauchy stress* tensor – where the definition of $\mathrm{dev}\, \boldsymbol{\sigma}$ is given as

$$\mathrm{dev}\, \boldsymbol{\sigma} = \boldsymbol{\sigma} - \frac{1}{3} \mathrm{tr}\, \boldsymbol{\sigma}\, \boldsymbol{1} \quad . \tag{2.60}$$

Further, $K(\bar{\alpha})$ describes the nonlinear and isotropic hardening curve. In this thesis, $K(\bar{\alpha})$ is composed of a linear and an exponential part as

$$K(\bar{\alpha}) = \sigma_0 + h\bar{\alpha} + (\sigma_\infty - \sigma_0)\left(1 - e^{-\omega\bar{\alpha}}\right) \quad . \tag{2.61}$$

In Eq. (2.61), σ_0 is the initial yield stress, h the linear hardening parameter, σ_∞ the saturation stress, and ω the hardening exponent. Further, $\bar{\alpha}$ defines an internal variable which is often referred to as equivalent plastic strain. Finally, the associative flow rule describing the evolution of the plastic strain is given as

$$\dot{\boldsymbol{\varepsilon}}_p = \dot{\gamma}\frac{\partial\Phi(\boldsymbol{\sigma}, \bar{\alpha})}{\partial\boldsymbol{\sigma}} \quad , \tag{2.62}$$

where $\gamma \geq 0$ denotes the nonnegative plastic multiplier.

2.3.4 Finite strain plasticity

In the following, we give a brief explanation of the material model assuming J_2 flow theory of plasticity for problems in finite strain. A more detailed description of the theory can be found in [81–84].

The formulation starts with the multiplicative decomposition of the deformation gradient $\boldsymbol{F}$ into an elastic $\boldsymbol{F}_e$ and a plastic part $\boldsymbol{F}_p$ as

$$\boldsymbol{F} = \boldsymbol{F}_e\boldsymbol{F}_p \quad . \tag{2.63}$$

In doing so, an isotropic and compressible neo-Hookean model is applied in order to describe the elastic part of the deformation. The elastic material model is based on the following strain energy density function

$$W_e = W_e(I_1, I_3) = \frac{\lambda}{4}(I_3 - 1 - \ln I_3) + \frac{\mu}{2}(I_1 - 3 - \ln I_3) \quad , \tag{2.64}$$

where λ and μ denote the Lamé parameters. Further, I_1 and I_3 represent the first and third invariant, which can be computed with respect to the *elastic right Cauchy-Green* tensor $\boldsymbol{C}_e$ as

$$I_1 = \operatorname{tr}\boldsymbol{C}_e \quad \text{and} \quad I_3 = \det\boldsymbol{C}_e \quad , \tag{2.65}$$

or, with respect to the *elastic left Cauchy-Green* tensor $\boldsymbol{b}_e$ as

$$I_1 = \operatorname{tr}\boldsymbol{b}_e \quad \text{and} \quad I_3 = \det\boldsymbol{b}_e \quad . \tag{2.66}$$

Utilizing the multiplicative decomposition of the deformation gradient in Eq. (2.63), the *elastic right Cauchy-Green* tensor $\boldsymbol{C}_e$ can then be computed as

$$\boldsymbol{C}_e = \boldsymbol{F}_e^T\boldsymbol{F}_e = \left(\boldsymbol{F}\boldsymbol{F}_p^{-1}\right)^T\left(\boldsymbol{F}\boldsymbol{F}_p^{-1}\right) = \boldsymbol{F}_p^{-T}\boldsymbol{F}^T\boldsymbol{F}\boldsymbol{F}_p^{-1} = \boldsymbol{F}_p^{-T}\boldsymbol{C}\boldsymbol{F}_p^{-1} \tag{2.67}$$

and the *elastic left Cauchy-Green* tensor $\boldsymbol{b}_e$ as

$$\boldsymbol{b}_e = \boldsymbol{F}_e\boldsymbol{F}_e^T = \left(\boldsymbol{F}\boldsymbol{F}_p^{-1}\right)\left(\boldsymbol{F}\boldsymbol{F}_p^{-1}\right)^T = \boldsymbol{F}\boldsymbol{F}_p^{-1}\boldsymbol{F}_p^{-T}\boldsymbol{F}^T = \boldsymbol{F}\boldsymbol{C}_p^{-1}\boldsymbol{F}^T \quad . \tag{2.68}$$

Consequently, the constitutive equations relating the different stress measures with $\boldsymbol{C}_e$ and $\boldsymbol{b}_e$ are obtained as follows

$$\boldsymbol{P} = \frac{\partial W_e}{\partial \boldsymbol{F}_e} = 2\boldsymbol{F}_e \frac{\partial W_e}{\partial \boldsymbol{C}_e} \tag{2.69}$$

$$\boldsymbol{S} = 2\boldsymbol{F}_e^{-1} \frac{\partial W_e}{\partial \boldsymbol{F}_e} = 2\frac{\partial W_e}{\partial \boldsymbol{C}_e} \tag{2.70}$$

$$\boldsymbol{\sigma} = \frac{1}{J_e}\frac{\partial W_e}{\partial \boldsymbol{F}_e}\boldsymbol{F}_e^T = \frac{2}{J_e}\boldsymbol{F}_e \frac{\partial W_e}{\partial \boldsymbol{C}_e}\boldsymbol{F}_e^T \tag{2.71}$$

$$\boldsymbol{\tau} = \frac{\partial W_e}{\partial \boldsymbol{F}_e}\boldsymbol{F}_e^T = 2\boldsymbol{b}_e \frac{\partial W_e}{\partial \boldsymbol{b}_e} \quad . \tag{2.72}$$

Further, in order to account for admissible stress states, the von Mises yield criterion is utilized

$$\Phi\left(\boldsymbol{\tau}, \bar{\alpha}\right) = \sqrt{\frac{3}{2}\boldsymbol{s} : \boldsymbol{s}} - K\left(\bar{\alpha}\right) \quad , \tag{2.73}$$

where $\boldsymbol{s}$ defines the deviatoric part of the *Kirchhoff stress* tensor $\boldsymbol{\tau}$

$$\boldsymbol{s} = \boldsymbol{\tau} - \frac{1}{3}\mathrm{tr}\,\boldsymbol{\tau}\,\boldsymbol{1} \quad . \tag{2.74}$$

Furthermore, $K\left(\bar{\alpha}\right)$ denotes the hardening curve. Here, we again assume the nonlinear and isotropic hardening function given in Eq. (2.61). In accordance with Korelc and Stupkiewicz [82], the associative flow rule describing the evolution of the plastic variables can be written in two equivalent forms as

$$\dot{\boldsymbol{C}}_p = 2\dot{\gamma}\boldsymbol{F}^T \frac{\partial \Phi\left(\boldsymbol{\tau}, \bar{\alpha}\right)}{\partial \boldsymbol{\tau}} \boldsymbol{F}^{-T}\boldsymbol{C}_p \tag{2.75}$$

and

$$\dot{\boldsymbol{C}}_p^{-1} = -2\dot{\gamma}\boldsymbol{F}^{-1} \frac{\partial \Phi\left(\boldsymbol{\tau}, \bar{\alpha}\right)}{\partial \boldsymbol{\tau}} \boldsymbol{F}\boldsymbol{C}_p^{-1} \quad , \tag{2.76}$$

where $\gamma \geq 0$ denotes the nonnegative plastic multiplier. Finally, the relation defining the evolution of the hardening variable $\bar{\alpha}$ is given as

$$\dot{\bar{\alpha}} = \dot{\gamma} \quad . \tag{2.77}$$

2.4 Strong and weak form of equilibrium

For the quasi-static analysis of a deformable body, it is necessary to solve a boundary value problem that is described by a system of partial differential equations coupling the kinematic relations, the local equilibrium conditions, and the constitutive equations of the underlying material model. However, finding an analytical solution of such boundary value problems is, in general, only possible for simple problems. Therefore, it is common to compute an approximate solution that is based on the weak (or variational) form of the equilibrium – also known as the principle of virtual work. In doing so, the strong form of the equilibrium is converted into a related weak form and, thus, the approximate solution is computed in the sense of a weighted residual whose integral over the computational domain vanishes. Moreover, the weak form holds also for the analytical solution of the problem.

2.4.1 Strong and weak form in the initial configuration

To derive the weak form, let us consider a deformable body with respect to the initial configuration, as depicted in Fig. 2.2. Thus, the boundary value problem can be described by the following set of equations

$$\operatorname{Div}\boldsymbol{P} + \rho_0 \boldsymbol{b} = \boldsymbol{0} \quad \text{on} \quad \Omega \tag{2.78}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on} \quad \Gamma_{\mathrm{D}} \tag{2.79}$$

$$\boldsymbol{P}\boldsymbol{N} = \bar{\boldsymbol{t}} \quad \text{on} \quad \Gamma_{\mathrm{N}} \quad , \tag{2.80}$$

where Eq. (2.78) defines the strong form of the equilibrium which has to be satisfied for every material point within the domain Ω. Further, the boundary value problem is complemented by the formulation of the Dirichlet boundary conditions in Eq. (2.79) and the Neumann boundary conditions in Eq. (2.80). Here, $\bar{\boldsymbol{u}}$ denote the prescribed displacements acting on the Dirichlet boundary Γ_{D}, and $\bar{\boldsymbol{t}}$ are the applied tractions acting on the Neumann boundary Γ_{N}.

As mentioned before, finding an analytical solution that satisfies the strong form and the boundary conditions is only possible for a selected set of simple problems. Due to this reason, it is common to compute an approximate solution. Usually, however, inserting the approximate solution into Eq. (2.78) results in a residual term of the equilibrium equations. Applying the weak form, the residual term is reduced to zero in a weak sense. To obtain the weak form, we multiply the strong form by a test (or trial) function $\boldsymbol{\eta}$ and integrate the resulting expression over the computational domain Ω

$$\int\limits_{\Omega} \operatorname{Div}\boldsymbol{P} \cdot \boldsymbol{\eta} \,\mathrm{d}V + \int\limits_{\Omega} \rho_0 \boldsymbol{b} \cdot \boldsymbol{\eta} \,\mathrm{d}V = 0 \quad . \tag{2.81}$$

Summarizing Eq. (2.81), an approximate solution can be computed in a weak sense by postulating that the integral of the weighted residual has to vanish where the test function $\boldsymbol{\eta}$ is interpreted as a weighting function. Further, we demand that the test function $\boldsymbol{\eta}$, also referred to as virtual displacements, vanishes at the Dirichlet boundary

$$\boldsymbol{\eta} = \{\boldsymbol{\eta} \mid \boldsymbol{\eta} = \boldsymbol{0} \quad \text{on} \quad \Gamma_{\mathrm{D}}\} \quad . \tag{2.82}$$

Now, applying integration by parts formula of the first term in Eq. (2.81) and making use of the divergence theorem results in the well-known expression of the weak form with respect to the initial configuration

$$G(\boldsymbol{\varphi}, \boldsymbol{\eta}) = \int\limits_{\Omega} \boldsymbol{P} \cdot \operatorname{Grad}\boldsymbol{\eta} \,\mathrm{d}V - \int\limits_{\Omega} \rho_0 \boldsymbol{b} \cdot \boldsymbol{\eta} \,\mathrm{d}V - \int\limits_{\Gamma_{\mathrm{N}}} \bar{\boldsymbol{t}} \cdot \boldsymbol{\eta} \,\mathrm{d}A = 0 \quad . \tag{2.83}$$

Further, the first term of the weak form $G(\boldsymbol{\varphi}, \boldsymbol{\eta})$ describes the internal and last two terms the external virtual work.

Moreover, by making use of the relation $\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S}$ we can substitute the *first Piola-Kirchhoff stress* tensor $\boldsymbol{P}$ resulting in an alternative formulation of Eq. (2.83)

$$G(\boldsymbol{\varphi}, \boldsymbol{\eta}) = \int\limits_{\Omega} \boldsymbol{S} \cdot \delta\boldsymbol{E} \,\mathrm{d}V - \int\limits_{\Omega} \rho_0 \boldsymbol{b} \cdot \boldsymbol{\eta} \,\mathrm{d}V - \int\limits_{\Gamma_{\mathrm{N}}} \bar{\boldsymbol{t}} \cdot \boldsymbol{\eta} \,\mathrm{d}A = 0 \tag{2.84}$$

given in terms of the *second Piola-Kirchhoff stress* tensor $\boldsymbol{S}$ and the variation of the *Green-Lagrange strain* tensor $\delta\boldsymbol{E}$.

2.4.2 Strong and weak form in the current configuration

The weak form with respect to the current configuration can be derived in the same manner. Considering a deformable body in the current configuration as depicted in Fig. 2.2, the formulation of boundary value problem is given by the following set of equations

$$\operatorname{div}\boldsymbol{\sigma} + \rho\boldsymbol{b} = \boldsymbol{0} \quad \text{on} \quad \varphi\left(\Omega\right) \tag{2.85}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on} \quad \varphi\left(\Gamma_{\mathrm{D}}\right) \tag{2.86}$$

$$\boldsymbol{\sigma}\boldsymbol{n} = \hat{\boldsymbol{t}} \quad \text{on} \quad \varphi\left(\Gamma_{\mathrm{N}}\right) \quad , \tag{2.87}$$

where Eq. (2.85) is the strong form of the equilibrium and Eq. (2.86) and (2.86) describe the corresponding Dirichlet and Neumann boundary conditions.

Starting from Eq. (2.85), we multiply the strong form of the equilibrium by a test function $\boldsymbol{\eta}$. Next, we convert the resulting expression into an integral equation over the domain $\varphi\left(\Omega\right)$, thus, after some manipulations we obtain the weak form of the equilibrium with respect to the current configuration

$$g\left(\boldsymbol{\varphi}, \boldsymbol{\eta}\right) = \int\limits_{\varphi(\Omega)} \boldsymbol{\sigma} \cdot \operatorname{grad}\boldsymbol{\eta}\,\mathrm{d}v - \int\limits_{\varphi(\Omega)} \rho\boldsymbol{b} \cdot \boldsymbol{\eta}\,\mathrm{d}V - \int\limits_{\varphi(\Gamma_{\mathrm{N}})} \hat{\boldsymbol{t}} \cdot \boldsymbol{\eta}\,\mathrm{d}a = 0 \quad . \tag{2.88}$$

For sake of completeness, it should be mentioned that the expression in Eq. (2.88) can be also obtained by applying a *push forward* operation of Eq. (2.83) which transforms the weak form of the initial configuration Ω into the current one $\varphi\left(\Omega\right)$.

Now, by taking advantage of the symmetry of the *Cauchy stress* tensor $\boldsymbol{\sigma}$, we can replace the spatial gradient of the test function $\boldsymbol{\eta}$ by its symmetric part

$$\nabla_x^S\boldsymbol{\eta} = \frac{1}{2}\left(\operatorname{grad}\boldsymbol{\eta} + \operatorname{grad}^T\boldsymbol{\eta}\right) \quad , \tag{2.89}$$

thus, the weak form in the current configuration simplifies to

$$g\left(\boldsymbol{\varphi}, \boldsymbol{\eta}\right) = \int\limits_{\varphi(\Omega)} \boldsymbol{\sigma} \cdot \nabla_x^S\boldsymbol{\eta}\,\mathrm{d}v - \int\limits_{\varphi(\Omega)} \rho\boldsymbol{b} \cdot \boldsymbol{\eta}\,\mathrm{d}V - \int\limits_{\varphi(\Gamma_{\mathrm{N}})} \hat{\boldsymbol{t}} \cdot \boldsymbol{\eta}\,\mathrm{d}a = 0 \quad . \tag{2.90}$$

2.5 Linearization of the weak form

For the analysis of quasi-static boundary value problems in nonlinear solid mechanics, in general, the finite element method – which is based on the weak (or variational) form – is applied to find an approximate solution. Due to its nonlinear character numerical algorithms have to be utilized to solve the weak form. Thereby, in most of the cases the Newton-Raphson method is used since it ensures a quadratic convergence rate close to a solution point. The starting point for the application of the Newton-Raphson method is based on the linearization of the weak form.

2.5.1 Linearized weak form in the initial configuration

To derive the linearized weak form, which is the basis for applying the Newton-Raphson procedure, let us consider the weak form in the initial configuration in Eq. (2.83). Next, we apply the Taylor series expansion of the weak form starting from an already known equilibrium state denoted by $\bar{\boldsymbol{\varphi}}$

$$G\left(\boldsymbol{\varphi},\boldsymbol{\eta}\right) = G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right) + \mathrm{D}G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)\cdot\Delta\boldsymbol{u} + r\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right) \quad . \tag{2.91}$$

Then, by neglecting the higher order terms denoted by $r\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)$, we obtain the linearized weak form

$$\mathrm{L}\left(G\right)_{\boldsymbol{\varphi}=\bar{\boldsymbol{\varphi}}} = G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right) + \mathrm{D}G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)\cdot\Delta\boldsymbol{u} \quad . \tag{2.92}$$

Now, for reasons of simplicity, we assume that the applied tractions $\bar{\boldsymbol{t}}$ and body forces $\boldsymbol{b}$ do not depend on the actual deformation. Therefore, the directional derivative of $G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)$ only depends on the first term of the weak form and, thus, can be computed in the direction of $\Delta\boldsymbol{u}$

$$\mathrm{D}G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)\cdot\Delta\boldsymbol{u} = \int\limits_{\Omega}\left(\mathrm{D}\boldsymbol{P}\cdot\Delta\boldsymbol{u}\right)\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V \quad . \tag{2.93}$$

Consequently, the directional derivative $\mathrm{D}G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)\cdot\Delta\boldsymbol{u}$ depends on the directional derivative of the *first Piola-Kirchhoff stress* tensor $\mathrm{D}\boldsymbol{P}\cdot\Delta\boldsymbol{u}$. Further, assuming that the *first Piola-Kirchhoff stress* tensor is a function of the deformation gradient, which in turn depends on the actual configuration, Eq. (2.93) can be recast as

$$\mathrm{D}G\left(\bar{\boldsymbol{\varphi}},\boldsymbol{\eta}\right)\cdot\Delta\boldsymbol{u} = \int\limits_{\Omega}\left(\mathrm{D}\boldsymbol{P}\left(\boldsymbol{F}\left(\bar{\boldsymbol{\varphi}}\right)\right)\cdot\Delta\boldsymbol{u}\right)\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V \quad . \tag{2.94}$$

Now, by applying the chain rule the directional derivative of the *first Piola-Kirchhoff stress* tensor in the direction of $\Delta\boldsymbol{u}$ can be computed as

$$\begin{aligned}
\mathrm{D}\boldsymbol{P}\left(\boldsymbol{F}\left(\bar{\boldsymbol{\varphi}}\right)\right)\cdot\Delta\boldsymbol{u} &= \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}}\,\frac{\mathrm{d}}{\mathrm{d}\epsilon}\left(\boldsymbol{F}\left(\bar{\boldsymbol{\varphi}}+\epsilon\Delta\boldsymbol{u}\right)\right)\bigg|_{\epsilon=0} && (2.95)\\
&= \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}}\,\frac{\mathrm{d}}{\mathrm{d}\epsilon}\left(\mathrm{Grad}\,\left(\bar{\boldsymbol{\varphi}}+\epsilon\Delta\boldsymbol{u}\right)\right)\bigg|_{\epsilon=0} && (2.96)\\
&= \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}}\,\mathrm{Grad}\,\Delta\boldsymbol{u} && (2.97)
\end{aligned}$$

The first term in Eq. (2.97) describes the *material tangent* tensor $\boldsymbol{A}$ which is obtained from the constitutive model – for hyperelastic material models $\boldsymbol{A}$ is also referred to as the *first elasticity* tensor. Thus, with the introduction of the *material tangent* tensor

$$\boldsymbol{A} = \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}} \quad , \tag{2.98}$$

finally, the linearized weak form with respect to the initial configuration can be summarized as

$$\int\limits_{\Omega}\left(\boldsymbol{A}\,\mathrm{Grad}\,\Delta\boldsymbol{u}\right)\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V = -\int\limits_{\Omega}\boldsymbol{P}\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V + \int\limits_{\Omega}\rho_0\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}V + \int\limits_{\Gamma_{\mathrm{N}}}\bar{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}A \quad . \tag{2.99}$$

Finally, it should be pointed out that the linearized weak form in the initial configuration can be also formulated in terms of the *second Piola-Kirchhoff stress* tensor $\boldsymbol{S}$ and the variation of the *Green-Lagrange strain* tensor $\delta\boldsymbol{E}$. Both formulations are equivalent.

2.5.2 Linearized weak form in the current configuration

The linearization of the weak form can be also formulated with respect to the current configuration. To this end, we apply a *push forward* operation of Eq. (2.99) resulting in the following expression

$$\begin{aligned}\int\limits_{\varphi(\Omega)} \frac{1}{J}\left(\boldsymbol{A}\,\mathrm{grad}\,\Delta\boldsymbol{u}\,\boldsymbol{F}\right)\cdot \mathrm{grad}\,\boldsymbol{\eta}\,\boldsymbol{F}\,\mathrm{d}v = &-\int\limits_{\varphi(\Omega)} \frac{1}{J}\boldsymbol{P}\cdot \mathrm{grad}\,\boldsymbol{\eta}\,\boldsymbol{F}\,\mathrm{d}v + \int\limits_{\varphi(\Omega)} \rho_0\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}v \\ &+\int\limits_{\varphi(\Gamma_\mathrm{N})} \hat{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}a \quad .\end{aligned} \tag{2.100}$$

Substituting the *first Piola-Kirchhoff stress* tensor by $\boldsymbol{P} = J\boldsymbol{\sigma}\boldsymbol{F}^{-T}$ and by taking advantage of the symmetry of the *Cauchy stress* tensor $\boldsymbol{\sigma}$ yields

$$\begin{aligned}\int\limits_{\varphi(\Omega)} \frac{1}{J}\left(\boldsymbol{A}\,\mathrm{grad}\,\Delta\boldsymbol{u}\,\boldsymbol{F}\right)\cdot \mathrm{grad}\,\boldsymbol{\eta}\,\boldsymbol{F}\,\mathrm{d}v = &-\int\limits_{\varphi(\Omega)} \boldsymbol{\sigma}\boldsymbol{F}^{-T}\cdot \nabla_x^S\boldsymbol{\eta}\,\boldsymbol{F}\,\mathrm{d}v + \int\limits_{\varphi(\Omega)} \rho\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}v \\ &+\int\limits_{\varphi(\Gamma_\mathrm{N})} \hat{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}a \quad ,\end{aligned} \tag{2.101}$$

where $\nabla_x^S\boldsymbol{\eta}$ denotes the symmetric part of the gradient of the test function, see Eq. (2.89). Due to the symmetry of $\nabla_x^S\boldsymbol{\eta}$, the following expression in Eq. (2.101) can be recast

$$\boldsymbol{\sigma}\boldsymbol{F}^{-T}\cdot\nabla_x^S\boldsymbol{\eta}\,\boldsymbol{F} = \boldsymbol{\sigma}\cdot\nabla_x^S\boldsymbol{\eta}\,\boldsymbol{F}^{-1}\boldsymbol{F} = \boldsymbol{\sigma}\cdot\nabla_x^S\boldsymbol{\eta} \quad . \tag{2.102}$$

Finally, using the relation from Eq. (2.102) and introducing the *spatial tangent* tensor $\boldsymbol{a}$ the linearized weak form in the current configuration leads to

$$\int\limits_{\varphi(\Omega)} \left(\boldsymbol{a}\,\mathrm{grad}\,\Delta\boldsymbol{u}\right)\cdot\mathrm{grad}\,\boldsymbol{\eta}\,\mathrm{d}v = -\int\limits_{\varphi(\Omega)} \boldsymbol{\sigma}\cdot\nabla_x^S\boldsymbol{\eta}\,\mathrm{d}v + \int\limits_{\varphi(\Omega)} \rho\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}v + \int\limits_{\varphi(\Gamma_\mathrm{N})} \hat{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}a \quad . \tag{2.103}$$

According to Neto et al. [78], the relation between the *material* and *spatial tangent* tensor is given as

$$a_{ijkl} = \frac{1}{J}A_{imkn}F_{jm}F_{ln} \quad . \tag{2.104}$$

3 The finite cell method

This chapter is intended to give a brief overview of the *finite cell method* (FCM) that is based on a combination of the fictitious domain approach and shape functions of higher order [19–22]. In Sec. 3.1, we therefore present the essential idea of the *fictitious domain approach* which provides an efficient and simple mesh generation strategy. Thereby, we show the formulation of the weak form and its linearization accounting for the fictitious domain. Next, in Sec. 3.2, the spatial discretization is explained. Here, the mapping relations based on rectangular cells are given, as well as the derivation of the discrete equation system. Sec. 3.3 focuses on the numerical integration process of the FCM. Here, we describe the Gaussian quadrature method as well as an adaptive Gaussian quadrature scheme utilizing a spacetree decomposition in order to resolve for the integration domain.

3.1 Fictitious domain approach

In the last decade, the interest in *fictitious domain methods* – such as the *finite cell method* [19–22], *CutFEM* [16, 17, 85], or *CutIGA* [18] – has increased significantly in the field of structural mechanics. This is due to the reason that these methods provide a simple and efficient mesh generation strategy for boundary value problems that possess a complex geometry – in contrast to body-conforming discretization methods like the classical *finite element method* (FEM).

In the following, we briefly describe the essential idea of the fictitious domain approach and its implementation within the context of the finite cell method. To this end, let us take a look to the boundary value problem depicted in Fig. 3.1. Here, we consider a body Ω that is subjected to Dirichlet boundary conditions $\bar{\boldsymbol{u}}$ on Γ_D and Neumann boundary conditions $\bar{\boldsymbol{t}}$ on Γ_N. Now, following the fictitious domain approach, the body Ω is immersed or embedded into a fictitious domain $\Omega_e \backslash \Omega$, which is why fictitious domain methods are also often referred to as *immersed* or *embedded domain methods*. In doing so, the resulting extended domain Ω_e is composed of a simple shape that can be easily discretized by employing structured meshes or Cartesian grids, for instance. To distinguish the boundary-nonconforming elements from standard boundary-conforming ones in the context of the FCM, we call these elements finite cells. Consequently, the underlying meshing strategy of the FCM allows to decouple the approximation of the primary variables, i.e. displacements, from the approximation of the geometry. Finally, in order to resolve for the domain of the body, the indicator function α is introduced – which is one for material points within the physical domain Ω and zero for points within the fictitious one $\Omega_e \backslash \Omega$

$$\alpha = \alpha\left(\boldsymbol{X}\right) = \begin{cases} 1 & \forall \boldsymbol{X} \in \Omega \\ 0 & \forall \boldsymbol{X} \in \Omega_e \backslash \Omega \quad . \end{cases} \tag{3.1}$$

Consequently, the FCM provides a simple mesh generation and analysis for a wide variety of geometric models such as voxel-based models provided by quantitative computer

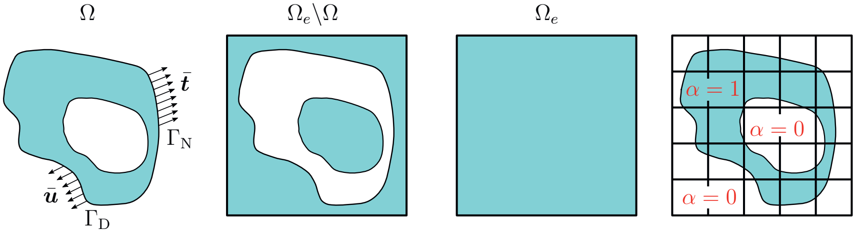


Figure 3.1: Basic concept of fictitious domain methods.

tomography scans (qCT-scans), B-rep models obtained from commercial CAD-programs, or level set representations. Moreover, note that finite cells which are completely located within the fictitious domain may be neglected for the analysis.

3.1.1 Weak forms

Next, to find an approximate solution of the boundary value problem using the finite cell method, we have to reformulate the weak form of equilibrium in Eq. (2.83) and Eq. (2.88) with respect to the extended domain Ω_e. Taking advantage of the indicator function α the weak form of the initial configuration can then be written as

$$G_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta}) = \int\limits_{\Omega_e} \alpha\,\boldsymbol{P}\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V - \int\limits_{\Omega_e} \alpha\,\rho_0\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}V - \int\limits_{\Gamma_\mathrm{N}} \bar{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}A = 0 \quad . \tag{3.2}$$

Further, starting with the relation in Eq. (2.88), we can derive the weak form of equilibrium with respect to the current configuration in an analogous manner

$$g_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta}) = \int\limits_{\varphi(\Omega_e)} \alpha\,\boldsymbol{\sigma}\cdot\mathrm{grad}\,\boldsymbol{\eta}\,\mathrm{d}v - \int\limits_{\varphi(\Omega_e)} \alpha\,\rho\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}V - \int\limits_{\varphi(\Gamma_\mathrm{N})} \hat{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}a = 0 \quad , \tag{3.3}$$

where $\varphi(\Omega_e)$ characterizes the current configuration of the extended domain. Consequently, the apparent difference of $G_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta})$ and $g_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta})$ with the weak forms $G(\boldsymbol{\varphi},\boldsymbol{\eta})$ and $g(\boldsymbol{\varphi},\boldsymbol{\eta})$ is characterized by the terms concerning the volume integrals. Note that – based on the definition of the indicator function α, which describes a discontinuous function – equivalence between the weak forms of the extended domain and the weak forms of the physical domain is ensured, see Dauge et al. [21].

3.1.2 Linearized weak forms

Since we are focusing on nonlinear problems in solid mechanics, approximation procedures have to be applied in order to solve the weak form of equilibrium. As stated in Sec. 2.4, we employ the Newton-Raphson method that is based on the linearization of the weak form. Then, using the relations of Eq. (2.94), (2.97), and (2.98) – which are derived assuming conservative loads that do not depend on the actual deformation – the directional derivative of $G_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta})$ in the direction of the displacement increment $\Delta\boldsymbol{u}$ reads

$$\mathrm{D}G_e^\alpha(\boldsymbol{\varphi},\boldsymbol{\eta})\cdot\Delta\boldsymbol{u} = \int\limits_{\Omega_e} \alpha\,(\boldsymbol{A}\,\mathrm{Grad}\,\Delta\boldsymbol{u})\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V \quad . \tag{3.4}$$

Thus, with the definition in Eq. (2.92), finally, the linearized weak form of equilibrium with respect to the initial configuration can be written as

$$\int_{\Omega_e} \alpha\left(\boldsymbol{A}\,\mathrm{Grad}\,\Delta\boldsymbol{u}\right)\cdot\mathrm{Grad}\,\boldsymbol{\eta}\,\mathrm{d}V = \int_{\Omega_e} \alpha\,\left(\rho_0\boldsymbol{b}\cdot\boldsymbol{\eta} - \boldsymbol{P}\cdot\mathrm{Grad}\,\boldsymbol{\eta}\right)\,\mathrm{d}V + \int_{\Gamma_\mathrm{N}} \bar{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}A \quad . \tag{3.5}$$

Moreover, to derive the linearized weak form in the current configuration, we apply a *push forward* operation of Eq. (3.5). Then, after some reformulations and by taking advantage of the symmetry of the *Cauchy stress* tensor, we obtain

$$\int_{\varphi(\Omega_e)} \alpha\left(\boldsymbol{a}\,\mathrm{grad}\,\Delta\boldsymbol{u}\right)\cdot\mathrm{grad}\,\boldsymbol{\eta}\,\mathrm{d}v = \int_{\varphi(\Omega_e)} \alpha\,\left(\rho\boldsymbol{b}\cdot\boldsymbol{\eta}\,\mathrm{d}v - \boldsymbol{\sigma}\cdot\nabla_x^S\boldsymbol{\eta}\right)\,\mathrm{d}v + \int_{\varphi(\Gamma_\mathrm{N})} \hat{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,\mathrm{d}a \quad . \tag{3.6}$$

3.2 Spatial discretization

In general, solving boundary value problems in solid mechanics is not straightforward. Due to this reason, an efficient approach is to spatially discretize the domain of the problem and then to find an approximate solution based on the chosen discretization. Applying the FCM, the spatial discretization is carried out by subdividing the undeformed extended domain into a set of n_c nonoverlapping finite cells Ω^c

$$\Omega = \bigcup_{c=1}^{n_c} \Omega^c \quad , \tag{3.7}$$

where each cell c incorporates a set of shape functions used for the local approximation of the unknown solution.

3.2.1 Mapping

In the context of the FCM, it is common to employ structured meshes or Cartesian grids as shown in Fig. 3.2. In the figure, Ω^c describes the initial and $\varphi\left(\Omega^c\right)$ the current configuration of a finite cell c. Further, $\Omega^c_\square$ characterizes the domain of the *parent cell* which is defined by the *local coordinates* $\boldsymbol{\xi}$. Furthermore, $\boldsymbol{Q}^c$ defines the mapping from the local to the initial coordinates

$$\boldsymbol{X}^c = \boldsymbol{X}^c\left(\boldsymbol{\xi}\right) = \boldsymbol{Q}^c\left(\boldsymbol{\xi}\right) = \boldsymbol{Q}^c \tag{3.8}$$

and $\boldsymbol{q}^c$ is the mapping from the local to the current coordinates

$$\boldsymbol{x}^c = \boldsymbol{x}^c\left(\boldsymbol{\xi}\right) = \boldsymbol{q}^c\left(\boldsymbol{\xi}\right) = \boldsymbol{q}^c \quad . \tag{3.9}$$

Moreover, the gradients of the mappings are given as

$$\boldsymbol{J}^c = \mathrm{Grad}_\xi\,\boldsymbol{X}^c = \frac{\partial\boldsymbol{X}^c}{\partial\boldsymbol{\xi}} \tag{3.10}$$

and

$$\boldsymbol{j}^c = \mathrm{Grad}_\xi\,\boldsymbol{x}^c = \frac{\partial\boldsymbol{x}^c}{\partial\boldsymbol{\xi}} \quad , \tag{3.11}$$

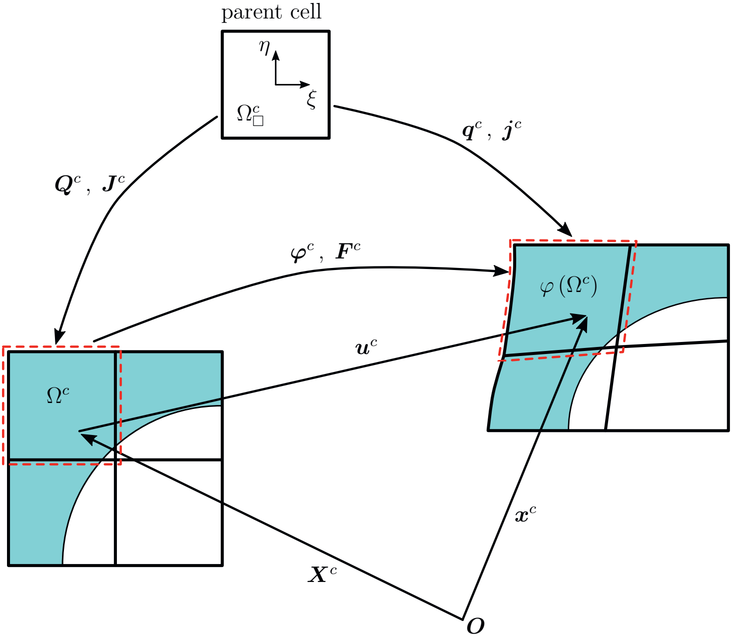


Figure 3.2: Spatial discretization of the extended domain.

respectively. Thus, with the definition of the deformation gradient $\boldsymbol{F} = \partial \boldsymbol{x} / \partial \boldsymbol{X}$, we can deduce the following kinematic relation between $\boldsymbol{j}^c$ and $\boldsymbol{J}^c$

$$\boldsymbol{j}^c = \frac{\partial \boldsymbol{x}^c}{\partial \boldsymbol{\xi}} = \frac{\partial \boldsymbol{x}^c}{\partial \boldsymbol{X}^c} \frac{\partial \boldsymbol{X}^c}{\partial \boldsymbol{\xi}} = \boldsymbol{F}^c \boldsymbol{J}^c \quad . \tag{3.12}$$

Since, we employ rectangular cells for the spatial discretization, the kinematic relations are simplified significantly, as demonstrated in the following. For reasons of convenience, the subsequent formulations are stated in the standard matrix notation. Therefore, in order to distinguish between matrix and tensor quantities, matrices are denoted by upright bold-faced symbols. In doing so, the mapping of a rectangular and hexahedral finite cell from the local to the initial coordinates reads

$$\mathbf{X}^c = \mathbf{Q}^c = \begin{bmatrix} X_{\min} + 1/2\,(1+\xi)\,H_X \\ Y_{\min} + 1/2\,(1+\eta)\,H_Y \\ Z_{\min} + 1/2\,(1+\zeta)\,H_Z \end{bmatrix} \quad , \tag{3.13}$$

where $X_{\min}$, $Y_{\min}$, and $Z_{\min}$ denote the minimum coordinates of the cell and H_X, H_Y, and H_Z are the lengths. Then, the derivative of the initial coordinates with respect to the local coordinates is given by the following Jacobi matrix

$$\mathbf{J}^c = \begin{bmatrix} X_{,\xi} & X_{,\eta} & X_{,\zeta} \\ Y_{,\xi} & Y_{,\eta} & Y_{,\zeta} \\ Z_{,\xi} & Z_{,\eta} & Z_{,\zeta} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} H_X & 0 & 0 \\ 0 & H_Y & 0 \\ 0 & 0 & H_Z \end{bmatrix} \quad . \tag{3.14}$$

Note that the Jacobi matrix $\mathbf{J}^c$ represents a constant matrix. Consequently, the determinant of $\mathbf{J}^c$ is also a constant

$$\det \mathbf{J}^c = \frac{1}{2} H_X H_Y H_Z \quad . \tag{3.15}$$

Moreover, with the relation given in Eq. (3.12), the Jacobi matrix $\mathbf{j}^c$ of the current configuration can be computed as

$$\mathbf{j}^c = \begin{bmatrix} x_{,\xi} & x_{,\eta} & x_{,\zeta} \\ y_{,\xi} & y_{,\eta} & y_{,\zeta} \\ z_{,\xi} & z_{,\eta} & z_{,\zeta} \end{bmatrix} = \mathbf{F}^c\,\mathbf{J}^c = \frac{1}{2} \begin{bmatrix} F_{11} H_X & F_{12} H_Y & F_{13} H_Z \\ F_{21} H_X & F_{22} H_Y & F_{23} H_Z \\ F_{31} H_X & F_{32} H_Y & F_{33} H_Z \end{bmatrix} \quad , \tag{3.16}$$

thus, its determinant follows as

$$\det \mathbf{j}^c = \det \mathbf{F}^c \det \mathbf{J}^c = \det \mathbf{F}^c \frac{1}{2} H_X H_Y H_Z \quad . \tag{3.17}$$

3.2.2 Discretization of the weak forms

Next, we define an ansatz on each finite cell that is used for the discretization of the displacement (trial) and virtual displacement (test) function. For reasons of convenience, the following formulations are given in standard matrix notation. In doing so, the displacement or trial function $\mathbf{u}$ is approximated within each finite cell c as

$$\mathbf{u} \approx \mathbf{u}^c = \mathbf{N} \mathbf{U}^c \quad \text{in} \quad \Omega^c \quad . \tag{3.18}$$

Following the Bubnov-Galerkin method, the same ansatz is used for the approximation of the virtual displacement or test function

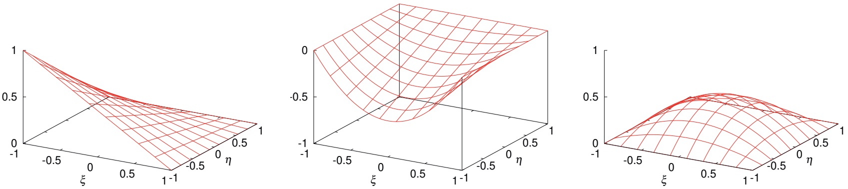
$$\boldsymbol{\eta} \approx \boldsymbol{\eta}^c = \mathbf{N}\mathbf{V}^c \quad \text{in} \quad \Omega^c \quad . \tag{3.19}$$

In Eq. (3.18) and (3.19), $\mathbf{N}$ denotes the matrix including the individual shape functions. Further, $\mathbf{U}^c$ is the displacement vector comprising the unknown displacement values and $\mathbf{V}^c$ is the virtual displacement vector containing the arbitrary virtual displacements. Considering the three-dimensional case, the matrix vector notation of the displacement $\mathbf{u}^c$ in Eq. (3.18) can be written as

$$\mathbf{u}^c = \mathbf{N}\mathbf{U}^c = \begin{bmatrix} N_1 & 0 & 0 & \dots & N_n & 0 & 0 \\ 0 & N_1 & 0 & \dots & 0 & N_n & 0 \\ 0 & 0 & N_1 & \dots & 0 & 0 & N_n \end{bmatrix} \begin{bmatrix} U_1 \\ V_1 \\ W_1 \\ \vdots \\ U_n \\ V_n \\ W_n \end{bmatrix} \quad , \tag{3.20}$$

where N_i denote the n individual shape functions and U_i, V_i, and W_i define the $3 \times n$ associated unknown displacements (with $i = 1, ..., n$). For the choice of N_i, it is possible to employ different shape functions such as Lagrange or Bernstein polynomials, B-splines, or NURBS – just to name a few. In this work, we employ hierarchic shape functions which provide high convergence rates for smooth problems of the FCM. For the *two-dimensional* case, the hierarchic shape functions are categorized into three groups: nodal, edge, and internal modes. A representative shape function of each group is plotted in Fig. 3.3. The extension of the hierarchic shape functions for the three-dimensional case is straightforward. Here, the shape functions are categorized into four groups: nodal, edge, face, and internal modes. A detailed description of the hierarchic shape functions is given in the textbook by Szabó and Babuška [86] or in the contribution by Szabó et al. [87].

In the following, we show how to derive the discrete equation system that needs to be solved. As mentioned before, we employ the Newton-Raphson method to find an approximate solution for the weak form of equilibrium. Then, inserting the ansatz of the displacement and the test function from Eq. (3.18) and (3.19) into the linearized weak form in Eq. (3.5) results in the following linear equation system that needs to be solved



(a) Nodal mode. **(b)** Edge mode. **(c)** Internal mode.

Figure 3.3: Hierarchic shape functions of a quadrilateral cell given in the parent domain.

within each Newton-Raphson iteration i

$$\mathbf{K}_T^i\left(\mathbf{U}^i\right)\,\Delta\mathbf{U}^{i+1} = \mathbf{F}_{\text{int}}^i\left(\mathbf{U}^i\right) - \bar{\lambda}\,\mathbf{F}_{\text{ext}} \quad . \tag{3.21}$$

Here, $\mathbf{K}_T^i\left(\mathbf{U}_k^i\right)$ denotes the *tangential stiffness* matrix, and $\mathbf{U}^i$ as well as $\Delta\mathbf{U}^{i+1}$ are the *displacement* and *displacement increment* vector, respectively. Further, $\mathbf{F}_{\text{int}}^i\left(\mathbf{U}^i\right)$ describes the *internal* and $\mathbf{F}_{\text{ext}}$ the *external load* vector. Furthermore, $\bar{\lambda}$ defines the accumulated load factor that is used to apply the load in an increment-wise manner. Moreover, the tangential stiffness matrix and the internal load vector are computed based on the displacement solution of the previous iteration.

Finally, since the support of the shape functions N_i is defined by the domain of the finite cells Ω^c, the global quantities in Eq. (3.21) are obtained during the *assembling process*

$$\mathbf{K}_T^i = \overset{n_c}{\underset{c=1}{\mathsf{A}}}\,\mathbf{k}^{i,c} \quad , \quad \mathbf{F}_{\text{int}}^i = \overset{n_c}{\underset{c=1}{\mathsf{A}}}\,\mathbf{f}_{\text{int}}^{i,c} \quad \text{and} \quad \mathbf{F}_{\text{ext}} = \overset{n_c}{\underset{c=1}{\mathsf{A}}}\,\mathbf{f}_{\text{ext}}^{c} \quad , \tag{3.22}$$

where A denotes the *assembling operator*. Thereby, for each finite cell c, the local stiffness matrix $\mathbf{k}^{i,c}$ is calculated as

$$\mathbf{k}^{i,c} = \int_{\Omega^c} \alpha\,\mathbf{G}^T\mathbf{A}\mathbf{G}\,\mathrm{d}V \quad , \tag{3.23}$$

the local internal load vector $\mathbf{f}_{\text{int}}^{i,c}$ as

$$\mathbf{f}_{\text{int}}^{i,c} = \int_{\Omega^c} \alpha\,\mathbf{G}^T\,\mathbf{P}\,\mathrm{d}V \quad , \tag{3.24}$$

and the local external load vector $\mathbf{f}_{\text{ext}}^c$ is computed as

$$\mathbf{f}_{\text{ext}}^c = \int_{\Omega^c} \alpha\,\rho_0\mathbf{N}^T\mathbf{b}\,\mathrm{d}V + \int_{\Gamma_{\text{N}}{}^c} \mathbf{N}^T\,\bar{\mathbf{t}}\,\mathrm{d}A \quad . \tag{3.25}$$

Note that these are the local cell quantities with respect to the initial configuration. Thereby, $\mathbf{G}$ denotes the *discrete gradient operator* which contains the derivatives of the individual shape functions with respect to the coordinates X, Y, and Z of the initial configuration

$$\mathbf{G} = \begin{bmatrix} N_{1,X} & 0 & 0 & \dots & N_{n,X} & 0 & 0 \\ N_{1,Y} & 0 & 0 & \dots & N_{n,Y} & 0 & 0 \\ N_{1,Z} & 0 & 0 & \dots & N_{n,Z} & 0 & 0 \\ 0 & N_{1,X} & 0 & \dots & 0 & N_{n,X} & 0 \\ 0 & N_{1,Y} & 0 & \dots & 0 & N_{n,Y} & 0 \\ 0 & N_{1,Z} & 0 & \dots & 0 & N_{n,Z} & 0 \\ 0 & 0 & N_{1,X} & \dots & 0 & 0 & N_{n,X} \\ 0 & 0 & N_{1,Y} & \dots & 0 & 0 & N_{n,Y} \\ 0 & 0 & N_{1,Z} & \dots & 0 & 0 & N_{n,Z} \end{bmatrix} \quad . \tag{3.26}$$

The discretization of the linearized weak form of the current configuration can be obtained in a similar manner or by applying corresponding *push forward* operations. Here, the interested reader is referred to the thesis of Gnegel [40].

3.3 Numerical integration

When it comes to computing the local cell quantities, special care has to be taken during the numerical integration process. Thereby, we have to distinguish between cut and noncut finite cells. For the numerical integration of the noncut cells, we employ Gaussian quadrature rules which define a widely used standard within the context of the finite element method. However, Gaussian quadratures do not perform well for the computation of the discontinuous integrals of the cut cells induced by the indicator function α. Due to this reason, the numerical integration of the cut cells is carried out, usually, applying adaptive Gaussian quadrature schemes.

3.3.1 Gaussian quadrature

Before we introduce the Gaussian quadrature method, let us start with the basic idea of the numerical computation of integrals considering the one-dimensional case – for a more detailed overview the interested reader is referred to the textbooks of Schwarz and Köckler [88] or Schaback and Wendland [89]. The following formulas are stated with respect to the local coordinate ξ on finite or infinite interval (a, b). In doing so, the integral of a function $g(\xi)$ is approximated by a weighted sum

$$\int_a^b g(\xi)\,\mathrm{d}\xi \approx \sum_{i=1}^{n} \lambda_i g(\xi_i) \quad \text{with} \quad a \leq \xi_1 < \ldots < \xi_n \leq b \quad . \tag{3.27}$$

In Eq. (3.27), n denotes the number of integration or quadrature points, ξ_i define the position of the integration points, and λ_i are the corresponding weights. The set of ξ_i and λ_i is also referred to as quadrature (or integration) points and weights. Now, for the numerical integration of polynomials, the maximum quadrature order p_q is limited by $p_q = 2n - 1$. This relation can be deduced by the following example. Suppose we want to compute the integral of a polynomial $p_{2n}(\xi)$ of order $2n$

$$p_{2n}(\xi) = \prod_{i=1}^{n} (\xi - \xi_i)^2 \quad \text{with} \quad a \leq \xi_1 < \ldots < \xi_n \leq b \tag{3.28}$$

where the roots of $p_{2n}(\xi)$ coincide with the position of the integration points. From the structure of $p_{2n}(\xi)$ one can easily deduce the relation

$$\int_a^b p_{2n}(\xi)\,\mathrm{d}\xi > 0 \quad . \tag{3.29}$$

However, the weighted sum of the quadrature yields

$$\sum_{i=1}^{n} \lambda_i\, p_{2n}(\xi_i) = 0 \quad . \tag{3.30}$$

Consequently, we obtain the following inequality

$$\int_a^b p_{2n}(\xi)\,\mathrm{d}\xi \neq \sum_{i=1}^{n} \lambda_i\, p_{2n}(\xi_i) \tag{3.31}$$

from which we can conclude that the maximal possible quadrature order is limited by $p_q = 2n - 1$.

Optimal points and weights are obtained by the Gaussian quadrature method, to be explained in more detail in the following. To this end, let us start with the more general form of the Gaussian approach where the integral of a function $g(\xi)$ is replaced by an equivalent weighted integral as

$$\int_a^b g(\xi)\,\mathrm{d}\xi = \int_a^b f(\xi) w(\xi)\,\mathrm{d}\xi \quad \text{with} \quad w(\xi) > 0 \quad \forall \xi \in (a,b) \quad . \tag{3.32}$$

In Eq. (3.32), $w(\xi)$ defines a positive weight function on interval (a, b). In accordance with Eq. (3.27), the integral is then approximated by a weighted sum as

$$\int_a^b f(\xi) w(\xi)\,\mathrm{d}\xi \approx \sum_{i=1}^{n} \lambda_i f(\xi_i) w(\xi_i) \quad \text{with} \quad a \leq \xi_1 < \ldots < \xi_n \leq b \quad . \tag{3.33}$$

Consequently, we have $2n$ unknowns: n unknown abscissa values ξ_i and n unknown weights λ_i. There are several possible approaches to determine the $2n$ unknowns. For example, one could solve a nonlinear equation system – but a technically more elegant approach is presented in this contribution, based on the usage of orthogonal polynomials. To this end, let us assume that $f(\xi) = f_{2n-1}(\xi)$ defines a polynomial function of order $2n - 1$. Further, we formulate a polynomial $p_n(\xi)$

$$p_n(\xi) = \prod_{i=1}^{n} (\xi - \xi_i) \tag{3.34}$$

where the roots of $p_n(\xi)$ coincide with the distinct abscissa values ξ_i. Then, applying Euclidean division of polynomials for $f_{2n-1}(\xi)$ and $p_n(\xi)$, we obtain the following relation

$$f_{2n-1}(\xi) = p_n(\xi) q_{n-1}(\xi) + r_{n-1}(\xi) \quad . \tag{3.35}$$

where quotient q_{n-1} and remainder r_{n-1} define polynomials of order $\leq (n - 1)$. Now, inserting Eq. (3.35) in Eq. (3.33) yields

$$\int_a^b f_{2n-1}(\xi) w(\xi)\,\mathrm{d}\xi = \int_a^b p_n(\xi) q_{n-1}(\xi) w(\xi) + r_{n-1}(\xi) w(\xi)\,\mathrm{d}\xi \quad . \tag{3.36}$$

Further, assuming orthogonality between $p_n(\xi)$ and any polynomial of order less than n on interval (a, b) with respect to the weight function $w(\xi)$ results in

$$\int_a^b p_n(\xi) q_{n-1}(\xi) w(\xi)\,\mathrm{d}\xi = 0 \tag{3.37}$$

and, finally, Eq. (3.36) simplifies to

$$\int_a^b f_{2n-1}(\xi) w(\xi)\,\mathrm{d}\xi = \int_a^b r_{n-1}(\xi) w(\xi)\,\mathrm{d}\xi \quad . \tag{3.38}$$

Next, let us have a look at the right-hand side of Eq. (3.33). Then, based on the assumption that ξ_i define the n distinct roots of $p_n(\xi)$, the weighted sum simplifies to

$$\sum_{i=1}^{n} \lambda_i f(\xi_i) = \sum_{i=1}^{n} \lambda_i \, p_n(\xi_i) q_{n-1}(\xi_i) + \lambda_i \, r_{n-1}(\xi_i) = \sum_{i=1}^{n} \lambda_i \, r_{n-1}(\xi_i) \quad , \tag{3.39}$$

Consequently, with Eq. (3.38) and (3.39), we can derive the following relation

$$\int_a^b f_{2n-1}(\xi) w(\xi) \, \mathrm{d}\xi = \int_a^b r_{n-1}(\xi) w(\xi) \, \mathrm{d}\xi \approx \sum_{i=1}^{n} \lambda_i \, r_{n-1}(\xi_i) \quad . \tag{3.40}$$

Thus, the only thing left to be done is to compute the n unknown weights λ_i. However, this can easily be carried out by employing the Lagrange polynomials through points ξ_i to interpolate the remainder $r_{n-1}(\xi)$

$$r_{n-1}(\xi) = \sum_{i=1}^{n} r_{n-1}(\xi_i) l_i(\xi) \quad . \tag{3.41}$$

Thus, inserting Eq. (3.41) in Eq. (3.40), finally, the weights λ_i can be computed as

$$\lambda_i = \int_a^b l_i(\xi) w(\xi) \, \mathrm{d}\xi \quad . \tag{3.42}$$

Quadrature formulas that follow this approach are denoted as Gaussian quadrature formulas, e.g. Gauss-Legendre, Gauss-Laguerre, Gauss-Hermite, Gauss-Chebyshev, or Gauss-Jacobi quadrature. From among these formulas, the most commonly used one is the Gauss-Legendre quadrature, which is based on Legendre polynomials $L_i(\xi)$ and a positive weight function $w(\xi) = 1$ on interval $(-1, 1)$.

For the numerical integration of the local cell quantities of noncut cells, we employ the Gauss-Legendre quadrature. In doing so, a multidimensional quadrature rule is obtained by simply applying a tensor product of the one-dimensional quadrature points. In the following, this is demonstrated for the computation of the local stiffness matrix $\mathbf{k}^c$. To this end, we start off by performing a change of variables from initial (X, Y, Z) to local coordinates (ξ, η, ζ)

$$\mathbf{k}^c = \int_{\Omega^c} \alpha \, \mathbf{G}^T \mathbf{A} \mathbf{G} \, \mathrm{d}V = \int_{\Omega^c} \mathbf{G}^T \mathbf{A} \mathbf{G} \, \mathrm{d}V = \int_{\Omega^c_\square} \alpha \, \mathbf{G}^T \mathbf{A} \mathbf{G} \, \mathrm{det} \mathbf{J}^c \, \mathrm{d}\Omega_\square \tag{3.43}$$

$$= \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \mathbf{G}^T \mathbf{A} \mathbf{G} \, \mathrm{det} \mathbf{J}^c \, \mathrm{d}\xi \mathrm{d}\eta \mathrm{d}\zeta \quad , \tag{3.44}$$

where $\mathrm{det}\mathbf{J}^c$ defines the determinant of the Jacobian matrix which is constant, see Eq. (3.15). Note that, since the cell is not cut, the value of the indicator function α is one. Consequently, the entries of $\mathbf{k}^c$ are computed as

$$\mathbf{k}^c = \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \mathbf{G}^T \mathbf{A} \mathbf{G} \, \mathrm{det} \mathbf{J}^c \, \mathrm{d}\xi \mathrm{d}\eta \mathrm{d}\zeta \approx \sum_{r=1}^{n} \sum_{s=1}^{n} \sum_{t=1}^{n} \mathbf{G}^T_{\boldsymbol{\xi}_i} \mathbf{A}_{\boldsymbol{\xi}_i} \mathbf{G}_{\boldsymbol{\xi}_i} \, \mathrm{det} \mathbf{J}^c \, \lambda_r \lambda_s \lambda_t \tag{3.45}$$

$$= \sum_{i=1}^{n_g} \mathbf{G}^T_{\boldsymbol{\xi}_i} \mathbf{A}_{\boldsymbol{\xi}_i} \mathbf{G}_{\boldsymbol{\xi}_i} \, \mathrm{det} \mathbf{J}^c \, \lambda_i \tag{3.46}$$

with

$$\boldsymbol{\xi}_i = \begin{bmatrix}\xi_r & \zeta_s & \eta_t\end{bmatrix}^T \quad \text{and} \quad \lambda_i = \lambda_r \lambda_s \lambda_t \tag{3.47}$$

denoting the position and the weights of the n_g quadrature points. Moreover, expressions of the following form $\mathbf{G}_{\boldsymbol{\xi}_i}$ represent the value of a quantity at integration point $\boldsymbol{\xi}_i$. In this case, the value of the *discrete gradient operator* $\mathbf{G}$ at quadrature point $\boldsymbol{\xi}_i$.

3.3.2 Adaptive Gaussian quadrature scheme

In order to compute the integrals of the cut cells, special care has to be taken during the numerical integration process. This is due to the fact that the indicator function α transforms the continuous integrals into discontinuous integrals. However, standard Gaussian quadrature rules, introduced in the previous section, show a weak performance facing such kind of integrals. Due to this reason, it is common to apply adaptive schemes for the numerical integration of cut finite cells. These schemes are based on spacetree decomposition to resolve for the physical and the fictitious subdomain, respectively. In the context of the FCM, the subdomains are usually resolved by a quadtree (2D) or an octree (3D) subdivision. In the following, we briefly explain the basic idea of the adaptive integration method.

To this end, let us consider the FCM problem depicted in Fig. 3.4 with respect to the initial configuration. For reasons of simplicity, we consider the two-dimensional case. Here, standard Gaussian quadrature rules are applied for the three noncut cells. However, for the cut cell Ω^c on the bottom right, an adaptive integration scheme based on a quadtree subdivision is employed. In doing so, the cut cell is subdivided on its parent cell domain $\Omega^c_\square$ with local coordinates (ξ, η). Consequently, the parent cell defines the root of the quadtree. Then, at the first refinement level ($k = 1$), the cut cell is subdivided into four uniform subcells. This procedure is repeated for each cut subcell $\Omega^{sc}_\square$ until the final refinement level ($k = 4$) has reached. As it can be seen from the figure, with increasing tree depth (or refinement) level of the quadtree, the accuracy of the resolution of the physical and the fictitious subdomain is improved. Finally, the numerical integration of the cut cell is conducted by employing a standard Gaussian quadrature on each subcell *sc* of the quadtree mesh. Thereby, the numerical integration is carried out on the parent subcell domain $\Omega^{sc}_\blacksquare$ with local coordinates (r, s). From this procedure, it is obvious that the approximation of the geometry is carried out using the integration mesh instead of the mesh applied for the discretization of the computational domain. Consequently, the simple mesh generation approach of the FCM is accompanied by a more elaborate quadrature method used for the numerical computation of cut finite cells. However, note that the generation of the integration mesh can be performed easily and in an automatic manner. Moreover, since it is used only for integration purposes, the integration mesh is subjected to much less restrictions as compared to the computational mesh, e.g. hanging nodes are no problem.

After addressing the basic idea of the adaptive integration method in a schematic manner, let us once again return to the three-dimensional case. In doing so, the subcell mesh is generated employing an octree, which defines the analogue of the quadtree (2D) for the three-dimensional case. Then, the computation of the local stiffness matrix $\mathbf{k}^c$ can be carried out as follows

$$\mathbf{k}^c = \int\limits_{\Omega^c} \alpha\, \mathbf{G}^T\mathbf{A}\mathbf{G}\, \mathrm{d}V = \int\limits_{\Omega^c_\square} \alpha\, \mathbf{G}^T\mathbf{A}\mathbf{G}\, \det\mathbf{J}^c\, \mathrm{d}\Omega_\square \tag{3.48}$$

$$= \int\limits_{-1}^{1}\int\limits_{-1}^{1}\int\limits_{-1}^{1} \alpha\, \mathbf{G}^T\mathbf{A}\mathbf{G}\, \det\mathbf{J}^c\, \mathrm{d}\xi\mathrm{d}\eta\mathrm{d}\zeta \tag{3.49}$$

$$\approx \int\limits_{-1}^{1}\int\limits_{-1}^{1}\int\limits_{-1}^{1} \sum_{sc=1}^{n_{sc}} \alpha\, \mathbf{G}^T\mathbf{A}\mathbf{G}\, \det\mathbf{J}^c \det\mathbf{J}^{sc}\, \mathrm{d}r\mathrm{d}s\mathrm{d}t \tag{3.50}$$

$$\approx \sum_{sc=1}^{n_{sc}} \sum_{i=1}^{n_g} \alpha_{\boldsymbol{\xi}_i}\, \mathbf{G}^T_{\boldsymbol{\xi}_i}\mathbf{A}_{\boldsymbol{\xi}_i}\mathbf{G}_{\boldsymbol{\xi}_i}\, \det\mathbf{J}^c \det\mathbf{J}^{sc}\, \lambda_i \tag{3.51}$$

To this end, the integral with respect to the initial coordinates (X, Y, Z) is first transformed to the local coordinates (ξ, η, ζ) of the parent cell domain $\Omega^c_\square$, similar to Eq. (3.44). However, since a standard Gaussian quadrature is applied on subcell level, the integration process of the cut cells is accompanied by an additional integral transformation from the local coordinates (ξ, η, ζ) of the cell c to local coordinates (r, s, t) of each subcell sc. Thereby, $\det\mathbf{J}^{sc}$ represents the determinant of the Jacobian matrix $\mathbf{J}^{sc}$ depending on the mapping relation between the cell c and the subcell sc. Due to the rectangular shape, the mapping relating the local coordinates (ξ, η, ζ) of the cell with the local coordinates of the subcell (r, s, t) is simple and can be carried out in a similar way to Eq. (3.13)

$$\boldsymbol{\xi} = \begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \mathbf{Q}^{sc} = \begin{bmatrix} \xi_{\min} + 1/2\,(1+r)\, H_\xi \\ \eta_{\min} + 1/2\,(1+s)\, H_\eta \\ \zeta_{\min} + 1/2\,(1+t)\, H_\zeta \end{bmatrix} \quad . \tag{3.52}$$

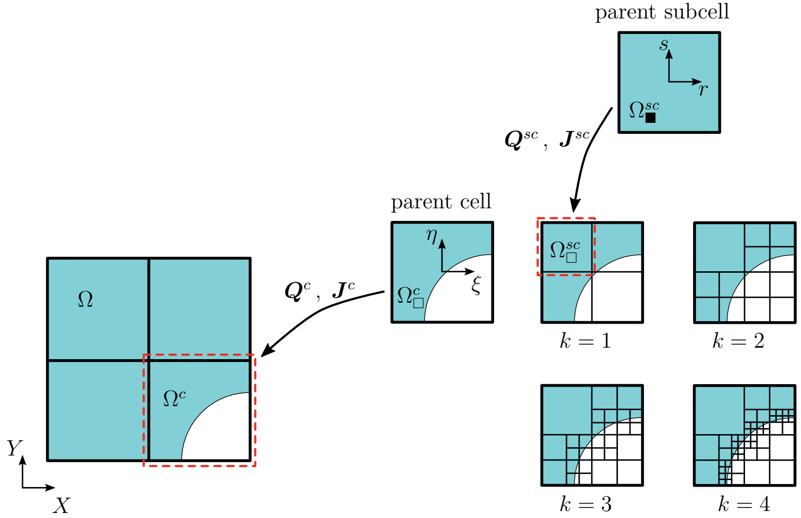


Figure 3.4: Adaptive integration scheme based on a quadtree decomposition.

Here, $\xi_{\min}$, $\eta_{\min}$, and $\zeta_{\min}$ define the minimum coordinates of the subcell with respect to the local coordinates of the parent cell. Further, the length dimensions of the subcell are given by H_ξ, H_η, and H_ζ. As a result of the simple mapping relation, the corresponding Jacobian matrix $\mathbf{J}^{sc}$ is constant and diagonal

$$\mathbf{J}^{sc} = \frac{1}{2}\begin{bmatrix} H_\xi & 0 & 0 \\ 0 & H_\eta & 0 \\ 0 & 0 & H_\zeta \end{bmatrix} \quad . \tag{3.53}$$

Thus, finally, the determinant of $\mathbf{J}^{sc}$ only depends on the length dimensions of the subcells

$$\det\mathbf{J}^{sc} = \frac{1}{2} H_\xi H_\eta H_\zeta \quad . \tag{3.54}$$

4 Moment fitting quadratures

In nonstandard discretization methods – such as the generalized or extended finite element method (GFEM/XFEM) [90–94], or fictitious domain methods [8, 11, 13, 14] like the finite cell method (FCM) [19–22] or the CutFEM [16, 17] – particular consideration has to be given to the numerical integration process. This is due to the fact that the computational mesh is decoupled from the description of the geometry or geometrical features – e.g. voids, material inclusions, or cracks – of the underlying problem, thus providing a fast and simple mesh generation. However, this simplification in the mesh generation results in elements that are cut by material or crack interfaces or by the physical boundary. As a consequence, these elements exhibit integrals including weak or strong discontinuities or singularities. Consequently, the integrals of these elements are not smooth anymore. Thus, standard Gauss quadrature rules cannot be applied, since they show a weak performance for such kind of integrals.

In the scope of this chapter, we focus on the numerical integration process of the finite cell method. As mentioned in Sec. 3.3, standard Gauss quadrature formulas are applied for the noncut cells. Here, we thus focus especially on integration methods that are employed for the computation of the discontinuous integrals of the cut cells. These cells include integrals of the following form

$$\int_{\Omega^c} \alpha\, f(\boldsymbol{X})\,\mathrm{d}V \quad . \tag{4.1}$$

Here, $f(\boldsymbol{X})$ defines a continuous and sufficiently smooth function on cell domain Ω^c with respect to the initial coordinates (X, Y, Z). Further, α represents the indicator function which is one for points within the physical subdomain $\Omega^{c,\mathrm{phy}}$ of the cell and zero for the fictitious subdomain $\Omega^{c,\mathrm{fic}}$. Consequently, the indicator function transforms the continuous integrals of the cell into discontinuous ones. Several integration methods have been proposed for the computation of these discontinuous integrals, which are discussed in the following.

A rather simple commonly applied numerical integration approach is based on a subdivision of a cut cell. In doing so, the discontinuous integrals are split into two separate integrals which consider the physical $\Omega^{c,\mathrm{phy}}$ and the fictitious subdomain $\Omega^{c,\mathrm{fic}}$ of a cell individually

$$\int_{\Omega^c} \alpha\, f(\boldsymbol{X})\,\mathrm{d}V = \int_{\Omega^{c,\mathrm{phy}}} 1\, f(\boldsymbol{X})\,\mathrm{d}V + \int_{\Omega^{c,\mathrm{fic}}} 0\, f(\boldsymbol{X})\,\mathrm{d}V = \int_{\Omega^{c,\mathrm{phy}}} f(\boldsymbol{X})\,\mathrm{d}V \quad . \tag{4.2}$$

Since we are only interested in the computation of the integral over the physical subdomain, the integral over the fictitious subdomain vanishes. In the context of the FCM, this is ensured by the definition of the indicator function α. Next, in order to compute the integral over the physical subdomain, a local integration mesh is introduced. This local mesh is composed of elements which resolve for $\Omega^{c,\mathrm{phy}}$. In the context of the FCM, these elements are denoted as subcells, since an individual integration mesh is generated for each cut cell. Consequently, the relation in Eq. (4.2) can be written as

$$\int_{\Omega^c} \alpha\, f(\boldsymbol{X})\,\mathrm{d}V = \int_{\Omega^{c,\mathrm{phy}}} f(\boldsymbol{X})\,\mathrm{d}V = \sum_{sc=1}^{n_{sc}} \int_{\Omega^{sc,\mathrm{phy}}} f(\boldsymbol{X})\,\mathrm{d}V \quad , \tag{4.3}$$

where n_{sc} is the number of the subcells and $\Omega^{sc,\mathrm{phy}}$ defines the physical subdomain of each subcell sc. Since the local subcell meshes are needed for integration purposes only, they are subjected to less restrictions as compared to the computational mesh. Neighboring subcells are not restricted to any continuity conditions. Thus, hanging nodes are allowed, for instance. The only requirement to each subcell is that it occupies a domain where standard quadrature rules can be applied, e.g. Gaussian quadrature formulas. There are several possible strategies for the generation of the subcell meshes. In [19], the integration mesh is composed of subcells applying a uniform subdivision of a cut cell. A more efficient strategy is provided by employing spacetrees that are based on a quadtree or an octree subdivision [20, 95–97]. Further mesh generation strategies are based on low-order tesselation utilizing triangles or tetrahedrons [98–101]. Moreover, in [102–110] high-order subcells are employed which allow to represent curved edges and faces, thus allowing to reduce the number of subcells for a high resolution of complex geometry. Generally, the adaptive integration scheme based on a quadtree or an octree subdivision is applied in the context of the FCM – which has been explained in more detail in Sec. 3.3.2. This is due to the reason that this integration scheme performs fully automatic considering complex geometries. Further, it allows to control the error in integration. Furthermore, since the generation of the quadtree or octree mesh only requires the information whether a point is inside or outside of the integration domain, it can be applied easily to a wide variety of geometric models – such as voxel-based models provided by quantitative computer tomography scans (qCT-scans), B-rep models obtained from commercial CAD-programs, or level-set representation. However, the disadvantage of this adaptive integration scheme is that it commonly results in a high number of quadrature points, making the numerical integration process computationally expensive.

In order to reduce the number of the quadrature points and thus to perform the computation of the integrals more efficiently, Ventura and Benvenuti [111, 112] introduced an innovative integration approach based on equivalent polynomials. In this integration method, the discontinuous function is replaced by an equivalent polynomial. In doing so, the discontinuous integrals are transformed into continuous ones so that standard Gauss quadrature rules can be applied. For the construction of the equivalent polynomial, a linear equation system has to be solved for the unknown polynomial coefficients. This integration method for applications of the FCM was extended in [96]. Moreover, in [113], Abedian and Düster introduced a modified variant based on equivalent Legendre polynomials (ELP). In this method, the monomial basis used for the construction of the equivalent polynomial is replaced by the Legendre basis. Then, by taking advantage of the orthogonality property of the Legendre polynomials the resulting equation system becomes diagonal, thus avoiding the necessity of solving an equation system. This fact is of special interest considering equivalent polynomials of higher order where the equation system based on the monomial basis, generally, results in high condition numbers.

Another promising integration method to perform the computation of the discontinuous integrals in a more efficient manner is based on the moment fitting approach [114–120]

– which is in the focus of this chapter. This integration method allows to generate individual quadrature rules for arbitrary domains by solving the moment fitting equation system, which is generally a nonlinear system. In [121] Mousavi and Sukumar applied the moment fitting approach together with Lasserre's method to construct quadrature rules to compute the integrals of polynomials on irregular convex polygonal or polyhedral elements. Thereby, the set of quadrature points is predefined in such a way that the nonlinear moment fitting equation system turns into a linear and underdetermined system. Further, in order to compute the integrals of elements that are cut by crack interfaces, they proposed a moment fitting approach which incorporates the generalized Heaviside function. In doing so, the numerical integration could be performed on element level, thus avoiding the necessity of employing any subdivision scheme. A similar strategy has recently been proposed by Düster and Allix [122], where the basis functions used for the moment fitting are enriched by appropriate functions accounting for weak and strong discontinuities with arbitrary shape of the interface. In the context of the enriched partition of unity methods (EPUM), Sudhakar and Wall [123] extended the moment fitting method for the computation of integrals on arbitrary convex or concave volumes. In doing so, they followed the approach by predefining the position of the quadrature points as suggested by Mousavi and Sukumar [121]. However, since Lasserre's method is restricted to convex volumes, the integration of the basis function is performed using the divergence theorem. A further version of the moment fitting approach has been presented by Müller et al. [124] for cells that are cut by implicitly described interfaces by means of level set functions. In this method, the divergence theorem is applied together with divergence-free basis functions. For planar interfaces, the divergence-free basis provides quadrature rules with a high accuracy. However, the accuracy reduces with an increasing curvature. Further, in [125], Thiagarajan and Shapiro presented an adaptively weighted integration method based on moment fitting, the divergence theorem, and shape-sensitive analysis.

In this chapter, we propose different versions of the moment fitting method and study their performance in terms of accuracy and robustness for linear and nonlinear applications of the finite cell method [31, 33, 122, 126–130]. To this end, in the first moment fitting method, we follow the approach suggested by Mousavi and Sukumar [121] and predefine the position of the quadrature points a priori – which turns the nonlinear moment fitting equation system into a linear one. Thereby, we present two different point distribution schemes. In the first distribution scheme, the points are adaptively distributed within the physical subdomain of the cut cell, based on a uniform subdivision. In the second approach, we use the position of the standard Gauss-Legendre points. As a consequence, the points may be located within the physical and the fictitious subdomain. Further, in a second moment fitting version, we solve the nonlinear system applying an optimization procedure. In doing so, the number of the points could be reduced further, thus resulting in more efficient moment fitting quadratures. Moreover, to increase the robustness for nonlinear problems of the FCM, we propose an adaptive scheme using moment fitting. In this scheme, the moment fitting based on Gauss-Legendre points is applied on cell or subcell level of cut cells. In order to reduce the effort of generating the quadratures, we introduce a moment fitting version that avoids the necessity of solving the moment fitting equation system, which is in general the most expensive part.

4.1 Moment fitting approach

In the following, we briefly describe the basic idea of the moment fitting approach. For reasons of simplicity, the subsequent formulations are stated with respect to the local coordinates, since the numerical integration is generally performed on the domain of the parent cell. In doing so, let us consider the cut cell $\Omega^{c}_{\square}$ depicted in Fig. 4.1 – where $\Omega^{c,\mathrm{phy}}_{\square}$ and $\Omega^{c,\mathrm{fic}}_{\square}$ define the physical and the fictitious subdomain. Then, the goal of the moment fitting approach is to set up an individual quadrature rule in order to approximate the integral over the physical subdomain by a weighted sum. Note that there are no restrictions to the integration domain $\Omega^{c,\mathrm{phy}}_{\square}$, so it can be of arbitrary shape or even composed of individual disconnected domains. In the more general form of the moment fitting formula, the integral of a function $g(\boldsymbol{\xi})$ is then replaced by an equivalent weighted integral

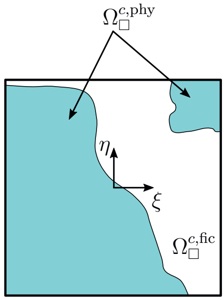


Figure 4.1: Cut cell.

$$\int\limits_{\Omega^{c,\mathrm{phy}}_{\square}} g(\boldsymbol{\xi})\,\mathrm{d}\Omega_{\square} = \int\limits_{\Omega^{c,\mathrm{phy}}_{\square}} f(\boldsymbol{\xi})w(\boldsymbol{\xi})\,\mathrm{d}\Omega_{\square} \approx \sum_{i=1}^{n} f(\boldsymbol{\xi}_i)w(\boldsymbol{\xi}_i)\lambda_i \quad , \tag{4.4}$$

where $w(\boldsymbol{\xi})$ defines a weight function. In Eq. (4.4), n denotes the number of quadrature points. Further, $\boldsymbol{\xi}_i$ and λ_i define the position and the corresponding weight of the integration points, respectively. Next, in order to compute the position and the weights, the function $f(\boldsymbol{\xi})$ is first approximated by means of a set of m linear independent basis functions $\varphi_j(\boldsymbol{\xi})$ as

$$f(\boldsymbol{\xi}) = \sum_{j=1}^{m} \beta_j\,\varphi_j(\boldsymbol{\xi}) \quad , \tag{4.5}$$

where $\beta_j \in \mathbb{R}$ denote the corresponding coefficients of each basis function. Introducing the approximation ansatz in Eq. (4.5) results in the well-known moment fitting equations

$$\sum_{i=1}^{n} w(\boldsymbol{\xi}_i)\varphi_j(\boldsymbol{\xi}_i)\,\lambda_i = \int\limits_{\Omega^{c,\mathrm{phy}}_{\square}} w(\boldsymbol{\xi})\varphi_j(\boldsymbol{\xi})\,\mathrm{d}\Omega_{\square} \quad , \quad j = 1,\dots,m \quad . \tag{4.6}$$

Here, the left-hand side represents the weighted sum of the quadrature rule and the right-hand side includes the integrals of the individual basis function over the integration domain, which are also referred to as the moments. Consequently, in order to obtain the quadrature rule, we have to solve a nonlinear equation system for the n unknown points $\boldsymbol{\xi}_i$ and the corresponding n unknown weights λ_i

$$\begin{bmatrix} w(\boldsymbol{\xi}_1)\varphi_1(\boldsymbol{\xi}_1) & \dots & w(\boldsymbol{\xi}_n)\varphi_1(\boldsymbol{\xi}_n) \\ \vdots & \ddots & \vdots \\ w(\boldsymbol{\xi}_1)\varphi_m(\boldsymbol{\xi}_1) & \dots & w(\boldsymbol{\xi}_n)\varphi_m(\boldsymbol{\xi}_n) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} \int\limits_{\Omega_\square^{c,\text{phy}}} w(\boldsymbol{\xi})\varphi_1(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \\ \vdots \\ \int\limits_{\Omega_\square^{c,\text{phy}}} w(\boldsymbol{\xi})\varphi_m(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \end{bmatrix} \quad . \tag{4.7}$$

In symbolic notation, the relation in Eq. (4.7) can be represented as

$$\mathbf{A}\,\mathbf{x} = \mathbf{b} \quad , \tag{4.8}$$

where $\mathbf{A}$ is the coefficient matrix, $\mathbf{x}$ denotes the vector containing the unknown weights, and $\mathbf{b}$ defines the vector including the individual moments. Note that the system is, in general, nonlinear in terms of $\boldsymbol{\xi}_i$ and linear dependent regarding the weights λ_i.

In the framework of this contribution, we are interested in the numerical integration of polynomials. Thus, the weight function is considered as $w(\boldsymbol{\xi}) = 1$, and the moment fitting equation system simplifies to

$$\begin{bmatrix} \varphi_1(\boldsymbol{\xi}_1) & \dots & \varphi_1(\boldsymbol{\xi}_n) \\ \vdots & \ddots & \vdots \\ \varphi_m(\boldsymbol{\xi}_1) & \dots & \varphi_m(\boldsymbol{\xi}_n) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} \int\limits_{\Omega_\square^{c,\text{phy}}} \varphi_1(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \\ \vdots \\ \int\limits_{\Omega_\square^{c,\text{phy}}} \varphi_m(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \end{bmatrix} \quad . \tag{4.9}$$

4.1.1 Basis functions

For the selection of the basis, different sets of functions may be chosen – such as monomials, or Legendre or Chebyshev polynomials. However, the use of monomials generally results in higher condition numbers of $\mathbf{A}$ as compared to the usage of the Legendre or the Chebyshev basis. In this contribution, we apply the Legendre basis. In doing so, the set of the basis functions for the three-dimensional case is composed of the tensor-product of the one-dimensional Legendre polynomials as

$$\mathcal{F} = \{L_u(\xi)L_v(\eta)L_w(\zeta), \quad u, v, w = 0, \dots, p_q\} \quad , \tag{4.10}$$

where p_q defines the order of the quadrature. The one-dimensional Legendre polynomials of order p are obtained employing a recursive formula

$$(p+1)L_{p+1}(\xi) = (2p+1)\xi L_p(\xi) - pL_{p-1}(\xi) \quad \text{with} \quad p = 1, 2, \dots \tag{4.11}$$

where $L_0(\xi) = 1$ and $L_1(\xi) = \xi$. Consequently, the number of the moment fitting basis functions for any dimension is given as

$$m = (p_q + 1)^d \quad , \tag{4.12}$$

where p_q defines the quadrature order and d denotes the dimension of the problem under consideration.

4.1.2 Point distribution schemes

In order to simplify the setup of the moment fitting equation system, we follow the approach suggested by Mousavi and Sukumar [121] and fix the position quadrature points a priori. In doing so, we transform the nonlinear moment fitting equation system into a linear one which only has to be solved for the unknown weights. Thereby, we consider two different point distribution strategies: an adaptive point distribution scheme and a distribution scheme using the standard Gauss-Legendre points. Both strategies are explained in the following.

Adaptive point distribution (APD)

In the *adaptive point distribution* scheme (APD), first presented in [127], we follow the strategy of distributing the moment fitting points within the physical subdomain $\Omega_{\square}^{c,\text{phy}}$ of a cut cell by employing a uniform subdivision scheme. Thereby, we ensure that the number of integration points n is greater or equal to the number of basis function m

$$n \geq m = (p_q + 1)^d \quad . \tag{4.13}$$

Consequently, the moment fitting system turns into a linear square or linear underdetermined system that has to be solved.

The basic idea of the adaptive point distribution is schematically depicted in Fig. 4.2 for the two-dimensional case. However, its implementation for the three-dimensional case is straightforward. In the figure, the APD is considered for a quadrature order $p_q = 3$, which is why at least $n = (p_q+1)^2 = 16$ points are required. For the distribution of the points, the cell is then uniformly subdivided into subcells, ensuring that sufficient number of subcells are completely located in the interior of the physical subdomain $\Omega_{\square}^{c,\text{phy}}$. For the situation in the figure, this is true for five subcells. Next, the APD distributes the points within the interior subcells. In doing so, the same number of points are distributed within each subcell. Thereby, practical experience has shown that employing a random distribution scheme on subcell level leads to better condition numbers of the resulting moment fitting system as compared to uniform distribution schemes. Due to this reason, each subcell includes an individual distribution scheme – as can be seen from the figure. Moreover, the number of created points ($n = 20$) is bigger than the number of basis function ($m = 16$).

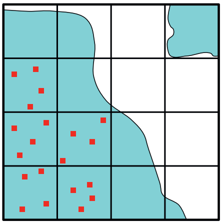


Figure 4.2: Adaptive point distribution scheme.

Gauss-Legendre points (GLP)

Although the adaptive point distribution scheme is simple to implement it often results in ill-conditioning problems of the moment fitting system and, thus, in badly conditioned quadrature rules. This fact is especially pronounced if the physical subdomain of the cut cell is relatively small, or in the case where moment fitting quadratures of higher order are considered. Due to this reason, we choose the position of the standard Gauss-Legendre points as the second point set, see Fig. 4.3. In doing so, the left-hand side of the moment fitting system in Eq. (4.9) is independent in terms of the composition of the cells – even whether the cell is cut or not. As a result, the coefficient matrix **A** is always the same and well-conditioned, thus leading to well-conditioned quadratures ensuring a high accuracy. Moreover, using the Gauss-Legendre points transforms the nonlinear moment fitting equations into a linear square system and, thus, the relation between the number of points and the number of basis functions holds $n = m$. Although applying the position of the Gauss-Legendre points implies that points are located within the physical as well as in the fictitious domain, we can show that this approach is well suited considering applications of the finite cell method to linear problems.

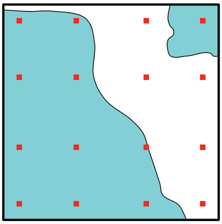


Figure 4.3: Gauss-Legendre points.

4.1.3 Computation of the moments

In order to compute the individual integrals of the right-hand side in Eq. (4.9), also referred to as moments, different approaches can be applied, see [127, 129]. One possibility is to perform the computation of the volume integrals by employing adaptive integration methods based on a subdivision of the cut cells. In doing so, a fine integration mesh can be used, since the computation of the moments is cheap as compared to the computation of the stiffness matrix, for instance. Moreover, in simple situations, the computations of the moment may be also performed symbolically.

Another possibility is to perform the computation of the moments on the surface of the physical subdomain. To this end, the volume integrals are transformed into surface ones by applying the divergence theorem. Thus, the right-hand side can be rewritten as

$$\int\limits_{\Omega_{\square}^{c,\text{phy}}} \varphi_j(\boldsymbol{\xi})\,\mathrm{d}\Omega_{\square} = \int\limits_{\Omega_{\square}^{c,\text{phy}}} \operatorname{div} \boldsymbol{h}_j(\boldsymbol{\xi})\,\mathrm{d}\Omega_{\square} = \int\limits_{\Gamma_{\square}^{c,\text{phy}}} \boldsymbol{h}_j(\boldsymbol{\xi}) \cdot \boldsymbol{n}(\boldsymbol{\xi})\,\mathrm{d}\Gamma_{\square} \quad , \tag{4.14}$$

where $\Gamma_{\square}^{c,\text{phy}}$ represents the closed surface of the physical subdomain $\Omega_{\square}^{c,\text{phy}}$. Further, $\boldsymbol{n}(\boldsymbol{\xi})$ denotes the normal of the surface pointing in outward direction and $\boldsymbol{h}_j(\boldsymbol{\xi})$ defines the anti-derivatives of the basis function $\varphi_j(\boldsymbol{\xi})$. For the three-dimensional case, the vector of the anti-derivatives can be computed as

$$\mathbf{h}_j(\boldsymbol{\xi}) = \frac{1}{3}\begin{bmatrix} \int \varphi_j(\boldsymbol{\xi})\,\mathrm{d}\xi \\ \int \varphi_j(\boldsymbol{\xi})\,\mathrm{d}\eta \\ \int \varphi_j(\boldsymbol{\xi})\,\mathrm{d}\zeta \end{bmatrix} \quad . \tag{4.15}$$

Note that, due to the standard matrix notation, we use an upright letter for the vector containing the individual anti-derivatives. Finally, in order to compute the integrals, we need a surface discretization. To this end, a rather simple approach is to discretize the surface using triangles and then to apply a standard Gauss quadrature on each triangle. Thereby, in order to improve the performance of the surface integration applying a fine discretization, reduced quadrature rules can be used.

4.1.4 Computation of the weights

After determining the coefficient matrix of the moment fitting equation system and computing the moments, we finally have to solve the resulting linear system for the unknown weights. In doing so, we have to distinguish between the two different point sets based on the APD and the GLP.

Applying the APD, in general, we ensure that the number of points exceeds the number of basis functions ($n > m$). As a consequence, the number of moment fitting equations is greater than the number of the unknown weights, so we obtain a linear underdetermined system. Then, assuming that this system is consistent, there exist an infinite number of solutions. In order to find an appropriate solution, we apply an optimization problem based on linear least squares

$$\min \|\mathbf{x}\|_2 \quad \text{subjected to} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad . \tag{4.16}$$

Summarizing, we find a solution based on the minimum Euclidean norm that solves the moment fitting equations system. In order to solve the optimization problem in Eq. (4.16), we employ the open-source linear algebra package LAPACK [131]. Thereby, different routines can be used – such as DGELSY, which computes the minimized norm solution based on a complete orthogonal factorization, or DGELSS, which computes the minimized norm solution based on a singular value decomposition.

Solving the moment fitting equation system using the GLP, on the other hand, is simple. Here, we just have to solve a linear square system which is well-conditioned for any situation of the cut cell and any order of the quadrature. Thus, there is always a unique solution that can be applied using common solvers.

4.1.5 Optimized points and weights

As mentioned at the beginning of this chapter, choosing the position of the moment fitting points a priori does not lead to optimal quadrature rules for cut cells. In this section, we therefore aim to provide an approach of how to solve the nonlinear moment fitting equation system, thus leading to optimized points and weights (OP). To this end, let us recall the discussion about optimal quadratures. In Sec. 3.3.1, we showed that Gaussian quadratures are optimal for the one-dimensional case – in the sense that the integrals of any polynomial of order $p \leq 2n-1$ can be computed exactly if n points and weights are used. However, applying the moment fitting based on preselected points commonly leads to nonoptimal quadrature rules of cut cells. In general, the integrals of polynomials of order $p \leq n-1$ can be computed exactly if n points are used. Consequently, to increase the order of the quadrature and, thus, to obtain more optimal points and weights, one has to solve the nonlinear moment fitting equation system, the number of points is therefore lower than the number of basis functions ($n < m$) – which means that we have to solve an overdetermined system of nonlinear equations. A general approach to solve such a system is based on an optimization problem that minimizes the Euclidean norm of the residual of the system

$$\min_{\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_n,\lambda_1,\ldots,\lambda_n} \left\| \mathbf{r}\left(\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_n,\lambda_1,\ldots,\lambda_n\right)\right\|_2^2 \quad , \tag{4.17}$$

where the definition of the residual is given as

$$r_j = \int\limits_{\Omega_\square^{c,\mathrm{phy}}} \varphi_j(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square - \sum_{i=1}^{n} \varphi_j(\boldsymbol{\xi}_i)\,\lambda_i \quad , \quad j = 1,\ldots,m \tag{4.18}$$

or, in standard matrix notation, as

$$\begin{bmatrix} r_1 \\ \vdots \\ r_m \end{bmatrix} = \begin{bmatrix} \int\limits_{\Omega_\square^{c,\mathrm{phy}}} \varphi_1(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \\ \vdots \\ \int\limits_{\Omega_\square^{c,\mathrm{phy}}} \varphi_m(\boldsymbol{\xi})\,\mathrm{d}\Omega_\square \end{bmatrix} - \begin{bmatrix} \varphi_1(\boldsymbol{\xi}_1) & \ldots & \varphi_1(\boldsymbol{\xi}_n) \\ \vdots & \ddots & \vdots \\ \varphi_m(\boldsymbol{\xi}_1) & \ldots & \varphi_m(\boldsymbol{\xi}_n) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} \quad . \tag{4.19}$$

In order to find an approximate solution for the optimization problem in Eq. (4.17), which defines a nonlinear least squares problem, several well-established iterative methods exist. In this thesis, we employ the MATLAB function lsqnonlin [132] using the Levenberg-Marquardt algorithm that, essentially, is based on a combination of the Gauss-Newton method and the gradient descent method. Another frequently used method is the Sequential Quadratic Programming.

Before starting to solve the optimization problem, we have to set up a set of n initial points and weights. Therefore, we first have to estimate the minimal number of points that is not known a priori. To this end, let us define the lower and upper bound for n. From the structure of the problem, it is obvious that the lower bound is defined by one. Further, assuming that there exist a set of $n = m$ distinct points such that the moment fitting system results into a linear and square system of full rank, the upper bound is then defined by $m - 1$ where m denotes the number of the basis functions. Next, in order to find an estimation for n, we apply a binary search method based on the lower and upper bound. Having defined a value for n within the predefined bounds, in the next step, we have to initialize the position of the points. Here, it turns out that a random distribution within the domain of the parent cell performs more robust in the context of the applied optimization procedure. Based on the position of the points the corresponding initial weights are computed by means of linear least squares. Further, the performance of the optimization method is very sensitive with respect to the initial points – meaning that it may result in a slow decrease of the residual and, thus, in a high number of iterations.

Due to this reason, we abort the procedure after a certain number of iterations has been reached. Finally, the iterative binary search algorithm is performed until the optimization procedure finds a set of optimized points and weights such that the Euclidean norm of the residual is zero within machine precision. Consequently, several runs have to be performed until an optimized moment fitting quadrature is found.

4.1.6 Numerical examples

In this section, we discuss the performance of the proposed moment fitting quadratures. To this end, we investigate the influence of the different point distribution schemes in terms of the condition number and the residual of the moment fitting system for different orders of the quadrature. Further, we study the condition number of the moment fitting quadratures as well as the accuracy in the numerical integration of polynomial functions.

4.1.6.1 Cell cut by a sphere

The first example is intended to provide a detailed investigation of the presented moment fitting methods in terms of accuracy and efficiency. To this end, we consider a cell that is cut by a sphere, see Fig. 4.4. Thereby, the geometry of the cell is described by a regular hexahedron of the domain

$$\Omega^c = [0,1]^3 \quad . \tag{4.20}$$

Further, the sphere is defined by a level set function

$$\phi(\boldsymbol{x}) = (x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 - R^2 \quad , \tag{4.21}$$

for which the geometry parameters are listed in Tab. 4.1. Consequently, the sphere cuts the cell in such a way that the integration domain is given by an eighth of the sphere.

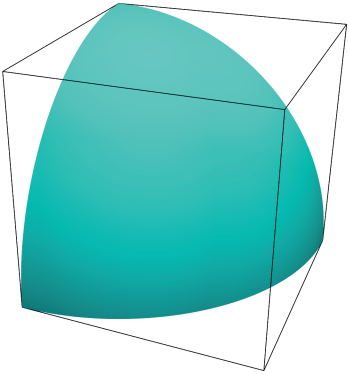


Figure 4.4: Cell cut by a sphere.

Table 4.1: Sphere. Geometry parameters.

parameter	variable	value
x center coordinate	x_c	0.0
y center coordinate	y_c	0.0
z center coordinate	z_c	0.0
radius	R	1.0

Moment fitting based on APD and GLP

In order to study the performance of the moment fitting quadratures, we start off by considering the approach based on the two distinct strategies for the predefinition of the position of the points: the adaptive point distribution scheme (APD) and the Gauss-Legendre points (GLP). Thereby, in order to obtain exact moment fitting quadratures for the eighth of the sphere, essentially, two conditions have to be satisfied. The first condition is linked to the issue that the chosen point distribution strategy has to provide a point set that results into a well-conditioned system of full rank. Thus, the solvability of the moment fitting equation system within machine precision can be ensured. The second condition concerns to the exact computation of the integrals of the moments. In order to compute these integrals, neither an adaptive integration based on an octree subdivision nor a surface integration based on a triangulation are exact. Due to this reason, we compute the integrals of the moments *symbolically* using Wolfram Mathematica [133].

To study the performance of the point distribution strategies, we start off by investigating the influence of the different point sets on the condition number κ of the coefficient matrix $\mathbf{A}$. Therefore, we employ the LAPACK routine DGELLSS to solve the moment fitting system which provides the computation of the singular values. Thus, we compute the condition number κ of the coefficient matrix $\mathbf{A}$ based on the following definition

$$\kappa = \frac{\sigma_{\max}}{\sigma_{\min}} \quad , \tag{4.22}$$

where $\sigma_{\max}$ denotes the maximum and $\sigma_{\min}$ the minimum singular value, respectively.

The values of the condition number applying the APD and the GLP are plotted in Fig. 4.5. Here, we consider the evolution of the condition number κ for different orders of the moment fitting quadratures ($p_q = 0, \ldots, 16$) since, in the context of the FCM, we are interested in the numerical integration of high-order shape functions. From the figure, it can be seen that the moment fitting based on GLP results in a much better conditioning of the coefficient matrix as compared to the APD. This fact becomes especially evident when quadratures of higher order are considered. For $p_q > 8$, we could improve the condition number by more than 12 orders of magnitude. Moreover, note that applying the GLP yields the same condition number for any arbitrary topology of $\Omega_{\square}^{c,\mathrm{phy}}$. However, employing the APD has a strong dependence on the composition of $\Omega_{\square}^{c,\mathrm{phy}}$ and, thus, has a considerable influence on κ, on the other hand. Especially for cases where the integration domain $\Omega_{\square}^{c,\mathrm{phy}}$ is relatively small, the APD results in high condition numbers.

From the results in Fig. 4.5, we can conclude that applying the GLP improves the solvability of the moment fitting equations system, which is due to the much better conditioning of the coefficient matrix. In order to demonstrate this fact, in the following, we study the performance of the different point distribution schemes with respect to the residual $\mathbf{r}$ of the moment fitting equations – where the definition of $\mathbf{r}$ is given in Eq. (4.18) and (4.19), respectively. To this end, we consider the Euclidean norm of the residual $\|\mathbf{r}\|_2$ for different quadrature orders ($p_q = 2, \ldots, 16$), see Fig. 4.6. If the GLP is applied, the norm of the residual is close to zero, i.e. within machine precision – as can be seen from the figure. On the other hand, $\|\mathbf{r}\|_2$ increases when the points provided by the APD are used. This is especially true for the quadratures of higher order. For $p_q > 9$, we could thus reduce $\|\mathbf{r}\|_2$ by 4 to 5 orders of magnitude by employing the GLP.

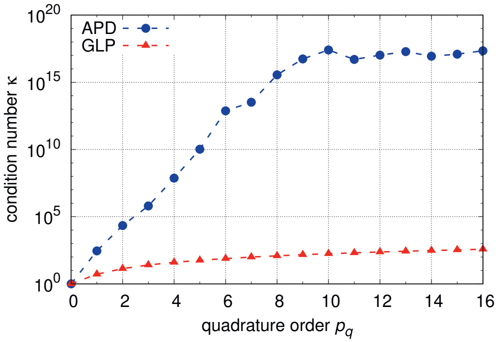


Figure 4.5: Condition number of the coefficient matrix for different orders of the moment fitting quadratures.

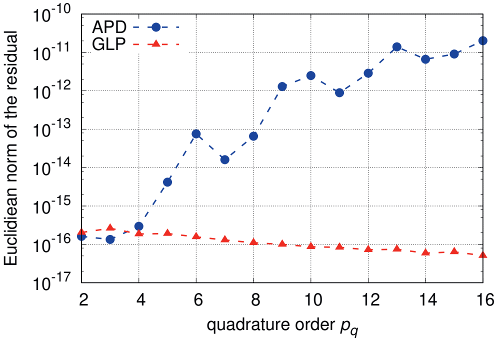


Figure 4.6: Euclidean norm of the residual of the moment fitting equation system for different orders of the quadrature.

Next, we study the conditioning of the generated quadrature rules based on moment fitting. To this end, the most frequently used measurement to evaluate the quality of a quadrature is provided by the sum of the absolute values of the n weights [134]

$$\kappa_q = \sum_{i=1}^{n_g} |\lambda_i| \quad . \tag{4.23}$$

In Eq. (4.23), κ_q is also known as the condition number of a quadrature rule. In order to facilitate the investigation of the quality of the quadratures, we consider $\bar{\kappa}_q$

$$\bar{\kappa}_q = \frac{\kappa_q}{\Omega_{\square}^{c,\mathrm{phy}}} \tag{4.24}$$

which normalizes κ_q by the volume of the integration domain $\Omega_{\square}^{c,\mathrm{phy}}$. In doing so, $\bar{\kappa}_q = 1$ represents an optimally conditioned quadrature – which implies that all weights are nonnegative. Consequently, the appearance of negative weights results in higher values for $\bar{\kappa}_q$ and, thus, in less well-conditioned quadrature rules. Next, in order to study the influence of the GLP and the APD on the conditioning of moment fitting quadratures, Fig. 4.7 shows the normalized condition number $\bar{\kappa}_q$ for different orders of p_q. As it can be seen from the figure, the moment fitting quadratures based on the APD results in high condition numbers. This applies in particular for the quadratures of higher order. Here, $\bar{\kappa}_q$ deviates by several orders of magnitude from the optimal value, which is one. The deviations from the optimal condition number can be deduced from the fact that the quadratures include negative weights. Further, the greater the absolute values of the negative weights the higher the condition number and, thus, the stronger the deviation from the optimal value. On the other hand, considering the condition numbers of the moment fitting quadrature based on the GLP results in much lower values even for the high order quadratures. Here, $\bar{\kappa}_q$ oscillates between 1 and 1.54, thus resulting in much better conditioned quadratures. For $\bar{\kappa}_q > 1$ the quadratures still imply the occurrence of negative weights, but the absolute value of these weights is much smaller than compared to the APD.

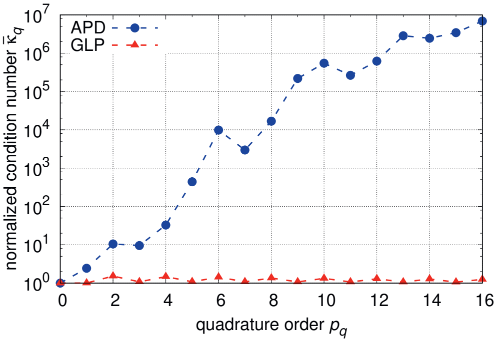


Figure 4.7: Normalized condition number of the moment fitting quadratures for different orders of the quadrature.

Next, we study the accuracy of the two distinct moment fitting approaches by computing the integrals of polynomial functions. To this end, we consider four different orders of the moment fitting quadratures $p_q = 4, 7, 11, 16$. For the integrands, we further choose polynomials of order $p_i = 0, \ldots, 17$. Due to the fact that the moments in Eq. (4.9) are computed symbolically using Wolfram Mathematica [133], the generated quadratures have to compute the integrals of the polynomials exactly for all cases where the polynomial order is less or equal to the quadrature order ($p_i \le p_q$). In order to capture the accuracy of the quadratures, we compute the relative error

$$e_r = \left| \frac{I_{\text{ex}} - I_q}{I_{\text{ex}}} \right| \quad . \tag{4.25}$$

In Eq. (4.25), I_q defines the value of the integral provided by the moment fitting quadrature and I_{ex} represents the *exact* value obtained by means of a symbolic integration using Wolfram Mathematica. Fig. 4.8 and 4.9 illustrate the relative error in integration applying the APD and the GLP, respectively. Considering the results, we can see that within the order of the quadrature the relative error in integration fluctuates around a certain error level $p_q < p_i$. Further, a large jump for e_r arises if the polynomial order of the integrand exceeds the quadrature order ($p_i > p_q$). Moreover, if the APD is applied, the error level increases with increasing order of the quadrature. While e_r for $p_q = 4$ is close to zero, it increases by orders of magnitude for $p_q = 7, 11, 16$. This behavior originates from the high condition number of the coefficient matrix, which adversely affects the solvability of the moment fitting system and, thus, results in weights of lower accuracy. If the GLP is applied, on the other hand, the error in integration remains zero within machine precision.

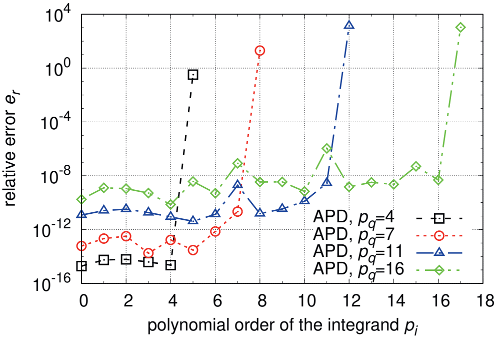


Figure 4.8: Relative error in integrating polynomials applying the moment fitting method based on the APD.

Finally, the total number of integration points n_g is plotted in Fig. 4.10 for the different orders of the moment fitting quadratures. As it can be seen from the figure, the difference in n_g for the APD and the GLP is not high. Moreover, to give an impression of the position of the integration points provided by the distinct distribution strategies, Fig. 4.11a and 4.11b show the points based on the APD and the GLP using a quadrature order $p_q = 4$, respectively.

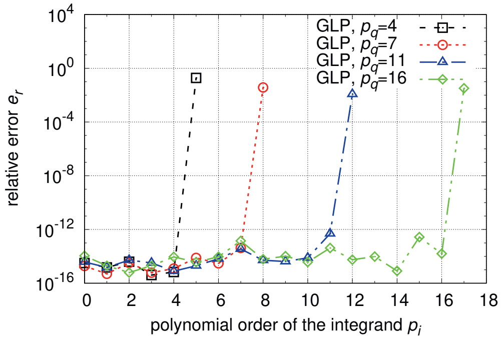


Figure 4.9: Relative error in integrating polynomials applying the moment fitting method based on the GLP.

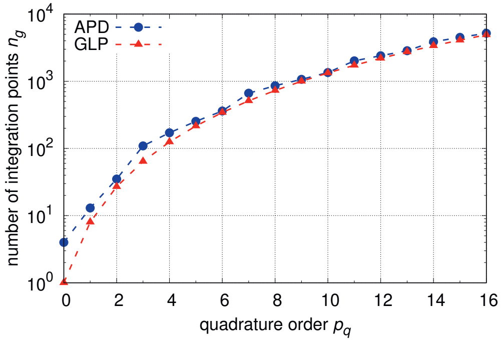


Figure 4.10: Total number of integration points applying different order of the moment fitting quadrature.

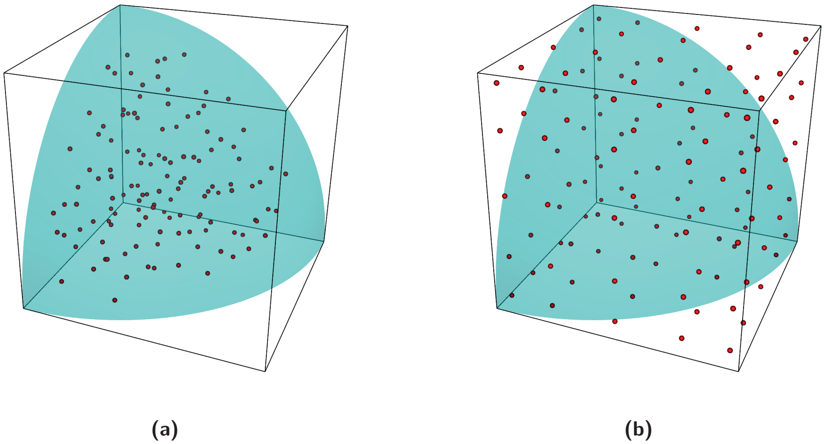


Figure 4.11: Moment fitting points for a quadrature of order $p_q = 4$. **(a)** Employing the adaptive point distribution (APD). **(b)** Based on the position of Gauss-Legendre points (GLP).

Moment fitting based on OP

Next, we investigate the performance of the moment fitting approach based on the optimization problem given in Sec. 4.1.5. For this purpose, we consider quadratures of order $p_q = 2, \ldots, 7$ since the optimization procedure becomes severely time-consuming for higher orders. In doing so, we compare the results of the optimized points (OP) with the results obtained by the GLP. To this end, we start off by studying the Euclidean norm of the residual $\|\mathbf{r}\|_2$, which is depicted in Fig. 4.12. Here, it can be seen that the values of $\|\mathbf{r}\|_2$ computed by the optimization procedure are slightly higher than those obtained by the GLP. However, it can be seen from the figure that the optimization procedure solves the moment fitting equations with a high level of precision. As the next quantity, we measure the quality of the optimized weights by considering the normalized condition number $\bar{\kappa}_q$ – for which the definition of $\bar{\kappa}_q$ is given in Eq. (4.24). The values of $\bar{\kappa}_q$ are plotted in Fig. 4.13. A comparison between the results of the OP and those of the GLP shows that the GLP results in better conditioned quadrature rules. Here, applying the moment fitting based on the OP, $\bar{\kappa}_q$ increases with increasing order of the quadratures – while $\bar{\kappa}_q$ oscillates between 1 and 1.54 if the GLP is used. From the results, we can therefore conclude that the moment fitting based on the OP leads to negative weights with a higher absolute value than the negative weights of the GLP.

Next, in Fig. 4.14 we study the accuracy of the OP by integrating polynomial functions of different order p_i by regarding the relative error e_r defined in Eq. (4.25). Therefore, we consider the quadrature rules of order $p_q = 4, 5, 6, 7$. As it can be seen from the figure, the OP results in quadratures with a high accuracy. Here, the relative error e_r is zero within machine precision for all polynomial integrands having a order smaller or less than the order of the quadrature ($p_i \leq p_q$). Further, a large jump in e_r arises if the polynomial order exceeds the quadrature order ($p_i > p_q$).

Finally, Fig. 4.15 shows the number of the integration points n_g for the different moment fitting quadratures. Here, it can be seen that the moment fitting based on the OP results in a lower number of integration points as compared to the GLP. However, the difference in the number of integration points is not high. Thus, considering the expenditure of time to solve the optimization problem, the benefit regarding the number of points does not pay off.

Finally, in order to provide an impression of the position of the points, Fig. 4.16 shows the OP points for a quadrature of order $p_q = 4$. Here, Fig. 4.16a shows the initial location of the points and Fig. 4.16b depicts the final position determined by the optimization procedure.

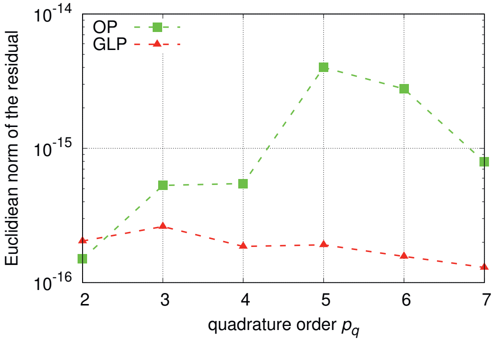


Figure 4.12: Euclidean norm of the residual of the moment fitting equation system for different orders of the quadrature.

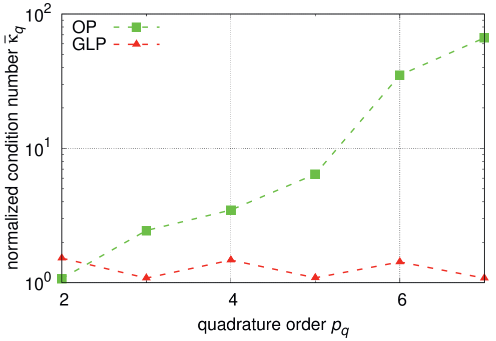


Figure 4.13: Normalized condition number of the moment fitting quadratures for different orders of the quadrature.

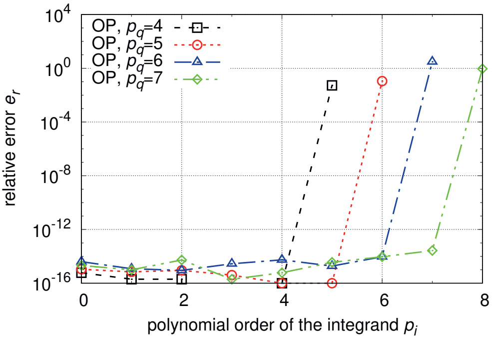


Figure 4.14: Relative error in integrating polynomials applying the moment fitting method based on optimized points.

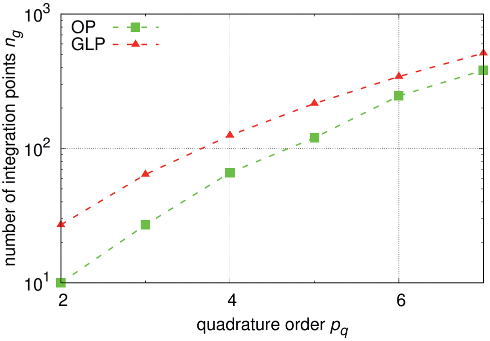


Figure 4.15: Total number of integration points applying different orders of the moment fitting quadrature.

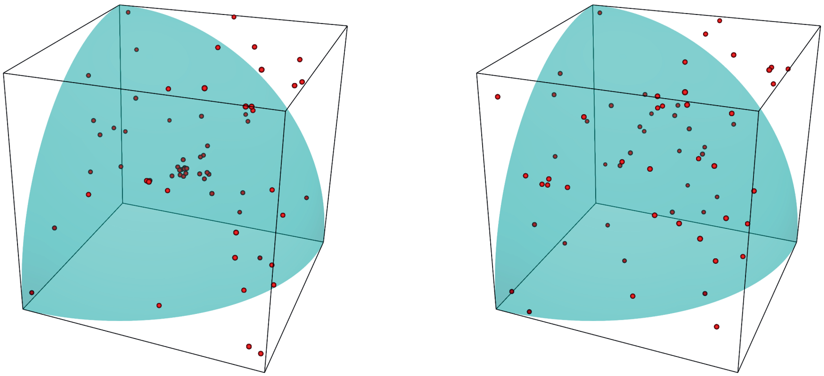


Figure 4.16: Moment fitting points for a quadrature of order $p_q = 4$ based on the optimization procedure. **(a)** Initial points based on a random distribution. **(b)** Optimized points.

4.1.6.2 Recovery of the Gauss-Legendre quadrature

In the second example, we study the influence of the moment fitting based on the GLP for the special case where the fictitious domain vanishes. Consequently, this means that the integration domain coincides with the domain of the cell, see Fig. 4.17 depicting a noncut finite cell with 125 Gauss-Legendre points. Since the integration domain is described by a regular hexahedron, the computation of the moments can be performed exactly applying standard Gauss-Legendre quadrature rules. Further, since standard Gauss-Legendre points are chosen for the position of the moment fitting points, this example is intended to study the deviation in the weights obtained by the moment fitting with the standard Gauss-Legendre weights. In doing so, we compute the relative error e_w in the weights by the following definition

$$e_w = \frac{1}{n_g}\sqrt{\sum_{i=1}^{n_g}\left(\frac{w_i^{\mathrm{GL}} - w_i^{\mathrm{MF}}}{w_i^{\mathrm{GL}}}\right)^2} \quad . \tag{4.26}$$

Here, w_i^{GL} denotes the standard Gauss-Legendre weight at point $\mathbf{x}_i$ and w_i^{MF} is the corresponding moment fitting weight at point $\mathbf{x}_i$. Fig. 4.18 shows the relative error in weights considering different orders of the moment fitting quadratures $p_q = 0, 1, \ldots, 16$. As it can be seen from the results, the values of the moment fitting weights are almost the same as those of the standard Gauss-Legendre weights. Consequently, this implies that the resulting moment fitting quadratures have the same accuracy as the standard Gauss-Legendre quadrature rules. This means that, for the one-dimensional case, the moment fitting quadratures are able to integrate any polynomial integrand within order $2n_g - 1$ exactly. Moreover, a mathematical explanation showing that moment fitting weights exactly result in the Gauss-Legendre weights – for the special case of a noncut integration domain

– is provided in Sec. 4.2.1.

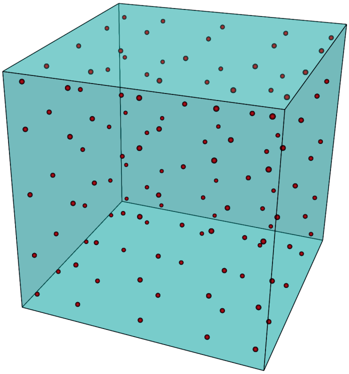


Figure 4.17: Cell by an infinitesimal fictitious domain including 125 Gauss-Legendre points.

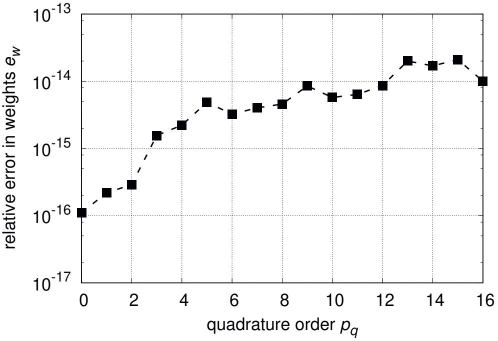


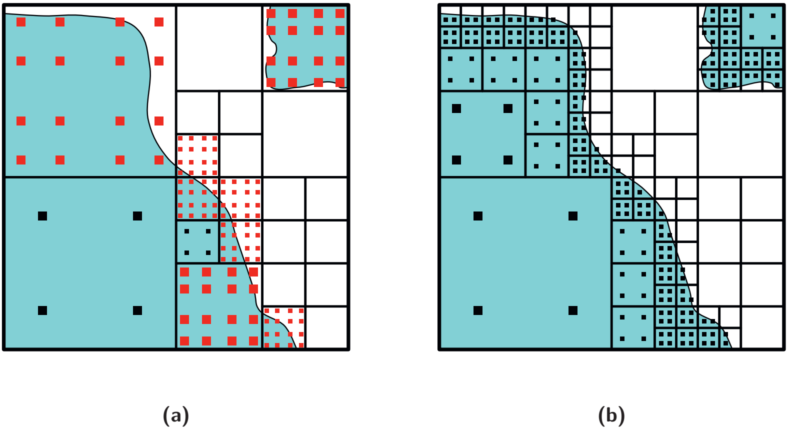
Figure 4.18: Relative error in weights applying different orders of the moment fitting quadratures.

4.2 Adaptive moment fitting

The moment fitting approach based on the GLP shows an excellent performance for linear applications in structural mechanics, see [128]. Considering nonlinear applications, however, there exist only a few examples in which the moment fitting performs as robust as the adaptive Gaussian quadrature scheme – where the numerical integration is performed on a quadtree (2D) or an octree (3D) mesh. In [33], it was shown that the Newton-Raphson method fails more often if the the moment fitting is applied. This reduction in the robustness of the Newton-Raphson procedure for nonlinear applications appears especially in cases where cut cells occur that include a small volume fraction of the physical domain. In this section, we present an adaptive integration scheme based on moment fitting in order to overcome the problem. In the following, this moment fitting approach is referred to as *adaptive moment fitting*. Further, to reduce the overhead in the generation of the moment fitting quadratures, in [31, 33], we introduced an approach that avoids the necessity to solve the moment fitting equations. This moment fitting approach is explained in more detail in Sec. 4.2.1.

In order to explain the basic idea of the *adaptive moment fitting* method, let us consider the situation depicted in Fig. 4.19 – which describes the same problem depicted in Fig. 4.1. In doing so, Fig. 4.19a shows the quadrature points of the adaptive moment fitting and Fig. 4.19b illustrates the quadrature points of the adaptive Gaussian method, respectively. For both quadrature methods we assume a geometry resolution based on a quadtree using a refinement level of $k = 4$. Here, however, as can be seen from Fig. 4.19a, the quadtree differs from the quadtree in Fig. 4.19b. While cut subcells occur only on the finest tree depth level $k = 4$ when using the adaptive Gaussian method, the adaptive moment fitting leads to cut subcells also on coarser levels $k < 4$. Further, the maximum tree depth level of the adaptive moment fitting is limited by 3 levels of refinement. Consequently, the quadtree of the adaptive moment fitting is based on another subdivision scheme than the standard quadtree. In order to distinguish between both quadtrees, we denote the tree depth level of the adaptive moment fitting as k_a and the tree depth level of the standard quadtree as k, respectively.

In the following, we explain the basic procedure of the adaptive moment fitting which



(a) (b)

Figure 4.19: Quadrature points of adaptive integration methods with $p_q = 3$. **(a)** Adaptive moment fitting method with $k_a = 1, 2, 3$. **(b)** Adaptive Gaussian method based on a quadtree using a refinement level $k = 4$.

is based on a criterion measuring the volume fraction of the integration domain. To this end, let us again consider the situation depicted in Fig. 4.19a. At first, we approximate the volume of the integration (or physical) domain $\Omega_{\square}^{\mathrm{c,phy}}$ of the cell. To this end, we employ an adaptive Gaussian integration using a quadtree with a refinement level $k = 2$ and a quadrature order of $p_q = 5$ – where points within the fictitious domain are neglected. Next, we check whether the volume fraction of the cell ($k_a = 0$) is greater or equal than a predefined threshold. If this is true, we apply the moment fitting on cell level – otherwise, we subdivide the cell into subcells, as depicted in the figure. Then, we check the volume fraction of the integration domain on each cut subcell on level $k_a = 1$. In doing so, cut subcells with a volume fraction smaller than a given threshold are further subdivided. Otherwise, we apply the moment fitting – as it is the case for the subcell on the upper left corner. This subdivision scheme is repeated until there is no remaining subcell with a smaller volume fraction than the predefined threshold, or until we reach the maximum tree depth level $k_a = 3$. Finally, for all cut cells that are left on level $k_a = 3$ we apply the moment fitting, as sketched in the figure. Moreover, on noncut subcells, we perform the numerical integration using standard Gaussian quadrature rules. In order to distinguish between the points provided by the two different quadrature rules, moment fitting quadrature points are marked by red dots and standard Gaussian quadrature points are denoted by black dots. Moreover, we employ different threshold values in order to decide whether a cell or a subcell has to be subdivided further. We choose a threshold of 0.85 for tree depth level $k_a = 0$, and we use 0.7 as the threshold for the volume fraction for $k_a = 1, 2$. These values are determined based on numerical experiments. Note that, in order to approximate the integration (or physical) domain of cut subcells, we again employ an adaptive Gaussian integration scheme with $k = 2$ and $p_q = 5$. Further, considering the performance of the different quadrature methods, we can see that the adaptive moment fitting leads to more efficient quadratures as compared to the adaptive Gaussian scheme. This is due to the fact that the adaptive moment fitting results in a lower number of subcells and, thus, in a lower number of integration points – which is in particular the case for the subcell on the top left where the moment fitting is applied on tree depth level $k_a = 1$, for instance.

4.2.1 Moment fitting without solving an equation system

In this section, we present a moment fitting approach that circumvents having to solve an equation system which is, in general, the most expensive part in the generation of the moment fitting quadratures of higher order. The main idea of this approach, which was first introduced in [33], is based on replacing the Legendre basis given in Eq. (4.10) by an equivalent basis composed of Lagrange polynomials. Thereby, the Lagrange polynomials are constructed using the standard Gauss-Legendre points (GLP) as the supporting points.

In the following, the usage of the Lagrange basis in combination with the Gauss-Legendre points (GLP) as the preselected position of the moment fitting points is described in more detail. For this purpose, the approach is explained on cell level ($k_a = 0$) with the integration (or physical) domain $\Omega_{\square}^{\mathrm{c,phy}}$ and for the one-dimensional case ($d = 1$) in order to simplify the notation. Its extension to the two- and three-dimensional case ($d = 2$ and $d = 3$), however, as well as its application on subcell level ($k_a > 0$) is straightforward. In doing so, the corresponding basis functions for the 1D case are defined as

$$\varphi_j(\xi) = l_j(\xi) \quad \text{with} \quad l_j(\xi) = \prod_{\substack{k=1 \\ k \neq j}}^{p_q+1} \frac{\xi - \xi_k^{\mathrm{GL}}}{\xi_j^{\mathrm{GL}} - \xi_k^{\mathrm{GL}}} \tag{4.27}$$

where $l_j(\xi)$ are the $p_q + 1$ distinct Lagrange polynomials and ξ_j^{GL} as well as ξ_k^{GL} denote the position of the $p_q + 1$ distinct Gauss-Legendre points. Inserting the Lagrange basis in Eq. (4.6) and using $w(\xi) = 1$, the moment fitting equations then read

$$\sum_{i=1}^{p_q+1} l_j\left(\xi_i^{\mathrm{GL}}\right)\lambda_i = \int\limits_{\Omega_\square^{c,\mathrm{phy}}} l_j(\xi)\,\mathrm{d}\Omega \quad \text{with} \quad j = 1, \ldots, p_q + 1 \quad . \tag{4.28}$$

Since we use the Gauss-Legendre points as the supporting points of the Lagrange polynomials, we can take advantage of the Kronecker delta property, so that the coefficient matrix $\mathbf{A}$ results in the identity matrix

$$A_{ji} = l_j\left(\xi_i^{\mathrm{GL}}\right) = \delta_{ji} \quad . \tag{4.29}$$

Consequently, the corresponding $p_q + 1$ moment fitting weights λ_i may be obtained directly by computing the integrals of the individual Lagrange polynomials as

$$\lambda_i = \int\limits_{\Omega_\square^{c,\mathrm{phy}}} l_i(\xi)\,\mathrm{d}\xi \quad , \tag{4.30}$$

or in standard matrix notation as

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{p_q+1} \end{bmatrix} = \begin{bmatrix} \int_{\Omega_\square^{c,\mathrm{phy}}} l_1(\xi)\,\mathrm{d}\xi \\ \vdots \\ \int_{\Omega_\square^{c,\mathrm{phy}}} l_{p_q+1}(\xi)\,\mathrm{d}\xi \end{bmatrix} \quad . \tag{4.31}$$

Thus, the moment fitting approach avoids the necessity to solve a linear equation system, which allows to reduce the overhead in the generation of the quadrature rules significantly – since the solution of the system of equations, generally, represents the most expensive part.

For the special case where the cell is not cut, note that Eq. (4.30) reads

$$\lambda_i = \int_{-1}^{1} l_i(\xi)\,\mathrm{d}\xi \quad , \tag{4.32}$$

which is exactly the same expression as in Eq. (3.42) used for the computation of the Gauss-Legendre weights on interval $(-1, 1)$ with the positive weight function $w(\xi) = 1$. Consequently, in this particular case, the moment fitting quadrature achieves the optimal quadrature order $p_q = 2n_g - 1$. Thus, the proposed moment fitting method can be interpreted as a general form of the Gaussian quadrature where the integration domain within the interval $(-1, 1)$ may take any arbitrary shape. For cut cells, however, the orthogonality property of the Legendre polynomials does not hold anymore and, thus, the quadrature order of the moment fitting is generally limited by $p_q = n_g - 1$.

As mentioned before, extending the proposed moment fitting approach to the multi-dimensional case is simple. For this purpose, the basis functions $\varphi_j\left(\boldsymbol{\xi}\right)$ are constructed applying a tensor product of the p_q+1 one-dimensional Lagrange polynomials, and the corresponding points $\boldsymbol{\xi}_i$ are obtained by the tensor product grid of the p_q+1 one-dimensional Gauss-Legendre points. For the 3D case ($d=3$), the basis thus is given as

$$\mathcal{F} = \{l_u(\xi)l_v(\eta)l_w(\zeta), \quad u,v,w = 1,\ldots,p_q+1\} \quad , \tag{4.33}$$

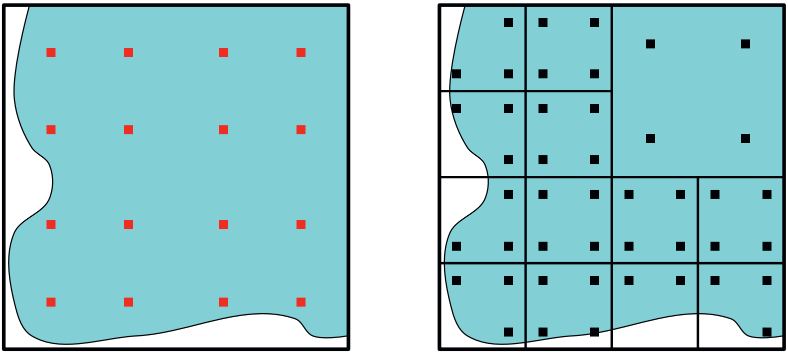
and the position of the moment fitting points as

$$\mathcal{X} = \left\{\left(\xi_u^{\mathrm{GL}}, \eta_v^{\mathrm{GL}}, \zeta_w^{\mathrm{GL}}\right), \quad u,v,w = 1,\ldots,p_q+1\right\} \quad . \tag{4.34}$$

4.2.2 Computation of the moment fitting weights

Using the aforementioned moment fitting approach, the final step missing is the computation of the moment fitting weights. In this contribution, we employ an adaptive Gaussian scheme based on a quadtree (2D) or an octree subdivision (3D) in order to perform the numerical integration of the moments – which corresponds to the weights, see Eq. (4.30). Consequently, an individual integration mesh is utilized for each cut subcell. Thereby, the tree depth level is chosen such that the total refinement level never exceeds a predefined resolution.

In the following, the computation of the moment fitting weights are explained considering the subcell on the upper right corner of the problem depicted in Fig. 4.19a. Here, the level of subcell is defined by $k_a = 2$ while the predefined resolution of the quadtree is given by 4 levels of refinement, see Fig. 4.19b. Then, for the computation of the moment fitting weights, we employ an adaptive Gaussian quadrature using a quadtree with 2 refinement levels. Consequently, the total refinement level corresponds to the predefined resolution (4 refinement levels). Further, since the maximal polynomial order of the basis functions is given by $p = 3$, we compute the moments applying a standard Gaussian quadrature using 4 Gauss-Legendre quadrature points on each subcell – where points within the fictitious domain are neglected. The moment fitting quadrature and the related adaptive Gaussian quadrature used for the computation of the weights are illustrated in Fig. 4.20a and 4.20b, respectively.



(a) (b)

Figure 4.20: Computation of the moment fitting weights of a subcell with $p_q = 3$. **(a)** Moment fitting points of the subcell. **(b)** Corresponding quadrature points of the adaptive Gaussian methods used for the computation of the moment fitting weights.

4.3 Applications to the finite cell method

In this section, we employ the presented moment fitting methods for various applications of the finite cell method considering three-dimensional problems. Thereby, we study the accuracy and efficiency as well as the robustness of the quadratures and compare the performance of the different moment fitting approaches to the adaptive Gaussian method based on an octree subdivision scheme – which is commonly applied to perform the numerical integration in the FCM.

4.3.1 Hydrostatic sphere

In the first example, we consider a sphere under hydrostatic stress state [20, 135]. The problem under investigation is depicted in Fig. 4.21a. For the analysis, the sphere is loaded on its surface by a uniform traction $\bar{t}_n = 1.0\,\mathrm{MPa}$ acting in normal direction, and the material behavior is assumed to be isotropic and linear elastic – where the Young's modulus is given as $E = 1.0\,\mathrm{MPa}$ and Poisson's ratio is $\nu = 0.3$. The geometry of the sphere is defined by the following level set function

$$\phi\left(\boldsymbol{x}\right) = \left(x - x_c\right)^2 + \left(y - y_c\right)^2 + \left(z - z_c\right)^2 - R^2 \tag{4.35}$$

with radius $R = 5.0\,\mathrm{mm}$ and fixed center coordinates $x_c = y_c = z_c = 0.0\,\mathrm{mm}$. Thanks to the simple setup of the problem, the analytical solution is defined by a linear displacement field

$$\mathbf{u} = \begin{bmatrix} Cx \\ Cy \\ Cz \end{bmatrix} \quad \text{with} \quad C = \frac{1}{E}(1-2\nu) \quad . \tag{4.36}$$

For the FCM analysis, we only have to consider one eighth of the sphere, which is due to the symmetry of the problem. Fig. 4.21b shows the FCM model and the discretization. Here, one finite cell is used for the discretization that is subjected to symmetry boundary conditions – fixing the back face of the cell in x-direction, the left face in y-direction, and the bottom face z-direction. Note that for the FCM only, in contrast to the standard finite element method, one finite cell with an ansatz order $p = 1$ is sufficient to solve the problem, provided that the numerical integration is carried out accurately enough. Further, in order to account for the inhomogeneous Neumann boundary conditions acting on the curved surface of the eighth of the sphere, we use the same procedure as proposed in [20]. In doing so, we employ a parametric description based on a triangulation. Then, the computation of the load vector is carried out applying a numerical integration on every triangle. We use the triangulation depicted in Fig. 4.22a by only taking the triangles of the curved surface into account.

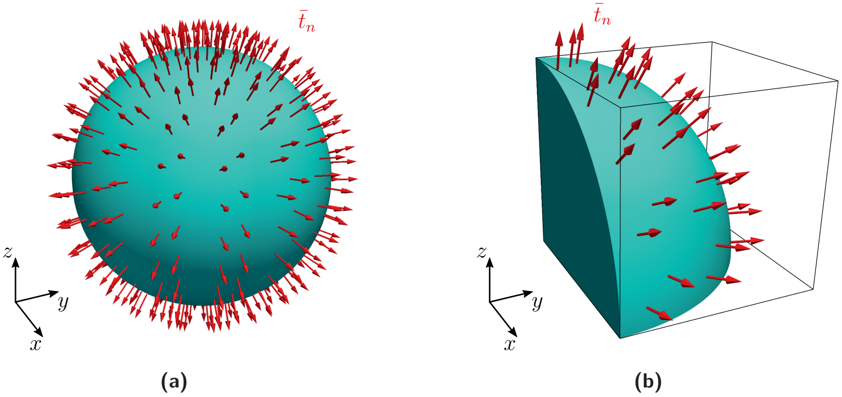


Figure 4.21: Sphere under hydrostatic stress. Geometry and boundary conditions.

For the numerical integration of the stiffness matrix, we employ the presented moment fitting methods and compare their performance with the adaptive Gaussian integration, which is usually used within the context of the FCM. In order to apply the moment fitting a geometric parameterization of the physical domain is needed to compute the moments. Here, we compute the moments by employing a triangulated surface mesh using $525,718$ triangles. Fig. 4.22a shows the utilized surface mesh, which is composed of a coarse triangulation for the planar faces and fine triangulation to capture the curved surface of the sphere. The triangular mesh results in a relative error in volume of about 10^{-6}. For the adaptive Gaussian integration based on an octree subdivision, we apply a quadrature order of $p_q = 2$ on each subcell. Further, in order to achieve the same error level as provided by the triangulation, we study the relative error in volume for different refinement levels of k – see Fig. 4.23. Here, each black dot represents a refinement level k starting from $k = 1$. As it can be seen from the figure, employing an octree with $k = 8$ refinement levels results in the same error level where the total number of integration points is $2,061,424$. Fig. 4.22b shows the corresponding subcell mesh of the octree subdivision based on 8 refinements.

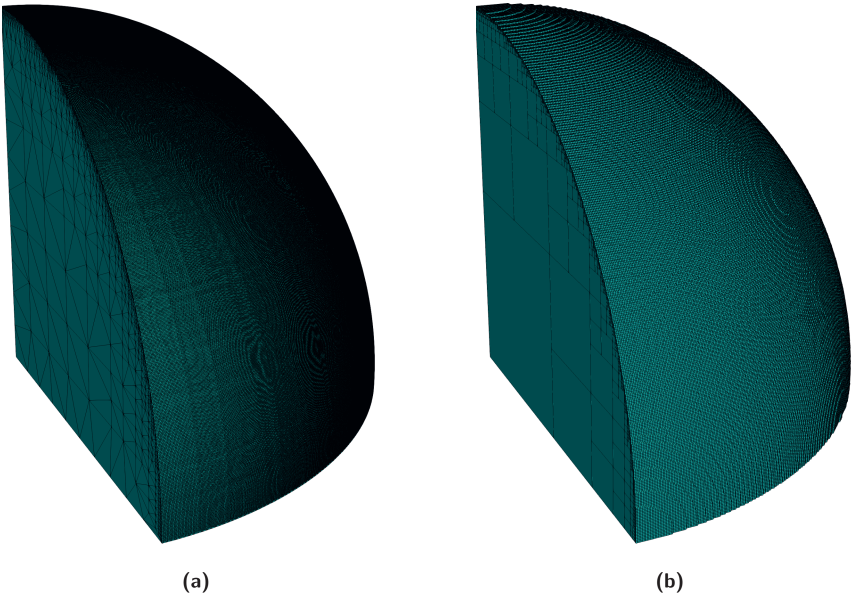


Figure 4.22: Geometry approximation of the eighth of the sphere used for the numerical integration. **(a)** Triangulated surface using $525,718$ triangles. **(b)** Octree mesh with 8 refinement levels.

In order to demonstrate the accuracy of the different quadratures, we study the error in von Mises stress σ_{vM}. Due to the setup of the problem, the sphere experiences a hydrostatic stress state. Consequently, the deviatoric part of the *Cauchy stress* $\boldsymbol{\sigma}$ vanishes. As a result, the von Mises stress σ_{vM} vanishes as well. Fig. 4.24a and 4.24b show the contour plots of σ_{vM} applying the different moment fitting methods and the adaptive Gaussian integration, respectively. In order to distinguish the different moment fitting approaches from each other, APD denotes the moment fitting employing the adaptive point distribution scheme, GLP is based on the standard Gauss-Legendre points, and OP represents the optimized points and weights. As it can be seen from the figures, the moment fitting quadratures lead to more accurate results in the von Mises stress as compared to the adaptive Gaussian quadrature.

In order to study the accuracy of the different integration methods in more detail, we next consider the relative error in the von Mises stress

$$e_{vM} = \left| \frac{\sigma_{vM}}{\bar{t}_n} \right| \tag{4.37}$$

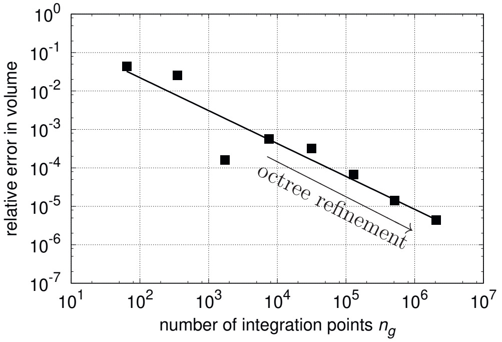


Figure 4.23: Relative error in volume, applying the adaptive Gaussian integration based on an octree subdivision considering different refinement levels k and a quadrature order of $p_q = 2$

along a diagonal cut line of the cell. The definition of the cutline is given as

$$x = y = z = r \quad \text{with} \quad 0.0\,\text{mm} \leq r \leq \frac{R}{\sqrt{3}} \quad . \tag{4.38}$$

The values of the relative error in the von Mises stress are plotted in Fig. 4.25. The figure shows that the error obtained by the moment fitting integration methods is almost zero, reaching a maximum error level of about 10^{-12}. The error achieved by the adaptive Gaussian integration, on the other hand, is higher by several orders of magnitude. Consequently, the moment fitting provides more accurate results.

To demonstrate the efficiency of the integration methods, let us take a look at the total number of integration points which are listed in Tab. 4.2. From the table, we can see that the moment fitting methods allows to reduce the number of integration points by 5 orders of magnitude, thus resulting in much more efficient quadrature rules. Moreover, the moment fitting based on optimized points and weights allows to further reduce the number of points. In order to provide a figurative imagination, the points of the different integration methods are illustrated in Fig. 4.26.

Table 4.2: Number of integration points

ansatz order p	n_g^{OT}	n_g^{APD}	n_g^{GLP}	n_g^{OP}
1	2,061,424	27	27	8

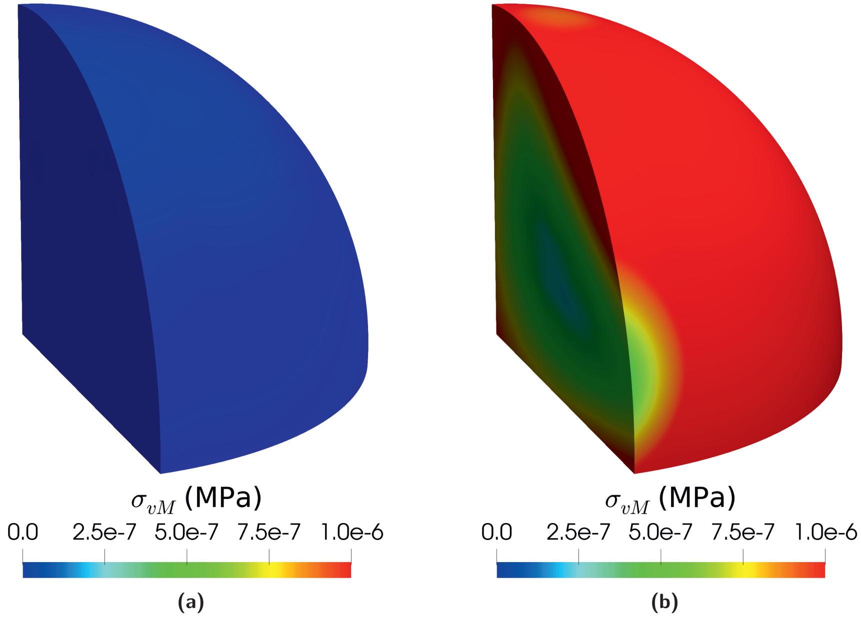


Figure 4.24: Contour plots of the von Mises stress. **(a)** Moment fitting for the triangulated surface mesh using APD, GLP, and OP. **(b)** Adaptive Gaussian integration using an octree mesh based on 8 refinement levels.

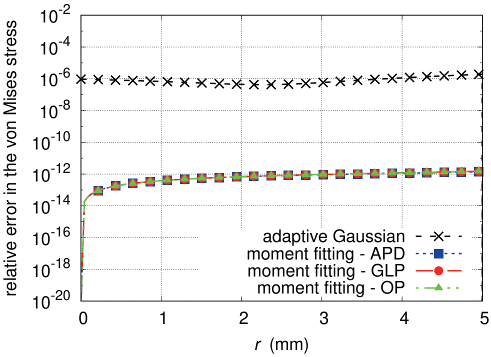


Figure 4.25: Relative error in the von Mises stress along the diagonal cutline.

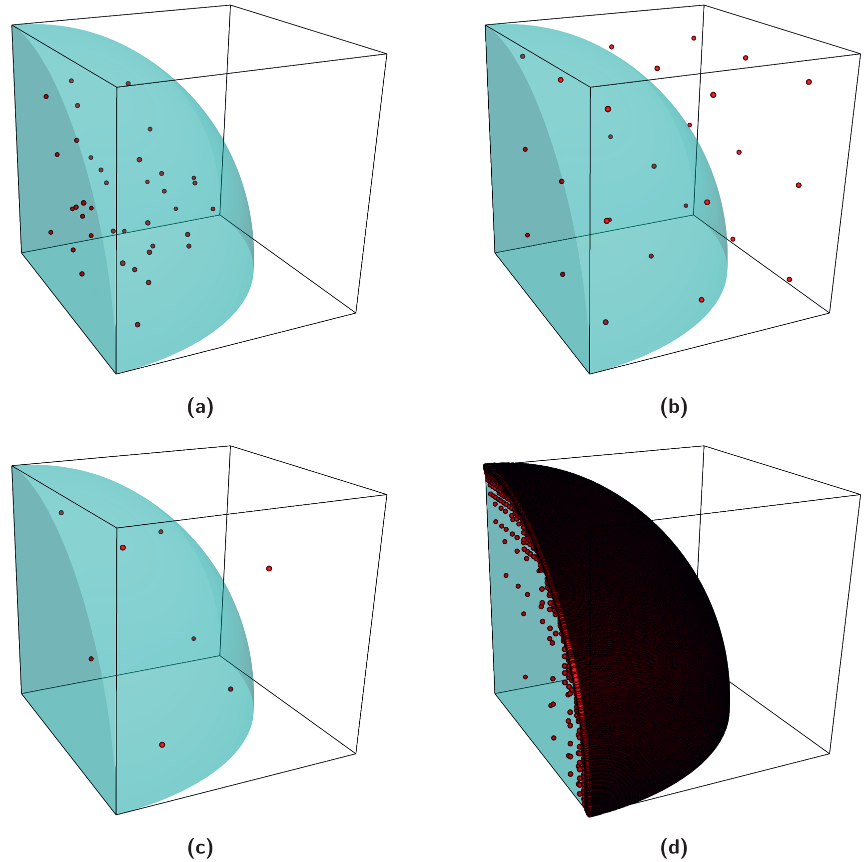


Figure 4.26: Integration points of the different quadratures. **(a)** Moment fitting using APD. **(b)** Moment fitting using GLP. **(c)** Moment fitting using OP. **(d)** Adaptive Gaussian integration using an octree mesh based on 8 refinement levels.

4.3.2 Porous material

In the next example, we study the performance and robustness of the moment fitting quadratures considering a more complex problem. To this end, we consider a porous material [127, 129, 136]. Fig. 4.27 shows the geometry of the problem under investigation. Here, we consider a cube including 27 ellipsoidal holes. The cube is described by a size of $10 \times 10 \times 10\text{mm}^3$ and the ellipsoidal holes are randomly distributed in the interior. For the investigation, we consider two cases. In the first test case, we employ an isotropic linear elastic material model, and an elastoplastic material behavior considering small strains is applied in the second test case.

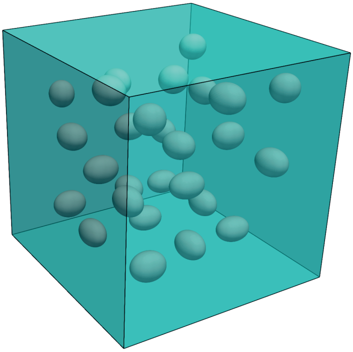


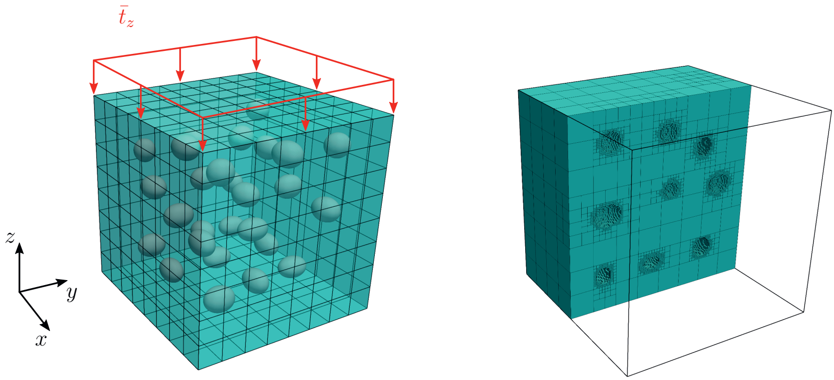
Figure 4.27: Porous material.

4.3.2.1 Linear elasticity

Fig. 4.28a shows the FCM model and the discretization of the first test case. Here, we employ a Cartesian grid using 512 finite cells of which 175 are intersected by the pores. For the material behavior of the porous domain, a linear elastic and isotropic material with Young's modulus $E = 5.0\,\text{GPa}$ and Poisson's ratio $\nu = 0.3$ is assumed. Further, the domain is subjected to symmetry boundary conditions – fixing the back face of the cube in x-direction, the right face in y-direction, and the bottom face in z-direction. A uniform pressure of $\bar{t}_z = 100.0\,\text{MPa}$ is applied at the top surface of the cube, acting in negative z-direction.

For the numerical integration of the stiffness matrix, we employ the moment fitting and the adaptive Gaussian integration using an octree with $k = 4$ refinement levels. Fig. 4.28b shows the corresponding octree mesh for a cut through the porous domain. The computation of the moments is carried out using the same adaptive Gaussian integration. Then, provided that the moment fitting weights can be computed with sufficient precision, both quadratures achieve the same accuracy.

Next, in order to investigate the performance of the various integration methods, we study the relative error in strain energy considering different orders of the ansatz. The reference value 1.065820653 J of the strain energy is obtained from an overkill solution. Fig. 4.29 shows the results of the various integration methods. Here, the error is plotted



(a) **(b)**

Figure 4.28: Porous material. **(a)** FCM model and discretization. **(a)** Octree mesh with 4 refinement levels.

regarding the number of degrees of freedom in a double logarithmic diagram. As it can be seen from the figure, the results of the moment fitting methods and the adaptive Gaussian scheme are in good agreement. However, the moment fitting based on the optimized points and weights (OP) is only studied for $p = 1, \ldots, 4$. This is because the optimization procedure becomes more and more expensive with increasing order of the ansatz. Further, the moment fitting based on the APD is only studied for $p = 1, \ldots, 8$. This is due to the fact that for cut cells including a small physical domain the points provided by the APD results in a bad condition number of the coefficient matrix, which is why the system cannot be solved with sufficient accuracy anymore. Consequently, the moment fitting based on the GLP is the preferable choice. Moreover, note that the moment fitting weights can be computed directly if the GLP is used together with the Lagrange basis. Thus, we circumvent having to solve the moment fitting equation system.

Finally, we investigate the efficiency of the different integration methods. To this end, the total number of quadrature points is plotted in Fig. 4.30. Here, it can be seen that the moment fitting allows to reduce the number of points by one order of magnitude.

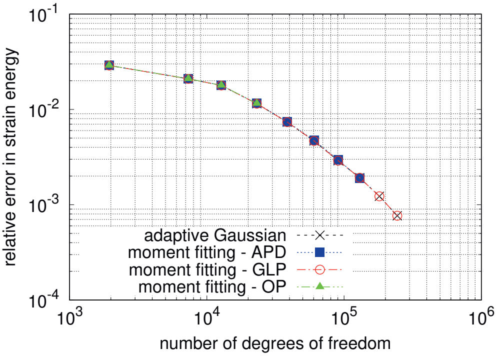


Figure 4.29: Relative error in strain energy applying different orders of the ansatz.

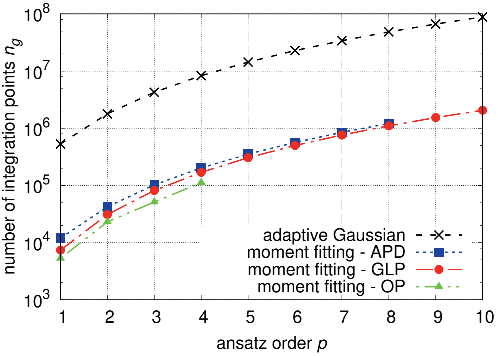


Figure 4.30: Total number of integration points applying different orders of the ansatz.

4.3.2.2 Small strain elastoplasticity

In the second test case, we study the performance of the moment fitting considering a nonlinear material behavior. The FCM model and the discretization depicted in Fig. 4.31a. For the spatial discretization, we again employ the Cartesian grid utilizing 512 finite cells of which 175 are cut. Concerning the material behavior, we assume an elastoplastic material model based on the J_2 flow theory for small strains with nonlinear isotropic hardening. A brief description of the material model is provided in Sec. 2.3.3. The elastoplastic material parameters are listed in Tab. 4.3. For the boundary conditions, we fix the back face of the cube in x-direction, the right face in y-direction, and the bottom face in z-direction. Further, a prescribed displacement $\bar{u}_z = 0.5\,\mathrm{mm}$ is applied on the top surface of the cube, acting in positive z-direction.

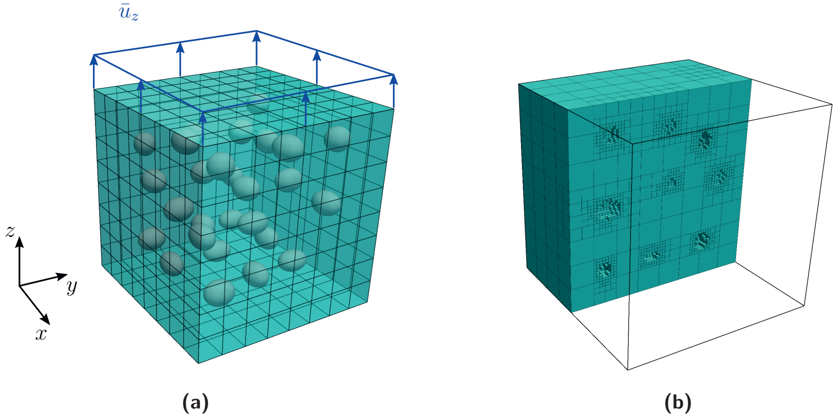


Figure 4.31: Porous material. **(a)** FCM model and discretization. **(a)** Octree mesh with 3 refinement levels.

Table 4.3: Elastoplastic material parameters.

parameter	variable	value	unit
bulk modulus	K	164.206	GPa
shear modulus	μ	80.194	GPa
initial yield strength	σ_0	450.0	MPa
saturation strength	σ_∞	715.0	MPa
linear hardening parameter	h	129.24	MPa
hardening exponent	ω	16.93	-

For the computation of the stiffness matrix, we employ the moment fitting approach based on the GLP and the Lagrange basis as well as the adaptive Gaussian integration. Thereby, we utilize an octree with $k = 3$ refinement levels in order to reduce the computational effort of the adaptive Gaussian scheme. Fig. 4.31 shows the corresponding octree mesh for a cut through the porous domain. For the computation of the moments, we use the same adaptive Gaussian integration. Considering the efficiency of both quadrature methods, Fig. 4.32 shows the total number of integration points applying different orders $p = 3, \dots, 8$ of the ansatz quadratures. As it can be seen from the figure, the moment fitting allows to reduce the number of points by a factor of about 10 for each order of the ansatz. Moreover, note that, in contrast to the adaptive Gaussian scheme, the number of the moment fitting points is maintained with increasing refinement level of the octree.

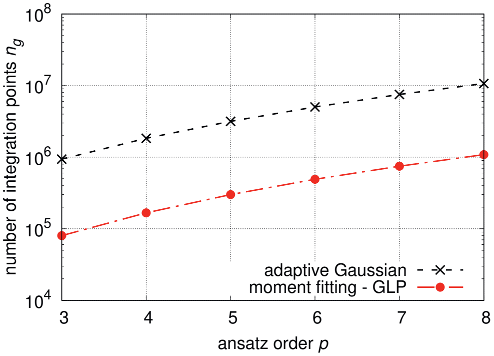


Figure 4.32: Total number of integration points applying different orders of the ansatz.

The following investigation addresses the aspect of choosing the material model for the moment fitting points that are located within the fictitious domain. For this, we consider two different versions of the moment fitting. In the first version, we assume exactly the same elastoplastic material model that is utilized for the points within the physical domain. In the second version, we simply assume a linear elastic material behavior for the moment fitting points in the fictitious domain. To this end, however, we take the same material model and assume an infinite yield stress.

Fig. 4.33 shows the load-displacement curves applying an ansatz of $p = 8$. As it can be inferred from the figure, the moment fitting of the first version (without asterisk) – assuming the same material model for the physical and the fictitious points – shows a good agreement with the results obtained by the adaptive Gaussian scheme. However, the results of the second moment fitting version (with asterisk) – assuming a simple linear elastic material behavior for the moment fitting points within the fictitious domain – deviate from the load values of the adaptive Gaussian scheme after plastic deformation arises.

Next, we investigate the effect of the different moment fitting versions on the stress behavior. For this, we consider the von Mises stress σ_{vM} of the final displacement $\bar{u}_z = 0.5\,\text{mm}$ along a diagonal cutline through the cube. In doing so, the definition of the cutline is given as

$$x = y = z = r \quad \text{with} \quad 0\,\text{mm} \leq r \leq 10\,\text{mm} \quad . \tag{4.39}$$

Fig. 4.34 shows the von Mises stress of the different integration methods. In the figure, the physical domain is highlighted by the cyan boxes and the fictitious one by the white

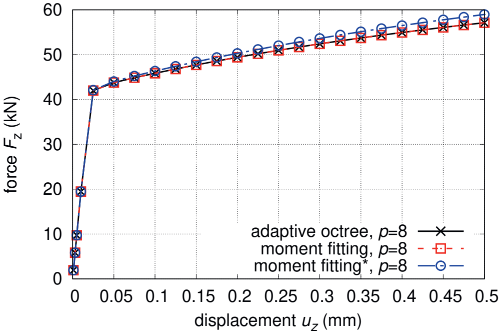


Figure 4.33: Load-displacement curves applying an ansatz of order $p = 8$.

boxes. It can be seen that the results of the first moment fitting version (without asterisk), assuming the same material behavior for all moment fitting points, are in a good agreement with the results obtained by the adaptive Gaussian scheme. Moreover, the results of the second moment versions again lead to less accurate results. Consequently, the investigations confirm that it is preferable to assume the same material model of the physical domain for the moment fitting points which are located within the fictitious domain.

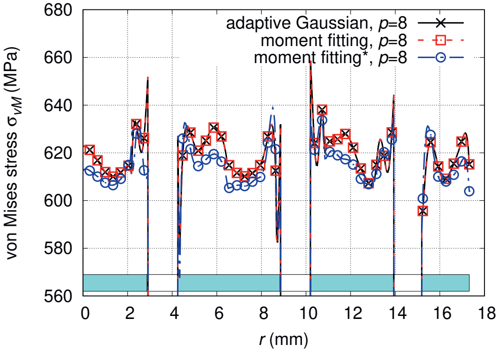
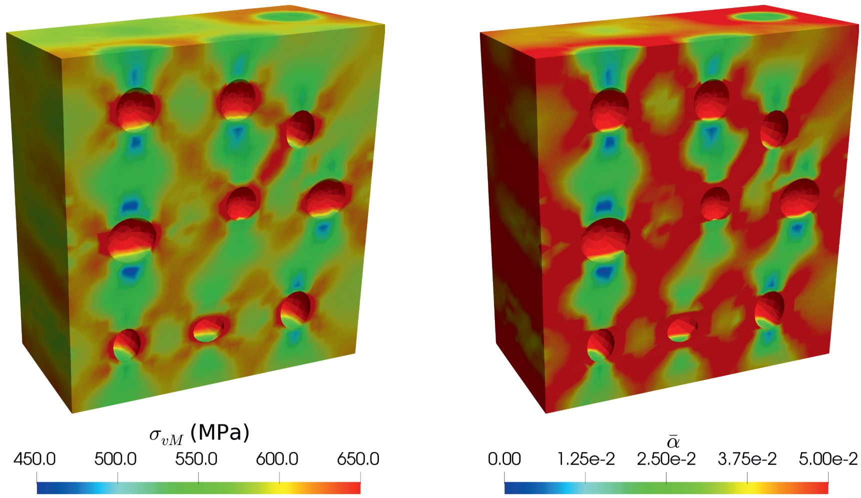


Figure 4.34: The von Mises stress σ_{vM} along a diagonal cutline for $\bar{u}_z = 0.5\,\text{mm}$ and an ansatz of order $p = 8$.

Finally, the contour plots of the von Mises stress σ_{vM} and the equivalent plastic strain $\bar{\alpha}$ for $\bar{u}_z = 0.5\,\text{mm}$ and an ansatz order $p = 8$ are illustrated in Fig. 4.35a and 4.35b, respectively. Here, it can be seen that the applied load results in a plastic zone that spreads out over the entire domain where the maximum values of the von Mises stress are located within the vicinity of the pores.



(a) **(b)**

Figure 4.35: Contour plots of the porous material for $\bar{u}_z = 0.5\,\mathrm{mm}$ and an ansatz of order $p = 8$. **(a)** The von Mises stress σ_{vM}. **(a)** The equivalent plastic strain $\bar{\alpha}$.

4.3.3 Cube with a cylindrical hole

In the previous example, the moment fitting shows the same robustness behavior for non-linear problems as the adaptive Gaussian scheme. However, this is not generally the case. To this end, the following example serves to point out the necessity of an adaptive moment fitting scheme. We consider a simple problem: a cube that is cut by a cylindrical hole. The size of the cube is given as $10 \times 10 \times 10\mathrm{mm}^3$, and the cylindrical hole is defined by the level set function

$$\phi\left(\boldsymbol{x}\right) = \left(y - y_c\right)^2 + \left(z - z_c\right)^2 - R^2 \quad . \tag{4.40}$$

The center coordinates and the radius of the cylinder read

$$y_c = 10.0\,\mathrm{mm} \quad , \quad z_c = 10.0\,\mathrm{mm} \quad , \quad \text{and} \quad R = 9.0\,\mathrm{mm} \quad . \tag{4.41}$$

Fig. 4.36a shows the FCM model and the discretization. Here, we employ one finite cell. Further, the cube is subjected to symmetry boundary conditions: fixing the back face of the cube in x-direction, the left face in y-direction, and the bottom face in z-direction. On the top surface, we apply a prescribed displacement of $\bar{u}_z = 0.5\,\mathrm{mm}$ acting in positive z-direction. For the material behavior, we apply an elastoplastic material model based on the J_2 flow theory for small strains, see Sec. 2.3.3. The elastoplastic material parameters are listed in Tab. 4.3.

For the numerical integration of the stiffness matrix, we apply the moment fitting and the adaptive moment fitting using the approach based on the GLP and Lagrange polynomials. In doing so, we study the performance of the two moment fitting schemes and compare it with the performance of the adaptive Gaussian integration. Further, for the resolution of the physical domain, we utilize an octree mesh based on $k = 5$ refinement levels. This octree mesh is also the starting point for the computation of the moments in order to achieve quadratures of the same accuracy. The subcell mesh of the corresponding octree is illustrated in Fig. 4.36b.

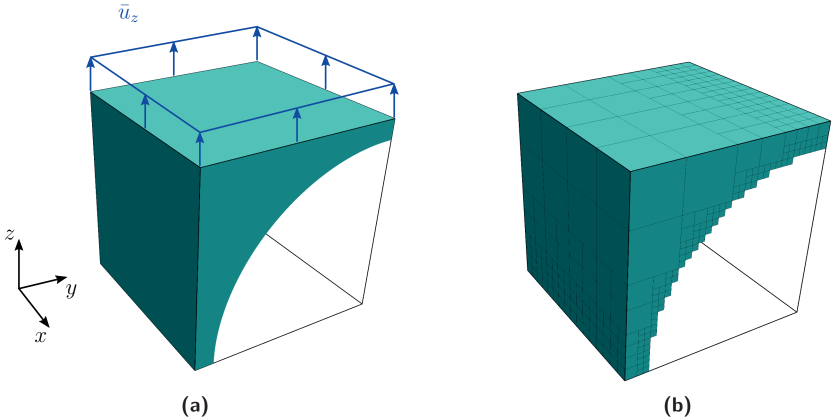


Figure 4.36: Cube with a cylindrical hole. **(a)** FCM model and discretization. **(a)** Octree mesh with 5 refinement levels.

Next, we study the robustness of the different integration methods. To this end, let us take a look at Fig. 4.37 where the load-displacement curves are plotted applying an ansatz of order $p = 8$. As it can be inferred from the figure, the moment fitting method performs less robust. Here, we have to use $\alpha = 10^{-q}$ with parameter $q = 2$ in order to increase the robustness of the FCM analysis and, thus, to reach the final load step. However, the adaptive moment fitting as well as the adaptive Gaussian integration show a much better behavior. Here, a parameter $q = 12$ is sufficient to achieve the final load step. The difference in the α value has an effect on the results of the analyses. As it can be seen from the figure, the moment fitting leads to higher load values – which is due to the fact that the higher α value increases the stiffness of the cut cell significantly.

Next, we study the von Mises stress σ_{vM} along a diagonal cutline of the final load step. The definition of the cutline is given as

$$x = y = z = r \quad \text{with} \quad 0\,\text{mm} \leq r \leq 10\,\text{mm} \quad . \tag{4.42}$$

Fig. 4.38 shows the values of the von Mises stress. Thereby, the cyan boxes represent the physical domain and the white box characterizes the fictitious one. Considering the transition from the fictitious to the physical domain on the right side of the figure, it can be seen the results of the moment fitting applying a high $\alpha = 10^{-2}$ value shows a notable difference in the von Mises stress while the values obtained by the adaptive moment fitting and the adaptive Gaussian integration are in excellent agreement.

In order to study the efficiency of the different integration methods, let us have a look at Fig. 4.39. Here, the total number of the integration points n_g is plotted applying various

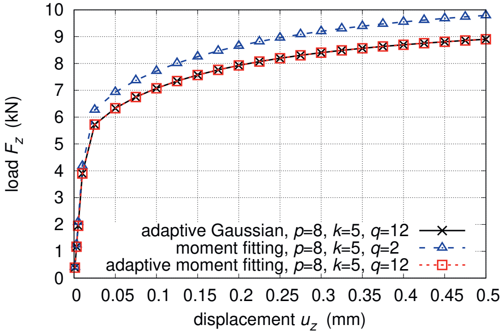


Figure 4.37: Load-displacement curves applying an ansatz of order $p = 8$.

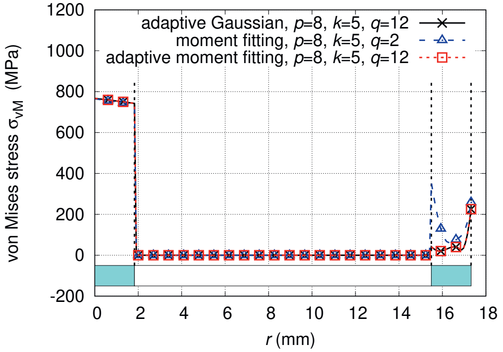


Figure 4.38: The von Mises stress σ_{vM} along a diagonal cutline for $\bar{u}_z = 0.5\,\mathrm{mm}$ and an ansatz of order $p = 8$.

orders of the ansatz $p = 1, \dots, 8$. Thereby, we consider two different resolutions of the octree $k = 5, 6$ in order to resolve for the physical domain. From the figure, it can be seen that the number of the points applying the moment fitting as well as the adaptive moment fitting does not increase with increasing tree depth level k of the octree, since the octree mesh is only used to compute the moment fitting weights.

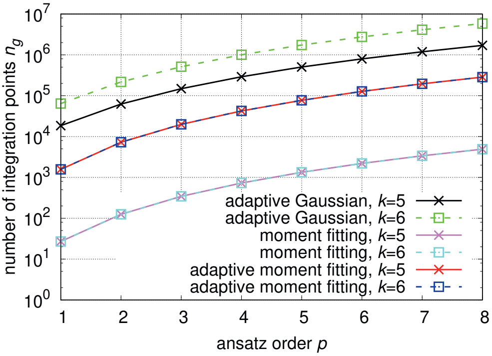


Figure 4.39: Total number of integration points applying different ansatz orders p.

Consequently, considering the performance of the different integration methods in terms of robustness, accuracy, and efficiency, we can conclude that the adaptive moment fitting is the preferred numerical integration method for nonlinear applications of the FCM.

4.3.4 Thick-walled plate with a circular hole

In the last example, we study the performance of the adaptive moment fitting investigating a square thick-walled plate with a circular hole that was also defined as a benchmark in [80]. In the context of the FCM, this problem has been studied in more detail by Abedian et al. in [28, 29].

The FCM model and the discretization are depicted in Fig. 4.40. Taking advantage of the symmetry of the problem, only one eighth has to be considered. The length of the square plate is given as $a = 100.0\,\text{mm}$ and it has a thickness of $t = 10.0\,\text{mm}$. The circular inclusion is defined by the level set function

$$\phi(\boldsymbol{x}) = (x - x_c)^2 + (y - y_c)^2 - R^2 \quad , \tag{4.43}$$

where the radius is $R = 10.0\,\text{mm}$ and the center coordinates are given as $x_c = 100.0\,\text{mm}$ and $y_c = 0.0\,\text{mm}$. Further, the plate is subjected to symmetry boundary conditions by fixing the back face of the plate in z-direction, the right face in x-direction, and the bottom face in y-direction. A uniform pressure load $\bar{t}_y = 100.0\,\text{MPa}$ is applied to the top face of the plate, acting in negative y-direction. For the analysis the pressure is monotonously raised up to a load factor of $\lambda = 4.15$ using 61 load steps. Further, for the spatial discretization, we employ a structured mesh using 399 finite cells of which 3 are cut by the physical boundary, see Fig. 4.40.

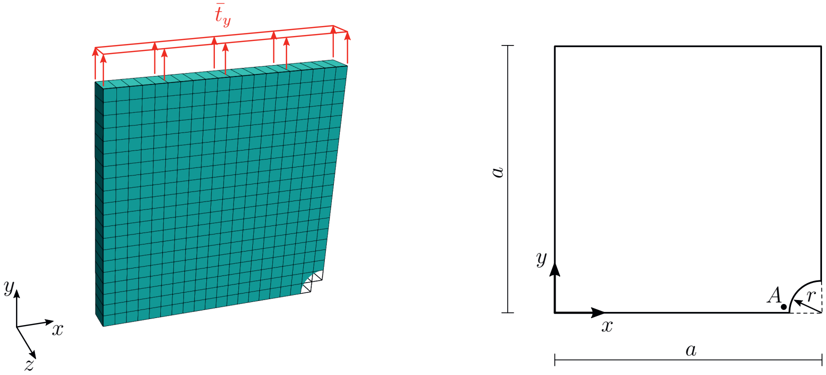


Figure 4.40: Plate with a circular hole. Geometry and boundary conditions.

For the numerical integration of the stiffness matrix, we employ the adaptive moment fitting as well as the adaptive Gaussian scheme – since the moment fitting performs less robust, as demonstrated in the previous example. Both quadratures are based on an octree resolution using $k = 5$ refinement levels. Further, to increase the robustness of the FCM analysis, we use $\alpha = 10^{-q}$ where the parameter is chosen as $q = 5$.

At first, we study the efficiency of both integration methods. For this, the total number of the integration points is plotted in Fig. 4.41 applying various ansatz orders $p = 3, \ldots, 8$. From the figure, we can see that the adaptive moment fitting allows to reduce the number of integration points by approximately one order of magnitude.

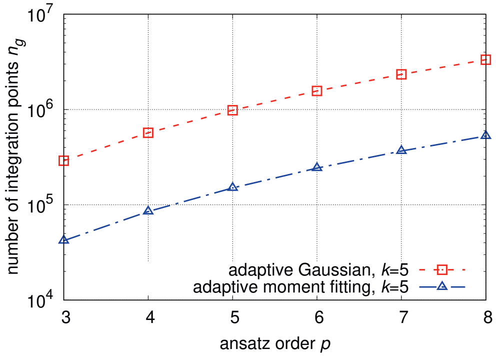


Figure 4.41: Total number of integration points applying different ansatz orders p.

In order to study the accuracy, we investigate the displacement u_x and the stress σ_{xx} at point A. The coordinates of A are given as $(89.98\,\text{mm}, 0.01\,\text{mm}, 0.13\,\text{mm})$. The results of the displacement and the stress are plotted in Fig. 4.42 and 4.43 for each load step applying an ansatz order $p = 8$. The reference solution is obtained by Wieners [80]. Comparing the results, we can see that the FCM solutions obtained by both integration methods are in a good agreement with the reference solution. Here, only small deviations occur. Moreover, the values achieved by the adaptive moment fitting and the adaptive Gaussian scheme show an excellent agreement.

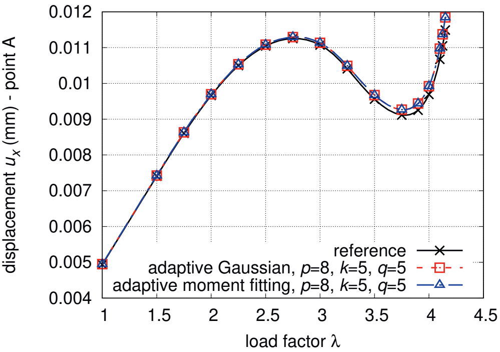


Figure 4.42: Displacement u_x at point A.

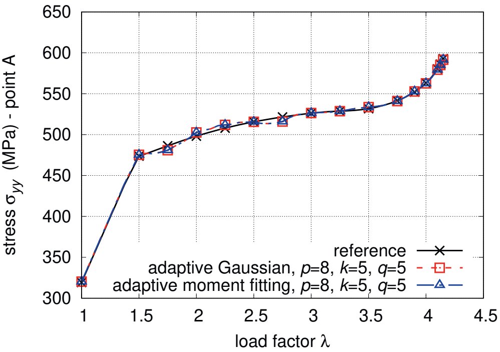
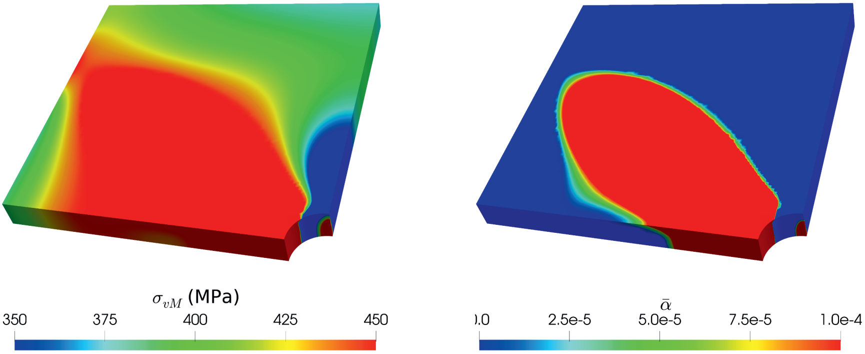


Figure 4.43: Stress σ_{yy} at point A.

Finally, Fig. 4.44a and 4.44b illustrate the contour plots of the von Mises stress and the equivalent plastic strain of the final load factor $\lambda = 4.15$, respectively. The figures show that the plastic region spreads over a large area of the plate.



(a) **(b)**

Figure 4.44: Contour plots of the porous material for $\bar{u}_z = 0.5\,\mathrm{mm}$ and an ansatz of order $p = 8$. **(a)** The von Mises stress σ_{vM}. **(a)** The equivalent plastic strain $\bar{\alpha}$.

5 Basis function removal for the FCM

Nonstandard discretization methods such as fictitious domain methods, the generalized or extended finite element method (GFEM/XFEM), or the B-Spline finite element method suffer from ill-conditioning of the resulting global stiffness matrix. The cause of this ill-conditioning behavior originates from cut finite elements/cells with a small volume fraction. As a result, these cut elements/cells lead to small entries of the stiffness matrix and, thus, to high condition numbers. Further, cut elements/cells may lead to nearly linearly dependent basis functions – which means that the global stiffness matrix does not have a positive definite character anymore. To overcome this problem of ill-conditioning, several approaches have been proposed to improve the robustness of these discretization methods.

In the context of the *finite cell method* (FCM) [19, 20] – which combines the fictitious domain technique with high order shape functions – a simple approach has been suggested that is based on a modification of the weak form. In this approach, the authors assigned a *fictitious material model* to the fictitious domain. In doing so, the conditioning behavior of the global stiffness matrix is improved by introducing artificial stiffness. Thereby, the introduction of the artificial stiffness is performed on cell level manipulating the local stiffness matrix of cut cells. In order to avoid modifying the initial boundary value problem significantly, the parameters of the *fictitious material model* have to be scaled properly. For linear elastic problems of the FCM, this scaling process is applied by assigning small values to the indicator function $\alpha = 10^{-q}$ and, then, using the reduced Young's modulus αE for the *fictitious material*. In doing so, moderate values have to be chosen for parameter q. Thereby, practical experience has shown that values within the range $q = 5, ..., 12$, usually, ensure reasonable results. For a detailed investigation of the influence of this approach on the solution the reader is referred to Dauge et al. [21]. A major advantage of the *fictitious material* approach is its simple implementation – since it is sufficient to add a new set of Gaussian points to the fictitious domain. Another benefit is that problem-dependent material models can be assigned in order to control the solution of the fictitious domain. This is especially of interest for nonlinear problems – e.g. considering elastoplasticity for small and large deformations [28–30].

Further, Burman et al. [16, 17, 137] presented the *ghost penalty* technique in order to overcome the problem in ill-conditioning of fictitious domain methods. The idea of this approach is to introduce *ghost penalty* terms into the weak form that penalize the jumps of normal derivatives between cut and neighboring noncut elements within the boundary zone of the physical domain. In doing so, the *ghost penalty* stabilization weakly enforces a higher continuity between the basis functions of these elements. Consequently, the basis functions of cut elements with small support are related to the basis functions of interior noncut elements. Summarizing, the solution of cut elements is controlled via the gradients of the neighboring noncut elements.

Another approach to improve the conditioning behavior is based on the usage of preconditioning techniques. Here, the system is manipulated in an algebraic sense, and the modified problem thus results in a well-conditioned system without modifying the solution. In the context of the extended finite element method, Béchet et al. [138] proposed a special preconditioner for enriched finite elements. In this preconditioner technique, the regular and enriched shape functions are orthogonalized by applying a Cholesky decomposition of the submatrices. Thus, the presented preconditioner can be interpreted as a local preconditioning approach. Further, the preconditioner is complemented by an optional scaling scheme applied to the diagonal terms of the submatrices. However, Menk and Bordas [139] figured out that there exist situations in which the preconditioner presented by Béchet et al. [138] does not perform well. Such situations are, for instance, if a node is exactly intersected by a crack or if a node is located at the boundary of an element. Due to this reason, Menk and Bordas [139] proposed a robust preconditioning technique based on a domain decomposition together with additional continuity constraints. The approach is similar to the finite element tearing and interconnecting method (FETI) presented by Farhat and Roux [140]. Finally, ill-conditioning is avoided by applying several Cholesky decompositions and an LQ decomposition. In [141], Lehrenfeld and Reusken presented an additive subspace preconditioner based on the theory of subspace correction methods [142]. In this preconditioning technique, a standard multigrid preconditioner is applied within the subspace of the standard basis functions, and a simple Jacobi diagonal scaling is used for the discontinuous enriched basis functions. Thereby, they were able to show that the proposed preconditioner is optimal in the sense that the resulting condition number is independent in terms of the mesh size and the location of the interface. In the context of the *finite cell method*, de Prenter et al. [69] proposed a preconditioning technique referred to as *Symmetric Incomplete Permuted Inverse Cholesky* (SIPIC). In this approach, a diagonal scaling technique is combined with an orthonormalization process. Thereby, the diagonal scaling avoids ill-conditioning induced by cut finite cells that possess a small volume fraction. Further, to prevent nearly linear dependency behavior for the basis functions of cut cells, the preconditioning technique is complemented by an orthonormalization scheme based on the Gram-Schmidt procedure.

In the context of the generalized finite element method, Babuška and Banerjee [143] presented a stabilization approach – referred to as the *stable generalized finite element method* (SGFEM) – for the one-dimensional case. This stabilization approach is based on a simple local modification of the enrichment functions, thus improving the condition number of the system significantly. Thereby, the enrichment basis is modified in such a way that the modified basis ensures the same convergence behavior as the standard GFEM basis. Gupta et al. presented in [144] an extension of the SGFEM for two-dimensional problems in elastic fracture mechanics, thereby giving a detailed investigation of the SGFEM and studying the condition number of the system and the convergence behavior of global and local quantities. In [145], Gupta et al. further developed the approach of the SGFEM for three-dimensional problems in fracture mechanics. However, in [144, 145], the authors stated that the extension of the one-dimensional SGFEM, presented in [143], to problems in higher dimensions is not trivial.

Another stabilization approach within the scope of the extended finite element method was presented by Loehnert in [146]. The stabilization technique for singular extended finite elements proposed in this work is performed on element level. Thereby, an eigenvalue decomposition of the local stiffness matrix of enriched elements is applied. In doing so, the method distinguishes between physically meaningful zero eigenmodes and eigenmodes that are numerically close to zero due to the position of the crack interface. The last-mentioned eigenmodes, which cause ill-conditioning, are stabilized in a consistent way. To this end, a fictitious stiffness is added to the affected modes of the local stiffness matrix. In order to be consistent, a correction term is added to the right-hand side of the corresponding element.

In the context of the B-spline finite element method, Höllig et al. [107, 147, 148] proposed an approach based on weighted extended B-splines, also referred to as *web-splines* or *web-method*. In this approach, a modification of the basis is introduced in order to avoid ill-conditioning of the global stiffness matrix. Thereby, intersected B-splines with a small support within the physical domain are extrapolated by the inner B-splines. In doing so, the related coefficients of the extrapolation are computed by Lagrange polynomials which are evaluated at the corresponding knots of the B-spline basis. Further, the Lagrange polynomials must have the same degree as the B-spline basis to preserve the same convergence behavior. Finally, the modified basis is used for the discretization of the problem. Moreover, B-splines which have no physical support are neglected. In [149], Sanches et al. developed an alternative approach referred to as the immersed B-spline (i-spline) finite element method. Here, basis functions that are completely outside of the physical domain as well as basis functions with a small support within the physical domain are removed. Then, the remaining basis functions are modified applying a scaling technique based on a normalization procedure. In doing so, the basis functions have a function value of one at the boundary and form a partition of unity.

The *cell aggregation* (or *agglomeration*) technique describes an approach similar to the one proposed by Höllig et al. in [107, 147, 148]. In the context of discontinuous Galerkin methods, Johansson and Larson [150] developed a *cell aggregation* technique in which cut cells with a small volume fraction are associated with neighboring cells consisting of a sufficiently high volume fraction. In doing so, shape functions of cut cells with a small support are constrained by linear combinations of the shape functions of neighboring cells with a sufficiently higher support. Consequently, the underlying idea is similar to the stabilization approach presented by Höllig et al. [107, 147, 148]. Further, in [151] Badia et al. proposed an aggregation technique, referred to as *aggregated unfitted finite element method*, that can be applied to both continuous and discontinuous formulations.

Another promising and simple approach to improve the conditioning behavior is based on removing critical basis function from the ansatz. In [152], Reusken presented a basis function removal strategy, referred to as modified XFEM space, in which two-phase incompressible flow problems are investigated. For this strategy, a criterion was developed to eliminate basis functions with a 'small support'. Depending on the choice of the critical parameter of the criterion, Reusken showed that the modified XFEM space improves the conditioning of the system, whereas the convergence behavior is only reduced slightly. Embar et al. [153] as well as Sehlhorst [154] suggested a criterion based on the volume fraction. Here, cut cells with a volume fraction less than 10^{-6} are eliminated from the ansatz. In [155], Verhoosel et al. studied the influence of the volume fraction in terms of the condition number of the global system. To improve the condition number, they suggested a strategy similar to [153] in which basis functions with a volume fraction smaller than a specified threshold are removed from the ansatz. Elfverson et al. [18] further developed the basis function removal approach – in the scope of which they proposed the strategy of introducing a criterion based on the error in the energy norm. Thereby, basis functions with a sufficiently small value in the energy norm are removed from the ansatz. To this end, the energy norm of the functions is estimated by considering the diagonal elements of the global stiffness matrix.

In this chapter, we present a basis function removal scheme applied to the hierarchical basis of the finite cell method (FCM), which can be easily combined with the *fictitious material* approach. In doing so, we introduce a criterion considering the diagonal elements of the global stiffness matrix, similar to Elfverson et al. [18]. Thereby, in order to obtain a criterion that is independent in terms of the utilized material model, we estimate the stiffness contribution of the shape functions based on the discrete gradient operator. Further, the basis is modified in such a way that the rigid body modes are maintained. This is especially of interest when considering problems in large deformations. Thanks to the structure of the hierarchical basis, the rigid body modes can be easily preserved by only removing basis functions of higher order. By considering FCM problems of different complexity, we demonstrate that the modified basis allows to improve the conditioning of the global stiffness matrix significantly, thus increasing the robustness of the FCM analyses. Furthermore, we show that the basis function removal strategy can be easily combined with the approach based on a fictitious material model – which is commonly used within the framework of the FCM. This combination of the basis function removal and the fictitious material model is especially of interest considering problems for large deformations. Moreover, the approach based on the basis function removal and the fictitious material model enables to improve the performance of the *remeshing strategy* that is introduced in Chap. 6.

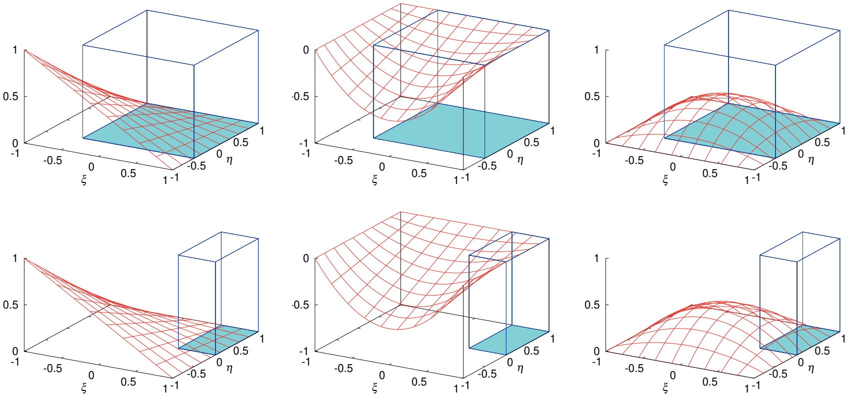
5.1 A simple function removal strategy for the hierarchical basis

In this section, we present a function removal strategy applied to the hierarchical basis of the finite cell method. Before we start to explain the basic idea of the proposed basis function removal strategy, let us briefly recall that the ill-conditioning in the FCM originates from the cut cells. To this end, Fig. 5.1 shows three representative hierarchical shape functions – nodal, edge, and internal mode – for two different cuts of one quadrilateral cell, where the cyan colored and rectangular areas characterize the physical domain and, thus, the support of the shape functions. From the figure, it can be seen that the basis functions of the cut cell in the upper row have a better support than the cut cell in the lower row.

Now, a rather simple approach would be to remove basis functions with a small support from the ansatz. However, such a criterion has two important shortcomings:

- firstly, it is performed only locally on cell level and
- secondly, it does not differentiate between the individual shape functions.

In this thesis, we consequently provide a basis function removal strategy that is based on a global criterion considering each shape function individually. Further, we ensure that the basis is modified in such a way that the shape functions representing the rigid body modes are preserved. In a first step, we thus categorize the shape functions into two groups: *affected* and *nonaffected* modes. The first group contains functions that have to be preserved (*nonaffected*), and the second group includes those functions (*affected*) that may be removed from the ansatz. In the second step, we remove those *affected modes* that do not satisfy the predefined criterion.



(a) Nodal mode **(b)** Edge mode **(c)** Internal mode

Figure 5.1: Support of hierarchical shape functions for two different cuts of a finite cell.

5.1.1 Affected and nonaffected modes of the hierarchical basis

In the following, we describe which modes of the hierarchical shape functions are *affected* and which modes are *nonaffected*. Thanks to the structure of the hierarchical basis, the rigid body modes are represented by the nodal modes (linear shape functions). This means that the nodal modes are labeled as *nonaffected* modes from the beginning. Although this implies that we have to preserve nodal modes of cut cells with a small volume fraction, we

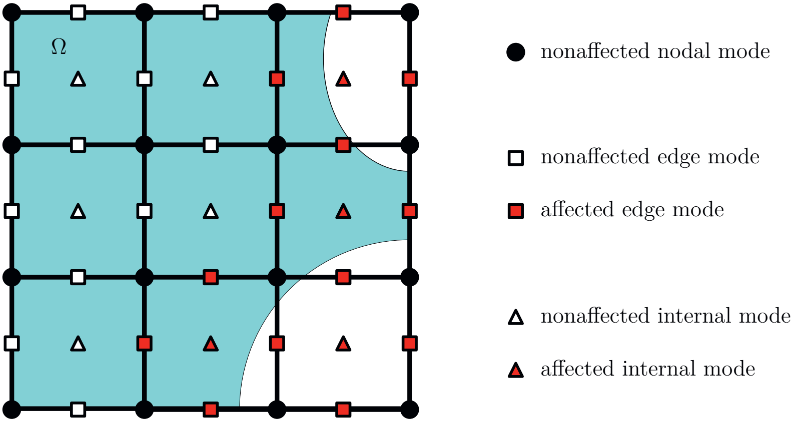


Figure 5.2: Affected and nonaffected shape functions of the hierarchical basis.

show that the presented basis function removal strategy still results in a good condition number since ill-conditioning is primarily caused by the higher order modes. For the two-dimensional case of the hierarchical basis, this means that ill-conditioning is mainly induced by the edge and internal modes of cut cells. Consequently, we have to distinguish between edge and internal modes that may be affected (removed) or not. To categorize edge and internal modes, let us consider the two-dimensional FCM problem depicted in Fig. 5.2, where the physical domain is characterized by the cyan colored area and the fictitious domain by the white area. In the figure, nodal modes are marked by dots, edge modes by squares, and internal modes by triangles. Since ill-conditioning is primarily induced by the higher order modes of cut cells, the edge and internal shape functions of these cells are labeled as *affected* modes. In the figure the *affected* modes are represented by red squares and red triangles. The remaining edge and internal modes – represented by white squares and white triangles – are labeled as *nonaffected* modes. Having categorized all *affected* shape functions, finally, we have to define a criterion that determines which of these basis functions have to be removed from the ansatz.

5.1.2 Removal criterion of affected modes

Next, we introduce a criterion that defines which of the affected shape functions have to be removed from the ansatz. Since some high order shape functions – edge modes (edge and face modes) for the two-dimensional (three-dimensional) case – may be shared by neighboring cells, we propose a criterion that eliminates affected shape functions in a global sense. The necessity of a global criterion can be explained by considering the two neighboring cut cells at the bottom of the problem depicted in Fig. 5.2, for instance. The shared edge mode is supported by one finite cell with a high volume fraction (high support) and a cell with a small volume fraction (small support). Thus, applying a local criterion based on the volume fraction would remove the edge mode from the ansatz although this edge mode may be well supported by the neighboring cell. Another reason against the usage of a volume fraction based criterion can be given by considering the cut cell on the top right of the problem. A criterion based on the volume fraction would not distinguish between the edge mode of the left and the right hand-side. For a certain threshold value, all edge modes would thus be eliminated, although the left edge mode is obviously well supported by the physical domain in a global sense.

In this thesis – due to the aforementioned reasons – we propose a global criterion by considering the diagonal elements of the global stiffness matrix, similar to Elfverson et al. [18]. To this end, let us recall the definition of the global stiffness $\mathbf{K}_T^i$ in Eq. (3.22), which is obtained during the assembly process of the local stiffness matrices $\mathbf{k}^{i,c}$. Thereby, the integrals of the local stiffness matrices are based on a product $\alpha \mathbf{G}^T \mathbf{A} \mathbf{G}$ where α is the indicator function, $\mathbf{G}$ defines the *discrete gradient operator*, and $\mathbf{A}$ denotes the tangential stiffness matrix of the material model. Since we are interested in the contribution of the shape functions to the stiffness global matrix, we develop a criterion that is based on the *discrete gradient operator* and the indicator function α. To this end, we set up a local auxiliary cell vector $\mathbf{q}^c$ estimating the stiffness contribution of the affected shape functions with respect to the physical domain

$$\mathbf{q}^c = \begin{bmatrix} q_1^1 & q_2^1 & q_3^1 & \dots & q_1^n & q_2^n & q_3^n \end{bmatrix} \quad , \tag{5.1}$$

where the entries of $\mathbf{q}^c$ are computed as

$$q_1^i = q_2^i = q_3^i = \int\limits_{\Omega_\square^c} \alpha \left(\frac{\partial N_i}{\partial X_1} \frac{\partial N_i}{\partial X_1} + \frac{\partial N_i}{\partial X_2} \frac{\partial N_i}{\partial X_2} + \frac{\partial N_i}{\partial X_3} \frac{\partial N_i}{\partial X_3} \right) \det \mathbf{J}^c \, \mathrm{d}\Omega_\square \quad . \tag{5.2}$$

Further, we compute a local auxiliary cell vector $\mathbf{h}^c$

$$\mathbf{h}^c = \begin{bmatrix} h_1^1 & h_2^1 & h_3^1 & \dots & h_1^n & h_2^n & h_3^n \end{bmatrix} \tag{5.3}$$

estimating the stiffness contribution of the affected shape functions with respect to the extended domain. Thus, the entries of $\mathbf{h}^c$ are calculated as

$$h_1^i = h_2^i = h_3^i = \int\limits_{\Omega_\square^c} \left(\frac{\partial N_i}{\partial X_1} \frac{\partial N_i}{\partial X_1} + \frac{\partial N_i}{\partial X_2} \frac{\partial N_i}{\partial X_2} + \frac{\partial N_i}{\partial X_3} \frac{\partial N_i}{\partial X_3} \right) \det \mathbf{J}^c \, \mathrm{d}\Omega_\square \quad . \tag{5.4}$$

Next, we introduce the corresponding two global auxiliary vectors $\mathbf{q}$ and $\mathbf{h}$ which are obtained during the assembly process by the local ones

$$\mathbf{q} = \mathop{\mathsf{A}}_{c=1}^{n_c} \mathbf{q}^c \quad \text{and} \quad \mathbf{h} = \mathop{\mathsf{A}}_{c=1}^{n_c} \mathbf{h}^c \quad . \tag{5.5}$$

Finally, we compute the ratio between the estimated stiffness contribution of the physical domain with respect to the extended domain

$$\mu_I = \frac{q_I}{h_I} \quad \text{where} \quad 0 < \mu_I \leq 1 \quad . \tag{5.6}$$

In Eq. (5.6), I characterizes the index of the global degree of freedom. Finally, we remove those affected basis functions that have a ratio which is smaller than a predefined threshold μ_t

$$\mu_I < \mu_t \quad \text{with} \quad 0 < \mu_t < 1 \quad . \tag{5.7}$$

Note that the integrand in Eq. (5.2) and (5.4) defines a nonnegative function. Consequently, the value of the integral in Eq. (5.2) and (5.4) increases with increasing volume fraction of a cut cell and reaches its maximum value if a cell is completely composed of physical domain. Therefore, the introduced criterion is based on a nonnegative measurement.

5.1.3 Implementation scheme

The implementation of the proposed basis function removal strategy is simple and can be considered as an additional step during the pre-processing that can be applied after the mesh generation and the setup of the basis functions, the mapping relation between local and global degrees of freedom, and the required quadrature rules. An implementation scheme is given in Alg. 1. For each finite cell, compute the entries of the local auxiliary vector $\mathbf{h}^c$ of the extended domain. If the cell is cut compute the entries of the local auxiliary vector $\mathbf{q}^c$ of the physical domain. Then, override the local entries related to the nodal shape functions of vector $\mathbf{q}^c_{N_i}$ with the entries of vector $\mathbf{h}^c_{N_i}$. If the cell is not cut, the physical domain corresponds with the cell domain and, thus, the local vectors $\mathbf{q}^c$ and $\mathbf{h}^c$

are equal. The related global vectors $\mathbf{q}$ and $\mathbf{h}$ are obtained applying the assembly process. For each global degree of freedom I, we finally compute the ratio μ_I of the physical and the extended domain. If the ratio is smaller than a given threshold μ_t, the global degree is removed by applying the general penalty method. Given the definition of the interval of μ_t in Eq. (5.7), preservation of the nodal modes is ensured by setting the vector containing the values of the nodal modes of the physical domain $\mathbf{q}^c_{N_i}$ to the vector containing the values of the embedded domain $\mathbf{h}^c_{N_i}$. Moreover, note that $\mathbf{q}^c_{N_i}$ and $\mathbf{h}^c_{N_i}$ represent a subset of the vectors $\mathbf{q}^c$ and $\mathbf{h}^c$.

Algorithm 1 Basis function removal strategy

```
for Each finite cell $c$ do
    Compute local auxiliary cell vector $\mathbf{h}^c$ of the extended domain
    if Cell $c$ is cut then
        Compute local auxiliary cell vector $\mathbf{q}c$ of the physical domain
        Set $\mathbf{q}^c_{N_i} = \mathbf{h}^c_{N_i}$
    else
        Set $\mathbf{q}^c = \mathbf{h}^c$
    end if
end for
Assemble global auxiliary vectors $\mathbf{q} = \mathop{\mathsf{A}}_{c=1}^{n_c} \mathbf{q}^c$ and $\mathbf{h} = \mathop{\mathsf{A}}_{c=1}^{n_c} \mathbf{h}^c$
for Each global degree of freedom $I$ do
    Compute ratio $\mu_I = q_I / h_I$
    if Ratio $\mu_I < \mu_t$ then
        Remove all global degrees of freedom related to the shape function $I$
    end if
end for
```

5.2 Benchmark problem

To study the performance of the proposed basis function removal strategy, let us introduce a simple benchmark problem. To this end, we consider a square thick-walled plate with a circular hole under uniform pressure. Due to the symmetry, it is sufficient to model only one eighth of the problem under investigation. The setup of the benchmark problem is given in Fig. 5.3. As it can be seen from the figure, the plate is loaded by a uniform pressure load $\bar{t}_y$ acting on top of the surface in y-direction. Further, symmetric boundary conditions are applied fixing the back face of the plate in z-direction, the right face in x-direction, and the bottom face in y-direction. The parameters describing the geometry comprising the side length a, the radius r of the circular inclusion, and the thickness t are listed in Tab. 5.1.

As the reference solution of the benchmark problem, we employ an overkill p-FEM solution based on a fine mesh using 3,200 elements and an ansatz of order $p = 5$. The p-FEM discretization is given in Fig. 5.4a. For the elements along the curved boundary, we thereby apply the blending function method [156, 157] for the mapping of the curved edges and faces. To study the performance of the basis function removal, we consider a coarse and a fine discretization for the FCM analysis. In doing so, the plate is discretized by 78

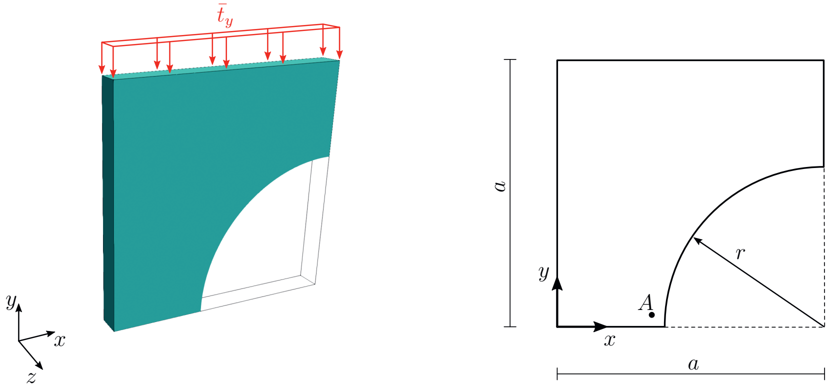
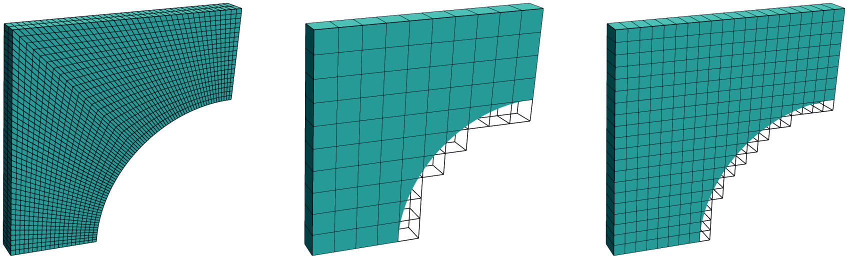


Figure 5.3: Plate with a circular hole. Geometry and boundary conditions.

Table 5.1: Geometric parameters

parameter	variable	value	unit
side length	a	100.0	mm
radius	r	60.0	mm
thickness	t	10.0	mm

finite cells for the coarse mesh and by 302 finite cells for the fine one. The discretizations of the FCM are depicted in Fig. 5.4b and 5.4c, respectively.


(a) p-FEM discretization using 3,200 elements.

(b) FCM discretization using 78 cells.

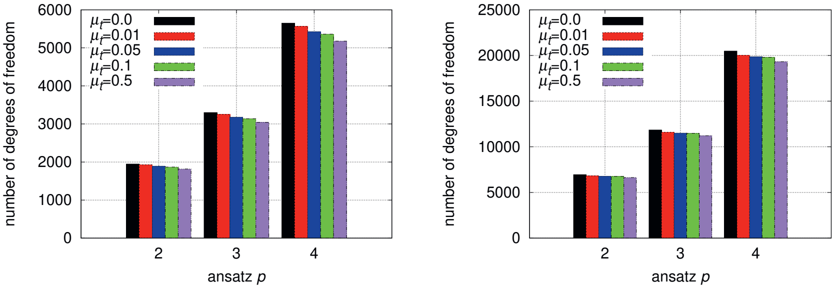
(c) FCM discretization using 302 cells.

Figure 5.4: Plate with a circular hole. Discretization used for p-FEM reference solution and FCM discretizations.

5.2.1 Linear elasticity

At first, we study the accuracy and the influence on the convergence behavior of the basis function removal assuming a linear elastic and isotropic material behavior. In doing so, the Young's modulus is given as $E = 50.0\,\mathrm{MPa}$, Poisson's ratio $\nu = 0.3$, and the value of the pressure load is $\bar{t}_y = 0.1\,\mathrm{MPa}$.

Now, to study the influence of the basis function removal, we consider different values for the criterion μ_t given in Eq. (5.7). In doing so, the number of degrees of freedom applying the coarse and the fine FCM discretization is plotted in Fig. 5.5a and 5.5b for different orders of the ansatz p, respectively. Since only high order basis functions are allowed to be removed from the ansatz, the number of degrees using $p = 1$ is not plotted. As it can be seen from the figures, the number of degrees of freedom decreases slightly with increasing values of the threshold μ_t.

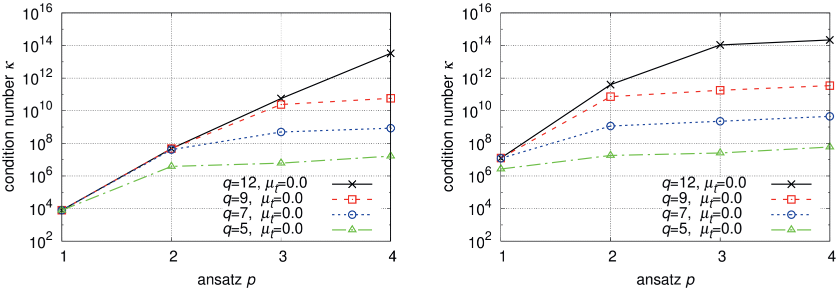


(a) FCM discretization using 78 cells.

(b) FCM discretization using 302 cells.

Figure 5.5: Plate with a circular hole. Number of degrees of freedom.

In the following, we study the influence of the basis function removal regarding the condition number κ of the global equation system for different orders of the ansatz p. For this, we first consider the case without basis function removal ($\mu_t = 0$). Thus, the conditioning behavior is improved by employing the approach based on a fictitious material only – where the Young's modulus of the fictitious domain αE is scaled by the indicator function α with $\alpha = 10^{-q}$. To this end, Fig. 5.6a and 5.6b show the evolution of the condition number for different orders of the ansatz p using $q = 12, 9, 7, 5$. As it can be seen from the figures, ill-conditioning is mainly induced by the high-order shape functions. Thus, if a low Young's modulus ($q = 12$) is used for the material of the fictitious domain, the condition number of the global stiffness matrix increases significantly with increasing order of the ansatz p. Moreover, by comparing the condition number for $p = 1$ of the coarse mesh in Fig. 5.6a with the one of the fine mesh in 5.6b, it can be seen that the conditioning of the fine mesh is worse. Here, the difference in the condition number is given by a factor between 10^2 and 10^3. This is due to the fact that more cut finite cells are present in the fine discretization. Now, in order to improve the condition number, the parameter q of the indicator function $\alpha = 10^{-q}$ is decreased systematically. Note that practical experience has shown that choosing $q = 7, ..., 5$, usually, leads to reasonable results for the FCM computations. From the figures, it can be seen that the condition



(a) FCM discretization using 78 cells.

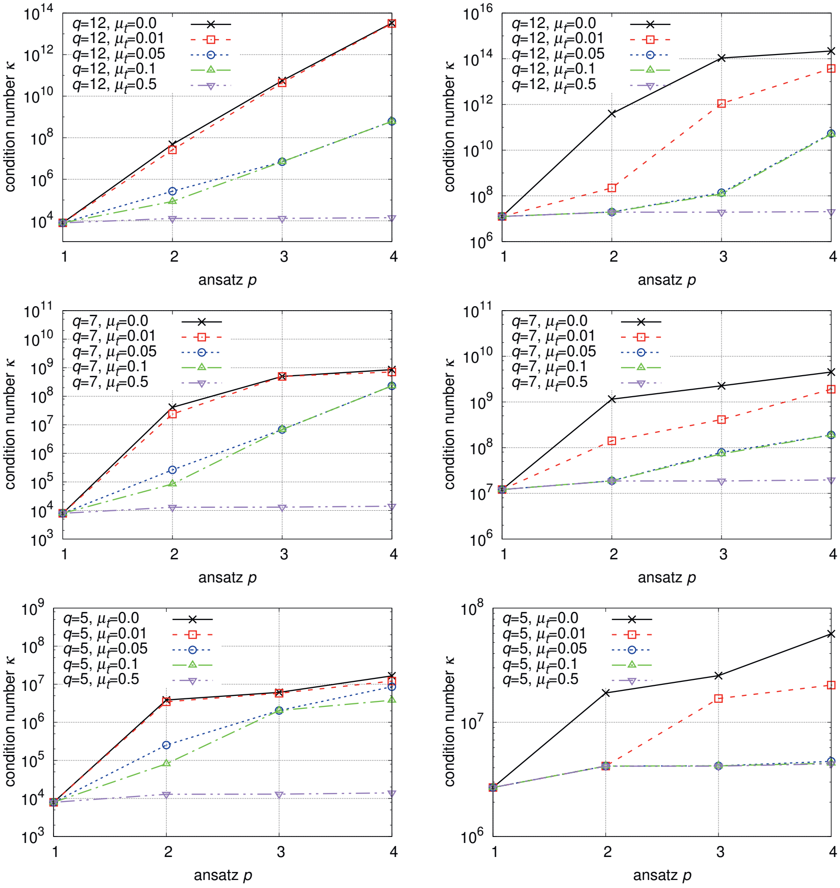
(b) FCM discretization using 302 cells.

Figure 5.6: Plate with a circular hole. Condition number of the global stiffness matrix for different values of parameter q without using basis function removal ($\mu_t = 0.0$).

number of the global stiffness matrix using an ansatz of higher order could be reduced considerably with decreasing values for the parameter q.

Next, we study the effect of the basis function removal on the condition number of the global stiffness matrix. To this end, we consider different combinations of the parameter q used to scale the material parameters of the fictitious domain with different values of the basis function removal criterion μ_t. Fig. 5.7a and 5.7b show the evolution of the condition number using the coarse and the fine discretization for the problem under investigation, respectively. Thereby, the parameter $q = 12$ assumes a fictitious material domain with a stiffness close to zero. Thus, the stiffness of the fictitious material is neglected. Further, parameter values $q = 5$ and $q = 7$ characterize typical values employed for the analysis of FCM problems. As it can be deduced from the figures, the basis function removal allows to improve the condition number of the global stiffness matrix significantly. Here, using the basis function removal could reduce the condition number of the system by several orders of magnitude as compared to the condition number without applying the basis function removal $\mu_t = 0.0$.

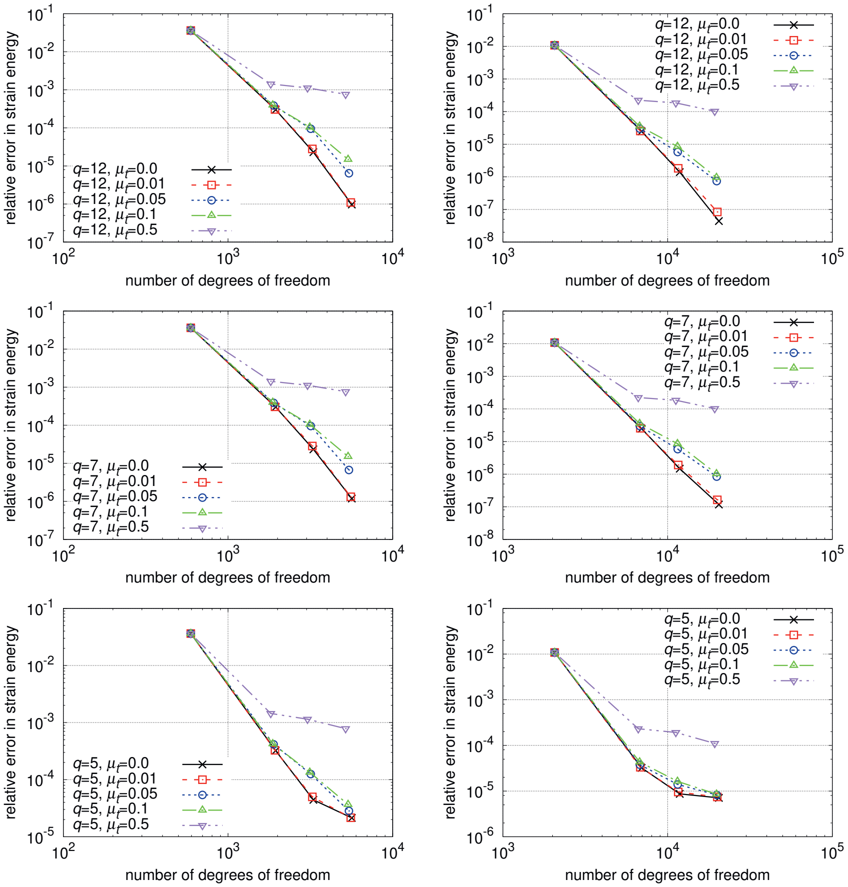
As next, we study the influence of the basis function removal on the FCM solution in terms of the accuracy and the convergence behavior. In doing so, we consider the relative error in strain energy where the reference solution is obtained by an overkill p-FEM computation applying an ansatz of order $p = 5$. The discretization of the p-FEM is given in Fig. 5.4a using 3,200 elements and the blending function method to describe the curved edges and faces along the circular boundary of the plate. Further, to avoid the occurrence of error due to numerical integration, we employ the moment fitting method given in Sec. 4.2.1. For the computation of the moment fitting weights, we thereby apply an octree integration of tree depth level $k = 10$ for the coarse mesh and an octree of tree depth level $k = 9$ for the fine mesh. Fig. 5.8a and 5.8b show the relative error in strain energy employing a coarse and a fine FCM discretization using 78 and 302 finite cells, respectively. From the figures, it can be seen that using a high value for the basis function removal criterion ($\mu_t = 0.5$) decreases the accuracy employing an ansatz of higher order deteriorating the convergence behavior. For moderate values of $\mu_t = 0.01, ..., 0.1$, however, the convergence is only reduced slightly.



(a) FCM discretization using 78 cells.

(b) FCM discretization using 302 cells.

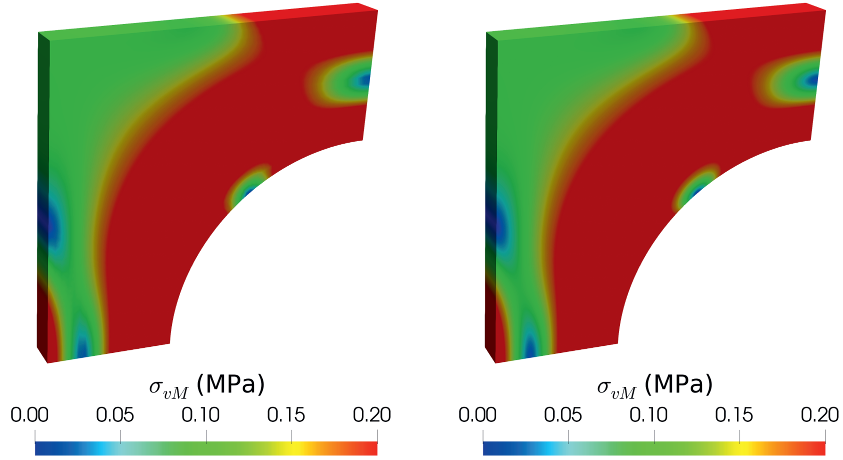
Figure 5.7: Plate with a circular hole. Condition number of the global stiffness matrix combining different values of parameter q together with different values of the basis function removal criterion μ_t.



(a) FCM discretization using 78 cells. **(b)** FCM discretization using 302 cells.

Figure 5.8: Plate with a circular hole. Relative error in strain energy.

Finally, Fig. 5.9a and 5.9b show the contour plots of the von Mises stress σ_{vM} of the p-FEM solution and the FCM solution discretized by 302 finite cells utilizing an ansatz of order $p = 4$. Further, the parameter q reads $q = 5$ and the basis function removal criterion is $\mu_t = 0.1$. Comparing both contour plots of the von Mises stress with each other, no visible difference could be observed between the FCM and the p-FEM solution.



(a) p-FEM discretization using 3,200 elements.

(b) FCM discretization using 302 cells.

Figure 5.9: Plate with a circular hole. Contour plots of the von Mises stress σ_{vM} of the p-FEM and the FCM solution.

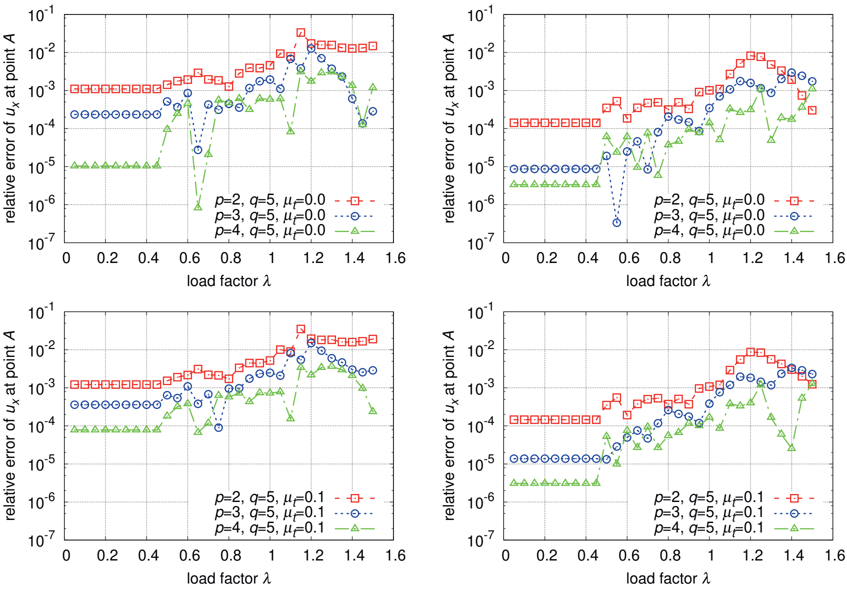
5.2.2 Small strain elastoplasticity

As the next test case for the benchmark problem, we study the influence of the basis function removal assuming an elastoplastic material behavior. In doing so, the material model is based on J_2 flow theory of plasticity for small strains with nonlinear isotropic hardening. The theory of the constitutive model is briefly described in Sec. 2.3.3. Tab. 4.3 lists the values of the material parameters comprising the bulk modulus K, the shear modulus μ, the initial yield strength σ_0, the saturation strength σ_∞, the linear hardening parameter h, and the hardening exponent ω. Further, the plate is subjected to a pressure load of $\bar{t}_y = 100.0\,\mathrm{MPa}$ which is applied incrementally in 30 load steps using a load factor of 0.05 for each load increment.

As the reference solution, we again employ an overkill p-FEM solution using 3,200 finite elements and an ansatz of order $p = 5$, see Fig. 5.4a. Thereby, curved edges and faces of the elements along the circular boundary of the plate are described applying the blending function method. For the FCM computation, we again consider the coarse and fine mesh discretized by 78 and 302 finite cells, respectively. The FCM discretizations are depicted in Fig. 5.4b and 5.4c, respectively. Further since the standard moment fitting does not perform well for nonlinear problems of the FCM, we employ the adaptive moment fitting – presented in Sec. 4.2 – in order to reduce the error induced by the numerical integration. For the resolution of the geometry, we thereby apply a refinement level of $k = 10$ for the coarse mesh and a refinement level of $k = 9$ for the fine one.

In the following, we study the influence of the basis function removal regarding the accuracy of the solution at a single point. To this end, we consider point A, which is located in the vicinity of the bottom, and the circular inclusion of the plate, see Fig. 5.3. The global coordinates of point A are given as $x_A = 39.0\,\text{mm}$, $y_A = 1.0\,\text{mm}$, and $z_A = 1.0\,\text{mm}$.

At first, we study the accuracy regarding the displacement u_x at point A. To this end, Fig. 5.10a and 5.10b show the relative error of displacement u_x for the coarse FCM mesh discretized by 78 finite cells and the fine one discretized by 302 finite cells using different orders of the ansatz p. For $\alpha = 10^{-q}$, we choose $q = 5$. Further, to investigate the effect of the basis function removal strategy, we consider different thresholds $\mu_t = 0.0$ and $\mu_t = 0.1$. Comparing the results of the coarse FCM discretization in Fig. 5.10a, it can be seen that for $p = 4$ the relative error during the elastic load steps is higher if the basis function removal is employed. For the plastic load steps, however, the relative error with and without basis function removal is within the same range. Comparing the results based on the fine FCM discretization in Fig. 5.10b with the coarse one in Fig. 5.10a, we can conclude that the effect of the basis function removal decreases if the number of finite cells is increased. This

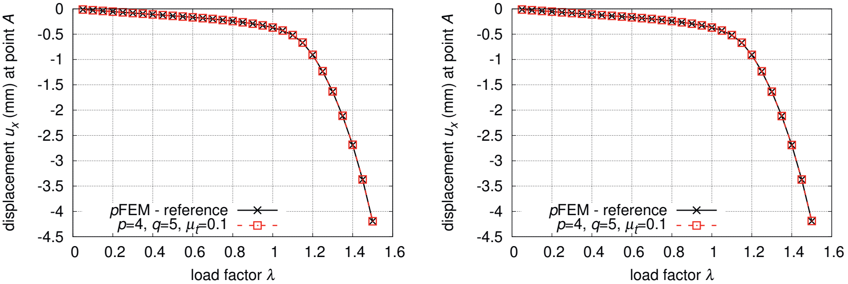


(a) FCM discretization using 78 cells.

(b) FCM discretization using 302 cells.

Figure 5.10: Plate with a circular hole. Error of displacement u_x at point 6.

behavior can be deduced from the fact that the finer the mesh, the smaller the support of the basis functions which are removed from the ansatz. Thus, the smaller the support of the removed basis functions, the smaller is the influence of the basis function removal on the solution. Finally, Fig. 5.11a and 5.11 show the displacement u_x of each load step for the coarse and the fine mesh using $q = 5$, $\mu_t = 0.1$, and $p = 4$. From the figures, it can be seen that the results obtained by the FCM solutions show a good agreement with the results of the p-FEM solution.



(a) FCM discretization using 78 cells.

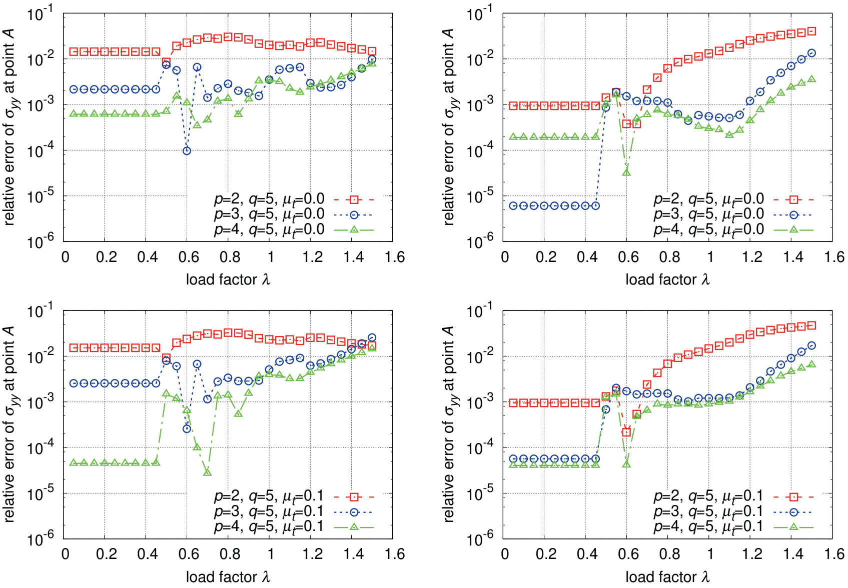
(b) FCM discretization using 302 cells.

Figure 5.11: Plate with a circular hole. Displacement u_y at point A.

Next, we study the influence of the basis function removal regarding the accuracy of the *Cauchy stress* which characterizes a more sensitive quantity as compared to the displacement. To this end, we consider the stress component σ_{yy} at point A. In doing so, the relative error of σ_{yy} is plotted in Fig. 5.12a and 5.12b employing the coarse and the fine FCM discretization, respectively. As it can be deduced from the figures, the basis function removal leads to lower values of the relative error of σ_{yy} applying an ansatz of order $p = 4$. This behavior is clearly visible within the elastic load steps. Here, the relative error of σ_{yy} could be reduced by approximately one order of magnitude.

Fig. 5.13a and 5.13b show the solutions of the stress σ_{yy} of both FCM discretizations and the p-FEM solution for each load step. For the FCM computations, we use an $\alpha = 10^{-q}$ with parameter $q = 5$, $\mu_t = 0.1$ as the threshold of the basis function removal criterion, and an ansatz of order $p = 4$. As it can be seen from the figures, the FCM solutions are in good agreement with the reference solution of the p-FEM computation.

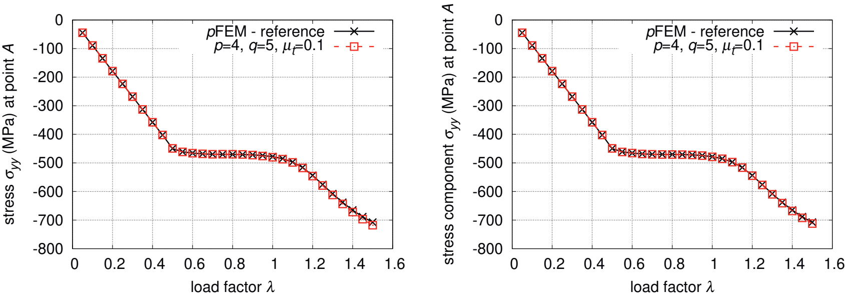
Finally, the evolution of the plastic region of the p-FEM solution and the FCM solution based on 302 finite cells is visualized by the contour plots in Fig. 5.14a and 5.14b, respectively. Here, we consider the von Mises stress σ_{vM} and the equivalent plastic strain $\bar{\alpha}$ of the final load step $\lambda = 1.5$. As it can be seen from the figures, two plastic regions on the right and on the bottom of the plate have developed significantly. Moreover, the contour plots of the FCM solution show a good agreement with the contour plots obtained by the p-FEM reference solution.



(a) FCM discretization using 78 cells.

(b) FCM discretization using 302 cells.

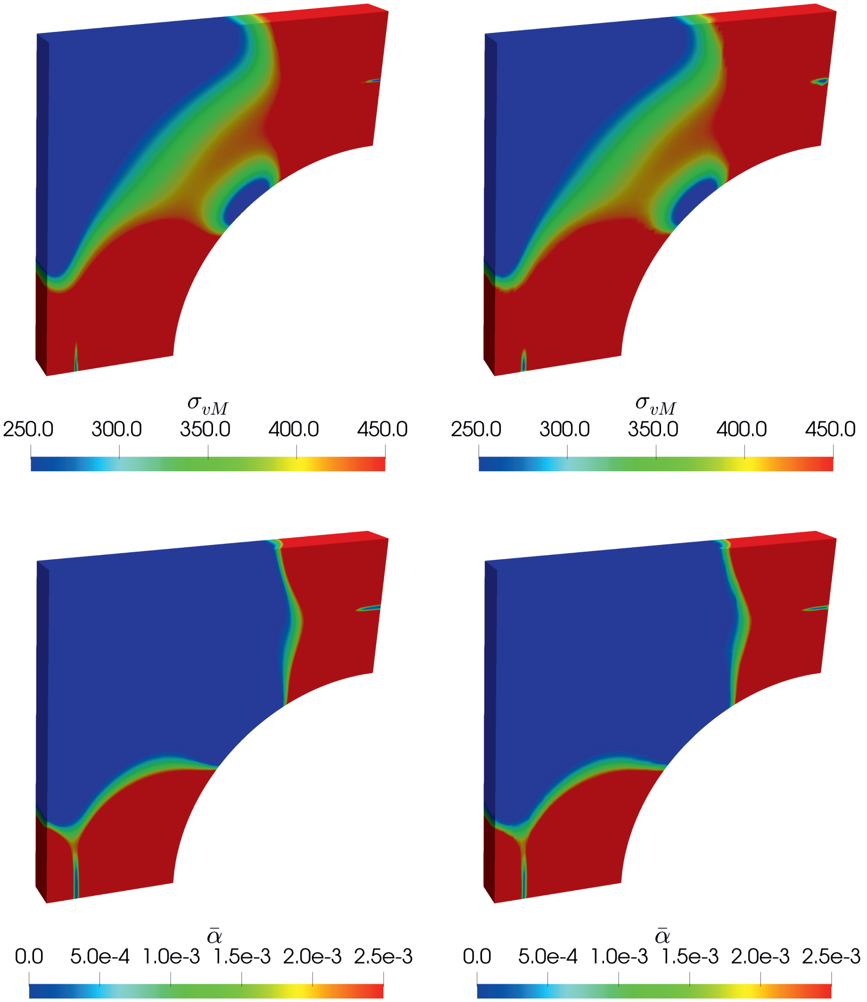
Figure 5.12: Plate with a circular hole. Error of stress component σ_{yy} at point 6.



(a) FCM discretization using 78 cells.

(b) FCM discretization using 302 cells.

Figure 5.13: Plate with a circular hole. Stress component σ_{yy} at point A.



(a) p-FEM discretization using 3,200 elements.

(b) FCM discretization using 302 cells.

Figure 5.14: Plate with a circular hole. Contour plots of the von Mises stress σ_{vM} and the equivalent plastic strain $\bar{\alpha}$ of the p-FEM and FCM solution of the last load step $\lambda = 1.5$.

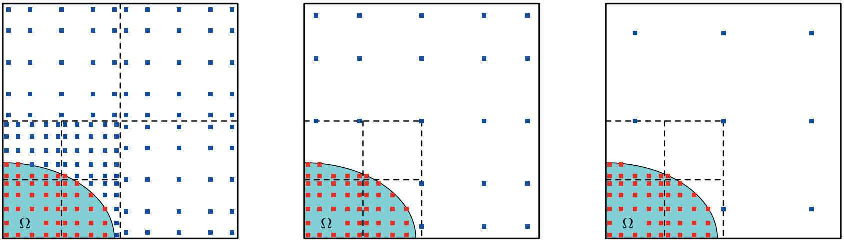
5.3 Finite strain problems

In this section, we study the effect of the basis function removal for finite strain problems. In general, FCM computations considering problems in large deformations fail due to severely distorted cut finite cells, causing the Newton-Raphson method to not converge anymore. This problem is especially pronounced when employing an ansatz p of high order. In this section, we therefore investigate the influence of the basis function removal – which only removes basis functions of higher order – in terms of the robustness of the FCM computations. To this end, we consider several problems of various complexity, based on two different material models. The first model is related to an isotropic hyperelastic material behavior based on a polyconvex strain energy density function. A brief description of the constitutive model is provided in Sec. 2.3.2. Further, the material parameters used for the subsequent analyses are given in Tab. 5.2. The second material model is related to the J_2 flow theory of plasticity for finite strains accounting for nonlinear isotropic hardening. A brief explanation describing the underlying equations of the constitutive model is given in Sec. 2.3.4. Further, for the material parameters of all subsequent analyses, we use the values that are listed in Tab. 4.3.

Table 5.2: Elastic material parameters

parameter	variable	value	unit
Lamé's first parameter	λ	28.846	MPa
shear modulus	μ	19.231	MPa

Before investigating the different examples, we would like to make an important remark on the quadrature of the fictitious domain used for the stabilization of the FCM. In the context of the FCM, various quadrature methods for the fictitious domain have been proposed. To this end, several authors [20, 25, 96] suggested a composed integration based on spacetrees where $(p+1)^d$ quadrature points are distributed within each subcell of spatial dimension d. This adaptive integration scheme is depicted in Fig. 5.15a considering a FCM problem of ansatz order $p = 4$ applying a quadtree of refinement level $k = 2$ and using $n_g = (p+1)^2 = 25$ quadrature points for each subcell. Thereby, the physical quadrature points are represented by red dots and the fictitious quadrature points are marked by blue dots. Since the fictitious quadrature points are only used for stabilization purposes, in [29] Abedian et al. proposed to use a refinement level of $k = 0$ for the quadrature employed for the fictitious domain, see Fig. 5.15b. In doing so, the number of integration points could be reduced significantly, thus, reducing the effort of the numerical integration process. In this thesis, we propose an additional quadrature scheme for the fictitious domain, depicted in Fig. 5.15c, that can be used for FCM analyses applying the basis function removal. Here, a reduced Gaussian quadrature is used for the fictitious domain applying a refinement level of $k = 0$. In doing so, $n_g = 3^d$ Gaussian points are distributed. Then, points that are located within the fictitious domain are taken into account. This reduced quadrature scheme can be used since high order basis functions with a small contribution to the global stiffness matrix are removed from the ansatz.



(a) (b) (c)

Figure 5.15: Different quadrature schemes used for the fictitious domain of the FCM employing an ansatz of order $p = 4$.

5.3.1 Single cube connector under pressure

As the first example, we consider a single cube connector under pressure. The geometry of the cube connector is given in Fig. 5.16, where its surface is described by means of a level set function [85] as

$$\begin{aligned}\phi(\boldsymbol{x}) = &\left[(x-x_\mathrm{c})^2+(y-y_\mathrm{c})^2-R^2\right]^2\\ &+\left[(y-y_\mathrm{c})^2+(z-z_\mathrm{c})^2-R^2\right]^2\\ &+\left[(x-x_\mathrm{c})^2+(z-z_\mathrm{c})^2-R^2\right]^2\\ &+\left[(x-x_\mathrm{c})^2-r^2\right]^2+\Big[(y-y_\mathrm{c})^2-r^2\Big)]^2+\left[(z-z_\mathrm{c})^2-r^2\right]^2-d \quad . \end{aligned} \tag{5.8}$$

Further, the cube connector is limited by the boundary of a cube with size $30 \times 30 \times 30\mathrm{mm}^3$. Furthermore, the values of the geometry parameters in Eq. (5.8) are listed in Tab. 5.3. Thereby, (x_c, y_c, z_c) represent the center coordinates, R is the outer radius, r denotes the inner radius, and d denotes an additional design parameter.

Due to the symmetry, it is sufficient to model only one eighth of the problem under investigation. In doing so, the setup of the FCM model is given in Fig. 5.17a. As it can be seen from the figure, a uniform displacement $\bar{u}_z$ is applied on the top surface acting in negative z-direction. Further, the left face is fixed in x-direction, the front face in y-direction, and the bottom face in z-direction. Finally, for the spatial discretization, we employ a FCM mesh using 129 finite cells. The spatial discretization of the problem is depicted in Fig. 5.17b.

For the computation of the integrals over the physical domain, we employ the adaptive moment fitting presented in Sec. 4.2 based on an octree refinement of tree depth level $k = 3$ to resolve for the physical domain. Further, for the volume fraction tolerances of the octree utilized by the adaptive moment fitting, we choose values of 0.85 on cell level $k_a = 0$, 0.7 for $k_a = 1$, and 0.9 for $k_a = 2$. For the integrals of the fictitious domain, we further choose the quadratures based on the fictitious integration points, denoted by the blue dots, given in Fig. 5.15b and 5.15c. Thereby, the quadrature points in Fig. 5.15c are used employing the basis function removal and the quadrature points in Fig. 5.15b are used without basis function removal.

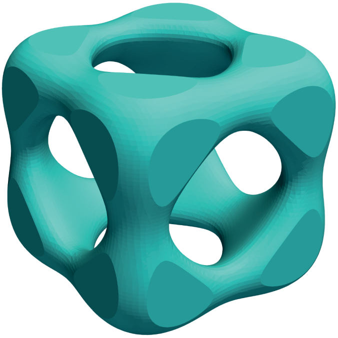
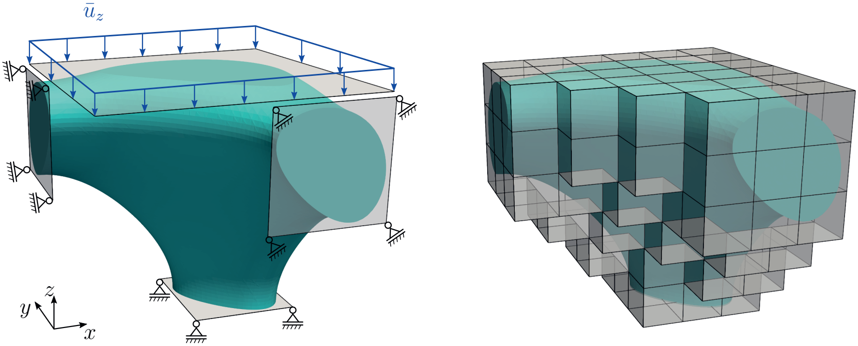


Figure 5.16: Single cube connector. Geometry.

Table 5.3: Single cube connector. Geometry parameters.

parameter	variable	value	unit
x center coordinate	x_c	0.0	mm
y center coordinate	y_c	0.0	mm
z center coordinate	z_c	0.0	mm
outer radius	R	15.0	mm
inner radius	r	11.25	mm
design parameter	d	46000.0	mm^4



(a) **(b)**

Figure 5.17: Single cube connector. FCM model and spatial discretization.

Finite strain hyperelasticity

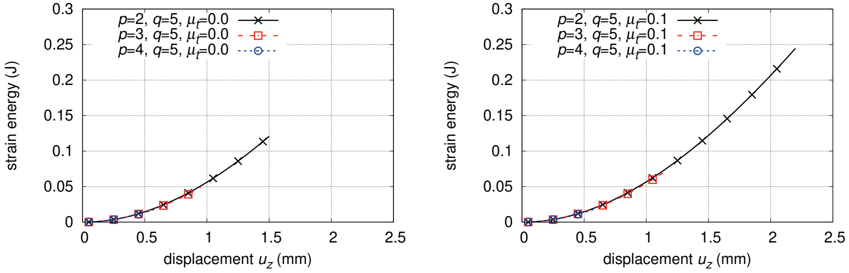
For the first test case, we assume an isotropic hyperelastic material behavior, see Sec. 2.3.2, where the material parameters are given in Tab. 5.2. Further, for the fictitious material model, we choose the same model applied to the physical domain. In doing so, we use $\alpha = 10^{-q}$ with $q = 5$ in order to scale the material parameters $\alpha\lambda$ and $\alpha\mu$ of the fictitious material model. Moreover, for the uniform prescribed displacement $\bar{u}_z$, we choose a displacement increment of 0.05 mm for each load step. Thereby, the displacement $\bar{u}_z$ is increased incrementally until the Newton-Raphson method (and, thus, the entire analysis) fails.

To study the effect of the basis function removal on the robustness of the FCM analysis, we consider different values for the basis function removal threshold μ_t where $\mu_t = 0.0$ means that no basis functions are removed from the ansatz. Based on the results in Sec. 5.2.1, we choose 0.1, 0.3, and 0.5 as the values of the basis function removal criterion. Moreover, we study the influence of the basis function removal considering different orders of the ansatz p.

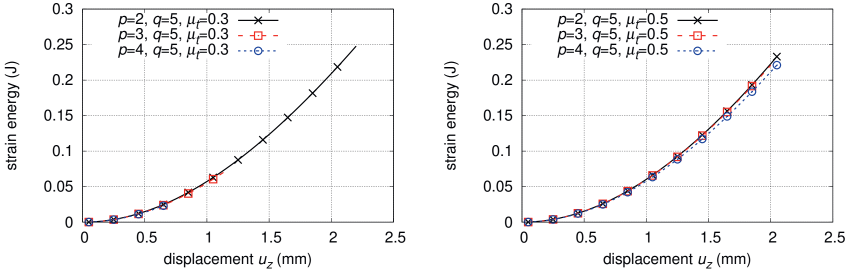
In doing so, Fig. 5.18a, 5.18b, 5.18c, and 5.18d show the energy-displacement curves of the physical domain for the different values of μ_t applying an ansatz order $p = 2, 3, 4$. As it can be seen from the figures, the basis function removal helps to improve the robustness of the FCM analyses. Moreover, employing a high value for the basis function removal criterion $\mu_t = 0.5$ also improves the robustness of the analyses using a higher order of the ansatz, see Fig. 5.18d.

To study the influence on the solution, in Fig. 5.19a, 5.19b, and 5.19c the energy-displacement curves are plotted employing the different values of μ_t for each order of the ansatz p in an individual figure, respectively. From the figures, it can be deduced that applying values of 0.1 and 0.3 for μ_t leads to almost no deviations from the solution obtained without basis function removal ($\mu_t = 0.0$). Further, only small deviations may be observed using $\mu_t = 0.5$ and an ansatz of order $p = 2$ and $p = 3$.

Finally, Fig. 5.20 shows the evolution of the von Mises stress σ_{vM} using an ansatz of $p = 2$ and a basis function removal criterion $\mu_t = 0.3$. To this end, the contour plots of σ_{vM} are depicted for different values of the prescribed displacement $\bar{u}_z$.

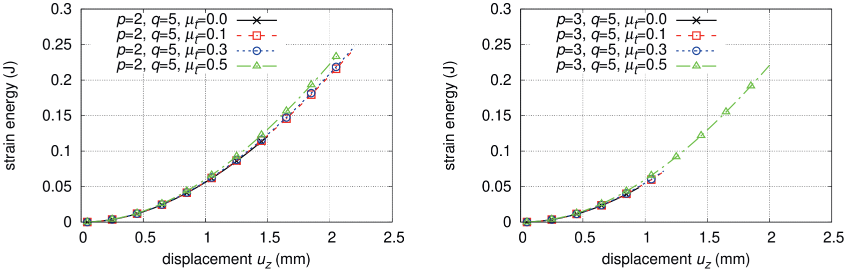


(a) $\mu_t = 0.0$ **(b)** $\mu_t = 0.1$

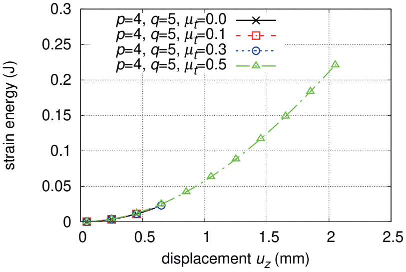


(c) $\mu_t = 0.3$ **(d)** $\mu_t = 0.5$

Figure 5.18: Single cube connector. Energy-displacement curves.

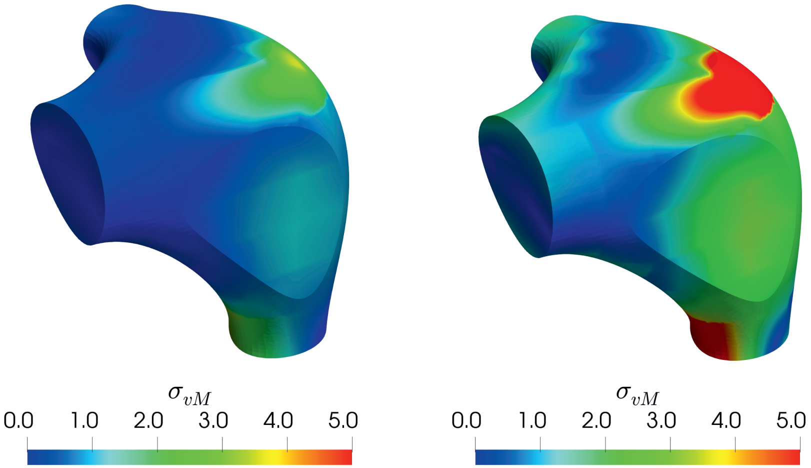


(a) $p = 2$ **(b)** $p = 3$

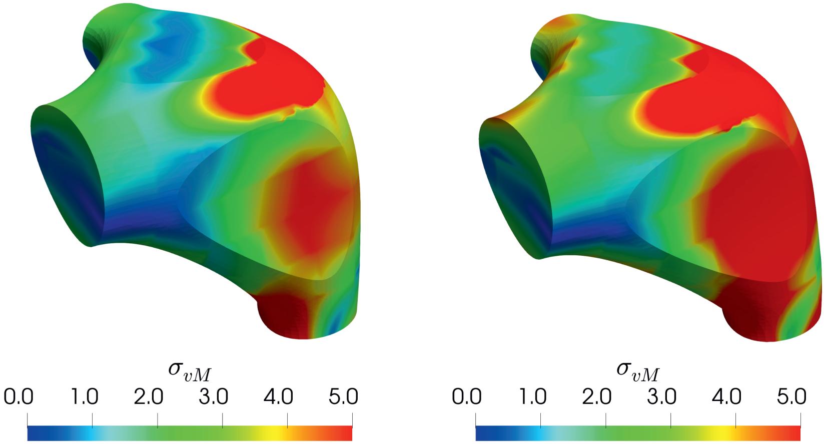


(c) $p = 4$

Figure 5.19: Single cube connector. Energy-displacement curves.



(a) $\bar{u}_z = 0.4\,\mathrm{mm}$ **(b)** $\bar{u}_z = 1.0\,\mathrm{mm}$



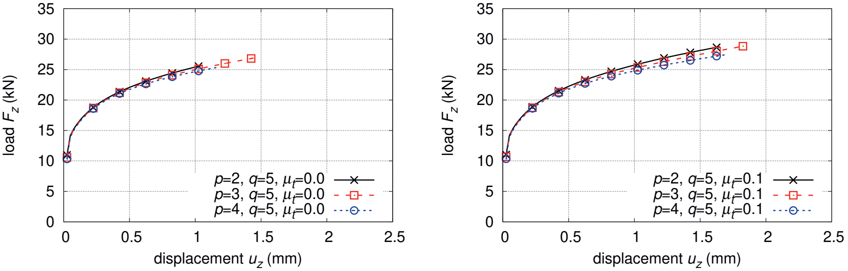
(c) $\bar{u}_z = 1.6\,\mathrm{mm}$ **(d)** $\bar{u}_z = 2.2\,\mathrm{mm}$

Figure 5.20: Single cube connector. Contour plots of the von Mises stress σ_{vM} for different load steps.

Finite strain elastoplasticity

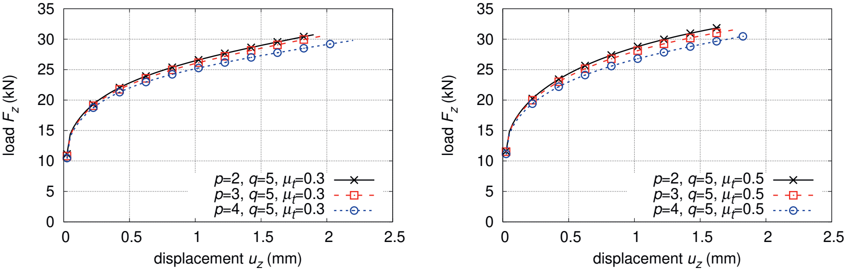
For the second test case of the single cube connector, we assume a material model based on the J_2 flow theory of plasticity for finite strains with nonlinear and isotropic hardening, see Sec. 2.3.4. The material parameters are listed in Tab. 4.3. To improve the robustness of the FCM analysis, we assign the same nonlinear material model of the physical domain to the fictitious one, but assuming an infinite yield stress. Thus, the fictitious material model is based only on the elastic strain energy density function of the elastoplastic material model. For $\alpha = 10^{-q}$, we thereby choose $q = 5$ in order to scale the material parameters of the fictitious domain. Further, analogous to the previous test case, the prescribed displacement $\bar{u}_z$ is increased incrementally until the Newton-Raphson method (and, thus, the entire analysis) fails. Thereby, a displacement increment of 0.025 mm is applied for each load step.

Next, to study the influence of the basis function removal on the robustness of the FCM analyses, Fig. 5.21a, 5.21b, 5.21c, and 5.21d show the load-displacement curves for fixed values of μ_t and different orders of the ansatz p. As it can be seen from Fig. 5.21a, where no basis functions are removed from the ansatz ($\mu_t = 0.0$), the robustness of the FCM depends on the choice of the ansatz order. Here, applying higher orders of the ansatz $p = 3$ and $p = 4$ results in more robust analyses than using $p = 2$, especially in the case of $p = 3$. Similar behavior appears when utilizing the basis function removal, see Fig. 5.21b,



(a) $\mu_t = 0.0$

(b) $\mu_t = 0.1$



(c) $\mu_t = 0.3$

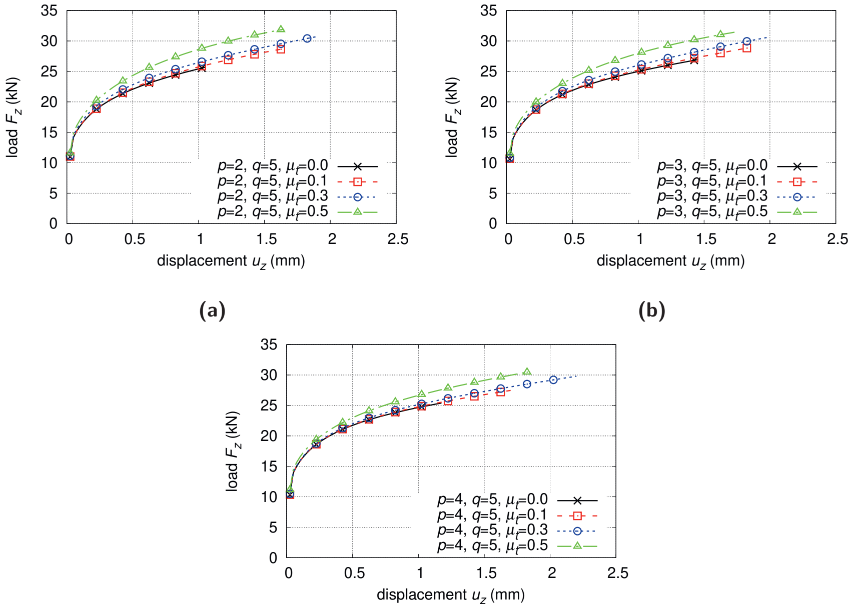
(d) $\mu_t = 0.5$

Figure 5.21: Single cube connector. Load-displacement curves.

5.21c, and 5.21d. Moreover, it can be seen from the figures that the robustness of the FCM analyses could be improved significantly as compared to the analyses without basis function removal, similar as in the previous test case in which we assumed a hyperelastic material model.

Next, we study the influence of basis function removal on the solution. To this end, we consider the different values of the basis function removal criterion μ_t for a fixed ansatz order p. In doing so, the load-displacement curves are plotted in Fig. 5.22a, 5.22b, and 5.22c. As it can be deduced from the figures, employing a high value of the basis function removal criterion $\mu_t = 0.5$ increases the load values significantly as compared to the results obtained without basis function removal ($\mu_t = 0.0$). However, the analyses based on $\mu_t = 0.1$ and $\mu_t = 0.3$ lead to reasonable results and a significant improvement in the robustness of the FCM analyses.

Finally, the evolution of the equivalent plastic strain $\bar{\alpha}$ during the loading is depicted in Fig. 5.23. To this end, contour plots of $\bar{\alpha}$ are given for different load steps using a basis function removal criterion of $\mu_t = 0.3$ and an ansatz of order $p = 4$.

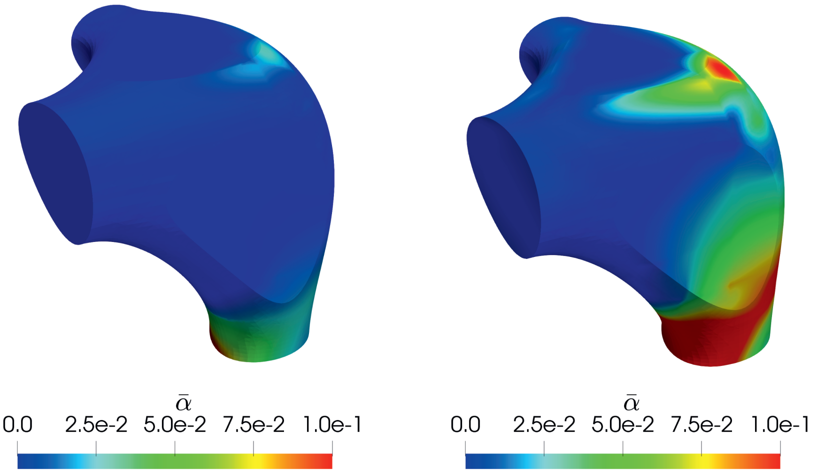


(a)

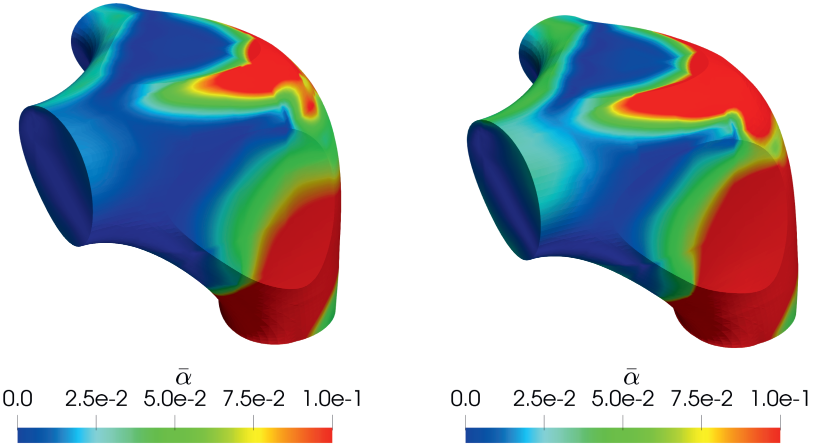
(b)

(c)

Figure 5.22: Single cube connector. Load-displacement curves.



(a) $\bar{u}_z = 0.15\,\mathrm{mm}$ **(b)** $\bar{u}_z = 0.825\,\mathrm{mm}$

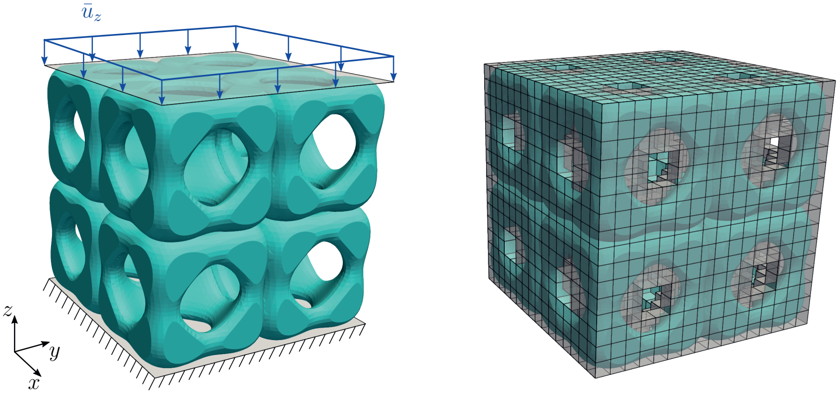


(c) $\bar{u}_z = 1.5\,\mathrm{mm}$ **(d)** $\bar{u}_z = 2.2\,\mathrm{mm}$

Figure 5.23: Single cube connector. Contour plots of the equivalent plastic strain $\bar{\alpha}$ for different load steps.

5.3.2 Complex cube connector under pressure

As the second example, we consider a cube connector with a more complex structure. Fig. 5.24a shows the model of the problem under investigation. As it can be seen from the figure, the structure of the complex cube connector is composed of eight individual single cube connectors. Thereby, each individual cube connector is defined by the level set function given in Eq. (5.8) and a bounding box of dimensions $30 \times 30 \times 30\mathrm{mm}^3$ where the parameters defining the surface of every cube connector are given in Tab. 5.4. As depicted in Fig. 5.24a, the complex cube connector is fixed at the bottom face. Further, a prescribed displacement $\bar{u}_z$ is applied at the top surface acting in negative z-direction. Furthermore, the top surface is fixed in x- and y-direction. Moreover, the spatial discretization used for the analysis is given in Fig. 5.24b. Here, we employ a FCM mesh using 2,912 finite cells.



(a) **(b)**

Figure 5.24: Complex cube connector. Geometry, boundary conditions, and discretization.

Table 5.4: Complex cube connector. Geometry parameters.

cube id	x_c mm	y_c mm	z_c mm	R mm	r mm	d mm^4
1	15.0	15.0	15.0	15.0	11.25	53000.0
2	45.0	15.0	15.0	15.0	11.25	49000.0
3	15.0	45.0	15.0	15.0	11.25	51000.0
4	45.0	45.0	15.0	15.0	11.25	47000.0
5	15.0	15.0	45.0	15.0	11.25	52000.0
6	45.0	15.0	45.0	15.0	11.25	48000.0
7	15.0	45.0	45.0	15.0	11.25	50000.0
8	45.0	45.0	45.0	15.0	11.25	46000.0

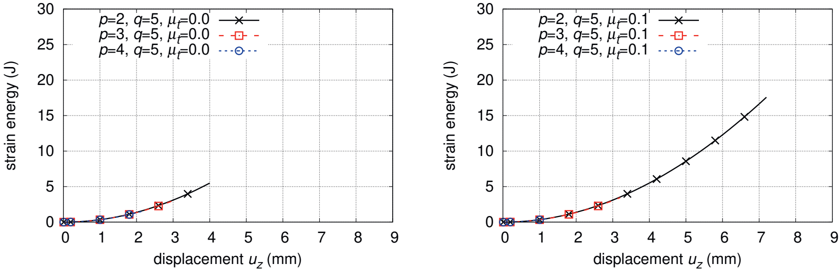
Finite strain hyperelasticity

As the first test case of the complex cube connector, we assume an isotropic hyperelastic material behavior, see Sec. 2.3.2. The material parameters are listed in 5.2. To study the robustness of the FCM analyses the prescribed displacement $\bar{u}_u$ is increased incrementally until the Newton-Raphson method (and, thus, the entire analysis) fails. Thereby, a load increment of 0.2 mm is applied for each load step. Further, for the computation of the integrals over the physical domain, we employ the adaptive moment fitting using an octree of refinement level $k = 3$ for the resolution of the geometry. For the volume fraction tolerances of the octree utilized by the adaptive moment fitting, we thereby choose a value of 0.85 at tree depth level $k_a = 0$, 0.7 at level $k_a = 1$, and 0.9 at level $k_a = 2$. Furthermore, for the computation of the integrals over the fictitious domain, we use the fictitious integration points depicted in Fig. 5.15b and 5.15c where the first point set is utilized without and the second one with the basis function removal. Moreover, for $\alpha = 10^{-q}$ we utilize a parameter of $q = 5$.

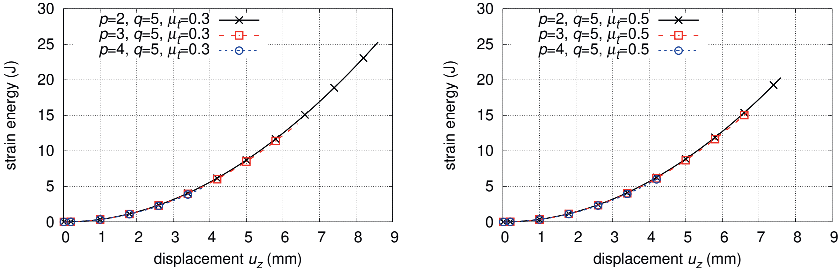
In Fig. 5.25a, 5.25a, 5.25a, and 5.25a, the energy-displacement curves are plotted for a fixed value of the basis function criterion μ_t and different orders of the ansatz p. As it can be deduced from the figures, the basis function removal improves the robustness of the FCM significantly. Applying $\mu_t = 0.3$ enables to increase the prescribed displacement $\bar{u}_z$ by a factor bigger than 2 for all orders of the ansatz p as compared to the values of the analyses without basis function removal $\mu_t = 0.0$.

To study the effect of the basis function removal on the solution in Fig. 5.26a, 5.26b, and 5.26c, the energy-displacement curves are plotted for a fixed order of the ansatz p and different values of the basis function removal criterion μ_t. As it can be seen from the figures, the results applying the basis function removal show a good agreement with the results using $\mu_t = 0.0$. Even the results utilizing a higher removal criterion $\mu_t = 0.5$ do not show a significant influence on the solution.

Finally, Fig. 5.27 shows the evolution of the von Mises stress σ_{vM} during the loading. Here, the contour plots of the von Mises stress are plotted for different load steps applying a basis function removal criterion of $\mu_t = 0.3$ and an ansatz of order $p = 2$.

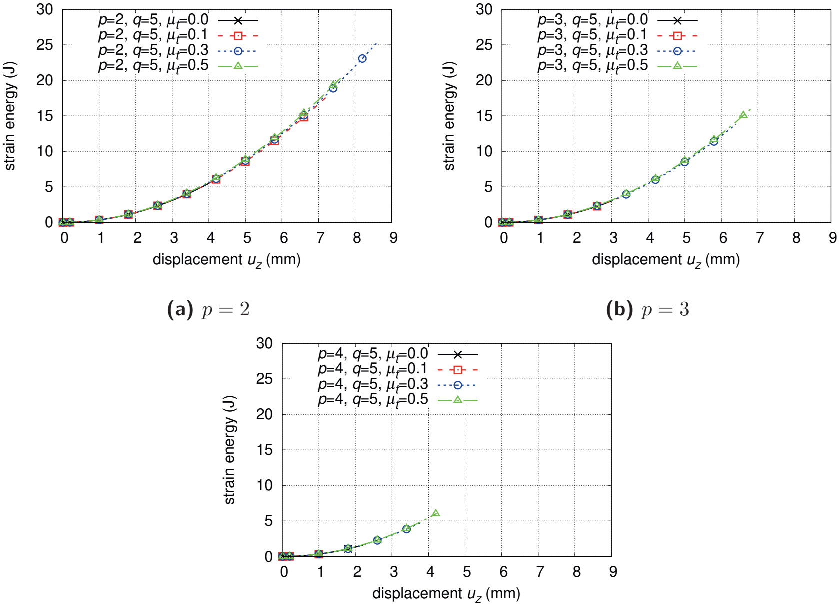


(a) $\mu_t = 0.0$ **(b)** $\mu_t = 0.1$



(c) $\mu_t = 0.3$ **(d)** $\mu_t = 0.5$

Figure 5.25: Complex cube connector. Energy-displacement curves.

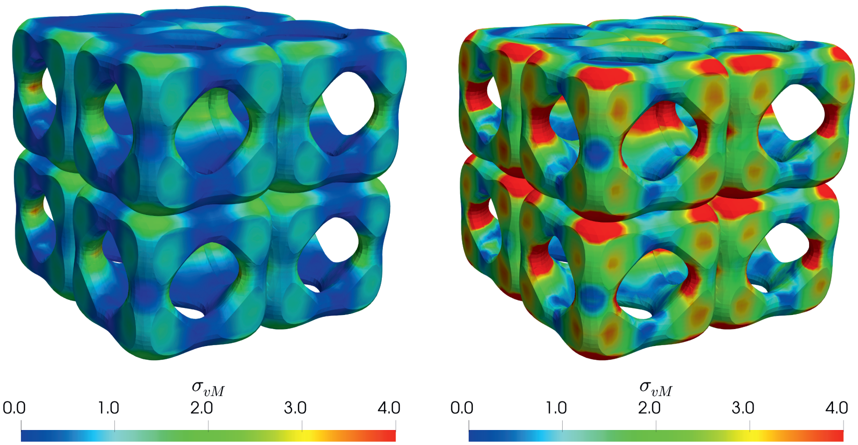


(a) $p = 2$

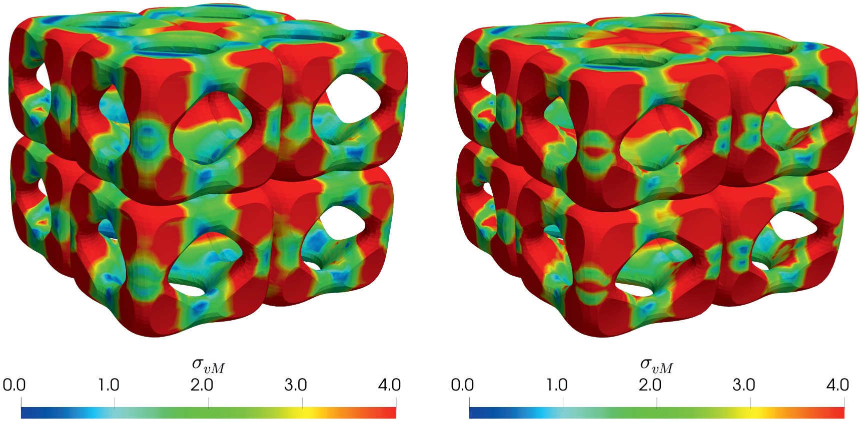
(b) $p = 3$

(c) $p = 4$

Figure 5.26: Complex cube connector. Energy-displacement curves.



(a) $\bar{u}_z = 1.4\,\mathrm{mm}$ **(b)** $\bar{u}_z = 3.8\,\mathrm{mm}$



(c) $\bar{u}_z = 6.2\,\mathrm{mm}$ **(d)** $\bar{u}_z = 8.6\,\mathrm{mm}$

Figure 5.27: Complex cube connector. Contour plots of the von Mises stress σ_{vM} for different load steps.

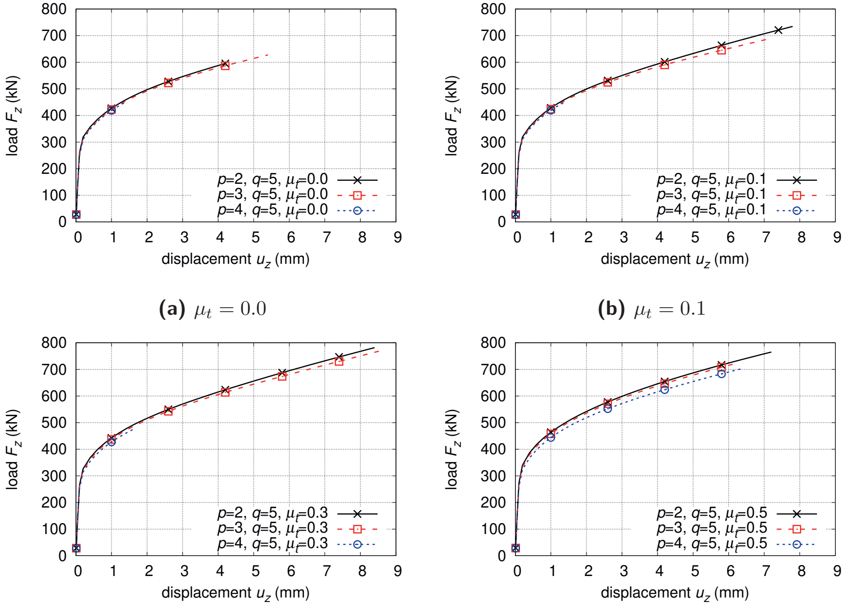
Finite strain elastoplasticity

As the second test case of the complex cube connector, we study the influence of the basis function removal assuming a material model based on the J_2 flow theory of plasticity, see Sec. 2.3.4. The material parameters are listed in Tab. 4.3. To improve the robustness of the FCM analysis, we employ the same material model for the fictitious domain using $\alpha = 10^{-q}$ with $q = 5$ in order to scale the material parameters of the fictitious material model appropriately. Moreover, we assume an infinite yield stress. Thus, the material behavior of the fictitious domain is defined by the hyperelastic strain energy density function of the constitutive model. For the numerical integration, we further employ the same quadratures as in the previous test case. Furthermore, analogous to the previous test case, the prescribed displacement $\bar{u}_z$ is increased incrementally until the analysis fails. To this end, for the first 5 displacement increments, we apply values of 0.01 mm, 0.02 mm, 0.02 mm, 0.05 mm, and 0.1 mm. For each load step bigger than 5, we then use a displacement increment of 0.2 mm.

To show the influence of the basis function removal, Fig. 5.28a, 5.28b, 5.28c, and 5.28d show the load-displacement curves for a fixed value of the basis function removal criterion μ_t and different orders of the ansatz p. As it can be seen from the figures, the basis function removal increases the robustness of the FCM analyses significantly. By using $\mu_t = 0.3$, the value of the prescribed displacement $\bar{u}_z$ could thus be increased by a factor of about 2 employing an ansatz of $p = 2$ and $p = 3$. Moreover, a high removal criterion $\mu_t = 0.5$ allows to increase the robustness of the analysis by applying an ansatz of order $p = 4$. Here, the prescribed displacement could be increased by a factor of about 5.

Next, to study the effect of the basis function removal on the solution, Fig. 5.29a, 5.29b, and 5.29c show the load-displacement curves for a fixed order of the ansatz p and different values of μ_t. As it can be seen from the figures, higher values of the basis function removal criterion $\mu_t = 0.3$ and $\mu_t = 0.5$ increase the deviation in the load values as compared to the results obtained without basis function removal ($\mu_t = 0.0$). When employing $\mu_t = 0.1$, however, the deviations in the load values are negligible.

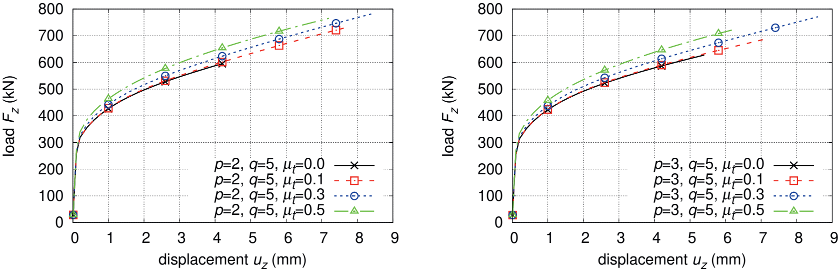
Finally, the evolution of the equivalent plastic strain during the loading is given in Fig. 5.30. Here, the contour plots of $\bar{\alpha}$ are depicted for different load steps applying an ansatz of $p = 2$ and a basis function removal criterion of $\mu_t = 0.3$.



(a) $\mu_t = 0.0$ **(b)** $\mu_t = 0.1$

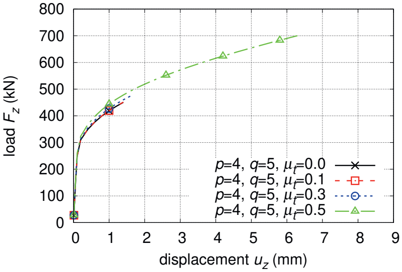
(c) $\mu_t = 0.3$ **(d)** $\mu_t = 0.5$

Figure 5.28: Complex cube connector. Load-displacement curves.



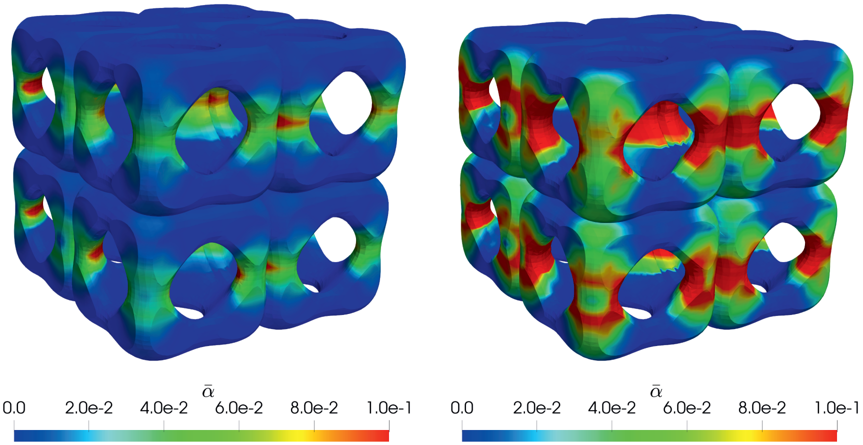
(a) $p = 2$

(b) $p = 3$

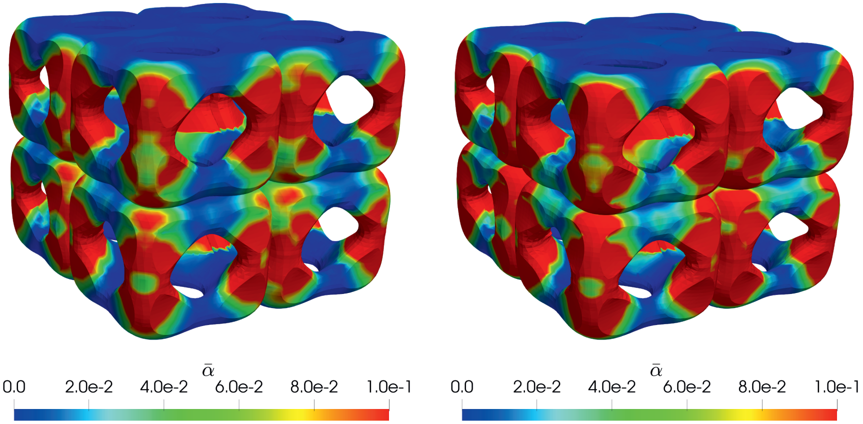


(c) $p = 4$

Figure 5.29: Complex cube connector. Load-displacement curves.



(a) $\bar{u}_z = 1.2\,\mathrm{mm}$ **(b)** $\bar{u}_z = 3.6\,\mathrm{mm}$

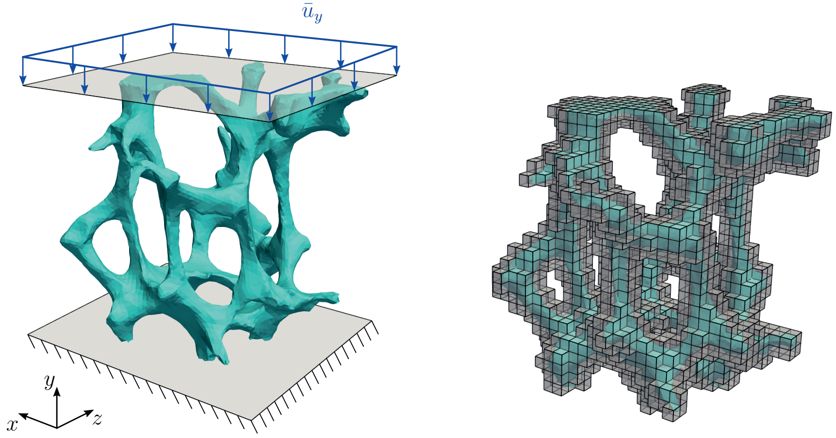


(c) $\bar{u}_z = 6.0\,\mathrm{mm}$ **(d)** $\bar{u}_z = 8.4\,\mathrm{mm}$

Figure 5.30: Complex cube connector. Contour plots of the equivalent plastic strain $\bar{\alpha}$ for different load steps.

5.3.3 Single pore of a foam-like structure under pressure

As the final example, we consider a single pore of a foam-like structure. The geometry and the boundary conditions of the problem are depicted in Fig. 5.31a. Here, the pore is embedded in a bounding box of dimensions $6.327 \times 6.424 \times 6.688\text{mm}^3$. Further, for the analysis, the pore is fixed at the bottom face and a prescribed displacement $\bar{u}_y$ is applied on its top surface acting in negative y-direction. Furthermore, the top face is fixed in x- and y-direction. Fig. 5.31b shows the spatial discretization of the problem. Here, we employ a Cartesian grid using 4,678 finite cells.



(a) **(b)**

Figure 5.31: Single pore of a foam-like structure. Geometry, boundary conditions, and discretization.

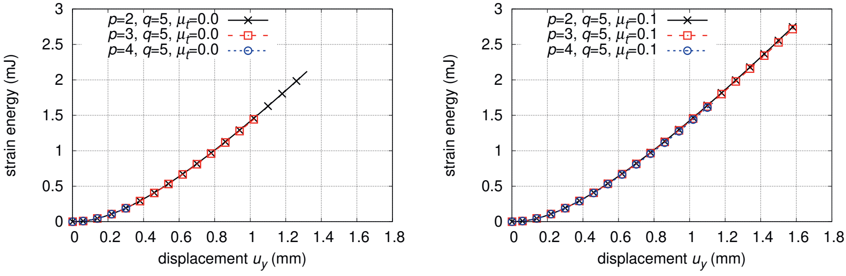
Finite strain hyperelasticity

For the first test case, we assume an isotropic hyperelastic material behavior, see Sec. 2.3.2. The material parameters are given in Tab. 5.2. To study the effect of the basis function removal, we increase the prescribed displacement $\bar{u}_y$ incrementally until the analysis fails. In doing so, we apply an increment of 0.001 mm for the first load step, an increment of 0.002 mm for the second and the third load step, and an increment of 0.005 mm for the fourth load step. For all load steps greater than 4, we apply a displacement increment of 0.01 mm. Further, for the computation of the integrals over the physical domain, we apply the adaptive moment fitting. In doing so, we employ an octree of tree depth level $k = 3$ for the resolution of the geometry. For the volume fraction tolerances of the octree utilized by the adaptive moment fitting, we thereby choose a value of 0.85 on cell level $k_a = 0$, 0.7 at level $k_1 = 1$, and 0.9 at level $k_a = 2$. Furthermore, for the computation of the integrals of the fictitious domain, we apply the fictitious integration points depicted in Fig. 5.15b and 5.15c where the first point set is utilized without and the second one with the basis function removal. Moreover, to improve the robustness of the FCM analysis, we use $\alpha = 10^{-q}$ with $q = 5$.

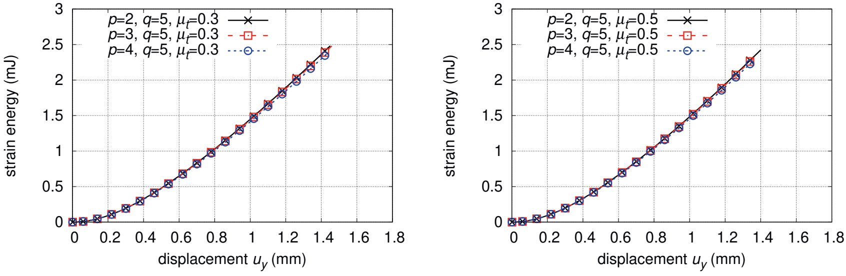
In order to study the influence of the basis function removal, Fig. 5.32a, 5.32b, 5.32c, and 5.32d show the energy-displacement curves for a fixed value of μ_t and different orders of the ansatz p. As it can be deduced from the figures, the analyses utilizing the basis function removal perform more robust than the analyses without ($\mu_t = 0.0$). This behavior is especially pronounced for the analyses applying an ansatz of order $p = 4$. By employing a criterion of $\mu_t = 0.3$, the prescribed displacement $\bar{u}_y$ could be increased by a factor greater than 4.

Further, to investigate the influence of the basis function removal on the solution, Fig. 5.33a, 5.33b, and 5.33b show the energy-displacement curves for a fixed ansatz order p and different values of the criterion μ_t. As it can be seen from the figures, the deviations in the energy values are negligibly small even when employing a high value for the basis function removal criterion ($\mu_t = 0.5$).

Finally, 5.34 shows the evolution of the von Mises stress σ_{vM} during the loading of the single pore. Here, the contour plots of σ_{vM} are depicted for different load values of the prescribed displacement $\bar{u}_y$. It can be seen from the figures that high values of the von Mises stress arise at the struts of the pore.

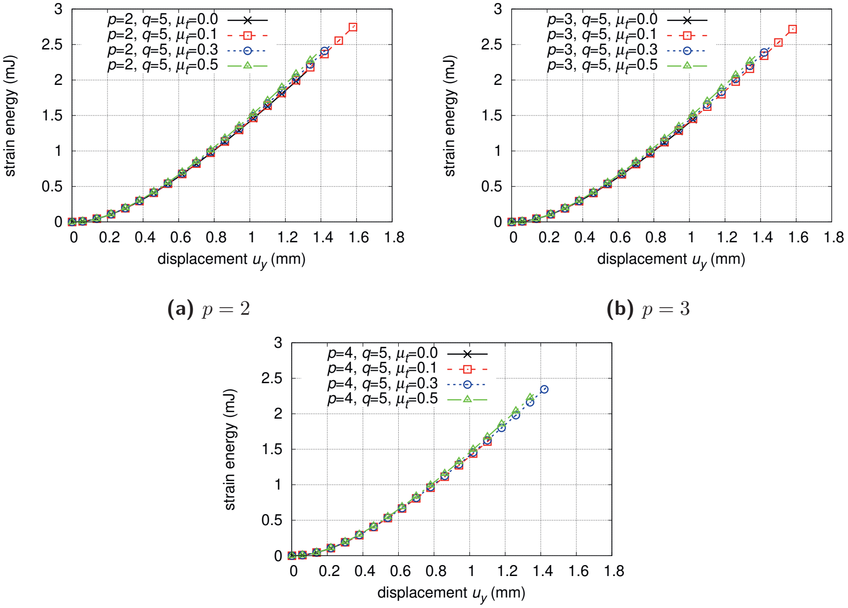


(a) $\mu_t = 0.0$ **(b)** $\mu_t = 0.1$



(c) $\mu_t = 0.3$ **(d)** $\mu_t = 0.5$

Figure 5.32: Single pore of a foam-like structure. Energy-displacement curves.

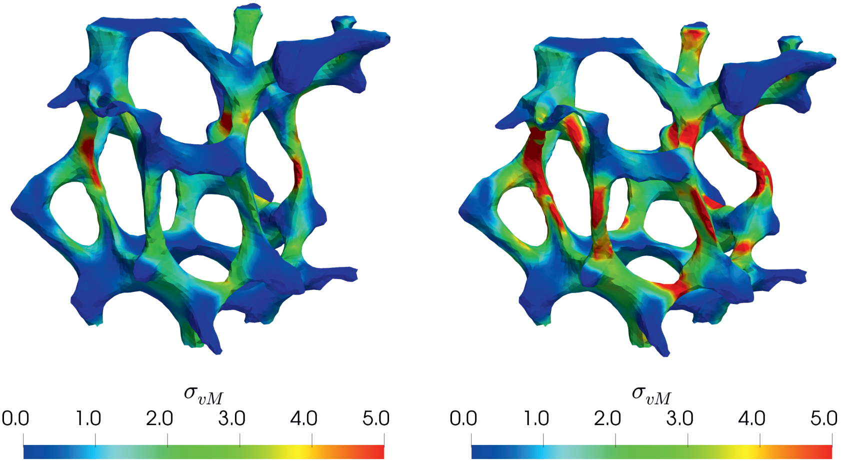


(a) $p = 2$

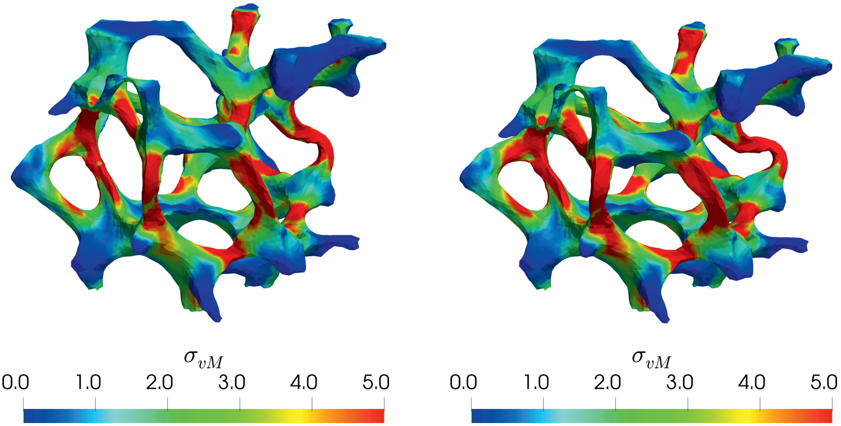
(b) $p = 3$

(c) $p = 4$

Figure 5.33: Single pore of a foam-like structure. Energy-displacement curves.



(a) $\bar{u}_y = 0.39\,\mathrm{mm}$ **(b)** $\bar{u}_y = 0.79\,\mathrm{mm}$



(c) $\bar{u}_y = 1.19\,\mathrm{mm}$ **(d)** $\bar{u}_y = 1.59\,\mathrm{mm}$

Figure 5.34: Single pore of a foam-like structure. Contour plots of the von Mises stress σ_{vM}.

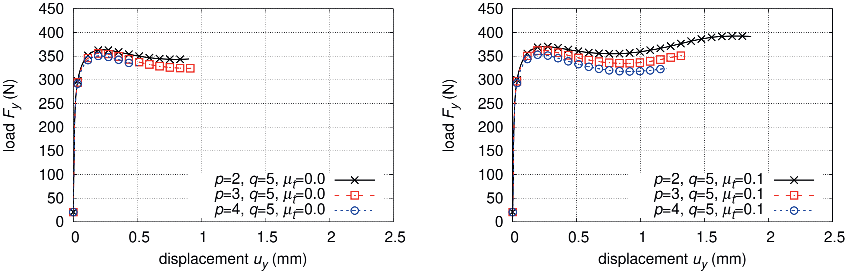
Finite strain elastoplasticity

As the second test case, we investigate the performance of the basis function removal assuming a material behavior based on the J_2 flow theory of plasticity, see Sec. 2.3.4. The material parameters are given in Tab. 4.3. Further, for the material model of the fictitious domain we assign the same model used for the physical domain. However, as we assume an infinite yield stress, the fictitious material is defined by the hyperelastic strain energy function only. To improve the robustness of the FCM analysis, we use $\alpha = 10^{-q}$ with $q = 5$. For the numerical integration, we employ the same quadratures as in the previous test case. Moreover, analogous to the previous test case the prescribed displacement $\bar{u}_y$ is increased incrementally until the analysis fails. For the first load step, we thereby choose a displacement increment of 0.001 mm, an increment of 0.002 mm for the second and the third load step, an increment of 0.005 mm from load step 4 to 9, and an increment of 0.01 mm for all following load steps.

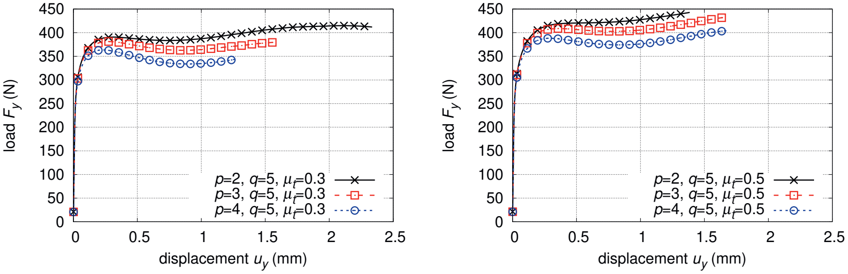
To study the effect of the basis function removal on the robustness of the FCM analyses, Fig. 5.35a, 5.35b, 5.35c, and 5.35d show the load-displacement curves applying a fixed value of the criterion μ_t and different orders of the ansatz p. As it can be deduced from the figures, the analyses employing the basis function removal behave much more robust than the ones without ($\mu_t = 0.0$). Here, the prescribed displacement could be increased by factors between 2 and 3.

Further, in order to study the influence of the basis function removal on the solution, Fig. 5.36a, 5.36a, and 5.36a show the load-displacement curves applying a fixed ansatz order p and different values of the criterion μ_t. As it can be inferred from the figures, applying high values of the basis function removal criterion ($\mu_t = 0.5$) results in large deviations of the load values. However, employing $\mu_t = 0.1$ and $\mu_t = 0.3$ leads to reasonable results. This relation is especially pronounced if an ansatz order of $p = 4$ is applied.

Finally, 5.37 shows the evolution of the plastic region during the loading for the analyses using an ansatz order $p = 2$ and a basis function removal criterion $\mu_t = 0.3$. To this end, the contour plots of the equivalent plastic strain $\bar{\alpha}$ are depicted for different load steps. As it can be seen from the figures, plastic regions develop at the struts of the single pore, reaching high values for the equivalent plastic strain.

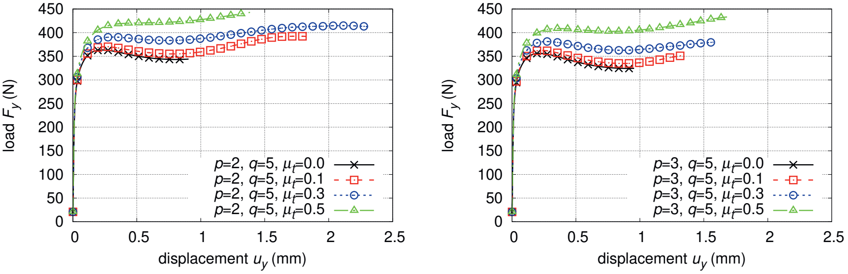


(a) $\mu_t = 0.0$ **(b)** $\mu_t = 0.1$

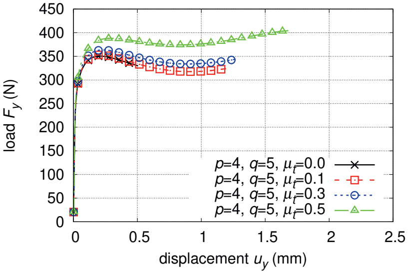


(c) $\mu_t = 0.3$ **(d)** $\mu_t = 0.5$

Figure 5.35: Single pore of a foam-like structure. Load-displacement curves.

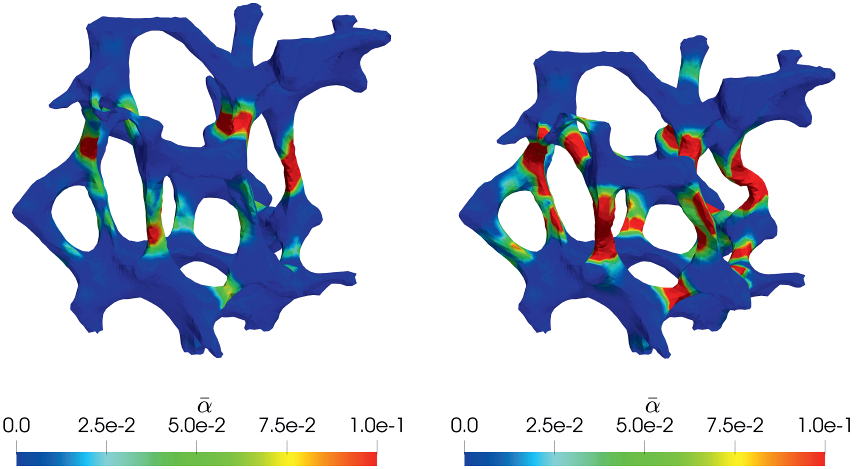


(a) $p = 2$ **(b)** $p = 3$

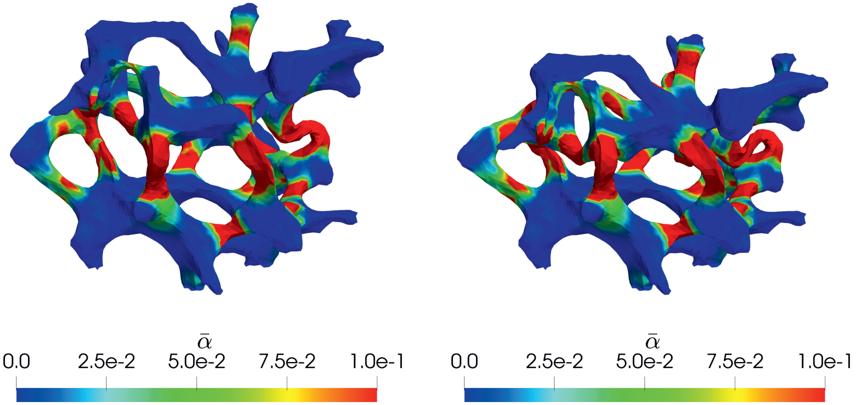


(c) $p = 4$

Figure 5.36: Single pore of a foam-like structure. Load-displacement curves.



(a) $\bar{u}_y = 0.235\,\mathrm{mm}$ **(b)** $\bar{u}_y = 0.935\,\mathrm{mm}$



(c) $\bar{u}_y = 1.635\,\mathrm{mm}$ **(d)** $\bar{u}_y = 2.335\,\mathrm{mm}$

Figure 5.37: Single pore of a foam-like structure. Contour plots of the equivalent plastic strain $\bar{\alpha}$.

6 A remeshing strategy for the FCM

For linear and nonlinear problems in small deformations the robustness of the FCM can be improved by employing a fictitious material model or by simply removing critical shape functions from the ansatz, as shown in the previous chapter. When considering problems in finite deformations, however, the robustness of the FCM suffers from degenerated cells. Due to the large deformation of the fictitious domain, badly cut finite cells get distorted severely during the deformation process, thus resulting in a bad parametric description of the deformation. Usually, the degenerated cells decrease the accuracy of the analysis and affect the convergence behavior of the Newton-Raphson procedure. In general, the analysis fails due to self-penetration of the cut cells.

The subject of degenerated elements when considering problems in large deformations, however, is a well-known issue in the standard finite element method as well. A promising approach to overcome this problem is based on remeshing [158–160]. The basic idea of the remeshing approach is simple. Whenever a finite element violates certain criteria, the simulation is stopped and a new mesh is generated, which captures the deformed structure. Next, essential field variables – needed to proceed the analysis – are transferred from the old to the new mesh. Such field variables are problem-dependent and may be stresses, history variables, or variables describing the current deformation of the body, for instance.

To overcome the problem of severely distorted cells in the FCM for structures undergoing large deformations, this chapter serves to present a remeshing strategy with respect to the total Lagrangian formulation [161]. The main idea of the proposed remeshing strategy is based on a decomposition of the deformation gradient. Moreover, the mesh generation during the remeshing procedure can be carried out by simply employing Cartesian grids or structured meshes, thanks to the fictitious domain approach. This is a main advantage compared to the standard FEM where we have to invest more effort to discretize the deformed geometry utilizing boundary-fitted elements. Finally, in order to transfer the field variables – like the deformation gradient or history variables in the case of a elastoplasticity – from the old to the new mesh, we introduce a local radial basis function interpolation scheme.

6.1 Kinematic relations

Before we present the remeshing procedure, let us start with the description of the essential kinematic relations. To this end, we consider the deformation process of a deformable body at various points in time, as depicted in Fig. 6.1. Thereby, a configuration of the body at time t is described by the mapping

$$\boldsymbol{\varphi}_t\left(\Omega\right) = \left\{\boldsymbol{\varphi}_t\left(\boldsymbol{X}\right) \middle| \boldsymbol{X} \in \Omega\right\} \tag{6.1}$$

where $\boldsymbol{\varphi}_0\left(\Omega\right) = \Omega$ characterizes the initial configuration. Consequently, following the motion of a specific material point its position with respect to the configuration at time t

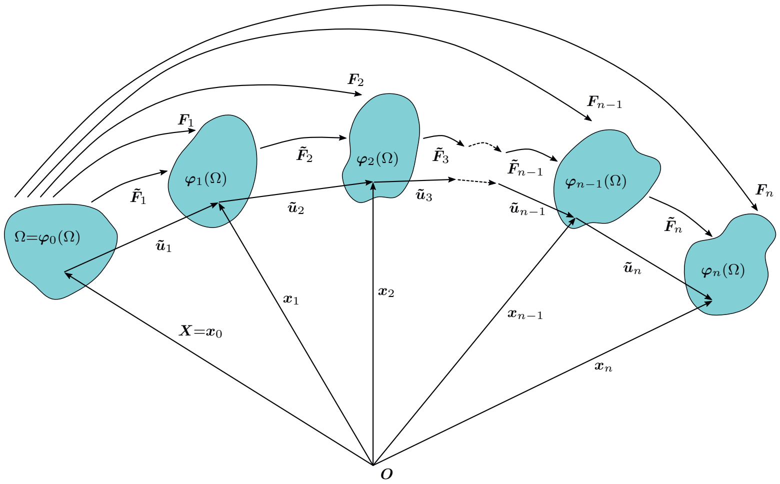


Figure 6.1: The motion and deformation of a body Ω.

is given as

$$\boldsymbol{x}_t = \boldsymbol{\varphi}_t(\boldsymbol{X}) \quad \text{with} \quad t = 0, 1, 2, \ldots, n-1, n \quad . \tag{6.2}$$

Thereby, the distance covered by the material point from one configuration to the next one is defined by the displacement

$$\tilde{\boldsymbol{u}}_t = \boldsymbol{x}_t - \boldsymbol{x}_{t-1} \quad . \tag{6.3}$$

Thus, the total displacement of the material point at time $t = n$ reads

$$\boldsymbol{u}_n = \tilde{\boldsymbol{u}}_1 + \tilde{\boldsymbol{u}}_2 + \ldots + \tilde{\boldsymbol{u}}_{n-1} + \tilde{\boldsymbol{u}}_n = \sum_{t=1}^{n} \tilde{\boldsymbol{u}}_t \quad . \tag{6.4}$$

Next, we consider the deformation process of the body. In doing so, the deformation gradient at time $t = n$ is given as a multiplicative decomposition

$$\boldsymbol{F}_n = \frac{\partial \boldsymbol{x}_n}{\partial \boldsymbol{X}} = \frac{\partial \boldsymbol{x}_n}{\partial \boldsymbol{x}_0} = \frac{\partial \boldsymbol{x}_n}{\partial \boldsymbol{x}_{n-1}} \frac{\partial \boldsymbol{x}_{n-1}}{\partial \boldsymbol{x}_{n-2}} \cdots \frac{\partial \boldsymbol{x}_2}{\partial \boldsymbol{x}_1} \frac{\partial \boldsymbol{x}_1}{\partial \boldsymbol{x}_0} = \prod_{t=1}^{n} \frac{\partial \boldsymbol{x}_{n+1-t}}{\partial \boldsymbol{x}_{n-t}} \quad . \tag{6.5}$$

With the introduction of $\tilde{\boldsymbol{F}}_t$ defining the deformation gradient from one configuration to the next one

$$\tilde{\boldsymbol{F}}_t = \frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{x}_{t-1}} \tag{6.6}$$

the relation in Eq. (6.5) can be rewritten as

$$\boldsymbol{F}_n = \tilde{\boldsymbol{F}}_n \tilde{\boldsymbol{F}}_{n-1} \ldots \tilde{\boldsymbol{F}}_2 \tilde{\boldsymbol{F}}_1 = \prod_{t=1}^{n} \tilde{\boldsymbol{F}}_{n+1-t} \quad . \tag{6.7}$$

Consequently, for the special case at time $t = 1$, the relations

$$\boldsymbol{u}_1 = \tilde{\boldsymbol{u}}_1 \quad \text{and} \quad \boldsymbol{F}_1 = \tilde{\boldsymbol{F}}_1 \tag{6.8}$$

hold for the displacement and the deformation gradient.

6.2 Remeshing procedure

In the following, we present a remeshing strategy suited for the FCM [161]. The motivation for this remeshing strategy is to improve the robustness of the FCM considering problems in finite strain. In general, the FCM analysis for such kind of problems suffers from large deformations of the fictitious domain, thus leading to severely distorted cut finite cells. These degenerated cells, in turn, have a negative effect on the convergence behavior of the Newton-Raphson scheme. Thereby, in most of the cases, the FCM analysis is aborted due to self-penetration of cut cells. In order to overcome this problem, we develop a remeshing strategy which takes advantage of the simple mesh generation process that is inherent to discretization methods based on the fictitious domain approach.

The basic procedure of the remeshing strategy is sketched in Fig. 6.2. In the figure, $\boldsymbol{\varphi}_0\left(\Omega\right)$ defines the initial configuration of the undeformed body and $\boldsymbol{\varphi}_2\left(\Omega\right)$ denotes the current configuration of the deformed body after reaching the final load step. Further, the figure shows that the computational mesh at configuration $\boldsymbol{\varphi}_1\left(\Omega\right)$ gets distorted severely during the loading which is due to large deformations of the cut finite cells. As a result of the mesh distortions, the analysis may be aborted, so that we cannot reach the desired final load anymore. In order to achieve the final load step, we remesh the deformed body at configuration $\boldsymbol{\varphi}_1\left(\Omega\right)$ and continue the simulation. Therefore, the mesh generation of the new mesh can be carried out in an efficient manner by employing Cartesian grids or structure meshes, for instance, thanks to the fictitious domain approach. Next, the essential field variables are transferred from the old mesh to the new one and, then, the simulation

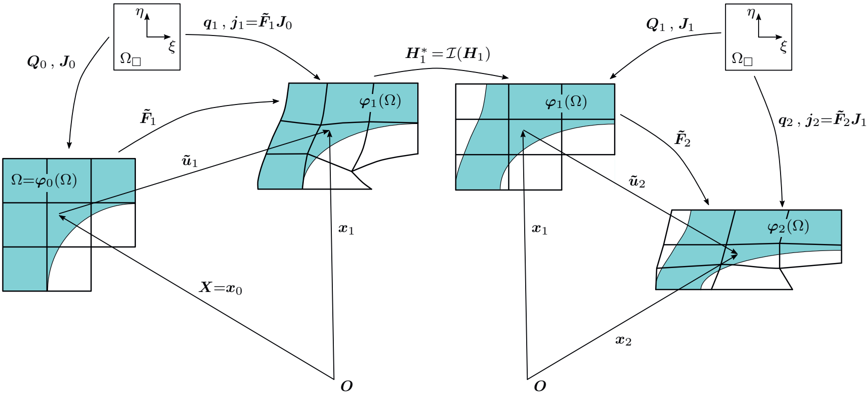


Figure 6.2: Remeshing procedure.

is continued until the final load is achieved at configuration $\boldsymbol{\varphi}_2(\Omega)$. Note that, due to the rectangular shape after remeshing, a good mesh quality is ensured at the beginning of each analysis.

In order to explain the general procedure of the remeshing strategy in more detail, let us take a look at the scheme provided in Algo. 2. At first, the initial mesh is created, together with further pre-processing steps required to run the analysis, e.g. boundary conditions, loads, etc. Next, the nonlinear global equation system in Eq. (3.21) has to be solved. To this end, the load is applied in an increment-wise manner starting from $k = 1$ and ending for the final load at $k = k_{\text{end}}$. For each load step k, we thereby employ the Newton-Raphson method in order to find a solution for the nonlinear problem. In doing so, in each Newton-Raphson iteration, the system in Eq. (3.21) is solved for the unknown displacement increment $\Delta\tilde{\mathbf{U}}^{i+1}$ and, then, the displacement is updated

$$\tilde{\mathbf{U}}_k^{i+1} = \tilde{\mathbf{U}}_k^{i} + \Delta\tilde{\mathbf{U}}^{i+1} \quad . \tag{6.9}$$

Note that, since the displacement is computed for different meshes in the case of the remeshing, we use a tilde to distinguish it from the total displacement vector $\mathbf{U}$. Further, keep in mind that standard vector or matrix quantities are represented by upright and bold symbols in order to distinguish them from tensor quantities. Next, we evaluate the remeshing criteria based on the updated solution $\tilde{\mathbf{U}}_k^{i+1}$. If all criteria are satisfied, we simply proceed with the standard Newton-Raphson procedure – meaning that we first check whether the solution of the Newton-Raphson iteration has converged and, if so, we continue with the next load step. If, however, one of the remeshing criteria fails, we go back to the converged solution of the previous load step $\tilde{\mathbf{U}}_{k-1}$. In doing so, we take the solution data and the geometric description of the deformed body of this load step as the starting point for the next analysis. Then, we create a new mesh for the deformed body.

Algorithm 2 Remeshing procedure scheme

```
Create initial mesh
for $k = 1$ to $k = k_{\text{end}}$ do                      ▷ Start load increment loop
    for $i = 1$ to $i = i_{\max}$ do                       ▷ Start Newton-Raphson iteration
        Solve system (3.21)
        Update solution (6.9)
        Check remeshing criteria
        if Check failed then
            Go back to the previous load step
            Create new mesh of the deformed body           ▷ Use last converged solution
            Transfer data from the old to the new mesh
            Set $k = k - 2$                                ▷ Enforce equilibrium step
            break                                          ▷ Redo previous load step
        end if
        Check convergence of Newton-Raphson iteration
        if converged then
            break                                          ▷ Proceed with next load step
        end if
    end for
end for
```

As the next step, we provide a data transfer from the old mesh to the new one. This data transfer step involves all essential field variables defining the deformed configuration of the body, history variables if history-dependent material models are employed, as well as boundary conditions and external loads such as tractions. After the data transfer, we reset the load increment to $k = k - 2$, break the loop of the Newton-Raphson iteration, and proceed with the loop of the load incrementation. In doing so, the new analysis is repeated for the last converged load step of the previous analysis. Consequently, the first load step of every new analysis can be considered as a balance step to recover the equilibrium, which might be violated due to the data transfer. This procedure is repeated until the final load step is achieved. In the subsequent section, we describe the remeshing criteria, the mesh generation of the deformed body, and the data transfer in more detail.

6.2.1 Remeshing criteria

In order to decide when the analysis has to be aborted and, thus, the remeshing has to be initiated, we consider several remeshing criteria. For this, the most obvious choice is to consider the convergence behavior of the Newton-Raphson procedure, meaning that the analysis is aborted if the Newton-Raphson iteration exceeds its maximum value. In the context of the FCM, this is mainly induced by cut finite cells – since these cells, usually, are susceptible to self-penetration ($\det \boldsymbol{F} < 0$) due to large deformations of the fictitious domain. However, in some cases it is of advantage to abort the analysis before the Newton-Raphson method fails. Consequently, specific criteria are needed to evaluate the quality of the deformed finite cells. Keep in mind that the undeformed mesh of each analysis ensures a good mesh quality since Cartesian grids or structured meshes are utilized. In the following, as a means to evaluate the quality of the deformed mesh, we propose three additional criteria to decide whether remeshing has to be initiated or not. Further, all criteria are based on the Jacobi matrix. For an analysis from configuration t to $t + 1$, the Jacobi matrix of the deformed mesh is consequently given as

$$\mathbf{J}_t = \begin{bmatrix} x_{t,\xi} & x_{t,\eta} & x_{t,\zeta} \\ y_{t,\xi} & y_{t,\eta} & y_{t,\zeta} \\ z_{t,\xi} & z_{t,\eta} & z_{t,\zeta} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_1 & \mathbf{G}_2 & \mathbf{G}_3 \end{bmatrix} \tag{6.10}$$

and the corresponding Jacobi matrix of the deformed mesh is defines as

$$\mathbf{j}_{t+1} = \tilde{\mathbf{F}}_{t+1}\mathbf{J}_t = \begin{bmatrix} x_{t+1,\xi} & x_{t+1,\eta} & x_{t+1,\zeta} \\ y_{t+1,\xi} & y_{t+1,\eta} & y_{t+1,\zeta} \\ z_{t+1,\xi} & z_{t+1,\eta} & z_{t+1,\zeta} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_1 & \mathbf{g}_2 & \mathbf{g}_3 \end{bmatrix} \quad . \tag{6.11}$$

Here, $\mathbf{G}_i$ and $\mathbf{g}_i$ are the columns of the Jacobi matrix and denote the tangential vectors. Keep in mind that, here, we use bold and upright symbols for the vector and matrix quantities in order to distinguish them from tensor ones. Finally, the criteria may be evaluated for different point sets – whereby, in this thesis, we choose the quadrature points $\mathbf{x}_k$ as the evaluation points. Further, in order to ease the subsequent formulations, we introduce the following abbreviations for the Jacobi matrices

$$\mathbf{J}^k = \mathbf{J}_t\left(\mathbf{x}_k\right) \quad \text{and} \quad \mathbf{j}^k = \mathbf{j}_{t+1}\left(\mathbf{x}_k\right) \quad , \tag{6.12}$$

the tangential vectors

$$\mathbf{G}_i^k = \mathbf{G}_i\left(\mathbf{x}_k\right) \quad \text{and} \quad \mathbf{g}_i^k = \mathbf{g}_i\left(\mathbf{x}_k\right) \quad , \tag{6.13}$$

and the deformation gradient

$$\tilde{\mathbf{F}}^k = \tilde{\mathbf{F}}_{t+1}(\mathbf{x}_k) \quad . \tag{6.14}$$

6.2.1.1 Ratio of Jacobians

In the first case we employ a criterion based on the ratio of Jacobians [162–164]. To this end, for each finite cell c, we determine the minimum and maximum values of the determinant of the Jacobi matrix of the deformed mesh and compute the ratio

$$R = \frac{\min\limits_k \det\mathbf{j}^k}{\max\limits_l \det\mathbf{j}^l} \quad . \tag{6.15}$$

Consequently, the ratio of Jacobians evaluates the ratio of volumetric deformations considering its extreme values of a predefined point set. Since the condition $\det\mathbf{j}^k > 0$ has to be satisfied for each point within the cell, the range of R is given as

$$0 < R \leq 1 \quad . \tag{6.16}$$

Thereby, $R = 1$ defines the optimal case and values close to zero characterize cells of a poor quality. Finally, note that employing Cartesian grids or structured meshes – which is generally the case when applying the FCM – the ratio of Jacobians simplifies to

$$R = \frac{\min\limits_k \det\tilde{\mathbf{F}}^k \det\mathbf{J}^k}{\max\limits_l \det\tilde{\mathbf{F}}^l \det\mathbf{J}^l} = \frac{\min\limits_k \det\tilde{\mathbf{F}}^k}{\max\limits_l \det\tilde{\mathbf{F}}^l} \tag{6.17}$$

since, with regard to the rectangular cells, the following condition holds for the ratio of Jacobians of the undeformed mesh

$$\frac{\det\mathbf{J}^k}{\det\mathbf{J}^l} = 1 \quad . \tag{6.18}$$

6.2.1.2 Orthogonality

In the second criterion, we introduce a measurement evaluating the mesh quality by considering the orthogonality property of each finite cell. Thereby, the cells of the undeformed mesh are optimal in this sense since they include a rectangular shape. Consequently, the criterion has to be developed for the cells of the deformed mesh. Therefore, we proceed as suggested in [165]. In doing so, we introduce the following orthogonality criterion

$$O = \min_{i,j,k} \frac{\sqrt{\left(\mathbf{g}_i^k \cdot \mathbf{g}_i^k\right)\left(\mathbf{g}_j^k \cdot \mathbf{g}_j^k\right) - \left(\mathbf{g}_i^k \cdot \mathbf{g}_j^k\right)^2}}{\sqrt{\left(\mathbf{g}_i^k \cdot \mathbf{g}_i^k\right)\left(\mathbf{g}_j^k \cdot \mathbf{g}_j^k\right)}} \quad . \tag{6.19}$$

which is based on the tangential vectors of the deformed Jacobi matrix, see Eq. (6.11). Since the condition $\det\mathbf{j}^k > 0$ holds at each point – which implies that the tangential vectors $\mathbf{g}_i^k$ are linearly independent – the range of the orthogonality criterion is defined as

$$0 < O \leq 1 \quad . \tag{6.20}$$

Thereby, $O = 1$ defines the optimal case and values close to zero indicate cells of a poor quality.

6.2.1.3 Inverse aspect ratio

As the third criterion, we consider a measurement based on the aspect ratio of the tangential vectors. To this end, we introduce a criterion relating the aspect ratio of the deformed cell with the inverse aspect ratio of the undeformed one. In doing so, the definition of the criterion is given as

$$A = \min_{i,j,k} \frac{\left\| \mathbf{g}_i^k \right\|_2}{\left\| \mathbf{g}_j^k \right\|_2} \frac{\left\| \mathbf{G}_j^k \right\|_2}{\left\| \mathbf{G}_i^k \right\|_2} \quad . \tag{6.21}$$

Since the tangential vectors are linearly independent – due to condition $\det \mathbf{j}^k > 0$ – the range of A is defined as

$$0 < A \leq 1 \quad . \tag{6.22}$$

6.2.1.4 Performance of the suggested remeshing criteria

Finally, let us have a look at the performance of the presented remeshing criteria. Therefore, we consider a single cube that is subjected to different load cases: uniaxial pressure, shear, and compression. For the investigations, the cube is discretized by one finite cell. Further, a hyperelastic and isotropic material behavior based on a polyconvex strain energy density function is assumed. A brief description of the material model is provided in Sec. 2.3.2. Further, the material parameters are listed in Tab. 5.2.

The values of the remeshing criteria of the different cases are plotted in Fig. 6.3, 6.4, and 6.5, respectively. Fig. 6.3 shows the values of the remeshing criteria considering the cube subjected to uniaxial pressure. From the figure it can be seen that due to the deformation of the cell the ratio of Jacobians ($R = 1$) and the orthogonality ($O = 1$) do not change during the loading. This is because the deformed cell has a rectangular shape and the determinant of the deformation gradient has the same value at each quadrature point. However, the inverse aspect ratio changes during the loading. Further, the values of the remeshing criteria for the cube under shear loading are plotted in Fig. 6.4. Here, the values of the orthogonality and the inverse aspect ratio change during loading, while the ratio of the Jacobians ($R = 1$) does not. Finally, Fig. 6.5 shows the results of the cube under compression. In this special case, all remeshing criteria are not changed during the loading.

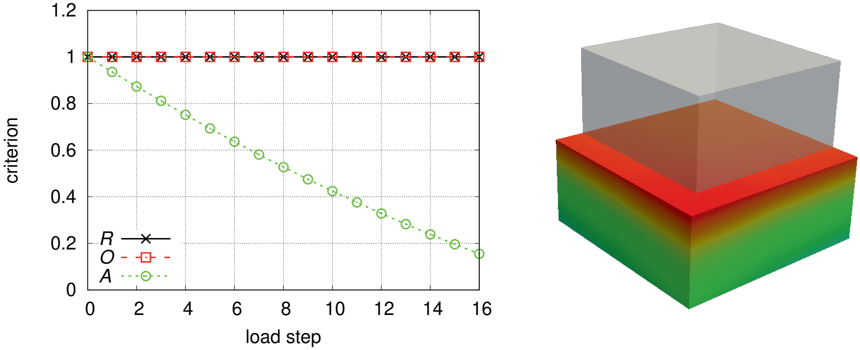


Figure 6.3: Remeshing criteria for a single cube under uniaxial pressure.

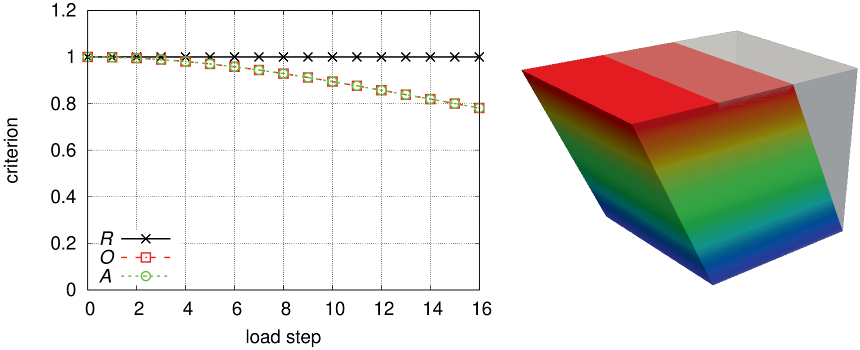


Figure 6.4: Remeshing criteria for a single cube under shear loading.

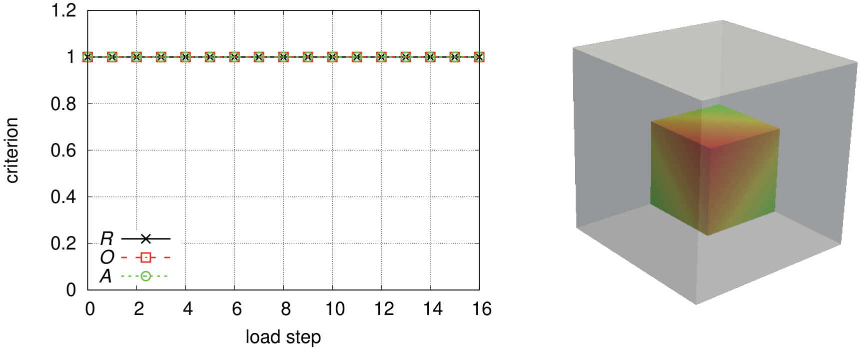


Figure 6.5: Remeshing criteria for a single cube under compression.

6.2.2 Mesh generation

After one of the remeshing criteria is activated and, thus, the analysis is aborted, the next step is to remesh the deformed body. Consequently, a geometry description of the body is needed during the simulation. Therefore, the geometric description of the body is taken into account by employing a surface triangulation. This strategy is illustrated in Fig. 6.6 for the two-dimensional case. Here, the boundary of the body is discretized by line segments forming a closed polygonal chain. In the figure, the boundary mesh is represented by purple lines and dots. Then, the mesh generation for each analysis is performed as follows during the simulation. Starting from the configuration $\boldsymbol{\varphi}_0\left(\Omega\right)$ of the undeformed body we create a bounding box enclosing the line segment mesh used to describe the body. Then, the bounding box is discretized by utilizing a structured mesh. Further, cells that are located completely within the fictitious domain are discarded. Next, the nodes of the line segments are mapped as post-grid points into the mesh. Thanks to the rectangular shape of the finite cells, the inverse mapping required to map the global position of the points $\mathbf{X}_p$ to the local position $\mathbf{r}_p$ of the corresponding cell is simple. Following Eq. (3.13), the inverse mapping for the three-dimensional case can be obtained in a straightforward manner as

$$\mathbf{r} = \mathbf{Q}^{-1} = \begin{bmatrix} {}^{2}\!/_{H_X}\left(\mathbf{X} - \mathbf{X}_{\min}\right) - 1 \\ {}^{2}\!/_{H_Y}\left(\mathbf{Y} - \mathbf{Y}_{\min}\right) - 1 \\ {}^{2}\!/_{H_Z}\left(\mathbf{Z} - \mathbf{Z}_{\min}\right) - 1 \end{bmatrix} \quad . \tag{6.23}$$

During the simulation, the node displacement values of the boundary mesh are evaluated by the element shape functions. Next, by adding the node displacement values to the node position, a geometric description of the deformed body is ensured during the analysis. Then, after remeshing is initiated, the deformed boundary mesh builds the basis for generating of the next mesh. In the figure this is illustrated for the deformed body at configuration $\boldsymbol{\varphi}_1\left(\Omega\right)$. At this configuration, remeshing is initiated and a new FCM mesh is generated by discretizing the boundary box of the deformed boundary mesh. Moreover, note that the boundary mesh builds the basis for the numerical integration of the cut cells. Here, it is used to check whether a quadrature point is within the physical domain or not [166].

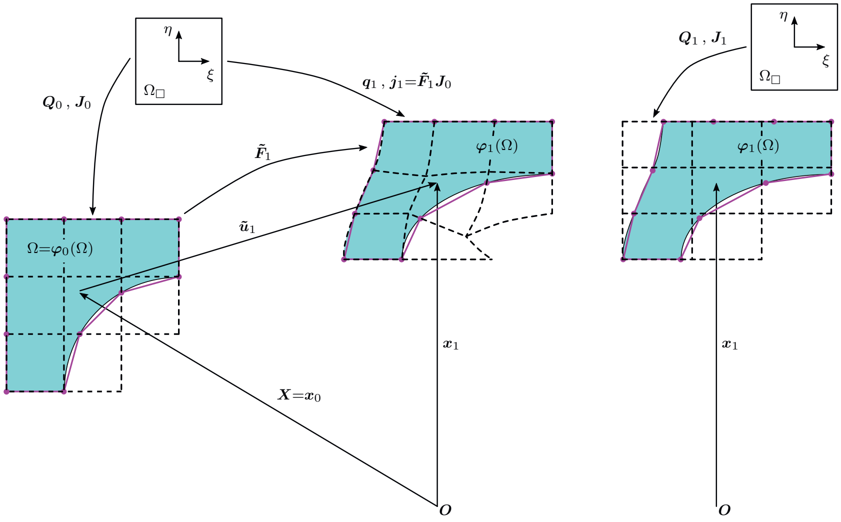


Figure 6.6: Mesh generation.

6.2.3 Data transfer

In order to proceed with the analysis, as the final step of the remeshing procedure, we have to transfer the necessary data from the old to the new mesh, e.g. displacement-based variables or history variables if nonlinear material models are involved. In this thesis, we introduce a *local* radial basis function (RBF) interpolation scheme [167–169] to perform the data transfer. Thereby, we consider the quadrature points of the old mesh as the source points and the quadrature points of the new mesh as the target points. Fig. 6.7 shows the graphical interpretation with $\mathcal{I}$ denoting the interpolation operator. Note that the fictitious quadrature points of the old mesh are excluded from the set of the source points, while the set of target points include the fictitious quadrature points. In the following, a source point is denoted as $\boldsymbol{x}^s$ and a target point as $\boldsymbol{x}^t$.

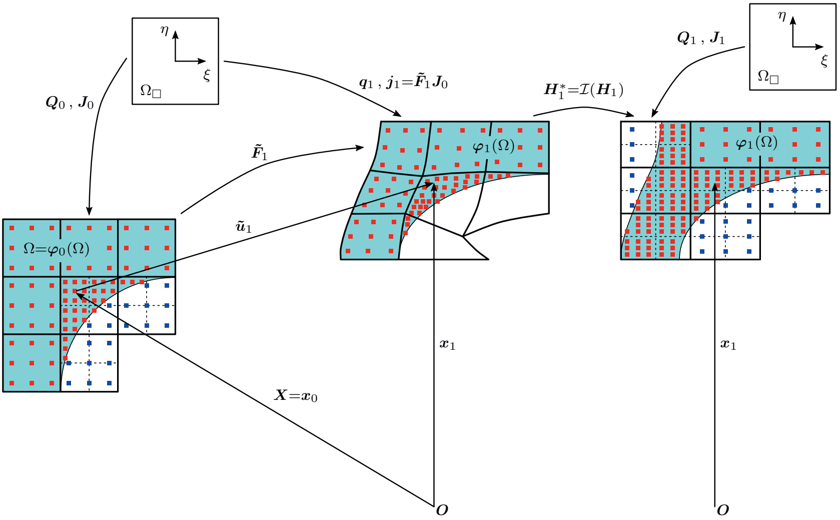


Figure 6.7: Data transfer between old and new mesh where the fictitious quadrature points of the old mesh are excluded from the set of the source points.

Next, we briefly describe the basic idea of the *local* RBF interpolation scheme. In order to compute the values of a target point $\boldsymbol{x}^t$, we start off by searching for the n^n nearest source points of $\boldsymbol{x}^t$ and deposit their indices in a set N. So as to find the nearest neighbor, we thereby employ a k-d tree [169, 170] in order to reduce the effort during the searching procedure. For the target point $\boldsymbol{x}^t$, we then set up an interpolation scheme that is based on the source points of set N. In the same way, we proceed with the remaining target points. Consequently, for each target point an individual interpolation is generated which can be applied in parallel. Further, in order to distinguish this variant from its global one – where only one global interpolation scheme is generated considering all source points at the same time – we call it *local* RBF interpolation.

Then, given the set N including the n^n nearest source points, an individual value v^t of a target point is computed by a weighted sum as

$$v^t = \sum_{i \in N} \lambda_i \, \theta \left(\left\| \boldsymbol{x}^t - \boldsymbol{x}_i^s \right\|_2 \right) \quad . \tag{6.24}$$

In Eq. (6.24) λ_i denotes the weight and $\theta(r)$ defines the related scalar-valued radial basis function where the argument is the Euclidean norm of the distance between the target point $\boldsymbol{x}^t$ and the source point $\boldsymbol{x}_i^s$. Consequently, in order to compute v^t, we first have to determine the unknown weights λ_i. To this end, the n^n unknown weights λ_i are computed by solving a linear system of equations

$$v_j^s = \sum_{i \in N} \lambda_i \, \theta \left(\left\| \boldsymbol{x}_i^s - \boldsymbol{x}_j^s \right\|_2 \right) \quad , \quad j \in N \tag{6.25}$$

that is obtained by incorporating the interpolation conditions of the source points – where v_j^s denotes the related value of source point $\boldsymbol{x}_j^s$. Further, as the choice for $\theta(r)$, commonly used radial basis functions, for instance, are

- the *Gaussian function* (GF):
$$\theta(r) = e^{-r^2} \quad , \tag{6.26}$$
- the *multiquadric* (MQ):
$$\theta(r) = \sqrt{1 + r^2} \quad , \tag{6.27}$$
- the *inverse multiquadric* (IMQ):
$$\theta(r) = \frac{1}{\sqrt{1 + r^2}} \quad , \tag{6.28}$$
- or the *thin plate spline* (TPS):
$$\theta(r) = r^2 \ln(r) \quad . \tag{6.29}$$

These RBFs are plotted in Fig. 6.8.

The choice of the RBF is problem-dependent. In the presented remeshing strategy, an important variable for the data transfer is given by the deformation gradient. At each remeshing step, we have to transfer the total deformation gradient of the old mesh to the new one in order to proceed with the simulation. In this thesis, the transfer of the deformation gradient is carried out by taking advantage of its relation with the displacement gradient ($\boldsymbol{F} = \boldsymbol{H} + \boldsymbol{I}$). Consequently, for the remeshing step illustrated in Fig. 6.7, the displacement gradient is interpolated as

$$\boldsymbol{H}_1^* = \mathcal{I}\left(\boldsymbol{H}_1\right) \tag{6.30}$$

where the superscript * is introduced in order to distinguish the approximated variable from its original one. In doing so, an approximation of the deformation gradient is obtained as

$$\boldsymbol{F}_1^* = \boldsymbol{H}_1^* + \boldsymbol{I} = \mathcal{I}\left(\boldsymbol{H}_1\right) + \boldsymbol{I} \quad . \tag{6.31}$$

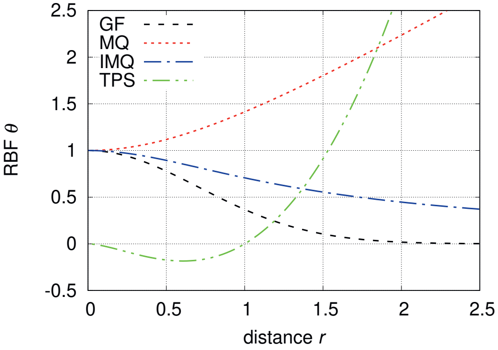


Figure 6.8: Commonly used radial basis functions.

The reason for interpolating the displacement gradient instead of the deformation gradient is that they have different characteristics. In the case of an undeformed body, the deformation gradient corresponds to the identity ($\boldsymbol{F} = \boldsymbol{I}$), while the displacement gradient, on the other hand, equals zero ($\boldsymbol{H} = \boldsymbol{0}$). Having recalled the characteristics of $\boldsymbol{F}$ and $\boldsymbol{H}$, next, let us again consider the case depicted in Fig. 6.7. As already mentioned, the fictitious quadrature points of the old mesh are excluded from the set of source points while the fictitious quadrature points in the new mesh are included in the set of target points. Consequently, the data transfer based on the *local* RBF can be differentiated in an interpolation phase (from physical source to physical target points) and an extrapolation phase (from physical source to fictitious target points). For the interpolation of the physical points, we aim to achieve target values that are close to the source values. During the extrapolation phase, on the other hand, we intend to obtain a smooth transition of the displacement gradient from the physical into the fictitious domain. In doing so, the goal is to achieve a zero displacement gradient ($\boldsymbol{H} = \boldsymbol{0}$) for fictitious target points that are placed far away from the physical source points. A radial basis function that complies well with the interpolation and extrapolation requirements of the displacement gradient is the *inverse multiquadric* RBF given in Eq. (6.28). In this thesis, we utilize a modification of the *inverse multiquadric* RBF in which the input argument is scaled for each source point individually as follows

$$\tilde{\varphi}_i\left(r\right) = \varphi\left(\beta \frac{r}{\bar{r}_i}\right) \quad . \tag{6.32}$$

In Eq. (6.32), β denotes a scaling factor and $\bar{r}_i$ defines the mean distance of source point $\boldsymbol{x}_i^s$ with respect to its n^r nearest neighbors. These additional parameters allow to further tune the RBF interpolation.

Summarizing, we set up an individual interpolation scheme for each target point $\boldsymbol{x}^t$, based on a modification of the *inverse multiquadric* RBF $\tilde{\varphi}_i\left(r\right)$. To this end, we have to define the following parameters:

- the number of source points per target point n^n whose indices are deposited in set N,

- the number of nearest neighbors n^r per source point in order to compute the mean distance $\bar{r}_i$, and
- the scaling factor β.

Thereby, practical experience has shown that $n^n = 50$, $n^r = 3$, and $\beta = 1, \ldots, 2$ are a good choice.

6.3 Finite strain problems

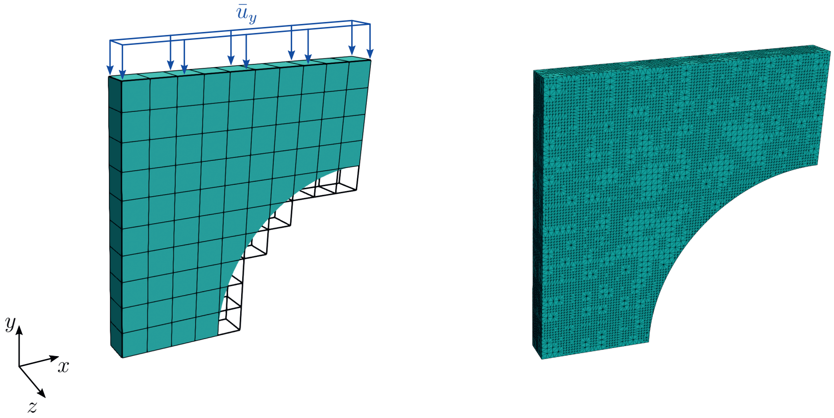
In this section, we investigate the performance of the remeshing strategy considering problems that undergo large deformations. For all examples, we assume a hyperelastic and isotropic material behavior based on a polyconvex strain energy density function. A brief explanation of the underlying equations of the constitutive model is provided in Sec. 2.3.2. Further, the material parameters used for all examples are listed in Tab. 5.2.

6.3.1 Plate with a circular hole

In the first example, we again consider the plate with a circular hole from Sec. 5.2. This time, however, the plate undergoes large deformations assuming a hyperelastic material behavior. For the investigation, the plate is subjected to symmetry boundary conditions and a prescribed displacement of $\bar{u}_y = 15.25\,\text{mm}$ is applied at the top surface. The FCM model as well as the discretization are depicted in Fig. 6.9a. Here, we employ a structured mesh using 78 finite cells. Further descriptions of the geometry are provided in Fig. 5.3 and Sec. 5.2. Moreover, the triangulated surface mesh in Fig. 6.9b represents the undeformed geometry that is the starting point for the mesh generation during the remeshing procedure.

The purpose of this example is to study the performance of the presented remeshing strategy in terms of accuracy. To this end, the prescribed displacement $\bar{u}_y = 15.25\,\text{mm}$ is applied utilizing 61 equal load steps. Thereby, every analysis is aborted after 10 load steps, meaning that a total number of 6 analyses are required to achieve the final displacement. Next, in order to study the effect of the remeshing strategy, the solutions of the FCM analyses are compared to a reference that is obtained by employing an overkill *p*FEM analysis. The *p*FEM discretization is illustrated in Fig. 5.4a. Here, we use 3,200 elements and an ansatz order of $p = 5$. Further, the blending function method is applied in order to represent the curved boundary more accurately. Moreover, to increase the robustness of the FCM analyses, we use an $\alpha = 10^{-q}$ value with parameter $q = 5$ and $\mu_t = 0.1$ as the threshold for the basis function removal criterion.

To study the effect of the remeshing, let us take a look at Fig. 6.10. Here, the strain energy is plotted versus the displacement applying different values of the scaling parameter $\beta = 1, 2$ – where each vertical black and dashed line represents a remeshing process. From the figure, we can see that the results of the FCM are in a good agreement with the reference solution. Next, in order to study the effect of the remeshing process in more detail, Fig. 6.11 shows the relative error in strain energy. Here, the green curve in the figure represents the FCM solution without remeshing. The FCM solution applying the remeshing, on the other hand, leads to a jump in the error – achieving a maximum error of about 10^{-2}, while the FCM solution without remeshing results in a maximum error value



(a) **(b)**

Figure 6.9: Plate with a circular hole. **(a)** FCM model and initial mesh. **(b)** Triangulated surface mesh of the undeforemed plate.

of about 10^{-4}. However, considering the strain energy curves in Fig. 6.10, the error of the remeshing is negligibly small.

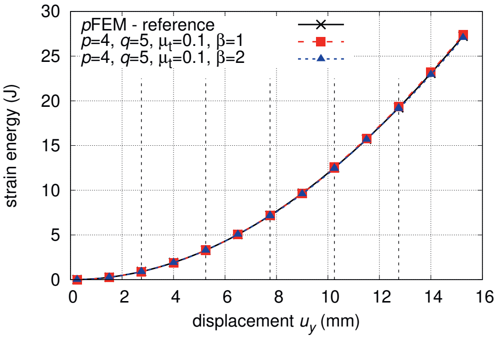


Figure 6.10: Energy-displacement curves.

Next, let us investigate the influence of the remeshing on the stress. To this end, we consider the stress component σ_{yy} at point A with coordinates $(39.0\,\text{mm}, 1.0\,\text{mm}, 1.0\,\text{mm})$. The stress values of σ_{yy} at point A are illustrated in Fig. 6.12. Considering the curves, we can see that the FCM results are in a good agreement with the reference solution. In order to obtain a better overview in terms of the accuracy, the relative error in σ_{yy} at point

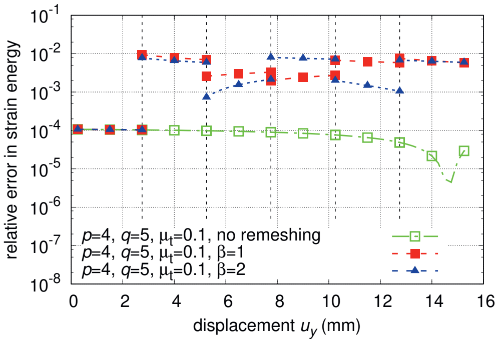


Figure 6.11: Relative error in strain energy.

A is plotted in Fig. 6.13. Again, the green curve represents the FCM solution without remeshing. As it can be seen from the figure, the relative error in the stress applying the remeshing is, in most of the cases, higher than without remeshing. Further, comparing the local error in the stress with the global error in the energy, we can see that the error in the stress is higher by approximately one order of magnitude. Moreover, the effect of the parameter β used for the interpolation is negligible. In the subsequent examples, we therefore choose $\beta = 1$ since – based on experiments – this choice shows a more robust behavior.

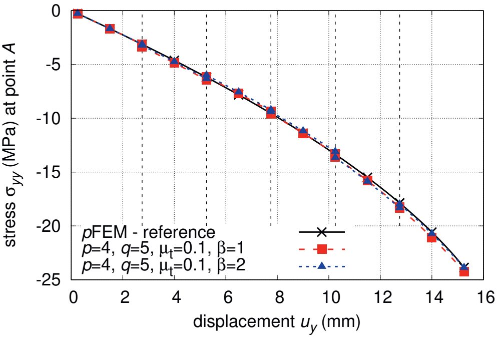


Figure 6.12: Stress-displacement curves at point A.

The contour plots of the pFEM solution and the FCM applying the remeshing are illustrated in Fig. 6.14a and 6.14b for the final displacement $\bar{u}_y = 15.25\,\text{mm}$, respectively. From the figures, it can be inferred that a combination of the FCM and the presented remeshing strategy provides accurate results.

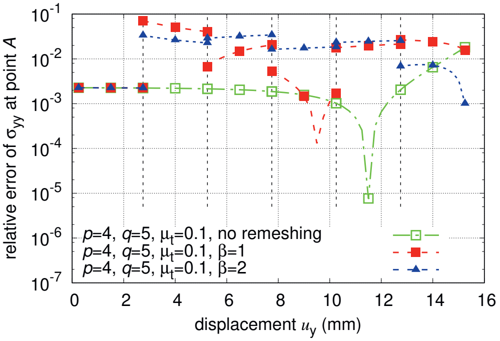
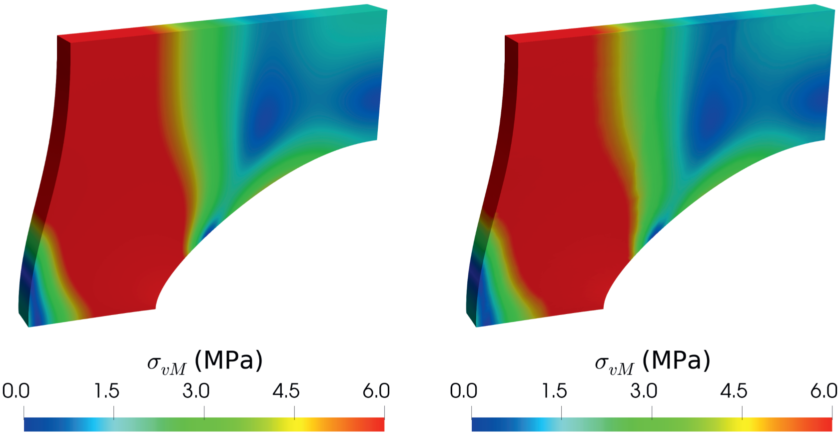


Figure 6.13: Relative error in σ_{yy} at point A.



(a) **(b)**

Figure 6.14: Contour plots of the von Mises stress σ_{vM} at the final displacement $\bar{u}_y = 15.25\,\mathrm{mm}$. **(a)** pFEM solution. **(b)** FCM solution applying the remeshing.

6.3.2 Single cube connector

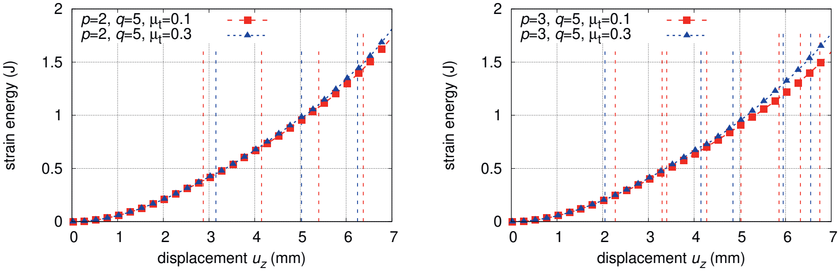
In the next example, we again study the single cube connector from Sect. 5.3.1, subjected to symmetric boundary conditions. The setup of the problem is depicted in Fig. 5.17a. At the top surface, we apply a prescribed displacement of $\bar{u}_z = 7.0\,\mathrm{mm}$ acting in negative z-direction. Since the geometry is approximated by a triangulated surface, the problem is changed slightly as compared to the analysis in Sec. 5.3.1 where we consider the level set function. For the remeshing procedure, the bounding box of the single cube connector is discretized employing a structured mesh with 6 finite cells in each spatial direction. This meshing strategy is applied at every remeshing step. The initial mesh as well as the triangulated surface of the undeformed body are depicted in Fig. 6.18a. Moreover, for the numerical integration, we employ the adaptive Gaussian scheme utilizing an octree based on $k = 3$ refinement levels.

In order to study the performance of the presented remeshing strategy, we apply various ansatz orders $p = 2, 3, 4$ for different thresholds $\mu_t = 0.1, 0.3$ of the basis function removal criterion. For each analysis, we employ an $\alpha = 10^{-q}$ value with parameter $q = 5$. In this example for the initiation of the remeshing, we furthermore set the thresholds of all remeshing criteria – ratio of Jacobians (R), orthogonality (O), and inverse aspect ratio (A) – to zero. Fig. 6.15a, 6.15b, and 6.15c show the energy-displacement curves of the different analyses for a fixed ansatz order – where each vertical and dashed line represents a remeshing process. As it can be seen from the figures, the remeshing improves the robustness of the FCM significantly. For $p = 2$, the remeshing allows to increase the value of the prescribed displacement by a factor greater than 2. This behavior is even more pronounced for the analyses employing an ansatz of higher order $p = 3, 4$. Here, we can increase the value of the prescribed displacement by factors of about 3.3 and 4.5, respectively. Considering the influence of the basis function removal, we can see that applying a higher threshold value $\mu_t = 0.3$ improves the robustness in such a way that less remeshing steps are required. Although the higher threshold of the basis function removal criterion leads to small deviations in the results of the final load steps, we can see that the energy values are still in a good agreement as compared to the values applying a lower threshold $\mu_t = 0.1$, however.

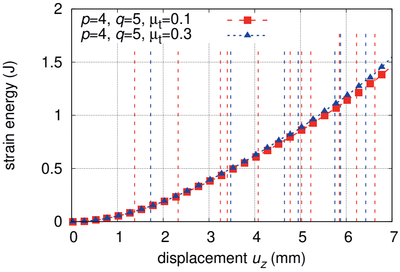
Next, we study the influence of the ansatz order on the solution. To this end, the energy-displacement curves applying $p = 2, 3, 4$ and a fixed threshold for the basis function removal criterion μ_t are plotted in Fig. 6.16a and 6.16b, respectively. The figures show that the higher ansatz orders provide a more accurate solution of the problem, especially at the final load steps.

In order to measure the quality of the computational mesh during the deformation process, Fig. 6.17 shows the values of all remeshing criteria. For each analysis, we thereby employ a basis function removal threshold of $\mu_t = 0.3$ an ansatz order of $p = 3$. Moreover, in addition to the remeshing criteria (R, O, and A), the figure also shows the determinant of the total as well as the current deformation gradient – denoted by $\det\boldsymbol{F}$ and $\det\tilde{\boldsymbol{F}}$, respectively. In most of the cases, the analysis apparently fails due to the fact that $\det\boldsymbol{F} \leq 0$ or $\det\tilde{\boldsymbol{F}} \leq 0$ – thus initiating a new remeshing step. Further, we can see that the remeshing leads to a significant improvement in the quality of the new computational mesh, especially in terms of the orthogonality property.

Fig. 6.18 shows the four different computational meshes that are required to achieve the final displacement. For each analysis, we thereby employ a basis function removal thresh-

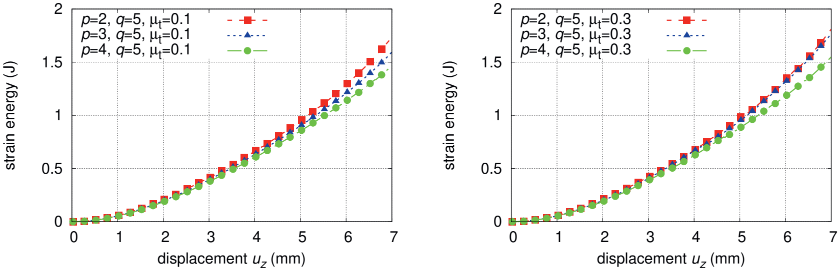


(a) $p = 2$ **(b)** $p = 3$



(c) $p = 4$

Figure 6.15: Energy-displacement curves applying different thresholds for the basis function removal criterion and a fixed ansatz order.



(a) $\mu_t = 0.1$ **(b)** $\mu_t = 0.3$

Figure 6.16: Energy-displacement curves applying different ansatz orders and a fixed threshold for the basis function removal criterion.

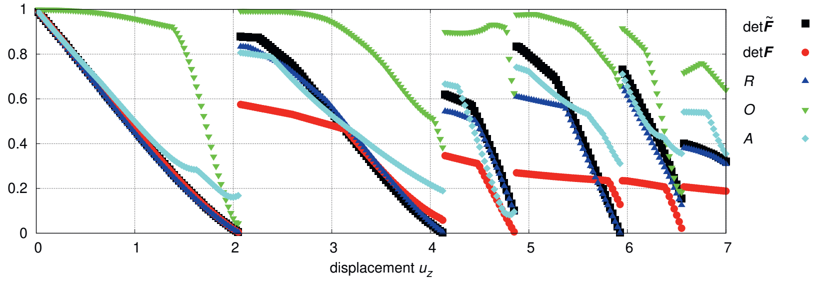
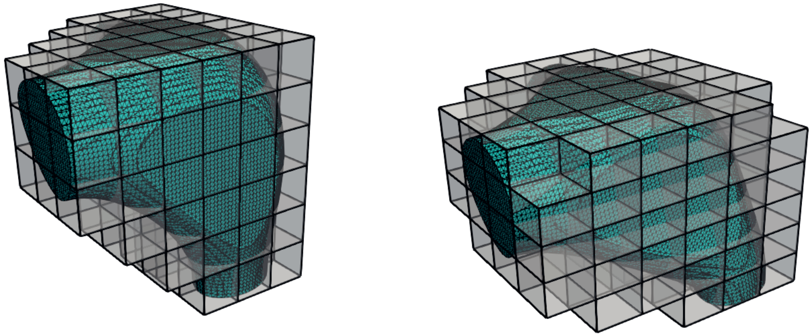


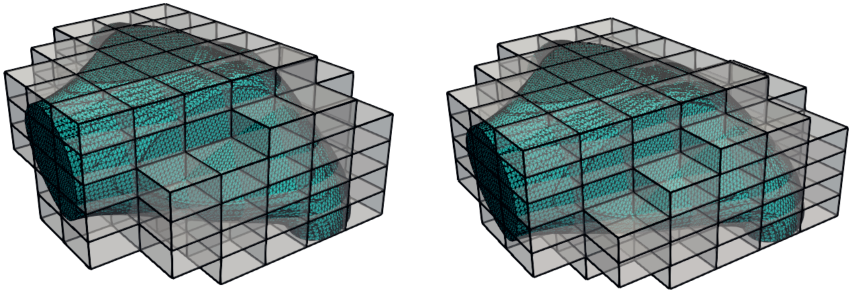
Figure 6.17: Remeshing criteria applying a basis function removal threshold of $\mu_t = 0.3$ and an ansatz of order $p = 3$.

old of $\mu_t = 0.3$ and an ansatz order of $p = 2$. Moreover, in addition to the computational meshes, also the corresponding surface triangulation of the undeformed or deformed geometry is illustrated. Finally, in order to obtain a visual representation of the deformation, Fig. 6.19 shows the von Mises stress σ_{vM} for different values of the prescribed displacement $\bar{u}_z$.



(a) 129 finite cells.

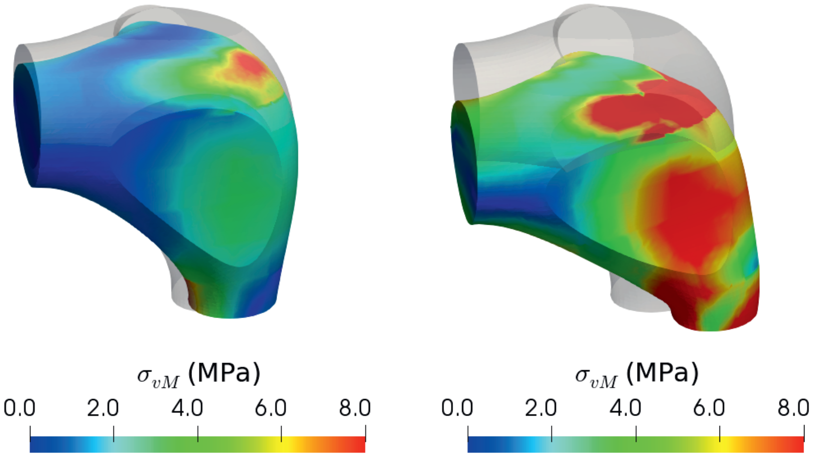
(b) 151 finite cells.



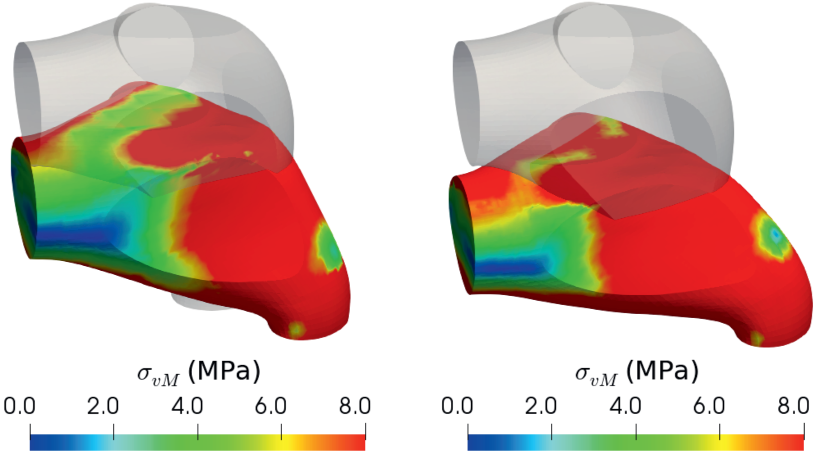
(c) 148 finite cells.

(d) 156 finite cells.

Figure 6.18: Computational meshes and corresponding surface triangulation of the undeformed or deformed geometry applying a basis function threshold of $\mu_t = 0.3$ and an ansatz order of $p = 2$.



(a) $\bar{u}_z = 1.0\,\mathrm{mm}$ **(b)** $\bar{u}_z = 3.0\,\mathrm{mm}$



(c) $\bar{u}_z = 5.0\,\mathrm{mm}$ **(d)** $\bar{u}_z = 7.0\,\mathrm{mm}$

Figure 6.19: Contour plots of the von Mises stress σ_{vM} for different values of the prescribed displacement $\bar{u}_z$.

6.3.3 Complex cube connector

In the last example, we again consider the complex cube connector from Sec. 5.3.2, which is composed of eight individual cube connectors. The setup of the problem is depicted in Fig. 5.24a and the geometry parameters are listed in Tab. 5.4. As illustrated in the figure, the complex cube connector is fixed at the bottom, and a prescribed displacement of $\bar{u}_z = 18.0\,\mathrm{mm}$ is applied at its top surface, acting in negative z-direction. Further, the top surface is fixed in x- and y-direction. The initial mesh as well as the surface triangulation of the undeformed geometry are depicted in Fig. 6.22a. For the discretization of the bounding box of the triangulated surface mesh, we apply a structured mesh utilizing 16 finite cells in each spatial direction resulting in 2,968 finite cells. For the analysis, we further apply $\alpha = 10^{-q}$ with $q = 5$ and an order of $p = 2$ for the ansatz. Furthermore, the numerical integration is carried out by the adaptive Gaussian scheme utilizing an octree with $k = 3$ refinement levels. During the simulation, the thresholds of all remeshing criteria – ratio of Jacobians (R), orthogonality (O), and inverse aspect ratio (A) – are set to zero.

Fig. 6.20 shows the energy-displacement curves for different thresholds $\mu_t = 0.1, 0.4$ of the basis function removal criterion. As it can be seen from the figure, the results are in a good agreement. However, the higher value of the basis function removal criterion allows to reduce the number of remeshing steps. Here, we only require two remeshing steps to apply the final value of the prescribed displacement $\bar{u}_z$ while 6 remeshing steps are needed employing a lower threshold for the basis function removal criterion ($\mu_t = 0.1$).

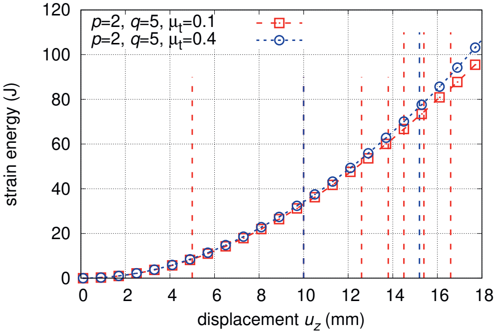


Figure 6.20: Energy-displacement curves.

For the simulation applying $p = 2$, $\mu_t = 0.4$, and $q = 5$, Fig. 6.21 shows the corresponding values of the remeshing criteria as well as the determinant of the total and the current deformation gradient ($\det\boldsymbol{F}$ and $\det\tilde{\boldsymbol{F}}$). From the figure, it can be inferred that remeshing is initiated due to self-penetration of the mesh ($\det\boldsymbol{F} \leq 0$ and $\det\tilde{\boldsymbol{F}} \leq 0$). Moreover, considering the values of the criteria after remeshing it is interesting to see that the quality of the mesh improves during some of the load steps.

The different meshes applying $\mu_t = 0.4$ as well as the corresponding surface triangulation of the undeformed or the deformed structure are illustrated in Fig. 6.22. Moreover, in order to obtain an impression of the deformation, Fig. 6.23 shows the contour plots of the von Mises stress σ_{vM} for different values of the prescribed displacement $\bar{u}_z$.

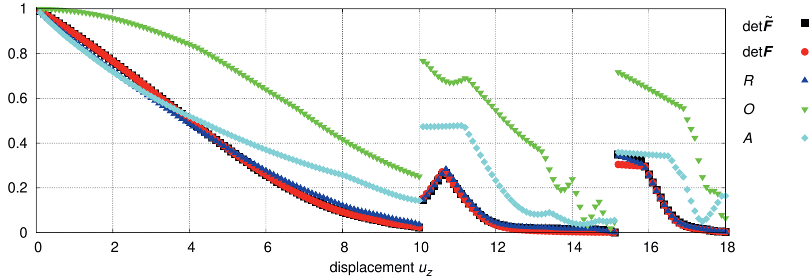
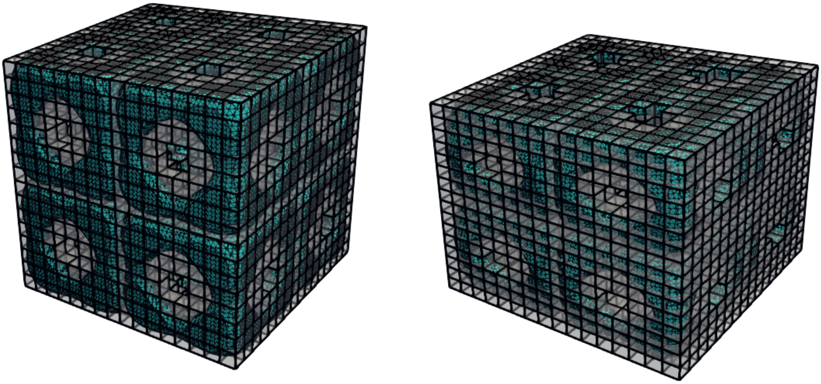
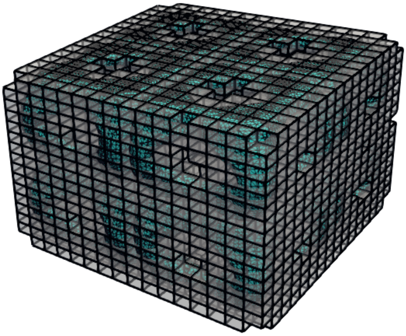


Figure 6.21: Remeshing criteria applying a basis function removal threshold of $\mu_t = 0.4$ and an ansatz of order $p = 2$.



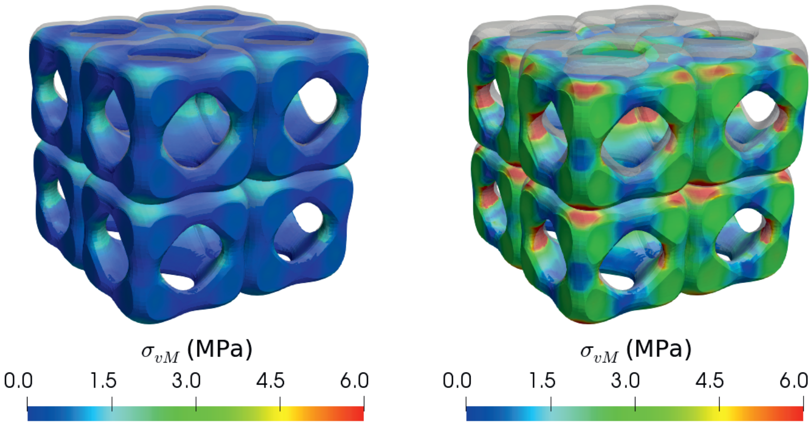
(a) 2,968 finite cells.

(b) 3,182 finite cells.

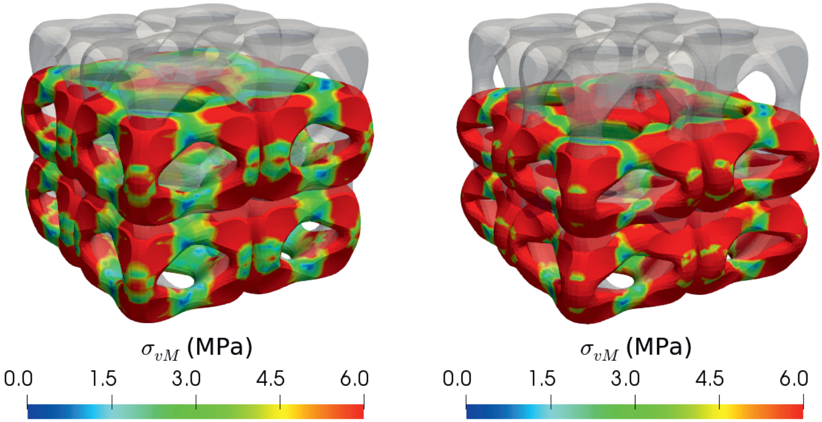


(c) 3,329 finite cells.

Figure 6.22: Computational meshes and corresponding surface triangulation of the undeformed or deformed geometry applying a basis function threshold of $\mu_t = 0.4$ and an ansatz order of $p = 2$.



(a) $\bar{u}_z = 1.0\,\mathrm{mm}$ **(b)** $\bar{u}_z = 6.0\,\mathrm{mm}$



(c) $\bar{u}_z = 12.0\,\mathrm{mm}$ **(d)** $\bar{u}_z = 18.0\,\mathrm{mm}$

Figure 6.23: Contour plots of the von Mises stress σ_{vM} for different values of the prescribed displacement $\bar{u}_z$.

7 Summary and outlook

In the present thesis, the *finite cell method*, which is a powerful simulation tool for considering problems with complex geometries, was successfully extended by novel approaches in order to improve its efficiency and robustness, in particular for nonlinear applications. To this end, we focused on three essential difficulties which are typically inherent to non-standard discretization methods based on the fictitious domain approach. In the following, these three important problematic aspects are briefly summarized.

- In the first topic this thesis, we discussed the **numerical integration of finite cells**. In the FCM, it is common to apply adaptive Gaussian schemes – which results in a large number of quadrature points, thus leading to computationally expensive numerical integration processes. To perform the numerical integration more efficiently, we proposed novel quadrature methods based on the **moment fitting approach**. In our first investigations, we thus studied a moment fitting version in which the positions of the points are defined a priori. In doing so, the nonlinear moment fitting equations turn into linear ones, thus simplifying the solution process considerably. Following this approach, we studied two distinct point distribution schemes. In the first scheme, the points are distributed within the physical domain of cut finite cells. To this end, we developed an adaptive point distribution scheme based on a uniform subdivision. Applying a random scheme, the points are then distributed in those subcells that are completely located within the physical domain. In the second distribution scheme, we chose the position of the standard Gauss-Legendre points. Moreover, we developed another moment fitting version based on an optimization procedure in order to solve the nonlinear moment fitting equation system. Comparing all moment fitting versions with each other in terms of accuracy as well as the conditioning of the resulting quadrature rules, the moment fitting method using the Gauss-Legendre points turned out to be the most promising approach. Considering the performance of the different moment fitting version regarding several applications of the FCM revealed the same findings. Further, comparing the efficiency of the moment fitting methods concerning the adaptive Gaussian scheme, which is commonly used within the context of the FCM, we were able to show that the number of quadrature points could be reduced significantly. However, for nonlinear problems of the FCM, the moment fitting quadratures proved to be less robust in cases where cut finite cells with a small physical domain appeared. Due to this reason, we proposed an adaptive moment fitting method. In this method, cut cells are subdivided using a spacetree if the volume fraction of the physical domain is below a predefined threshold. Consequently, the moment fitting is employed on cell or subcell level. In doing so, we demonstrated that the adaptive moment fitting results in more efficient quadrature rules than the adaptive Gaussian scheme, but achieving the same accuracy and robustness. Further, in order to reduce the effort in the generation of the moment fitting quadratures, we introduced an efficient approach based on Lagrange polynomials and standard Gauss-Legendre points. Following this approach, the Kronecker delta property enables to compute the weights immediately, thus avoiding the necessity of having to solve the moment fitting equation system which is, usually, the most expensive part. Furthermore, we discussed the treatment of the material behavior for moment fitting points that are located within the fictitious domain regarding FCM applications in plasticity. Here, it turned out that using the same nonlinear material model and material parameters as utilized for the physical points leads to more accurate results.

- The second topic of this thesis was the **ill-conditioning of the global system matrix** – a well-known problem which is inherent to fictitious domain methods. Within the context of the FCM, in general, an approach based on a fictitious material model is applied in order to improve the conditioning behavior. To further improve the conditioning behavior of the global system matrix, we proposed a novel **basis function removal** strategy applied to the hierarchical shape functions, which can be easily combined with the fictitious material approach. The essential idea of this strategy is to remove shape functions with a small support to the solution from the ansatz. To this end, we categorized the global shape functions into affected and nonaffected modes. In doing so, modes that are related to cut finite cells are labeled as affected ones. Further, the remaining modes were assigned as nonaffected modes. Furthermore, shape functions related to the rigid body modes were also labeled as nonaffected. In doing so, we ensure that the modified basis maintains the representation of the rigid body modes, which is an important issue, especially considering applications in finite strain problems. Thanks to the structure of the hierarchical shape functions, this can be easily assured by preserving the linear nodal modes. Next, in order to decide which of the affected modes have to be removed from the ansatz, we proposed a global criterion estimating the contribution of the shape functions regarding the diagonal entries of global system matrix. In doing so, the introduced criterion defines a nonnegative measurement that is based on the discrete gradient operator and, thus, is independent in terms of the applied material model. Then, basis functions with a value smaller than a specified threshold of the criterion were removed from the ansatz. The performance of the proposed basis function removal strategy was studied in detail considering linear and nonlinear benchmarks in structural mechanics. Here, it was proven that the conditioning behavior of the global stiffness matrix could be improved significantly while still ensuring highly accurate results. Moreover, considering several finite strain applications in elasticity as well as in plasticity, we could show that the presented basis function removal improves the robustness of the FCM for analyses including large strains.

- In the third and last topic of this thesis, we discussed the issue of **severely distorted finite cells for applications in finite strain**, which is caused by large deformations of the fictitious domain. In order to overcome this problem and, thus, to further improve the robustness of the FCM considering analyses in large strains, we developed a new **remeshing strategy**. To this end, we proposed a remeshing approach within the framework of a total Lagrangian formulation that is based on a multiplicative decomposition of the deformation gradient. The essential idea of the presented strategy is that to create a new mesh whenever an analysis fails because of strong distortions – and then to continue the simulation. Further, the mesh generation could be carried out easily thanks to the fictitious domain approach. To this end, the deformed shape of the body is taken into account by employing a triangulated surface as a post-grid mesh. In doing so, whenever the analysis fails, the bounding box of the triangulated post-grid mesh is spatially discretized by utilizing a structured mesh. Furthermore, the numerical integration is carried out using the same surface triangulation. Moreover, for the data transfer from the old mesh to the new one, we introduced a local radial basis function interpolation scheme. The performance of the presented remeshing strategy was demonstrated by considering several finite strain problems in elasticity. Here, it could be shown that the remeshing approach allows to improve the robustness of the FCM significantly, especially in combination with the presented basis function removal.

Finally, let us close the thesis by giving an outlook into further research topics regarding the finite cell method.

- In this thesis, we demonstrated that the presented moment fitting quadratures improves the **numerical integration of finite cells** significantly in terms of efficiency. Further, it was shown that the moment fitting approach based on Gauss-Legendre points performs robust for linear elastic applications of the FCM. For nonlinear problems, however, an adaptive moment fitting scheme is required in order to provide the same robustness behavior as the adaptive Gaussian scheme. The decrease in the robustness behavior of the moment fitting is attributed to the negative weights. Consequently, to further improve the numerical integration process within the FCM, future works should focus on moment fitting quadratures that ensure nonnegative weights. From the author's point of view, a promising approach could be oriented closely to the moment fitting version based on Gauss-Legendre points and Lagrange shape functions. Assuming one can construct a basis composed of nonnegative functions that ensure the Kronecker delta property, this basis would then result into nonnegative weights. The simplest implementation to construct such a basis would be to take the absolute value of the Lagrange polynomials. However, this approach results in a basis with bad interpolation properties and, thus, in quadrature rules of low accuracy. Another approach could be oriented to the essential idea of the standard Gaussian approach which is based on a nonnegative weight function in order to ensure an orthogonality property of the applied basis functions within the domain of interest. For simple shapes, the implementation of this approach is easy. In the context of cut finite cells, this would mean that one has to construct a nonnegative weight function that ensures an orthogonality property of the basis functions with respect to the physical domain of the cut cell. The constructed weight function can then be incorporated in a similar way as suggested in the approach presented in [122]. A further promising research issue in order to provide nonnegative weights could be to develop of moment fitting approaches based on point distribution schemes in combination with nonnegative least squares. In doing so, different distribution schemes can be developed to distribute the points within the physical as well as the fictitious domain. Moreover, another interesting research topic regarding the numerical integration could be oriented towards the approach based on equivalent Legendre polynomials [113]. Here, one could think about the construction of a nonnegative and smooth function for the replacement of the discontinuity. In doing so, a modification of the standard Gaussian weights would result to nonnegative modified weights.

- The basis function removal approach proposed in this thesis allows to improve the **ill-conditioning of the resulting global system matrix** significantly. Thereby, we introduced a material-independent criterion based on the discrete gradient operator that considers the contribution of the individual shape functions regarding the diagonal entries of the global stiffness matrix. Consequently, future works could focus on the development of new criteria which may also account for the constitutive models. Moreover, we combined the basis function removal together with the fictitious material approach to further improve the robustness of the FCM for nonlinear applications. In order to further improve the conditioning behavior and thus the robustness of the FCM regarding nonlinear problems, one could focus on combining the basis function removal and the fictitious material method with additional approaches, e.g. the ghost-penalty method [137] or a technique based on a singular value decomposition of the local system matrices [146]. Moreover, one could study the effect of preconditioning techniques as suggested in [69] in terms of the robustness behavior of the FCM considering nonlinear applications.

- Moreover, we proposed a promising remeshing strategy in order to overcome the problem of **severely distorted finite cells for applications in finite strain**. To this end, several examples assuming a hyperelastic material behavior were studied – showing that the presented strategy leads to a significant improvement in the robustness of the FCM for simulations involving large deformations. Consequently, future works could focus on the application of the presented remeshing strategy to account for finite strain plasticity models. Further, new interpolations schemes and criteria could be developed and studied in more detail.

Bibliography

[1] Abaqus. `https://www.3ds.com/`. visited on: 05/01/2020.

[2] ANSYS. `https://www.ansys.com/`. visited on: 05/01/2020.

[3] LS-DYNA. `https://www.dynamore.de/`. visited on: 05/01/2020.

[4] MSC-Software. `https://www.mscsoftware.com/`. visited on: 05/01/2020.

[5] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric analysis: Towards Integration of CAD and FEM*. John Wiley & Sons, 2009.

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

[7] V. K. Saul'ev. A method for automatization of the solution of boundary value problems on high performance computers. *Dokl. Akad. Nauk SSSR 144 (1962), 497-500 (in Russian). English translation in Soviet Math. Dokl.*, 3:763–766, 1963.

[8] V. K. Saul'ev. On solution of some boundary value problems on high performance computers by fictitious domain method. *Siberian Mathematical Journal*, 4:912–925, 1963.

[9] P. Neittaanmäki and D. Tiba. An embedding of domains approach in free boundary problems and optimal design. *SIAM Journal on Control and Optimization*, 33(5):1587–1602, 1995.

[10] C. Peskin. The Immersed Boundary Method. *Acta Numerica*, 11:1–39, 2002.

[11] S. Del Pino and O. Pironneau. A fictitious domain based general pde solver. In P. Neittanmaki Y. Kuznetsov and O. Pironneau, editors, *Numerical methods for scientific computing variational problems and applications*, CIMNE, Barcelona, Spain, 2003.

[12] R. Mittal and G. Iaccarino. Immersed Boundary Method. *Annual Review Fluid Mechanics*, 37:239–260, 2005.

[13] R. Glowinski and Y. Kuznetsov. Distributed Lagrange multipliers based on fictitious domain method for second order elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 196:1498–1506, 2007.

[14] I. Ramière, P. Angot, and M. Belliard. A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 196:766–781, 2007.

[15] I. Ramière, P. Angot, and M. Belliard. A general fictitious domain method with immersed jumps and multilevel nested structured meshes. *Journal of Computational Physics*, 225:1347–1387, 2007.

[16] E. Burman and P. Hansbo. Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method. *Computer Methods in Applied Mechanics and Engineering*, 199(41-44):2680–2686, 2010.

[17] E. Burman and P. Hansbo. Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Applied Numerical Mathematics*, 62(4):328–341, 2012.

[18] D. Elfverson, M. G. Larson, and K. Larsson. CutIGA with basis function removal. *Advanced Modeling and Simulation in Engineering Sciences*, 5:2213–7467, 2018.

[19] J. Parvizian, A. Düster, and E. Rank. Finite cell method – h- and p-extension for embedded domain problems in solid mechanics. *Computational Mechanics*, 41:121–133, 2007.

[20] A. Düster, J. Parvizian, Z. Yang, and E. Rank. The finite cell method for three-dimensional problems of solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 197:3768–3782, 2008.

[21] M. Dauge, A. Düster, and E. Rank. Theoretical and numerical investigation of the finite cell method. *Journal of Scientific Computing*, 65:1039–1064, 2015.

[22] D. Schillinger and M. Ruess. The Finite Cell Method: A Review in the Context of Higher-Order Structural Analysis of CAD and Image-Based Geometric Models. *Archives Of Computational Methods In Engineering*, 22:391–455, 2015.

[23] D. Schillinger, A. Düster, and E. Rank. The *hp*-*d*-adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering*, 89:1171–1202, 2012.

[24] D. Schillinger, M. Ruess, A. Düster, and E. Rank. The Finite Cell Method for large deformation analysis. *Proceedings in Applied Mathematics and Mechanics*, 11:271–272, 2011.

[25] D. Schillinger, M. Ruess, N. Zander, Y. Bazilevs, A. Düster, and E. Rank. Small and large deformation analysis with the p- and B-spline versions of the finite cell method. *Computational Mechanics*, 50:445–478, 2012.

[26] D. Schillinger, A. Düster, and E. Rank. The *hp*-*d*-adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering*, 89:1171–1202, 2012.

[27] D. Schillinger. *The p- and B-spline versions of the geometrically nonlinear finite cell method and hierarchical refinement strategies for adaptive isogeometric and embedded domain analysis.* Dissertation, Chair for Computation in Engineering, TU-München, 2012.

[28] A. Abedian, J. Parvizian, A. Düster, and E. Rank. The finite cell method for the J_2 flow theory of plasticity. *Finite Elements in Analysis and Design*, 69:37–47, 2013.

[29] A. Abedian, J. Parvizian, A. Düster, and E. Rank. Finite cell method compared to h-version finite element method for elasto-plastic problems. *Applied Mathematics and Mechanics*, 35(10):1239–1248, 2014.

[30] A. Taghipour, J. Parvizian, S. Heinze, and A. Düster. The finite cell method for nearly incompressible finite strain plasticity problems with complex geometries. *Computers & Mathematics with Applications*, 75:3298–3316, 2018.

[31] S. Hubrich and A. Düster. Adaptive numerical integration of broken finite cells based on moment fitting applied to finite strain problems. *Proceedings in Applied Mathematics and Mechanics*, 18:e201800089, 2018.

[32] S. Hubrich and A. Düster. Numerical integration for nonlinear problems of the finite cell method using an adaptive scheme based on moment fitting. *Computers & Mathematics with Applications*, https://doi.org/10.1016/j.camwa.2018.11.030, 2018.

[33] S. Hubrich and A. Düster. Numerical integration for nonlinear problems of the finite cell method using an adaptive scheme based on moment fitting. *Computers & Mathematics with Applications*, 77:1983–1997, 2019.

[34] A. Düster, H.-G. Sehlhorst, and E. Rank. Numerical homogenization of heterogeneous and cellular materials. *Computational Mechanics*, 50:413–431, 2012.

[35] S. Heinze, M. Joulaian, H. Egger, and A. Düster. Efficient computation of cellular materials using the finite cell method. *Proceedings in Applied Mathematics and Mechanics*, 14:251–252, 2014.

[36] S. Heinze, M. Joulaian, and A. Düster. Numerical homogenization of hybrid metal foams using the finite cell method. *Computers & Mathematics with Applications*, 70:1501–1517, 2015.

[37] S. Heinze, Z. Chen, A. Jung, S. Diebels, and A. Düster. Numerical analysis of Ni/Al hybrid metal foams using the finite cell method. *Proceedings in Applied Mathematics and Mechanics*, 15:299–300, 2015.

[38] S. Heinze, A. Jung, S. Diebels, and A. Düster. Experimental and numerical investigation of metal foams undergoing large deformations. *Proceedings in Applied Mathematics and Mechanics*, 16:345–346, 2016.

[39] S. Heinze, T. Bleistein, A. Düster, S. Diebels, and A. Jung. Experimental and numerical investigation of single pores for identification of effective metal foams properties. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 98:682–695, 2018.

[40] S. Gnegel. *The finite cell method for the computation of cellular materials.* PhD thesis, Fachgebiet für Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik (M-10), TU Hamburg, 2019.

[41] J. Parvizian, A. Düster, and E. Rank. Topology optimization using the finite cell method. *Optimization and Engineering*, pages 1–22, 2011. 10.1007/s11081-011-9159-x.

[42] S. Cai, W. Zhang, J. Zhu, and T. Gao. Stress constrained shape and topology optimization with fixed mesh: A B-spline finite cell method combined with level set function. *Computer Methods in Applied Mechanics and Engineering*, 278:361–387, 2014.

[43] M. Joulaian and A. Düster. Local enrichment of the finite cell method for problems with material interfaces. *Computational Mechanics*, 52:741–762, 2013.

[44] M. Joulaian and A. Düster. The hp-d version of the finite cell method with local enrichment for multiscale problems. In *Proceedings in Applied Mathematics and Mechanics*, volume 13, pages 259–260, 2013.

[45] M. Joulaian, N. Zander, T. Bog, S. Kollmannsberger, E. Rank, and A. Düster. A high-order enrichment strategy for the finite cell method. *Proceedings in Applied Mathematics and Mechanics*, 15:207–208, 2015.

[46] M. Joulaian. *The hierarchical finite cell method for problems in structural mechanics.* PhD thesis, Fachgebiet für Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik, TU Hamburg-Harburg, 2017.

[47] M. Elhaddad, N. Zander, T. Bog, L. Kudela, S. Kollmannsberger, J. Kirschke, T. Baum, M. Ruess, and E. Rank. Multi-level hp-finite cell method for embedded interface problems with application in biomechanics. *International Journal for Numerical Methods in Biomedical Engineering*, 34(4):e2951, 2018.

[48] Tino Bog. The finite cell method for contact problems in solid mechanics. In *6th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS)*, 2012.

[49] T. Bog, N. Zander, S. Kollmannsberger, and E. Rank. The Finite Cell Method for Contact Problems in Solid Mechanics. In *3rd International Conference on Computational Contact Mechanics (ICCCM 2013)*, Lecce, Italy, Juli 2013.

[50] Tino Bog, Nils Zander, Stefan Kollmannsberger, and Ernst Rank. A formulation for frictionless contact using a material model and high order finite elements. In *Sixth International Workshop on High-Order Finite Element and Isogeometric Methods*, Frauenchiemsee Island, Germany, 2014.

[51] Tino Bog, Nils Zander, Stefan Kollmannsberger, and Ernst Rank. Normal contact with high order finite elements and a fictitious contact material. *Computers & Mathematics with Applications*, 70(7):1370–1390, Oktober 2015.

[52] T. Bog. *Frictionless contact simulation using the finite cell method.* PhD thesis, Technische Universität München, 2017.

[53] T. Bog, N. Zander, S. Kollmannsberger, and E. Rank. Weak imposition of frictionless contact constraints on automatically recovered high-order, embedded interfaces using the finite cell method. *Computational Mechanics*, 61(4):385–407, 2018.

[54] A. Konyukhov, C. Lorenz, and K. Schweizerhof. Various contact approaches for the finite cell method. *Computational Mechanics*, 56:331–351, 2016.

[55] N. Zander. The Finite Cell Method for Linear Thermoelasticity. Master's thesis, Technische Universität München, 2011.

[56] N. Zander, S. Kollmannsberger, M. Ruess, Z. Yosibash, and E. Rank. The Finite Cell Method for Linear Thermoelasticity. *Computers & Mathematics with Applications*, 64(11):3527–3541, 2012.

[57] N. Zander, P. Erbts, S. Kollmannsberger, A. Düster, and E. Rank. The Finite Cell Method for Transient, Non-linear Heat Conduction. In *Proceedings of the 3rd European Seminar on Computing*, 2012.

[58] N. Zander, P. Erbts, S. Kollmannsberger, A. Düster, and E. Rank. High-Order hp-FEM: High Order Mesh Adaptivity without the Difficulties of Hanging Nodes. In *11th. World Congress on Computational Mechanics (WCCM XI), Barcelona, Spain*, Barcelona, Spain, 2014.

[59] N. Zander, T. Bog, S. Kollmannsberger, D. Schillinger, and E. Rank. Multi-Level hp-Adaptivity: High-Order mesh adaptivity without the difficulties of constraining hanging nodes. *Computational Mechanics*, 55(3):499–517, 2015.

[60] N. D. Zander. *Multi-level hp-FEM: dynamically changing high-order mesh refinement with arbitrary hanging nodes*. PhD thesis, Technische Universität München, 2017.

[61] S. Kollmannsberger, A. Özcan, M. Carraturo, N. Zander, and E. Rank. A hierarchical computational model for moving thermal loads and phase changes with applications to selective laser melting. *Computers & Mathematics with Applications*, 75(5):1483–1497, 2018.

[62] A. Özcan, S. Kollmannsberger, J. Jomo, and E. Rank. Residual stresses in metal deposition modeling: Discretizations of higher order. *Computers & Mathematics with Applications*, 78(7):2247–2266, 2019.

[63] N. Zander, M. Ruess, T. Bog, S. Kollmannsberger, and E. Rank. Multi-level hp-adaptivity for cohesive fracture modeling. *International Journal for Numerical Methods in Engineering*, 2016.

[64] S. Nagaraja, M. Elhaddad, M. Ambati, S. Kollmannsberger, L. De Lorenzis, and E. Rank. Phase-field modeling of brittle fracture with multi-level *hp*-FEM and the finite cell method. *Computational Mechanics*, 63(6):1283–1300, 2019.

[65] S. Duczek, M. Joulaian, A. Düster, and U. Gabbert. Numerical analysis of Lamb waves using the finite and spectral cell method. *International Journal for Numerical Methods in Engineering*, 99:26–53, 2014.

[66] M. Joulaian, S. Duczek, U. Gabbert, and A. Düster. Finite and spectral cell method for wave propagation in heterogeneous materials. *Computational Mechanics*, 54:661–675, 2014.

[67] M. Joulaian, S. Duczek, U. Gabbert, and A. Düster. Efficient simulation of wave propagation in heterogeneous materials. *Proceedings in Applied Mathematics and Mechanics*, 14:715–716, 2014.

[68] F. Mossaiby, M. Joulaian, and A. Düster. The spectral cell method for wave propagation in heterogeneous materials simulated on multiple GPUs and CPUs. *Computational Mechanics*, 63:805–819, 2019.

[69] F. de Prenter, C. V. Verhoosel, G. J. van Zwieten, and E. H. van Brummelen. Condition number analysis and preconditioning of the finite cell method. *Computer Methods in Applied Mechanics and Engineering*, 2017.

[70] N. Zander, T. Bog, M. Elhaddad, F. Frischmann, S. Kollmannsberger, and E. Rank. The multi-level hp-method for three-dimensional problems: Dynamically changing high-order mesh refinement with arbitrary hanging nodes. *Computer Methods in Applied Mechanics and Engineering*, 310:252–277, 2016.

[71] I. Babuška. The finite element method with penalty. *Mathematics of Computation*, 27:221–228, 1973.

[72] M. Ruess, D. Schillinger, Y. Bazilevs, V. Varduhn, and E. Rank. Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method. *International Journal for Numerical Methods in Engineering*, 95(10):811–846, 2013.

[73] S. Kollmannsberger, A. Özcan, J. Baiges, M. Ruess, E. Rank, and A. Reali. Parameter-free, weak imposition of Dirichlet boundary conditions and coupling of trimmed and non-conforming patches. *International Journal for Numerical Methods in Engineering*, 101(9):1–30, 2014.

[74] P. Wriggers. *Nonlinear Finite-Element-Methods*. Springer-Verlag, 2009.

[75] T. Belytschko, W.K. Liu, and B. Moran. *Nonlinear finite elements for continua and structures*. John Wiley & Sons, 2000.

[76] J. C. Simo and T. J. R. Hughes. *Computational Inelasticity*. Springer-Verlag, 1998.

[77] J. Bonet and R.D. Wood. *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press, New York, 2008.

[78] E. A. de Souza Neto, D. Perić, and D. R. J. Owen. *Computational Methods for Plasticity, Theory and Applications*. John Wiley & Sons, 2008.

[79] P. G. Ciarlet. *Mathematical Elasticity*, volume 1. Elsevier, 1988.

[80] E. Stein, editor. *Error-Controlled Adaptive Finite Elements in Solid Mechanics*. John Wiley & Sons, 2002.

[81] P. Wriggers and B. Hudobivnik. A low order virtual element formulation for finite elasto-plastic deformations. *Computer Methods in Applied Mechanics and Engineering*, 327:459–477, 2017.

[82] J. Korelc and S. Stupkiewicz. Closed-form matrix exponential and its application in finite-strain plasticity. *International Journal for Numerical Methods in Engineering*, 98(13):960–987, 2014.

[83] J. Korelc and P. Wriggers. *Automation of Finite Element Methods*. Springer, 2016.

[84] J.C. Simo and C. Miehe. Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation. *Computer Methods in Applied Mechanics and Engineering*, 98(1):41–104, 1992.

[85] E. Burman, S. Claus, P. Hansbo, M. G. Larson, and A. Massing. CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104:472–501, 2015.

[86] B.A. Szabó and I. Babuška. *Finite Element Analysis*. John Wiley & Sons, 1991.

[87] B.A. Szabó, A. Düster, and E. Rank. The p-version of the Finite Element Method. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, volume 1, chapter 5, pages 119–139. John Wiley & Sons, 2004.

[88] H.R. Schwarz and N. Köckler. *Numerische Mathematik*. Vieweg+Teubner Verlag, 8. edition, 2011.

[89] R. Schaback and H. Wendland. *Numerische Mathematik*. Springer, 5. edition, 2005.

[90] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45:601–620, 1999.

[91] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 64:131–150, 1999.

[92] C. A. Duarte, I. Babuška, and J. T. Oden. Generalized finite element method for three-dimensional structural mechanics problems. *Computers & Structures*, 77(2):215–232, 2000.

[93] T. Strouboulis, K. Copps, and I. Babuška. The generalized finite element method: an example of its implementation and illustration of its performance. *International Journal for Numerical Methods in Engineering*, 47:1401–1417, 2000.

[94] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253–304, 2010.

[95] B. Müller, F. Kummer, M. Oberlack, and Y. Wang. Simple multidimensional integration of discontinuous functions with application to level set methods. *International Journal for Numerical Methods in Engineering*, 92:637–651, 2012.

[96] A. Abedian, J. Parvizian, A. Düster, H. Khademyzadeh, and E. Rank. Performance of different integration schemes in facing discontinuities in the finite cell method. *International Journal of Computational Methods*, 10(3):1350002/1–24, 2013.

[97] A. Abedian and A. Düster. An extension of the finite cell method using boolean operations. *Computational Mechanics*, 59:877–886, 2017.

[98] N. Sukumar, N. Moës, B. Moran, and T. Belytschko. Extended finite element method for three-dimensional crack modelling. *International Journal for Numerical Methods in Engineering*, 48:1549–1570, 2000.

[99] N. Sukumar, D.L. Chopp, N. Moës, and T. Belytschko. Modeling holes and inclusions by level sets in the extended finite-element method. *Computer Methods in Applied Mechanics and Engineering*, 190:6183–6200, 2001.

[100] N. Moës, M. Cloirec, P. Cartraud, and J.-F. Remacle. A computational approach to handle complex microstructure geometries. *Computer Methods in Applied Mechanics and Engineering*, 192:3163–3177, 2003.

[101] S. Loehnert, D. S. Mueller-Hoeppe, and P. Wriggers. 3D corrected XFEM approach and extension to finite deformation theory. *International Journal for Numerical Methods in Engineering*, 86:431–452, 2011.

[102] T. Strouboulis, K. Copps, and I. Babuška. The generalized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 190:4081–4193, 2001.

[103] K.W. Cheng and T.-P. Fries. Higher-order XFEM for curved strong and weak discontinuities. *International Journal for Numerical Methods in Engineering*, 82:564–590, 2009.

[104] L. Kudela, N. Zander, T. Bog, S. Kollmannsberger, and E. Rank. Efficient and Accurate Numerical Integration for High Order Immersed Boundary Methods. In *Proceedings of Isogeometric Analysis: Integrating Design and Analysi*, Austin, Texas, 2014.

[105] L. Kudela, N. Zander, T. Bog, S. Kollmannsberger, and E. Rank. Efficient and accurate numerical quadrature for immersed boundary methods. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):1–22, Juni 2015.

[106] L. Kudela, N. Zander, S. Kollmannsberger, and E. Rank. Smart octrees: Accurately integrating discontinuous functions in 3D. *Computer Methods in Applied Mechanics and Engineering*, 306:406–426, Juli 2016.

[107] K. Höllig. *Finite Element Methods with B-Splines*. Frontiers in Applied Mathematics. SIAM Society for Industrial and Applied Mathematics, 2003.

[108] K. Höllig and J. Hörner. Programming finite element methods with weighted B-splines. *Computers & Mathematics with Applications*, 70(7):1441–1456, 2015.

[109] T.-P. Fries and S. Omerović. Higher-order accurate integration of implicit geometries. *International Journal for Numerical Methods in Engineering*, 106(5):323–371, 2016.

[110] T.-P. Fries, S. Omerović, D. Schöllhammer, and J. Steidl. Higher-order meshing of implicit geometries–Part I: Integration and interpolation in cut elements. *Computer Methods in Applied Mechanics and Engineering*, 313:759–784, 2017.

[111] G. Ventura. On the elimination of quadrature subcells for discontinuous functions in the eXtended Finite-Element Method. *International Journal for Numerical Methods in Engineering*, 66:761–795, 2006.

[112] G. Ventura and E. Benvenuti. Equivalent polynomials for quadrature in Heaviside function enrichment elements. *International Journal for Numerical Methods in Engineering*, 102:688–710, 2015.

[113] A. Abedian and A. Düster. Equivalent Legendre polynomials: Numerical integration of discontinuous functions in the finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 343:690–720, 2019.

[114] J. N. Lyness and D. Jespersen. Moderate degree symmetric quadrature rules for the triangle. *Journal of the Institute of Mathematics and Its Applications*, 15:19–32, 1975.

[115] J. N. Lyness and G. Monegato. Quadrature rules for regions having regular hexagonal symmetry. *SIAM Journal on Numerical Analysis*, 14:283–295, 1977.

[116] D.A. Dunavant. Economical symmetrical quadrature rules for complete polynomials over a square domain. *International Journal for Numerical Methods in Engineering*, 21:1777–1784, 1985.

[117] D.A. Dunavant. High degree efficient symmetrical Gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, 21:1129–1148, 1985.

[118] S. E. Mousavi and N. Sukumar. Generalized Gaussian quadrature rules for discontinuities and crack singularities in the extended finite element method. *Computer Methods in Applied Mechanics and Engineering*, 199(49–52):3237–3249, 2010.

[119] S. E. Mousavi, H. Xiao, and N. Sukumar. Generalized Gaussian quadrature rules on arbitrary polygons. *International Journal for Numerical Methods in Engineering*, 82(1):99–113, 2010.

[120] H. Xiao and Z. Gimbutas. A numerical algorithm for the construction of efficient quadrature rules in two and higher dimensions. *Computers & Mathematics with Applications*, 59(2):663 – 676, 2010.

[121] S. E. Mousavi and N. Sukumar. Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. *Computational Mechanics*, 47:535–554, 2011.

[122] A. Düster and O. Allix. Selective enrichment of moment fitting and application to cut finite elements and cells. *Computational Mechanics*, 2019.

[123] Y. Sudhakar and W. A. Wall. Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods. *Computer Methods in Applied Mechanics and Engineering*, 258:39–54, 2013.

[124] B. Müller, F. Kummer, and M. Oberlack. Highly accurate surface and volume integration on implicit domains by means of moment-fitting. *International Journal for Numerical Methods in Engineering*, 96:512–528, 2013.

[125] V. Thiagarajan and V. Shapiro. Adaptively weighted numerical integration over arbitrary domains. *Computers & Mathematics with Applications*, 67(9):1682–1702, 2014.

[126] S. Hubrich, M. Joulaian, and A. Düster. Numerical integration in the finite cell method based on moment-fitting. In *Proceedings of 3rd ECCOMAS Young Investigators Conference, 6th GACM Colloquium*, pages 1–4, Aachen, Germany, 2015.

[127] M. Joulaian, S. Hubrich, and A. Düster. Numerical integration of discontinuities on arbitrary domains based on moment fitting. *Computational Mechanics*, 57:979–999, 2016.

[128] S. Hubrich, M. Joulaian, P. Di Stolfo, A. Schröder, and A. Düster. Efficient numerical integration of arbitrarily broken cells using the moment fitting approach. *Proceedings in Applied Mathematics and Mechanics*, 16:201–202, 2016.

[129] S. Hubrich, P. Di Stolfo, L. Kudela, S. Kollmannsberger, E. Rank, A. Schröder, and A. Düster. Numerical integration of discontinuous functions: moment fitting and smart octree. *Computational Mechanics*, 60:863–881, 2017.

[130] V. Thiagarajan and V. Shapiro. Adaptively Weighted Numerical Integration in the Finite Cell Method. *Computers & Mathematics with Applications*, 311:250–279, 2016.

[131] LAPACK – Linear Algebra PACKage. `http://www.netlib.org/lapack/`. visited on: 12/11/2019.

[132] The MathWorks, Inc MATLAB. `https://de.mathworks.com/`. visited on: 12/11/2019.

[133] Wolfram Mathematica. `https://www.wolfram.com/mathematica/`. visited on: 12/11/2019.

[134] A.R. Krommer and C.W. Ueberhuber. *Numerical Integration on Advanced Computer Systems.* Lecture Notes in Computer Science. Springer-Verlag, Berlin, Heidelberg, 1994.

[135] H. Bröker. *Integration von geometrischer Modellierung und Berechnung nach der p-Version der FEM.* PhD thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München, 2001.

[136] A. Düster, H.-G. Sehlhorst, and E. Rank. Numerical homogenization of heterogeneous and cellular materials utilizing the finite cell method. *Computational Mechanics*, 50:413–431, 2012.

[137] E. Burman. Ghost penalty. *Comptes Rendus Mathematique*, 348(21):1217–1220, 2010.

[138] E. Béchet, H. Minnebo, N. Moës, and B. Burgardt. Improved implementation and robustness study of the X-FEM for stress analysis around cracks. *International Journal for Numerical Methods in Engineering*, 64:1033–1056, 2005.

[139] A. Menk and S. P. A. Bordas. A robust preconditioning technique for the extended finite element method. *International Journal for Numerical Methods in Engineering*, 85(13):1609–1632, 2011.

[140] C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32(6):1205–1227, 1991.

[141] C. Lehrenfeld and A. Reusken. Optimal preconditioners for Nitsche-XFEM discretizations of interface problems. *Numerische Mathematik*, 135(2):313–332, 2017.

[142] J. Xu. Iterative Methods by Space Decomposition and Subspace Correction. *SIAM Review*, 34(4):581–613, 1992.

[143] I. Babuška and U. Banerjee. Stable generalized finite element method (SGFEM). *Computer Methods in Applied Mechanics and Engineering*, 201:91–111, 2012.

[144] V. Gupta, C.A. Duarte, I. Babuška, and U. Banerjee. A stable and optimally convergent generalized FEM (SGFEM) for linear elastic fracture mechanics. *Computer Methods in Applied Mechanics and Engineering*, 266:23–39, 2013.

[145] V. Gupta, C.A. Duarte, I. Babuška, and U. Banerjee. Stable gfem (sgfem): Improved conditioning and accuracy of gfem/xfem for three-dimensional fracture mechanics. *Computer Methods in Applied Mechanics and Engineering*, 289:355–386, 2015.

[146] S. Loehnert. A stabilization technique for the regularization of nearly singular extended finite elements. *Computational Mechanics*, 54:523–533, 2014.

[147] K. Höllig, U. Reif, and J. Wipper. Weighted Extended B-Spline Approximation of Dirichlet Problems. *SIAM Journal on Numerical Analysis*, 39(2):442–462, 2001.

[148] K. Höllig, C. Apprich, and A. Streit. Introduction to the Web-method and its applications. *Advances in Computational Mechanics*, 23(1):215–237, 2005.

[149] R.a.K. Sanches, P.B. Bornemann, and F. Cirak. Immersed b-spline (i-spline) finite element method for geometrically complex domains. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1432–1445, 2011.

[150] A. Johansson and M. G. Larson. A high order discontinuous galerkin nitsche method for elliptic problems with fictitious boundary. *Numerische Mathematik*, 123:607–628, 2013.

[151] S. Badia, F. Verdugo, and A. F. Martín. The aggregated unfitted finite element method for elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 336:533–553, 2018.

[152] A. Reusken. Analysis of an extended pressure finite element space for two-phase incompressible flows. *Computing and Visualization in Science*, 11:293–305, 2008.

[153] A. Embar, J. Dolbow, and I. Harari. Imposing Dirichlet boundary conditions with Nitsche's method and spline-based finite elements. *International Journal for Numerical Methods in Engineering*, 83(7):877–898, 2010.

[154] H.-G. Sehlhorst. *Numerical homogenization strategies for cellular materials with applications in structural mechanics*. PhD thesis, Fachgebiet für Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik, TU Hamburg-Harburg, 2011.

[155] C. V. Verhoosel, G. J. van Zwieten, B. Rietbergen, and R. de Borst. Image-based goal-oriented adaptive isogeometric analysis with application to the micro-mechanical modeling of trabecular bone. *Computer Methods in Applied Mechanics and Engineering*, 284:138–164, 2015.

[156] W.J. Gordon and Ch.A. Hall. Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, 7:461–477, 1973.

[157] W.J. Gordon and Ch.A. Hall. Transfinite element methods: Blending function interpolation over arbitrary curved element domains. *Numerische Mathematik*, 21:109–129, 1973.

[158] A. M. Habraken and S. Cescotto. An automatic remeshing technique for finite element simulation of forming processes. *International Journal for Numerical Methods in Engineering*, 30(8):1503–1525, 1990.

[159] N. S. Lee and K. J. Bathe. Error indicators and adaptive remeshing in large deformation finite element analysis. *Finite Elements in Analysis and Design*, 16:99–139, 1994.

[160] J. Mediavilla, R. H.J. Peerlings, and M. G.D. Geers. A robust and consistent remeshing-transfer operator for ductile fracture simulations. *Computers & Structures*, 84:604–623, 2006.

[161] W. Garhuom, S. Hubrich, L. Radtke, and A. Düster. A remeshing strategy for large deformations in the finite cell method. *Computers & Mathematics with Applications*, 80:2379–2398, 2020.

[162] W. Kwok and Z. Chen. A Simple and Effective Mesh Quality Metric for Hexahedral and Wedge Elements. In *Proceedings of the 9th International Meshing Roundtable, IMR*, pages 325–333, New Orleans, Louisiana, USA, 2000.

[163] W. Lowrie, V. S. Lukin, and U. Shumlak. A priori mesh quality metric error analysis applied to a high-order finite element method. *Journal of Computational Physics*, 230:5564–5586, 2011.

[164] M. Bucki, C. Lobos, Y. Payan, and N. Hitschfeld. Jacobian-based repair method for finite element meshes. *Engineering Computations*, 27:285–297, 2011.

[165] Ch. Sorger. *Generierung von Netzen für Finite Elemente hoher Ordnung in zwei und drei Raumdimensionen.* Dissertation, Chair for Computation in Engineering, TU-München, 2012.

[166] A. Düster, E. Rank, and B. Szabó. The p-Version of the Finite Element and Finite Cell Methods. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics Second Edition*, volume Part 1. Solids and Structures, chapter 4, pages 137–171. John Wiley & Sons, 2017.

[167] M. D. Buhmann. *Radial Basis Functions: Theory and Implementations*, volume 12 of *Cambridge monographs on applied and computational mathematics.* Cambridge University Press, 2009.

[168] A. de Boer, A. H. van Zuijlen, and H. Bijl. Radial basis functions for interface interpolation and mesh deformation. In Barry Koren and Kees Vuik, editors, *Advanced Computational Methods in Science and Engineering*, pages 143–178, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[169] M. König. *Partitioned Solution Strategies for Strongly-Coupled Fluid-Structure Interaction Problems in Maritime Applications.* PhD thesis, Fachgebiet für Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik (M-10), TU Hamburg, 2018.

[170] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

Die Reihen der Fortschritt-Berichte VDI:

1 Konstruktionstechnik/Maschinenelemente
2 Fertigungstechnik
3 Verfahrenstechnik
4 Bauingenieurwesen
5 Grund- und Werkstoffe/Kunststoffe
6 Energietechnik
7 Strömungstechnik
8 Mess-, Steuerungs- und Regelungstechnik
9 Elektronik/Mikro- und Nanotechnik
10 Informatik/Kommunikation
11 Schwingungstechnik
12 Verkehrstechnik/Fahrzeugtechnik
13 Fördertechnik/Logistik
14 Landtechnik/Lebensmitteltechnik
15 Umwelttechnik
16 Technik und Wirtschaft
17 Biotechnik/Medizintechnik
18 Mechanik/Bruchmechanik
19 Wärmetechnik/Kältetechnik
20 Rechnerunterstützte Verfahren (CAD, CAM, CAE CAQ, CIM ...)
21 Elektrotechnik
22 Mensch-Maschine-Systeme
23 Technische Gebäudeausrüstung

ISBN 978-3-18-3**35518**-1