

Spline-Based Methods for Fluid-Structure Interaction

Spline-basierte Methoden für Fluid-Struktur-Interaktionen

Von der Fakultät für Maschinenwesen
der Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades eines Doktors der
Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Norbert Michael Hosters

Berichter: Univ.-Prof. Marek Behr, Ph.D.
Prof. Dr. ir. Harald van Brummelen

Tag der mündlichen Prüfung: 04.04.2018

Vorwort

Die vorliegende Arbeit entstand während meiner Zeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für computergestützte Analyse technischer Systeme der RWTH Aachen University und wurde maßgeblich beeinflusst durch unzählige Diskussionen mit allen Kolleginnen und Kollegen. Deshalb gilt mein Dank zunächst jeder und jedem Einzelnen am Institut. Die freundschaftliche Atmosphäre am Institut habe ich sehr genossen, nicht nur in der alltäglichen Zusammenarbeit, sondern insbesondere in den obligatorischen Kaffeepausen und bei allen Veranstaltungen außerhalb des Instituts. Im Folgenden möchte ich die Chance nutzen, mich bei einzelnen Personen auch persönlich zu bedanken.

Zunächst möchte ich mich bei der Institutsleitung Prof. Marek Behr und Prof. Stefanie Elgeti bedanken, insbesondere für die Freiheiten, eine eigene Forschungsidee entwickeln und mich in dieser unabhängig entfalten zu können. Ich hoffe sehr, ihrer Geduld, ihrem Vertrauen und der mir entgegengebrachten Wertschätzung gerecht geworden zu sein und durch meine Arbeit einen nachhaltigen Beitrag zur Forschung des Instituts geleistet zu haben. Des Weiteren möchte ich mich bei Prof. Harald van Brummelen für die Begutachtung der Dissertation bedanken, und zusätzlich für alle wissenschaftlichen Impulse, die er während der Erstellung dieser Arbeit geliefert hat. Bei Prof. Ballmann bedanke ich mich für jegliche Unterstützung insbesondere während meines Einstieges in den Bereich Aeroelastik.

Meinen Kollegen der Arbeitsgruppe "Fluid-Struktur Interaktion" Manuel Brüderlin, Michel Make, Max von Danwitz und Emre Ongut gilt mein Dank für jede fachliche Diskussion und für alle gemeinsamen Entwicklungen. Die Zusammenarbeit mit Euch hat mir jederzeit großen Spaß gemacht. Mein Dank gilt auch meinen Vorgängern Lars Reimer und Georg Wellmer, auf deren gewissenhafte Vorarbeiten ich immer wieder aufbauen konnte. Ein wesentlicher Teil der im Verlauf dieser Arbeit realisierten Methoden basiert auf den Ideen, die gemeinsam mit Markus Frings und Roland Siegbert, später ergänzt durch Sebastian Eusterholz und Florian Zwicke, im Softwareprojekt "SplineLib" realisiert wurden. Allen mitwirkenden Kollegen gilt deshalb besonderer Dank. Für jeglichen fachlichen Impuls, sei es im Institut oder während der "Soersrunde", im Speziellen aber in Momenten, in denen die Kompetenzen eines Ingenieurs Unterstützung benötigten, möchte

ich mich bei Philipp Knechtges bedanken. Viele seiner Ideen haben die in dieser Arbeit verwirklichten Methoden direkt oder indirekt beeinflusst.

Besonderer Dank gilt meinen früheren Masterstudenten und jetzigen Kollegen Jan Helmig, Daniel Hilger, Fabian Key und Thomas Spenke, die wesentlich zum Erfolg dieser Arbeit beigetragen haben. Für die Zukunft wünsche ich Euch, dass Ihr ebenfalls so kompetente, engagierte und inspirierende Studenten findet, welche sich auch nicht aus der Bahn werfen lassen, wenn eigentlich doch gar nichts funktioniert. Dieser Dank gilt aber selbstverständlich auch allen anderen Studenten, die mich während meiner Promotion unterstützt haben.

Ein herzliches Dankeschön gilt meinen Eltern, welche mich und meine Geschwister dazu ermutigt haben, alle Herausforderungen anzunehmen und den Glauben an die eigenen Stärken auch in schweren Zeiten nicht zu verlieren. Ohne Ihre Unterstützung hätte ich den Weg, der durch diese Arbeit abgeschlossen wird, niemals beschreiten können. Nicht zuletzt möchte ich mich bei meiner Frau Annika bedanken. Ohne ihren steten und selbstlosen Rückhalt wäre diese Arbeit niemals zustande gekommen. Insbesondere in stressigen Zeiten hat es mir sehr geholfen, eine starke Partnerin an der Seite zu haben. Jetzt freue ich mich auf unsere gemeinsame Zukunft, vor allem auf gemeinsame Reisen in denen das mobile Büro nun hoffentlich wieder seinen ursprünglichen Zweck erfüllen kann.

Aachen, im April 2018

Norbert Hosters

Abstract

Computer-aided design uses Non-Uniform Rational B-Splines (NURBS) to represent and communicate geometry. Therefore, a NURBS description of a given design can generally be considered the exact description. The development of isogeometric methods has made this spline-based geometry description available for numerical analysis methods [43]. As a result of the possible combination with plate and shell theory, isogeometric analysis is already versatile in structural analysis. In fluid dynamics, on the other hand, three-dimensional analysis is usually indispensable. Admittedly, the application of isogeometric analysis is fundamentally independent of the spatial dimensions. However, it is still very difficult to create volume splines at the moment. Alternatively, classical finite element methods can be extended to take account of the given geometry at least at the limits. An example of these methods is the NURBS-enhanced finite element method [136, 137].

The thesis presents the solution of fluid-structure interactions using the same spline description at the interface. On the structural side, the spline is used in an isogeometric formulation. On the liquid side, the same spline is used in a NURBS-enhanced finite element method. The dissertation introduces the entire solution methodology of the surface-coupled problem, based on an implicit staggered approach. Both the usual approach of Dirichlet-Neumann coupling and Robin-Neumann coupling are discussed. The use of the identical spline representation on both sides enables a consistent and conservative transfer of the required information, whereby an individual discretization on both sides is possible. Existing acceleration techniques for partitioned approaches are discussed in the context of the spline-based method and combined to a modified variant.

The numerical advantages of the presented combination of isogeometric analysis and the NURBS-enhanced finite element method are verified by simple stationary and transient test cases. A test case with a closed domain with deformable boundaries proves that the proposed combination is able to exceed the accuracy of the usual coupling of Lagrangian finite elements and isogeometric analysis. By applying the methodology in a fluid-structure interaction with contact, it is shown that known advantages of

isogeometric analysis on the structural side can be transferred to fluid-structure interaction.

The spline-based description not only offers advantages in linking the individual fields. The spline-based boundary description is also advantageous for problems of fluid-structure interaction and free surfaces. Corresponding extensions of the boundary conditions are discussed. The methodology is demonstrated on a simple three-dimensional test case.

Zusammenfassung

Zur Darstellung und Kommunikation von Geometrien werden in der computergestützte Konstruktion üblicherweise Non-Uniform Rational B-Splines (NURBS) verwendet. Der NURBS einer gegebenen Geometrie kann deshalb als exakte Beschreibung angesehen werden. Durch die Entwicklung isogeometrischer Methoden kann diese Geometriebeschreibung nun direkt in die numerische Analyse integriert werden [43]. Eine mögliche Kombination mit der Platten- und Schalentheorie macht die isogeometrische Analyse in der Strukturanalyse bereits sehr vielseitig einsetzbar. In der Fluidodynamik hingegen sind dreidimensionale Analysen meist unverzichtbar. Zwar ist die Anwendung der isogeometrischen Analyse grundsätzlich unabhängig von den räumlichen Dimensionen, die Erstellung von Volumen-Splines ist allerdings momentan noch sehr aufwendig. Alternativ können klassische Finite-Elemente-Methoden so erweitert werden, dass die vorgegebene Geometrie zumindest an den Rändern berücksichtigt wird. Ein Beispiel für diese Methoden ist die NURBS-enhanced Finite-Elemente-Methode [136,137].

Die vorliegende Dissertation behandelt die Lösung von Fluid-Struktur-Interaktionen unter Verwendung des gleichen Splines am Interface. Der Spline wird auf der Strukturseite in einem isogeometrischen Verfahren verwendet, auf der Fluidseite dagegen in der NURBS-enhanced Finite-Elemente-Methode als Randbeschreibung genutzt. Die Dissertation stellt die gesamte Lösungsmethodik des oberflächen-gekoppelten Problems vor, welche auf einem implizit gestaffelten Ansatz basiert. Es wird sowohl der übliche Ansatz der Dirichlet-Neumann-Kopplung, als auch der Ansatz einer Robin-Neumann-Kopplung diskutiert. Die Verwendung des identischen Splines auf beiden Seiten ermöglicht eine konsistente und konservative Übertragung der benötigten Informationen, ohne die angepasste, individuelle Diskretisierung auf beiden Seiten einzuschränken. Bestehende Beschleunigungstechniken für partitionierte Ansätze werden im Rahmen der spline-basierten Methode diskutiert und in einer modifizierten Variante kombiniert.

Die numerischen Vorteile der Kombination aus isogeometrischer Analyse und der NURBS-enhanced Finite-Elemente-Methode werden durch einfache stationäre und transiente Testfälle verifiziert. Eine deutlich verbesserte Genauigkeit im Vergleich zur üblichen Kopplung von linearen Lagrange'schen Finite-Elementen und isogeometrischer Analyse kann

anhand eines geschlossenen Gebietes mit deformierbaren Rändern nachgewiesen werden. Durch die zusätzliche Berücksichtigung von Festkörperkontakt wird gezeigt, dass bekannte Vorteile der isogeometrischen Strukturanalyse in die Fluid-Struktur-Interaktion übertragen werden können. Darüber hinaus ist die Spline-basierte Beschreibung des Interfaces nicht nur für die Kopplung der Einzelfelder von Vorteil, sondern auch für Fluid-Struktur-Interaktionen mit freien Oberflächen. Die entsprechenden Erweiterungen der Randbedingungen werden diskutiert und an einem dreidimensionalen Testfall demonstriert.

Contents

List of Figures	xi
List of Tables	xiii
1 Introduction	1
1.1 Numerical Methods for FSI	2
1.2 Motivation	3
1.3 Overview	4
2 Physical background	7
2.1 Continuum mechanics	7
2.1.1 Kinematics	8
2.1.2 On stress and force	10
2.1.3 Fundamental laws of continuum mechanics	12
2.1.4 Constitutive equations	16
2.2 Summary with focus on FSI and FSF problems	19
3 Spline-based methods for fluid and structural mechanics	23
3.1 Splines	24
3.1.1 Non-uniform rational B-splines	24
3.2 Introduction to finite element methods	27
3.2.1 Short remark on numerical integration	28
3.2.2 Short remark on linearization of nonlinear problems	29
3.2.3 Short remark on the numerical domain	30
3.3 Finite dimensional ansatz spaces	30
3.3.1 Linear isoparametric Lagrangian finite elements	31
3.3.2 NURBS-based isogeometric analysis	32
3.3.3 NURBS-enhanced finite elements	34
3.4 Numerical method for the elastodynamic problem	43
3.4.1 Variational formulation	44
3.4.2 Time discretization	45

3.4.3	Linearization of the nonlinear problem	47
3.5	Numerical methods for fluid mechanical problem	48
3.5.1	Variational formulation	49
3.5.2	Deforming spatial domain/stabilized space-time method	50
3.5.3	Finite dimensional spaces for space-time	54
3.5.4	On moving boundaries	56
4	Numerical methods for the coupled problem	61
4.1	Temporal coupling between the space-time and semi-discrete approach	63
4.2	Spatial Coupling	65
4.2.1	Energy conservation and consistency	66
4.2.2	Finite Interpolation Element	67
4.2.3	Direct transfer between spline-based methods	72
4.3	Iterative solver	76
4.3.1	Field partitioning	77
4.3.2	Gauss-Seidel iteration	81
4.3.3	Newton-based schemes	84
4.3.4	Aitken's method	85
4.3.5	Remark on relaxation with splines	91
5	Influence of discretization methods to FSI test case	93
5.1	Cylinder	93
5.1.1	Rigid Cylinder	93
5.1.2	Flexible Cylinder	95
5.1.3	Influence of deformation to the benefit of NEFEM	98
5.2	2D flexible structure	100
6	Challenging problems for partitioned approaches	105
6.1	Balloon	105
6.2	Elastic pressure tube	109
6.2.1	Numerical test	110
7	Transfer of advantages of spline-based methods to FSI	115
7.1	Falling ring	115
7.1.1	Contact formulation	116
7.1.2	Configuration and results	117
7.2	Sloshing tank	122
7.2.1	On boundary conditions	122
7.2.2	Configuration and results	126

8 Conclusion and Outlook	131
8.1 Conclusion	131
8.2 Outlook	133
Bibliography	135

List of Figures

2.1	Motion of a body.	9
2.2	Stress relation on an internal surface and on two bodies with a common surface.	11
2.3	Sketch of a typical setup for fluid-structure interaction with free-surface flows.	19
3.1	General discretization schemes.	23
3.2	NURBS basis functions	26
3.3	Knot vector, control point coordinates and weights	26
3.4	Control polygon and related NURBS curve	26
3.5	Position of knots evaluated on NURBS curve	26
3.6	Mapping from physical space to reference element	28
3.7	Mapping and basis functions for linear isoparametric Lagrangian finite element	31
3.8	Mapping for isogeometric finite elements	32
3.9	Spatial discretization with NEFEM	35
3.10	Illustration of the principal NEFEM concepts	36
3.11	Dirichlet boundary conditions for Cartesian space-time NEFEM.	37
3.12	Shape functions for elements with curved edges.	38
3.13	The pFEM mapping	39
3.14	The Triangle-Rectangle-Triangle mapping	40
3.15	Comparison of the p-FEM and the TRT mapping	42
3.16	Discretized time slab Q_n	51
3.17	Interface tracking	56
3.18	Interface capturing	56
4.1	Temporal coupling between the fluid solver and the elastodynamic solver.	64
4.2	Finite Interpolation method for piecewise linear elements.	68
4.3	Finite Interpolation method for SFEM on fluid side and IGA on structural side.	68
4.4	Gapless geometry representation of the interface for NEFEM-IGA coupling.	72

4.5	Basis functions for NEFEM and NURBS-based IGA along the interface. .	72
5.1	Sketch of the rigid cylinder benchmark.	94
5.2	Grid convergence for rigid cylinder.	96
5.3	Sketch of the flexible cylinder test case.	97
5.4	Pressure distribution for the undeformed and the deformed cylinder. . . .	98
5.5	Convergence with respect to drag and displacement for the flexible cylinder.	98
5.6	Pressure field for different Young's moduli.	99
5.7	Differences in displacement between SFEM and NEFEM.	100
5.8	Sketch of the moving flag featflow benchmark.	101
5.9	Pressure distribution for deformed flag at time $t=13.6s$	102
5.10	Vertical displacement of point A	102
5.11	Horizontal displacement of point A	103
5.12	Grid convergence of amplitude of vertical displacement at point A	103
6.1	Balloon type test case: circular inflow enclosed by a circular thin structure.	106
6.2	Error in radius over time for $\Delta t = 0.001s$	108
6.3	Absolute error of the radius at $t=1.0$ s.	109
6.4	Definitions of the elastic pressure tube test case.	110
6.5	Elastic Tube: Propagating Shock	111
6.6	Coupling iterations for different types of acceleration.	112
7.1	Schematic view of contact problem between elastic body and rigid wall. .	116
7.2	Configuration and dimensions for fluid-structure interaction with contact.	118
7.3	Displacement of point A over time.	119
7.4	Close-up of displacement of point A over time.	119
7.5	Snapshots of contact situation.	121
7.6	Definitions of coordinate systems.	123
7.7	Initial configuration of the sloshing tank.	126
7.8	Snapshot series of sloshing tank.	128
7.9	Absolute displacement in x -direction.	129
7.10	Difference of displacement in x -direction with respect to excitation. . . .	129
7.11	Absolute displacement in y -direction.	130
7.12	Movement of two nodes on free surface.	130

List of Tables

2.1	Balance laws.	15
5.1	Rigid cylinder benchmark parameters.	94
5.2	Grids used for the rigid cylinder benchmark simulations.	95
5.3	Flexible cylinder parameters.	96
5.4	Grids used for the flexible cylinder simulations.	97
5.5	Moving flag featflow benchmark parameters.	101
5.6	Grids used for moving flag simulations.	101
6.1	Parameters of the balloon test case.	107
6.2	Material parameters of the elastic pressure tube test case.	110
6.3	Speed-up in terms of coupling iterations and computational time.	113
7.1	Falling ring test case parameters.	118
7.2	Parameters of the sloshing tank test case.	127

1 Introduction

— *"Wind is caused by the trees waving their branches"* — Odgen Nash

What a confusing quote! It is obviously the other way around, isn't it? But is it possible to disprove the quote by just observing the depicted scene? Welcome to the fascinating world of fluid-structure interaction! This term summarizes all problems with mutual dependencies between fluids and deformable solids. With respect to the quote, these would be air and the elastic trees. Still thinking of air as the acting fluid, one can easily identify many additional examples in daily life, for example, waving flags. The sculptures of Anthony Howe show [82], how amazing fluid-structure interaction can be.

In engineering, phenomena of fluid-structure interaction play an important role. A prominent example is aeroelasticity as a sub-area of aerospace engineering, where, e.g., the behaviour of elastic airplane wings has to be included in the design process in order to maximize the aerodynamic performance. If thermal effects are additionally regarded, the field of aerothermalelasticity emerges. It considers, for example, the interaction of the hot exhaust gas with the structure of rocket nozzles. But not only in aerospace engineering, also in other engineering areas one can find further examples: the blood flow through flexible arteries in bio-engineering, the deformation of wind turbine blades in energy engineering, the flexible hulls of ships exposed to wind and water in naval engineering, or the movement of flexible piston rings in automotive engineering. And is there any text on fluid-structure interaction that does not mention the failure of the Tacoma Narrows Bridge in 1940? Summarizing all these problems: The study of fluid-structure interaction is crucial if the single-field problems of fluid and solid mechanics cannot be analyzed independently.

In general, the highly non-linear characteristics of fluid-structure interaction do not allow for an analytical solution. But in the most cases, experimental studies of such phenomena are too expensive or simply not realistic, e.g., in case of airplane prototypes. However, computational analysis enables detailed studies of such problems. And with growing

computing performance, the complexity of problems that can be accurately solved with numerical methods is raised. These results can help to decrease safety margins in airplane design to increase the efficiency in the sense of fuel consumption per unit mass. However, computational capacities are always limited and, thus, efficient numerical methods are still essential in computational analysis.

1.1 Numerical Methods for FSI

The numerical methods to compute problems involving fluid-structure interaction are manifold. While monolithic approaches solve all involved equations together, partitioned approaches solve the single-field problems of fluid and solid mechanics separately by exchanging information at a common interface. In monolithic schemes, difficulties arise from the different timescales of the single-field problems and make tailored preconditioning techniques necessary, while in partitioned approaches, stability problems can lead to inefficient iterative processes. However, partitioned schemes are attractive, since linking of existing solvers into a total solver is possible with little effort. In both solution strategies, additional challenges emerge if the sub-problems of fluid and solid mechanics are discretized separately. Then, additional methods for recombining meshes with non-matching interface discretizations are needed. A more detailed categorization and literature survey of numerical methods for fluid-structure interaction is discussed in Ch. 4.

In recent years, with isogeometric analysis, a promising numerical method emerged that tries to close the gap between computer aided design (CAD) and numerical analysis [89]. While CAD is usually used for the design of technical objects like cars, airplanes, etc., the resulting spline-based geometry model is utilized within numerical analysis only for the domain definition. In a very time-consuming procedure, this domain is discretized into a computational grid on which the numerical analysis is conducted. Besides the computing resources, the generation of a computational mesh is the most expensive and time-consuming part of computational analysis, taking up to 80% of the overall time [89]. As in the standard finite element method, the isogeometric analysis considers the variational formulation of partial differential equations. Within the variational formulation, the unknown solution is approximated with interpolation functions. But instead of linear Lagrangian polynomials, the spline basis functions of the geometry description are utilized. With that, the spline-based geometry generated with CAD tools is directly involved in numerical analysis. An additional grid generation is not

needed anymore. Unfortunately, today's CAD tools provide surface splines only. Therefore, the automatic generation of volume splines is an ongoing topic in the isogeometric analysis community. To the author's knowledge, a tool providing such models does not exist and, thus, volume splines need to be generated manually. But since a volume discretization is mandatory in fluid mechanics, the applicability of isogeometric analysis is currently limited to problems with geometries of minor complexity. In structural mechanics the situation is different. In case of thin-walled structures, structural problems can be reduced to a surface problem, taking shell theory into account. Then, isogeometric analysis can be applied directly to the CAD surface spline. In addition to the simplified discretization, the higher-order smooth basis functions can be beneficial for numerical accuracy. A more detailed discussion of isogeometric analysis is given in Sec. 3.3.2.

Since isogeometric analysis was established, the method is applied for fluid-structure interaction [9,10]. Within that field, the development of the method was pushed forward mainly by two types of applications. On the one hand, bio-medical problems like blood flow in arteries [10], in the abdominal aorta [9], heart-valves [84,85] or ventricular assist devices [113] were analyzed with isogeometric analysis. On the other hand, the method was continuously developed within the scope of wind turbines [11,12,83,161]. Due to the thin-walled structures of the blades, isogeometric analysis was applied in this case on the structural side only; the fluid part was discretized with standard finite elements. But also in other engineering fields the method was used for the simulation of fluid-structure interaction. Isogeometric shell elements were used in fluid-structure interaction simulations in maritime engineering [160]. And the smooth description of the boundaries makes this method suitable for problems that are sensitive to geometrical errors, like the analysis of elasto-capillary fluids [153].

1.2 Motivation

The presented thesis continues the work initiated in SFB 401 on numerical methods for aeroelasticity and the software framework SOFIA [29]. The framework was implemented and applied to accompany wind tunnel tests in the HIRENASD [128] and ASDMAD project [38]. Later on, it was extended for free-flying aircrafts [158], in SFB TR40 for aerothermoelastic problems [80,81], and, parallel to the developments of this thesis, for aeroservoelastic problems with the focus on optimal control [33]. In the presented thesis,

the partitioned approach is expanded by spline-based methods to integrate the benefits of the smooth geometry description of the coupling interface.

As previously mentioned, the application of isogeometric analysis is currently limited to cases with moderate complexity. The generation of physical models is straight-forward, if surface models, i.e., shell theory, can be applied. Unfortunately, since CAD tools provide only surface models, the generation of volume splines is not trivial. Hence, in complex cases of fluid-structure interaction, isogeometric analysis is usually combined with Lagrangian finite elements. Unfortunately, the geometrical accuracy of the fluid-structure interface is lost in that case. A compromise between both methods is the NURBS-enhanced finite element method [136, 137], where the spline-based description of the geometry is combined with Lagrangian finite elements. The integration of splines into the method allows for integrating the exact representation of a domain boundary. Additionally, the computational grid can be created by existing software. Combining NURBS-enhanced finite elements with isogeometric analysis is promising, because the interface is now represented geometrically exact on both sides. The method can be attractive especially for problems where the fluid domain needs to be discretized with a volume grid, but the shell theory is sufficient for structural analysis. The main aim of the thesis is to investigate numerical advantages of that combination. Additionally, further advantages of the spline-based description in the context of fluid-structure interaction with free-surface flows are derived.

1.3 Overview

At the beginning, the presented thesis provides a brief but complete overview about the theoretical background needed to understand the derived strategies for solving fluid-structure interaction problems with spline-based methods. Thus, Chapter 2 introduces the physical models for the single-field problems of fluid mechanics and elastodynamics in terms of continuum mechanics. In the same context, the fluid-structure interaction problem is defined. In Chapter 3, the numerical methods to solve the single-field problems are introduced. These methods are the basis for the iteratively staggered solution approach presented in Chapter 4. In the following, results of numerical experiments are presented. Chapter 5 focuses on the effect to the accuracy by combining NURBS-enhanced finite elements and isogeometric analysis. Afterwards, some challenging problems for the partitioned solution approach are analyzed in Chapter 6. The numerical experiments are concluded in Chapter 7 by demonstrating further improvements of

methods for fluid-structure interaction if spline-based models are involved. Eventually, the thesis ends with a summary and further perspectives. In all phases, reference is made to literature which can provide a more detailed insight into the particular topics.

2 Physical background

The interaction between a fluid and an elastic structure is actually the interaction between matter in different states of aggregation, acting on two different domains, which are intersecting at a common surface. The field of Fluid-Structure Interaction (FSI) is relevant if the behaviour of a system cannot be studied without taking the mutual dependency between fluid and structure into account.

Both matter and the separating interface, rely on physical properties that can be observed at different scales. If molecular effects are neglected, the physical characteristics of matter can be studied by applying the hypothesis of continuum mechanics. This hypothesis allows to derive equations for motion, deformation, etc., of any material in a general way. The individual physical properties are taken into account by specific constitutive models. Finally, the continuum hypothesis allows to directly derive coupling conditions on the interface.

A complete introduction to continuum mechanics is not in the scope of this thesis; the interested reader is pointed to [37, 114]. In the following sections, the essential parts to derive the problem of fluid-structure interaction including the differences between the underlying theory for fluid and solid mechanics are introduced.

2.1 Continuum mechanics

As mentioned in the introduction above, the physical properties of matter can be studied on different scales: From the smallest scale, where matter is assembled from particles, i.e., molecules, atoms, etc., to the macroscopic scale, where the physical properties of matter are homogenized over the observed domain. Even if the analysis on microscopic scales is crucial to get detailed information about the internal structure of matter and its resulting properties, it is impossible to handle in numerical analysis on domain sizes on the scale of engineering problems. Even tiny domains contain millions of discontinuously distributed particles and, hence, millions of degrees of freedom. The macroscopic

viewpoint immediately leads to the field of continuum mechanics, which is clearly more convenient for numerical analysis. Even if such an approach decreases the accuracy of the analysis in comparison to the microscopic one, the remaining accuracy is still sufficient for studying engineering problems. The decision, whether continuum mechanics can be reliably applied, can be made based on the Knudsen number, allowing to distinguish between continuum and statistical mechanics.

The theory of continuum mechanics is based on the *continuum hypothesis* by Cauchy, who first introduced the concept of strain and stress. Following that hypothesis, the motion and deformation of a hypothetical physical object — called continuous medium or continuum — is studied under the assumption that the matter is continuously distributed over the entire object. Hence, equations are derived to describe the behaviour of the continuum under external effects. Initially, there is no distinction between the specific behaviour of the type of matter. The resulting field equations need to be closed by models for the specific matter to be investigated, i.e., fluids or solids. Even if the microscopic structure of the studied matter is not regarded explicitly, its properties are transferred to these constitutive laws.

The derivation of field equations in continuum mechanics can be subdivided into two main parts. While kinematics describe the motion purely in sense of geometry, balance laws describe the physics that cause the motion of the observed object.

2.1.1 Kinematics

Following the hypothesis of continuum mechanics, a body B_X is defined by a domain Ω_X with boundary Γ_X that is occupied by the observed portion of matter at time $t = 0$, cf. Fig. 2.1. This configuration is called *reference* or *material configuration*. Within the domain Ω_X , a material point is uniquely defined by $\mathbf{X}$. The material points are changing the position during time, forming at time t the body B_x on Ω_x with boundary Γ_x . This configuration is called *current* or *spatial configuration*. As long as discontinuities, e.g., cracks and voids are neglected, the spatial position $\mathbf{x}$ of a material point $\mathbf{X}$ at time t is specified by

$$\mathbf{x} = \phi(\mathbf{X}, t), \quad (2.1)$$

where the function ϕ is called the motion of the body. Based on the motion of a material point $\mathbf{X}$ its displacement is given by

$$\mathbf{d} = \phi(\mathbf{X}, t) - \mathbf{X}. \quad (2.2)$$

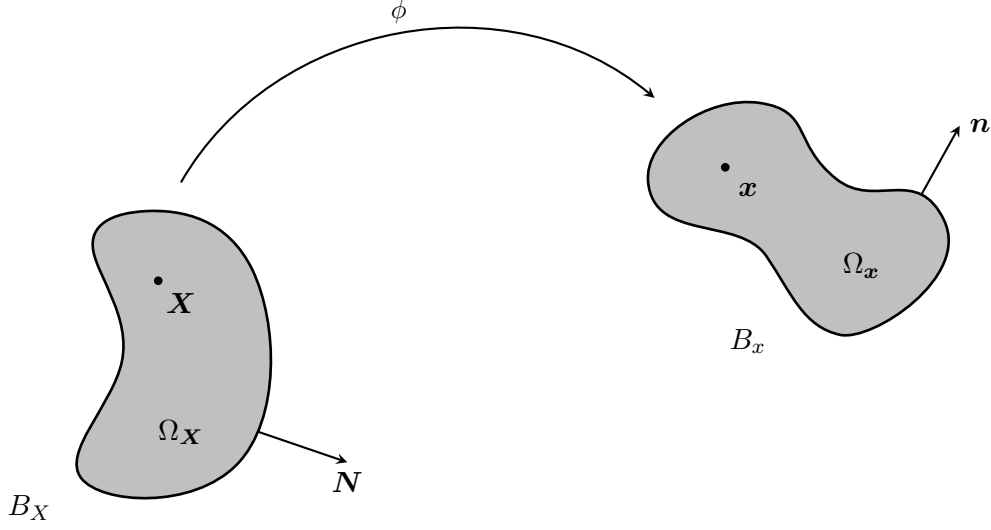


Figure 2.1: Motion of a body.

while the local deformation at that point can be measured by the deformation gradient

$$\mathbf{F} = \nabla_{\mathbf{X}} \mathbf{x} = \mathbf{I} + \nabla_{\mathbf{X}} \mathbf{d} \quad (2.3)$$

and its Jacobian determinant

$$J = \det(\mathbf{F}), \quad (2.4)$$

while $\nabla_{\mathbf{X}}$ represents the nabla operator with respect to $\mathbf{X}$. Based on this relation between material and spatial configuration, rates of change, including rates of motion, within the observed continuum can be observed from two different viewpoints:

Lagrangian viewpoint: The properties of a body B are observed in the reference/material configuration. The observer is following the material points $\mathbf{X}$, while the spatial position $\mathbf{x}$ of these points is given by the motion $\phi(\mathbf{X}, t)$ for all t . The reference configuration — often the initial configuration is chosen — is time-independent and fixed in space, i.e., a certain point $\mathbf{X}$ is always surrounded by the same set of points.

Eulerian viewpoint: The so-called spatial description is defined by a set of points with fixed location. The observer is located at a fixed non-moving spatial position $\mathbf{x}$, recognizing the change by all material points that are passing that position. The relation between the particle passing $\mathbf{x}$ at a certain time $t = t_A$ and its specific position in the initial configuration is given therefore by the inverse of motion $\mathbf{X} = \phi^{-1}(\mathbf{x}, t_A)$. For $t \neq t_A$ another particle is passing $\mathbf{x}$. With that, the phenomenon of advection

is introduced, i.e., the changes measured at $\mathbf{x}$ are not only the local changes of the particle reaching $\mathbf{x}$, but also the change of the changing particles transported through $\mathbf{x}$.

Based on these definitions and by exploiting the one-to-one mapping of the motion ϕ , the velocity of a particle is defined in both frameworks as follows:

$$\frac{\partial \phi(\mathbf{X}, t)}{\partial t} = \underbrace{\frac{\partial \mathbf{d}(\mathbf{X}, t)}{\partial t}}_{\text{velocity in Lagrangian viewpoint}} = \frac{\partial \mathbf{d}(\phi^{-1}(\mathbf{x}, t))}{\partial t} = \underbrace{\mathbf{u}(\mathbf{x}, t)}_{\text{velocity in Eulerian viewpoint}} \quad (2.5)$$

Additionally, the rate of change, also called material time derivative, for any scalar field $\Phi = \Phi(\mathbf{X}, t)$ in material configuration is given by:

$$\frac{D\Phi}{Dt} = \frac{\partial \Phi}{\partial t}, \quad (2.6)$$

while the material time derivative for any scalar field $\Psi = \Psi(\mathbf{x}, t)$ on a spatial point $\mathbf{x}$ is given by

$$\frac{D\Psi}{Dt} = \frac{\partial \Psi}{\partial t} + \frac{\partial \Psi}{\partial \mathbf{x}} \cdot \frac{\partial \mathbf{x}}{\partial t} = \frac{\partial \Psi}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{x}} \Psi, \quad (2.7)$$

where the second term on the right-hand side is the convective part. Here, $\nabla_{\mathbf{x}}$ represents the nabla operator with respect to $\mathbf{x}$. In particular, the acceleration passing $\mathbf{x}$ at time t can now be computed by

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{x}} \mathbf{u}. \quad (2.8)$$

2.1.2 On stress and force

The motion and deformation previously defined by kinematic relations are caused by external forces applied to the observed body. Before the relation between motion and forces is introduced, the general definition of forces on deformable bodies in the framework of continuum mechanics is shortly discussed. Later on, this will be essential for deriving coupling conditions at the interface between fluid and solid.

The total forces acting on a body can be subdivided into body and surface forces. Body forces, e.g., gravitational forces, are acting on the entire volume of material and can be specified by

$$\mathbf{f}_b = \int_{\Omega_x} \rho \mathbf{b} \, d\Omega_x, \quad (2.9)$$

where the vector $\mathbf{b}$ is the body force per unit of mass. Instead, surface forces are acting on a surface of a body that separates it from its environment. *Cauchy's stress postulate* introduces a stress vector field $\mathbf{s}(\mathbf{x}, t, -\mathbf{n})$, specified as the force per unit area depending on the position $\mathbf{x}$, time t and unit normal $\mathbf{n}$ on an oriented surface $\Gamma_{\mathbf{x}}$. Commonly, the outer normal on $\Gamma_{\mathbf{x}}$ is chosen, while $\mathbf{s} = \mathbf{s}(\mathbf{x}, t, \mathbf{n})$ is the external force. Hence, the total surface force on surface $\Gamma_{\mathbf{x}}$ can be written as:

$$\mathbf{f}_s = \int_{\Gamma_{\mathbf{x}}} \mathbf{s} d\Gamma_{\mathbf{x}}. \quad (2.10)$$

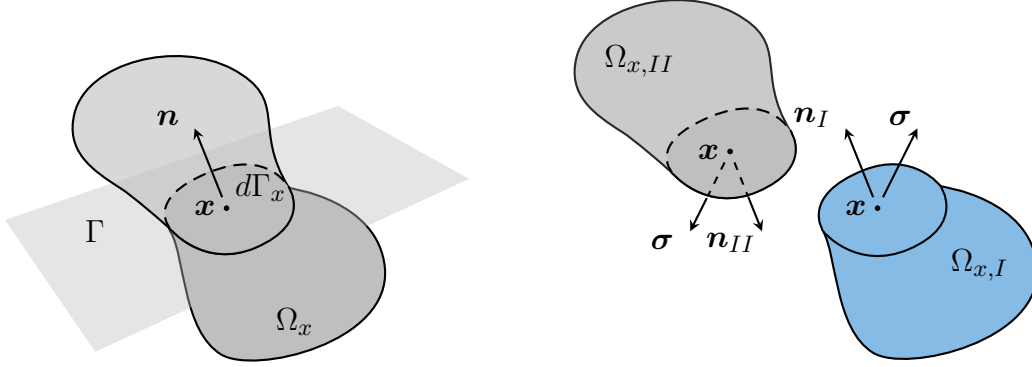


Figure 2.2: Stress relation on an internal surface (left) and on two bodies with a common surface (right).

As the vector field $\mathbf{s}$ introduced above is uniquely defined over the entire body, the definition of surface forces can be generalized by cutting the body with an imaginary surface Γ ; cf. Fig. 2.2. Since the resulting internal surface $d\Gamma_{\mathbf{x}}$ is completely enclosed by the body, there are two possible choices for the surface normal. Following Newton's third law of motion, it is postulated that the traction for these choices must be equal, and therefore:

$$\mathbf{s}(\mathbf{x}, t, -\mathbf{n}) = -\mathbf{s}(\mathbf{x}, t, \mathbf{n}), \quad (2.11)$$

which is also called *Cauchy's reciprocal relation*. The state of stress of a body at point $\mathbf{x}$ at time t is uniquely defined with *Cauchy's law* by a second-order tensor $\mathbf{T}$, called *Cauchy's stress tensor*, and the relation:

$$\mathbf{s}(\mathbf{x}, t, \mathbf{n}) = \mathbf{T}(\mathbf{x}, t)\mathbf{n}. \quad (2.12)$$

The stress tensor is strongly depending on the studied material and will be modelled later. Now, the domain Ω can be sub-divided into two non-overlapping subdomains Ω_I and Ω_{II} with

$$\overline{\Omega_I} \cup \overline{\Omega_{II}} = \Omega \quad \text{and} \quad \Omega_I \cap \Omega_{II} = \emptyset. \quad (2.13)$$

The splitting interface between both subdomains can be defined as

$$\overline{\Omega_I} \cap \overline{\Omega_{II}} = \Gamma_{I,II}. \quad (2.14)$$

The configuration subscripts are omitted here, since the relation holds in both material and spatial configuration.

As the presented thesis focuses on conventional solids and fluids by neglecting polar materials, the resulting bodies are coupled only across the surface. Then, the concept of coupled stress and the outer normals of opposite sides result in the relation

$$\mathbf{T}_I(\mathbf{x}, t) \mathbf{n}_I = \mathbf{T}_{II}(\mathbf{x}, t) \mathbf{n}_{II} \quad \text{or} \quad \mathbf{T}_I(\mathbf{x}, t) \mathbf{n} = -\mathbf{T}_{II}(\mathbf{x}, t) \mathbf{n}. \quad (2.15)$$

Hence, it is not necessary to distinguish between internal and external surface (contact) forces. The main difference is how the observed body is separated from its environment. In the following, this is useful when boundary conditions for fluid-structure interaction and free-surface flows are defined in Sec. 2.2.

2.1.3 Fundamental laws of continuum mechanics

While the description of motion and deformation is based on geometrical relations, the physical response of a continuum to external influences is specified by balance laws, defined in material or in spatial form. These balance laws can be derived from the fundamental mechanical principles. Their detailed derivation can be found in any continuum mechanics book, like [37, 114]. Here, only the fundamental principles are recalled shortly.

Mass balance: The mass of a closed body remains unchanged during its motion, following the principle of conservation of mass. Therefore, the following relation holds:

$$\int_{\Omega_{\mathbf{x}}} \rho_0 \, d\mathbf{X} = \int_{\Omega_{\mathbf{x}}} \rho \, d\mathbf{x}, \quad (2.16)$$

with mass density in reference configuration $\rho_0 = \rho_0(\Omega_{\mathbf{X}})$ and in current configuration $\rho = \rho(\Omega_{\mathbf{x}})$.

Linear Momentum: The material time rate of change of linear momentum of body B_x at time t is equal to all forces acting on the body, cf. Sec. 2.1.2, known as the principle of balance of linear momentum or Newton's second law of motion:

$$\frac{D}{Dt} \int_{\Omega_x} \rho \mathbf{u} \, d\Omega_x = \int_{\Gamma_x} \mathbf{T} \mathbf{n} \, d\Gamma_x + \int_{\Omega_x} \rho \mathbf{b} \, d\Omega_x. \quad (2.17)$$

Angular momentum: The material time rate of change of angular momentum about the origin of body B_x at time t — known as the law of balance of angular momentum — is equal to the resultant moment about the origin of the external forces acting on B_x .

$$\frac{D}{Dt} \int_{\Omega_x} \mathbf{x} \times \rho \mathbf{u} \, d\Omega_x = \int_{\Gamma_x} \mathbf{x} \times \mathbf{T} \mathbf{n} \, d\Gamma_x + \int_{\Omega_x} \mathbf{x} \times \rho \mathbf{b} \, d\Omega_x. \quad (2.18)$$

In the absence of body moments, the principle of angular momentum leads to the conclusion, that the stress tensor is symmetric, see [37, 114], and thus

$$\mathbf{T} = \mathbf{T}^T. \quad (2.19)$$

Since this equation is implicitly fulfilled by the considered constitutive laws, it is not included explicitly in numerical analysis. However, the previous equation does not hold for polar materials, which are not considered in this thesis.

Energy conservation: The total energy of a body B_x is the sum of internal and kinetic energy. The material time rate of change of the total energy is equal to the sum of the rate of work done by volume and surface forces and the rate of the amount of heat generated in the volume and crossing the boundary. This is known as the principle of balance of energy or the first law of thermodynamics. With the internal energy per unit of mass ϵ , the heat flux through the surface $\mathbf{q}$, the heat internally generated per unit volume r and with strain rate tensor $\mathbf{D}(\mathbf{u}) = \text{sym}(\nabla_{\mathbf{x}} \mathbf{u})$, it is given as:

$$\frac{D}{Dt} \int_{\Omega_x} \rho \epsilon + \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} \, d\Omega_x = \int_{\Omega_x} \mathbf{T} : \mathbf{D} \, d\Omega_x - \int_{\Gamma_x} \mathbf{q} \cdot \mathbf{n} \, d\Gamma_x + \int_{\Omega_x} r \, d\Omega_x. \quad (2.20)$$

The field equations of continuum mechanics are completed by the second law of thermodynamics and the related Clausius-Duhem inequality. Since thermal effects are not studied in this thesis, it is referred to [114] for more details.

All the fundamental laws of continuum mechanics, also called balance laws, previously introduced in Eulerian form can be generalized in terms of integral equalities [114]. Let $\hat{\mathbf{a}}$ be a vector of a conserved quantity, $\hat{\mathbf{b}}$ a volumetric source term and $\mathbf{C}_x$ a second rank tensor defining surface fluxes $\mathbf{C}_x \mathbf{n}$. Then the master balance law describes the change of the quantity in the observed domain Ω_x by fluxes over its boundary and source terms:

$$\frac{D}{Dt} \int_{\Omega_x} \hat{\mathbf{a}} d\Omega_x = \int_{\Gamma_x} \mathbf{C}_x \mathbf{n} d\Gamma_x + \int_{\Omega_x} \hat{\mathbf{b}} d\Omega_x. \quad (2.21)$$

By applying the transport and divergence theorem, the master balance law in Eulerian form can be rewritten as

$$\int_{\Omega_x} \left[\frac{\partial \hat{\mathbf{a}}}{\partial t} + \nabla_x \cdot (\hat{\mathbf{a}} \mathbf{u} - \mathbf{C}_x) - \hat{\mathbf{b}} \right] d\Omega_x = \mathbf{0}. \quad (2.22)$$

Hence, a local form of the master balance law in Eulerian form can be extracted:

$$\frac{\partial \hat{\mathbf{a}}}{\partial t} + \nabla_x \cdot (\hat{\mathbf{a}} \mathbf{u} - \mathbf{C}_x) - \hat{\mathbf{b}} = \mathbf{0}. \quad (2.23)$$

By applying the relations introduced in Sec. 2.1.1, the integral equation (2.22) can be transformed to the material domain, resulting in the master balance law in Lagrangian form, see [114]:

$$\int_{\Omega_X} \left[\frac{\partial J \hat{\mathbf{a}}}{\partial t} - \nabla_X \cdot (J \mathbf{F}^{-1} \mathbf{C}_x) - J \hat{\mathbf{b}} \right] d\Omega_X = \mathbf{0}. \quad (2.24)$$

Similar to the Eulerian form, a local form can be extracted:

$$\frac{\partial J \hat{\mathbf{a}}}{\partial t} - \nabla_X \cdot (J \mathbf{F}^{-1} \mathbf{C}_x) - J \hat{\mathbf{b}} = \mathbf{0}, \quad (2.25)$$

where the relation

$$J \mathbf{F}^{-1} \mathbf{C}_x = \mathbf{C}_X \quad (2.26)$$

is called Piola transformation. By applying (2.22)-(2.25) to the introduced balance laws, local forms of the mass, momentum and energy equations both in Eulerian and Lagrangian viewpoint can be derived. Both sets of equations are the basis for numerical analysis later and are summarized in Tab. 2.1. The stress tensor in material configura-

tion is given by the *second Piola-Kirchhoff stress* $\mathbf{S}$ following the relation:

$$\mathbf{S} = J\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}. \quad (2.27)$$

Material description (Lagrangian) on $\Omega_{\mathbf{X}}, \forall t \in (0, T)$	Spatial description (Eulerian) on $\Omega_{\mathbf{x}}, \forall t \in (0, T)$
Mass conservation	
$\rho_0 = \rho J \quad (2.28)$	$\frac{D\rho}{Dt} + \rho \nabla_{\mathbf{x}} \cdot \mathbf{u} = 0 \quad (2.29)$
Conservation of Linear Momentum	
$\rho \frac{D^2 \mathbf{d}}{Dt^2} = \nabla_{\mathbf{X}} \cdot (\mathbf{F} \mathbf{S}^T) + \rho_0 \mathbf{b} \quad (2.30)$	$\rho \frac{D\mathbf{u}}{Dt} = \nabla_{\mathbf{x}} \cdot \mathbf{T}^T + \rho \mathbf{b} \quad (2.31)$
Conservation of Angular Momentum	
$\mathbf{S}^T = \mathbf{S} \quad (2.32)$	$\mathbf{T}^T = \mathbf{T} \quad (2.33)$
Conservation of Energy	
$\rho_0 \frac{D\epsilon_0}{Dt} = \mathbf{F} \mathbf{S}^T \cdot (\nabla_{\mathbf{X}} \mathbf{u}) - \nabla_{\mathbf{X}} \cdot \mathbf{q}_0 + r_0 \quad (2.34)$	$\rho \frac{D\epsilon}{Dt} = \mathbf{T}^T \cdot \nabla_{\mathbf{x}} \mathbf{u} - \nabla_{\mathbf{x}} \cdot \mathbf{q} + r \quad (2.35)$

Table 2.1: Balance laws [122].

Although these laws are applicable to all continua regardless of their internal structure, the viewpoint finally chosen depends on the specific problem. For the analysis of elastodynamic problems, the Lagrangian viewpoint is commonly utilized, since the relative movement of neighboring particles is small. This is not the case in fluid dynamics, i.e., if vortices are present. Additionally, in case of fluid dynamics, engineers are more interested in the flow field within a defined domain than in the movement of specific particles.

Therefore, the Eulerian viewpoint is a proper choice in fluid dynamics. However, the Eulerian viewpoint is not sufficient anymore, if moving boundaries are involved, i.e., in case of fluid-structure interaction and free-surface flow problems. This is discussed in more detail in Sec. 3.5.4.

Since the unknowns in the balance law are density, velocity and energy, the equations need to be closed by constitutive models for the stress tensors $\mathbf{T}$ and $\mathbf{S}$, respectively. These models are representing the intrinsic physical properties of the considered continuum.

2.1.4 Constitutive equations

In the previous sections, the fundamental laws of continuum mechanics were introduced, together with the Eulerian and Lagrangian viewpoints. Following the continuum hypothesis, these laws can be applied to every physical object for which the assumption of a continuously distributed matter holds and a macroscopic viewpoint is convenient. Although these laws were introduced in different frameworks, the fundamental balance laws do not distinguish between different kinds of continua, i.e., gases, liquids or solids. For that reason, the previous laws need to be closed by mathematical models that reflect the physical properties of an observed continuum. Often, these models are derived using experimental data and mathematical generalizations of these data. One of the best known constitutive equation is the Hooke's law for linear elastic materials, that was derived in the 17th century by Robert Hooke based on observations of springs.

In the following, the constitutive equations applied in numerical studies of this thesis are presented. These are the models for hyperelastic solids and incompressible Newtonian fluids.

Models for structural mechanics

The constitutive model that is mainly applied to solids in the presented thesis relies on the theory of hyperelasticity. Assuming that the observed body is isothermal, energy is stored only in deformation, cf. Eq. (2.20). Hence, the elastic solid keeps its deformation as long as the applied outer forces are not varying. Assuming ideal elasticity and a stress-free initial configuration, the observed body deforms back to its initial configuration, when the outer forces are diminishing.

The simplest hyperelastic material, convenient for moderate strains, is based on Hooke's law for a general anisotropic solid, giving a linear relation between strain and stress tensor:

$$\mathbf{S}^s = \mathcal{C} : \mathbf{G}, \quad (2.36)$$

with the Green strain tensor

$$\mathbf{G} = \frac{1}{2} (\mathbf{F}^T \mathbf{F} - \mathbf{I}), \quad (2.37)$$

and a constant fourth-rank tensor of elastic moduli $\mathcal{C}$, that can be split into its invariants by using scalars α, β and γ :

$$\mathcal{C}_{ijkl} = \alpha \delta_{ij} \delta_{kl} + \beta \delta_{ik} \delta_{jl} + \gamma \delta_{il} \delta_{jk}. \quad (2.38)$$

Although $\mathcal{C}$ can be designed to represent various types of anisotropic materials [150], the simplest form assumes isotropy. Following that assumption, the constitutive law (2.36) can be rewritten by introducing the Lamé parameters $\lambda^s = \alpha$ and $2\mu^s = \beta + \gamma$ into (2.38):

$$\mathbf{S}^s = \lambda^s \text{tr}(\mathbf{G}) \mathbf{I} + 2\mu^s \mathbf{G}. \quad (2.39)$$

A complete derivation of the model based on the strain energy function is given, e.g., in [114]. Alternatively, the model can be expressed by using the Young's modulus E and the Poisson ratio ν as material properties:

$$\lambda^s = \frac{E}{2(1+\nu)}, \quad (2.40)$$

$$\mu^s = \frac{\nu E}{(1+\nu)(1-2\nu)}. \quad (2.41)$$

While the Young's modulus is comparable to the factor of proportionality in Hooke's law for springs, the Poisson ratio is the ratio of transversal to axial strain.

The resulting constitutive law is called *St. Venant-Kirchhoff model*. It is time-independent and does not rely on the rate of deformation. Without any additional restriction, i.e., by adding a direct relation to the mass conservation equation, the material is assumed to be compressible. It is physically valid in case of small strains [114] and moderate forceful compressions [141]. But since rigid-body motions produce no strains, it allows for large displacements. Finally, assuming an isothermal and compressible body, the equation of linear momentum is the only field equation to be taken into account to study motion in classical elasticity theory.

Models for fluids

The fundamental characteristic property distinguishing a fluid from a solid is its inability to sustain shear stress when it is at rest or uniform motion. In case of an external constant shear stress, a fluid will continuously flow, while a solid would reach a constant deformation, called static equilibrium. In contrast, the stress on a surface element consists only of normal pressure $-p$ if the fluid rests or is in uniform motion, while tangential stress is zero. This property needs to be reflected by the constitutive model. Thus, the fluid stress tensor is split into contributions by the hydrostatic pressure p , and, in case the fluid exerts shear stresses, viscous stresses T^v , following George G. Stokes (1845):

$$\mathbf{T}^f = -p\mathbf{I} + T^v. \quad (2.42)$$

Similar to the Green strain tensor (2.37), one can introduce a strain rate tensor:

$$\mathbf{D}(\mathbf{u}) = \frac{1}{2} (\nabla_{\mathbf{x}} \mathbf{u} + (\nabla_{\mathbf{x}} \mathbf{u})^T) = \text{sym}(\nabla_{\mathbf{x}} \mathbf{u}), \quad (2.43)$$

which disappears if the fluid rests or is in uniform motion. One common choice is to model the viscous stress tensor as a homogeneous function of $\mathbf{D}$, called Stokesian fluid. In case of a linear homogeneous function, it is called linear viscous fluid that can be written as:

$$T^v = \mathcal{A} : \mathbf{D} \quad (2.44)$$

As in (2.38), the fourth-rank tensor $\mathcal{A}$ can be split by using α , β and γ :

$$\mathcal{A}_{ijkl} = \alpha \delta_{ij} \delta_{kl} + \beta \delta_{ik} \delta_{jl} + \gamma \delta_{il} \delta_{jk}. \quad (2.45)$$

By defining the coefficients of viscosity under the assumption of isotropy with $\bar{\lambda} = \alpha$ and $2\bar{\mu} = \beta + \gamma$, with (2.42) the constitutive equation for so called *Newtonian viscous fluids* follows

$$\mathbf{T}^f = (-p + \bar{\lambda} \nabla_{\mathbf{x}} \cdot \mathbf{u}) \mathbf{I} + \bar{\mu} (\nabla_{\mathbf{x}} \mathbf{u} + (\nabla_{\mathbf{x}} \mathbf{u})^T). \quad (2.46)$$

Since the viscous terms have to dissolve in case of a fluid in uniform motion, the constitutive equation is completed by the relation $\bar{\lambda} = -\frac{2}{3}\bar{\mu}$. This model is called *Stokes' law* and was obtained by Stokes as a generalization of *Newton's law of viscosity*, derived by Newton in 1687 based on experimental observations. Using the introduced constitutive relation within the momentum equation leads to the well-known *Navier-Stokes equations*. These can be applied to model the mechanical behaviour of a wide variety of common liquids and gases.

In the presented thesis, exclusively incompressible fluids ($\rho = \text{const}$) are studied. In that case, mass conservation (2.29) simplifies to

$$\nabla_{\mathbf{x}} \cdot \mathbf{u} = 0, \quad (2.47)$$

and the final stress tensor for linear, isotropic, viscous, incompressible fluids can be given as

$$\hat{\mathbf{T}}^f(p, \mathbf{u}) = -p\mathbf{I} + 2\bar{\mu}\mathbf{D}(\mathbf{u}). \quad (2.48)$$

If the viscosity is additionally kept constant, the energy equation is decoupled from mass and momentum equations and does not need to be solved to obtain information about the flow field.

2.2 Summary with focus on FSI and FSF problems

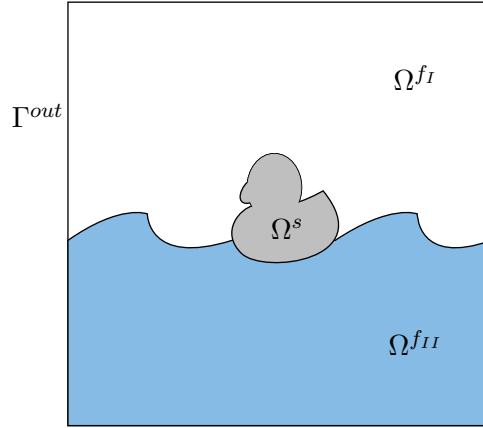


Figure 2.3: Sketch of a typical setup for fluid-structure interaction with free-surface flows.

With the recapped basics of continuum mechanics, the fundamental characteristic of fluid-structure interaction problems is the clear interface between both continua. The interface is uniquely defined as a sudden change in material properties. The same is valid also for free-surface flows, as long as two immiscible fluids are interacting. Both problems

can be generalized as a *surface-coupled problem*. Based on the previous sections, coupling conditions that need to be fulfilled on the interface between interacting continua can be introduced. The presented thesis is focusing on incompressible fluids and isothermal structures, so the only energy exchange between both parts are due to the mechanical loads.

Similar to the general description, the domain Ω can be sub-divided into three non-overlapping subdomains, two enclosing the immiscible fluids f_I and f_{II} , and one enclosing the solid s , as marked by the subscripts, cf. Fig. 2.3:

$$\overline{\Omega^{f_I}} \cup \overline{\Omega^{f_{II}}} \cup \overline{\Omega^s} = \Omega \quad \text{and} \quad \Omega^{f_I} \cap \Omega^{f_{II}} \cap \Omega^s = \emptyset. \quad (2.49)$$

The FSI interface is given by

$$\overline{\Omega^{f_i}} \cap \overline{\Omega^s} = \Gamma_{f_i s}, \quad i \in \{I, II\}, \quad (2.50)$$

and the free-surface is defined by

$$\overline{\Omega^{f_I}} \cap \overline{\Omega^{f_{II}}} = \Gamma_{f_I f_{II}}. \quad (2.51)$$

Finally, the contact line $\gamma_{f_I f_{II} s}$ between both fluids and the solid can be defined by

$$\overline{\Omega^{f_I}} \cap \overline{\Omega^{f_{II}}} \cap \overline{\Omega^s} = \gamma_{f_I f_{II} s}. \quad (2.52)$$

Following the kinematics introduction in 2.1.1, *kinematic coupling conditions*, ensuring a synchronized geometrical movement of all the interfaces, can be derived.

$$\mathbf{d}^\alpha = \mathbf{d}^\beta \text{ on } \Gamma^{\alpha\beta}, \quad \alpha, \beta \in \{s, f_I f_{II}\}, \alpha \neq \beta, \quad (2.53)$$

$$\mathbf{u}^\alpha = \mathbf{u}^\beta \text{ on } \Gamma^{\alpha\beta}, \quad \alpha, \beta \in \{s, f_I, f_{II}\}, \alpha \neq \beta, \quad (2.54)$$

$$\mathbf{a}^\alpha = \mathbf{a}^\beta \text{ on } \Gamma^{\alpha\beta}, \quad \alpha, \beta \in \{s, f_I, f_{II}\}, \alpha \neq \beta, \quad (2.55)$$

As the continua are in mutual dependency regarding stresses, Cauchy's reciprocal relation (2.12) leads to the *dynamic boundary conditions* for the FSI interfaces

$$\mathbf{T}^f \mathbf{n} = \mathbf{T}^s \mathbf{n} \text{ on } \Gamma^{fs}. \quad (2.56)$$

Here, additional contributions by surface tension at the fluid-structure interface are neglected. Since in case of fluid-structure interaction commonly the Eulerian — or ALE, cf. Sec. 3.5.4 — viewpoint is applied on fluid side and the Lagrangian one on structural side,

the coupling condition needs to be adapted by using relation (2.27):

$$\mathbf{T}^f \mathbf{n} = \frac{1}{J} \mathbf{F} \mathbf{S}^s \mathbf{N} \text{ on } \Gamma^{fs}. \quad (2.57)$$

As mentioned before, the same condition can be applied in case multiphase flows are involved. However, surface tension causing a pressure jump over the interface needs to be taken into account, which is usually neglected in case of solids. The mutual dependency/dynamic coupling condition on the fluid-fluid interface can be modeled by applying the Young-Laplace equation [6]:

$$(\mathbf{T}^{f_I} - \mathbf{T}^{f_{II}}) \mathbf{n} = \gamma \kappa \mathbf{n} \text{ on } \Gamma^{f_I f_{II}}, \quad (2.58)$$

where γ is the surface tension coefficient and κ the curvature of the surface $\Gamma^{f_I f_{II}}$. The outer boundary Γ_{out} is actually separating the observed domain from its environment. Hence, boundary conditions similar to the coupling conditions need to be defined. Since the external surface can be handled like internal boundaries, kinematic and dynamic constraints need to be prescribed at the interface.

By the definition of coupling conditions, the theoretical background of the FSI problem is completed. In macroscopic view, the problem is just a continuum mechanical problem with a discontinuous material change at a clearly defined surface. At that surface, the observed domain can be split into sub-bodies by requiring that the coupling conditions are fulfilled on the interface. The equations for the subproblems can be derived separately, i.e., by using different viewpoints and, of course, separate constitutive models. Later on, this divide-and-conquer strategy is also utilized to derive the applied staggered numerical solution scheme. But first, numerical schemes for the single-field problems are introduced in the next section, before these schemes are combined in a numerical scheme for the overall problem.

3 Spline-based methods for fluid and structural mechanics

In the previous section, the physical foundation of FSI problems was introduced based on continuum mechanics. In general, the resulting partial differential equations (PDEs) are not analytically solvable anymore. Thus, numerical methods allowing to solve these equations in a discrete manner are required. Literature offers a wide variety of numerical methods, such as finite elements, finite volumes, finite differences or even mesh-less methods. In all cases, the PDEs have to be discretized in both space and time. For that, different strategies can be applied, see Fig. 3.1. In case of *Method of Lines*, space is discretized first and time afterwards. The *Method of Rothe* is following exactly the opposite way. *Space-Time Methods* allow for a discretization of both directions simultaneously. In general, all these discretization strategies can be applied to the equations of motion, cf. 2.1, independent of the chosen viewpoint.

Within the present thesis, all methods are based on the finite element method, which is briefly summarized in Sec. 3.2. While the elastodynamic problem is solved following method of lines combining finite elements in space and finite differences in time, the fluid problem is solved with the space-time finite element method. In both cases, the discretization relies on the variational formulation and finite-dimensional function spaces.

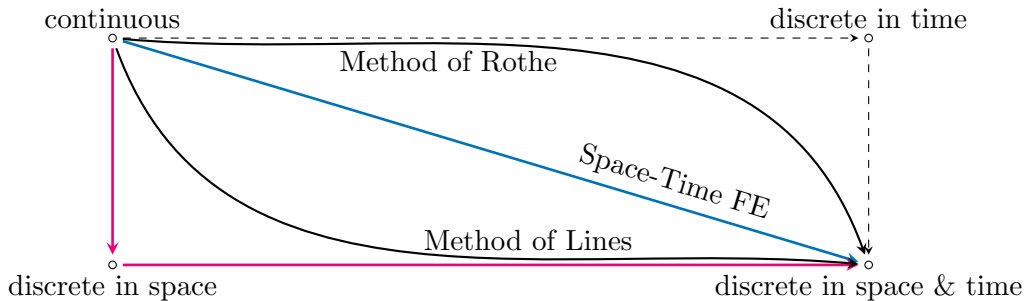


Figure 3.1: General discretization schemes; Method of Lines applied to elastodynamics (red); Space-Time FE applied to fluid dynamics (blue).

The concept of these spaces is introduced in Sec. 3.3, variational formulations for both problems are given in Sec. 3.4 and Sec. 3.5. Since the numerical methods involve splines, they are introduced first in the following.

3.1 Splines

In the past, splines were used by draftsmen as an engineering tool for construction of ships, cars, airplanes, etc. In these particular applications, splines were elastic strips made of, e.g., wood, that were spanned between two defined points and formed to get the smoothest possible shape restricted by given weights and nails. In physical sense, the draftspline has taken the shape that minimizes the elastic energy with respect to the applied constraints.

With direct reference to the designer's draftspline, the concept of splines as a piecewise polynomial, parametric function was introduced by Schoenberg in 1946 [134]. In the late 1950s the concept of splines was applied to geometry representation. Thus, it is not surprising that the further development of mathematical splines was driven later by computer-based construction and design, strongly influenced by engineers and mathematicians like de Casteljau, Bezier and de Boor. Today, the mathematical splines have nearly completely replaced the draftsplines, and are now the standard tool in computer-aided design (CAD). A detailed overview about the history of splines is given in [43].

In recent years, splines were getting more and more popular for solving PDEs in sense of isogeometric analysis, initiated by Hughes et. al in 2005 [89]. Here, splines are acting as finite-dimensional interpolation functions for finite elements. That is also how they will be used in the presented thesis. Since the thesis is focusing on *non-uniform rational B-splines (NURBS)*, that will be introduced in the following.

3.1.1 Non-uniform rational B-splines

Non-uniform rational B-splines (NURBS) were introduced by Versprille [155], based on the work of Riesenfeld [130] on B-Splines. In the following, the general concept of NURBS is introduced for a curve, and extended in a common way to higher dimensional manifolds as surfaces and volumes. A NURBS curve $\mathbf{x}(\theta)$ of degree p is defined

as

$$\mathbf{x}(\theta) = \sum_{i=0}^{n_{cp}} \mathbf{X}_i R_i^p(\theta), \quad (3.1)$$

which is a parametric function of the parametric coordinate θ , composed by a linear combination of piecewise rational basis functions, the NURBS basis functions R_i^p :

$$R_i^p(\theta) = \frac{\omega_i B_i^p(\theta)}{\sum_{\hat{i}=0}^{n_{cp}} \omega_{\hat{i}} B_{\hat{i}}^p(\theta)}. \quad (3.2)$$

Here, $\mathbf{X}_i$ are the coordinates of the control points, that span the control polygon as a convex hull around the curve. The dimensions of $\mathbf{X}_i$ and $\mathbf{C}$, respectively, depend on the Euclidian space the curve is embedded in. The ω_i are the corresponding weights of the control points and $B_i^p(\theta)$ the B-spline basis functions of degree p . They are defined recursively by the Cox-de Boor formula:

$$B_i^0(\theta) = \begin{cases} 1 & \text{if } \theta \in [\theta_i, \theta_{i+1}[\\ 0 & \text{elsewhere,} \end{cases} \quad (3.3)$$

$$B_i^k(\theta) = \frac{\theta - \theta_i}{\theta_{i+k} - \theta_i} B_i^{k-1}(\theta) + \frac{\theta_{i+k+1} - \theta}{\theta_{i+k+1} - \theta_{i+1}} B_{i+1}^{k-1}(\theta). \quad (3.4)$$

θ_i are referred to as knots. They are ordered monotonically increasing $\theta_0 \leq \theta_i \leq \theta_{i+1} \leq \theta_n$, defining the knot vector Ξ

$$\Xi = \{\underbrace{\theta_0, \dots, \theta_0}_{p+1}, \theta_{p+1}, \dots, \theta_{n_{cp}-p-1}, \underbrace{\theta_n, \dots, \theta_n}_{p+1}\}.$$

The multiplicity of the first and last knot in the sense of an open knot vector leads to the interpolatory property of NURBS at the first and last control point of a curve.

Since each NURBS basis function has a compact support — the interval where the function is not equal to zero — the influence of each basis function R_i^p is restricted to the specific interval $[\theta_i, \theta_{i+p+1}[$ in the parameter space. Hence, at most $p+1$ basis functions are non-zero on a knot span $[\theta_i, \theta_{i+1}[$, which is an important property with respect to efficient assembly strategies in IGA. The basis functions are always positive and, additionally, the sum of all basis functions evaluated at a parametric coordinate θ gives 1, which is known as partition of unity. The basis functions are p times continuously differentiable between knots. However, the continuity is changing with multiplicity of single knots within the knot vector. In that case, the continuity is decreasing by k at the individual

knot, where k is the knot multiplicity.

Based on the properties of the basis functions, properties of the curve itself can directly be deduced. First, all properties of continuity can directly be transferred to the NURBS curve. For example, the differentiability of the curve is defined by the multiplicity of a knot. Based on the locality of the NURBS basis functions, the change of a control point $\mathbf{X}_i$ affects only the curve on interval $[\theta_i, \theta_{i+p+1}[$. An example curve is given in Fig. 3.2–Fig. 3.5.

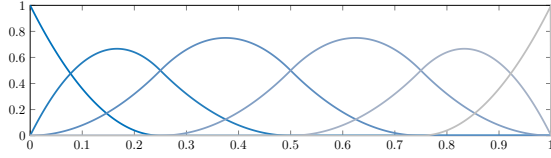


Figure 3.2: NURBS basis functions

$$\begin{aligned}\Xi &= [0 \quad 0 \quad 0 \quad 0.25 \quad 0.5 \quad 0.75 \quad 1 \quad 1 \quad 1] \\ X &= [0 \quad 0.5 \quad 1.5 \quad 2.5 \quad 3.5 \quad 4] \\ Y &= [1 \quad 1.5 \quad 1.5 \quad 0.5 \quad 0.5 \quad 1] \\ \omega &= [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]\end{aligned}$$

Figure 3.3: Knot vector, control point coordinates and weights

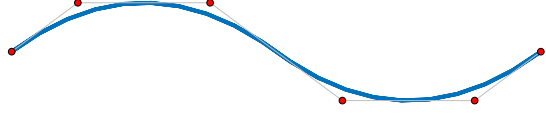


Figure 3.4: Control polygon and related NURBS curve



Figure 3.5: Position of knots evaluated on NURBS curve

NURBS surfaces and volumes can be constructed by tensor products of NURBS curves, resulting in multivariate functions. Consequently, the basis functions $R_{ij}^{p,q}(\theta, \eta)$ for NURBS surfaces are defined as

$$R_{i,j}^{p,q}(\theta, \eta) = \frac{\omega_{i,j} B_i^p(\theta) C_j^q(\eta)}{\sum_{\hat{i}=0}^{n_{cp}} \sum_{\hat{j}=0}^{m_{cp}} \omega_{\hat{i},\hat{j}} B_{\hat{i}}^p(\theta) C_{\hat{j}}^q(\eta)}, \quad (3.5)$$

and by another tensor product, the NURBS basis function for volumes as:

$$R_{i,j,k}^{p,q,r}(\theta, \eta, \zeta) = \frac{\omega_{i,j,k} B_i^p(\theta) C_j^q(\eta) D_k^r(\zeta)}{\sum_{\hat{i}=0}^{n_{cp}} \sum_{\hat{j}=0}^{m_{cp}} \sum_{\hat{k}=0}^{l_{cp}} \omega_{\hat{i},\hat{j},\hat{k}} B_{\hat{i}}^p(\theta) C_{\hat{j}}^q(\eta) D_{\hat{k}}^r(\zeta)}. \quad (3.6)$$

The properties mentioned above for NURBS basis functions and NURBS curves can

directly be transferred to these multivariate versions.

As for NURBS curves, with each combination of the indices used as subscripts of R , both, one basis function and one control point, are related. Omitting the superscript for the polynomial degrees, a common formulations for curves, surfaces and volumes can be given as

$$\mathbf{x}(\boldsymbol{\theta}) = \sum_{A=0}^{n_{cp}} R_A(\boldsymbol{\theta}) \mathbf{X}_A, \quad (3.7)$$

while using the subscript A as a unique identifier of the control points and basis functions.

More details on NURBS can be found in [43, 125], also including efficient algorithms for, e.g., the evaluation of NURBS, refinement and geometrical operations. An open-source library for different spline types is provided by CATS [70] and was already used in [79, 140, 168].

3.2 Introduction to finite element methods

The numerical methods applied in this thesis are different kinds of finite element methods, following all the same steps: 1) a variational formulation over the whole domain Ω , 2) a triangulation $\mathcal{T}$ decomposing the domain into non-overlapping subdomains Ω_e of finite size, and 3) the choice of a proper finite-dimensional interpolation space $\mathcal{S}_h$ as subspace of the actual solution space $\mathcal{S}$. A general definition of finite elements can be found in Ciarlet [40].

The specific finite element methods applied in this thesis are referred to in literature as (linear Lagrangian) *Finite Elements* [98, 164], *NURBS-enhanced Finite Elements* [136, 137] and *NURBS-based isogeometric Finite Elements* [8, 43]. The names are related to the chosen basis functions constructing the particular finite-dimensional function space. The specific functions spaces are introduced in Sec. 3.3.

Linear isoparametric Lagrangian finite elements are still the most widespread finite element class, in literature often just called finite elements. To avoid misunderstandings, the method is named standard finite elements (SFEM) in this thesis. The SFEM function spaces utilize linear Lagrangian polynomials on a triangulation by convex polygons (2D) or polyhedra (3D), e.g., triangles or tetrahedrons. The polynomials are defined as

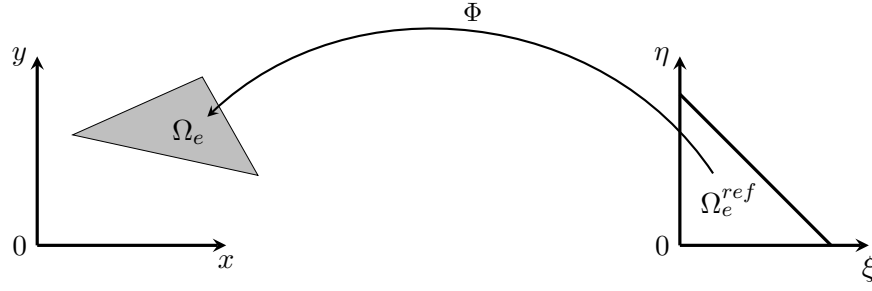


Figure 3.6: Mapping from physical space to reference element

ansatz functions on a unique reference element, that is used for numerical integration on all global elements via coordinate transformation. In case of NURBS-enhanced finite elements (NEFEM) this transformation is slightly modified for all elements in the vicinity of the boundaries by taking the spline-based geometry into account. In case of NURBS-based isogeometric finite elements, the ansatz functions are implicitly defined by the basis functions of the NURBS, elements are defined by the underlying knot vectors. This leads to one major difference between SFEM/NEFEM and IGA: While the basis functions are defined on element level in case of SFEM/NEFEM, they are defined globally in case of IGA. Hence, the SFEM/NEFEM basis functions can be summed up to a global interpolation function while the IGA basis function can be subdivided into element-wise basis functions.

3.2.1 Short remark on numerical integration

The finite element method involves the numerical evaluation of integrals over subdomains Ω_e . To simplify implementation, numerical integration is conducted on simple reference elements in reference coordinates, instead of an individual integration in the physical domain. In general, the integral of the continuous function $\mathbf{f}(\mathbf{x})$ over domain Ω_e can be transformed to an integral over a domain Ω_e^{ref} applying the mapping function $\mathbf{x} = \phi(\boldsymbol{\xi})$:

$$\int_{\Omega_e} \mathbf{f}(\mathbf{x}) \, d\Omega = \int_{\Omega_e^{ref}} \mathbf{f}(\phi(\boldsymbol{\xi})) \, |\det \mathbf{J}| \, d\Omega^{ref} \quad (3.8)$$

with the Jacobian $\mathbf{J} = \frac{\partial \phi(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}$, cf. Fig. 3.6. But even if the integration is computed on simple domains, it still needs to be computed numerically. For that reason, in all solvers

used in this thesis Gaussian integration is applied:

$$\int_{\Omega^{ref}} \mathbf{g}(\boldsymbol{\xi}) \, d\Omega \approx \sum_{l=0}^{n_{GP}} \mathbf{g}(\boldsymbol{\xi}_l) w_l, \quad \mathbf{g}(\boldsymbol{\xi}) = \mathbf{f}(\boldsymbol{\phi}(\boldsymbol{\xi})) |det \mathbf{J}|, \quad (3.9)$$

with Gauss weights w_l and Gauss points $\boldsymbol{\xi}_l$. The number of Gauss points depends on the degree of the polynomial function to integrate [46].

3.2.2 Short remark on linearization of nonlinear problems

Due to the displacement- or velocity-dependent stress tensors, the balance laws are nonlinear. Hence, also the numerical methods lead to a nonlinear system of equations, commonly written in residual form:

$$\mathbf{R}(\mathbf{d}) = \mathbf{0}. \quad (3.10)$$

To solve this problem numerically, the nonlinear system is solved by applying the Newton-Raphson method [27]. By using Taylor's theorem the residual is linearized at iteration $k + 1$ around the know solution k

$$\mathbf{R}(\mathbf{d}^{k+1}) = \mathbf{R}(\mathbf{d}^k) + D\mathbf{R}(\mathbf{d}^k) \Delta \mathbf{d} = \mathbf{0}. \quad (3.11)$$

Assuming that the Jacobian matrix $D\mathbf{R}(\mathbf{d}^k)$ has been computed, the nonlinear problem can be solved by:

$$D\mathbf{R}(\mathbf{d}^k) \Delta \mathbf{d} = -\mathbf{R}(\mathbf{d}^k), \quad (3.12)$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \Delta \mathbf{d}. \quad (3.13)$$

Actually, the convergence of the Newton-Raphson method strongly depends on the chosen initial value. If the current value is close to the final solution, the convergence is of second order [7]. The linear system can be solved with GMRES [131].

3.2.3 Short remark on the numerical domain

The difference between the observed physical domain Ω and the discretized domain Ω^h strongly depends on the applied method. In case of linear isoparametric Lagrangian finite elements, the relation is unique, since the domain and its boundaries are linearly interpolated.

$$\Omega \approx \Omega^h. \quad (3.14)$$

In case of NEFEM and NURBS-based IGA, the smooth boundary of the physical domain is represented by a spline. As long as the initial geometry was generated by a NURBS, both methods are initially geometrically exact. However, as soon as the domain is deformed based on numerical results, e.g., based on the computed results on the solid side in case of FSI, all methods are acting on an approximated state of the geometry. For these reasons, Ω^h is always used to denote the computational domain in the following.

3.3 Finite dimensional ansatz spaces

The variational formulation of the balance laws presented below is discretized in the sense of a finite element method by finite function spaces as an approximation of the continuous solution space $\mathcal{S}_h \subset \mathcal{S}$. These function spaces are defined by finite dimensional interpolation functions. A continuous function can be approximated by a linear combination of all basis functions N_a , reading

$$\mathbf{u} \approx \mathbf{u}^h = \sum_{a=1}^{nn} N_a \mathbf{u}_a, \quad (3.15)$$

with the nodal values $\mathbf{u}_a$. Hence, the numerical properties of a specific FE method depend strongly on the properties of the applied basis functions. A finite element method is called isoparametric, if the unknown function is interpolated with the same ansatz space as the geometry [87],

$$\mathbf{x}^h = \sum_{a=1}^{nn} N_a \mathbf{x}_a, \quad (3.16)$$

while choosing all coordinates $\mathbf{x}_a$ as basis points.

In the following sections, the finite-dimensional ansatz spaces applied in this thesis are introduced. They are all summarized in a finite-dimensional subspace $\mathcal{I}_0$ that is used in the discretization of the variational formulation by $\mathcal{I}_0 \subset C_0(\overline{\Omega^h})$.

3.3.1 Linear isoparametric Lagrangian finite elements

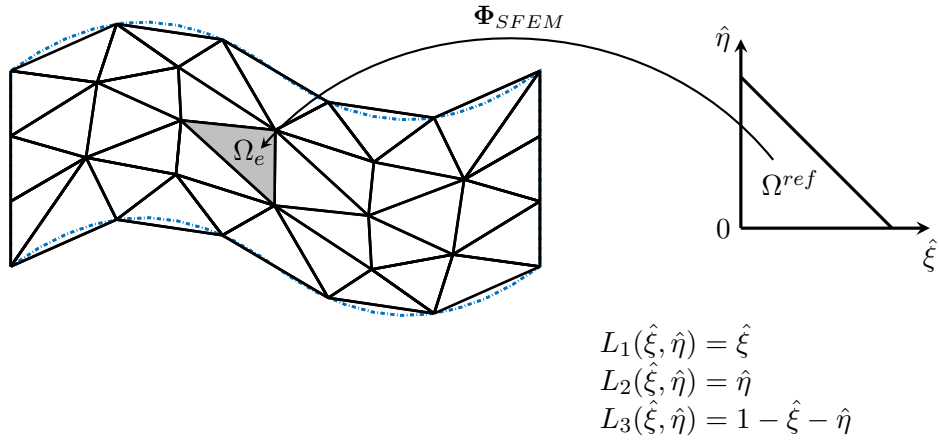


Figure 3.7: Mapping and basis functions for linear isoparametric Lagrangian finite element

In case of standard finite elements, interpolation functions are defined locally on reference elements Ω^{ref} given by a triangulation $\mathcal{T}$. The shape functions of Lagrangian finite elements are constructed by products of one-dimensional Lagrange polynomials, given in barycentric coordinates $\hat{\xi}$ by local basis functions and nodal values on the reference element. The mapping for the geometry of a specific element Ω_e with nen as the number of element nodes is given as

$$\begin{aligned}
 \Phi : \Omega^{ref} &\longrightarrow \Omega_e \\
 \hat{\xi} &\longmapsto \Phi_{SFEM}(\hat{\xi}) := \mathbf{x}_h = \sum_{a=1}^{nen} L_a(\hat{\xi}) \mathbf{x}_a^e.
 \end{aligned} \tag{3.17}$$

Following the isoparametric concept for any approximated solution $\mathbf{d}_h$ leads to

$$\mathbf{u}_h = \sum_{a=1}^{nen} L_a(\hat{\boldsymbol{\xi}}) \mathbf{u}_a^e. \quad (3.18)$$

Concerning SFEM, only linear elements are applied in this thesis, e.g., the linear triangles introduced first by Courant in 1943, see Fig. 3.7. Lagrangian finite elements are interpolative at basis points, in case of the linear version solution and geometry are interpolated linearly in between. The shape functions are strictly positive only in case of linear Lagrangian FE. The linear Lagrangian finite elements are differentiable once within the domain, the derivatives are discontinuous across element boundaries. A detailed derivation of linear and higher order elements based on Lagrangian polynomials can be found, e.g., in [87, 164].

3.3.2 NURBS-based isogeometric analysis

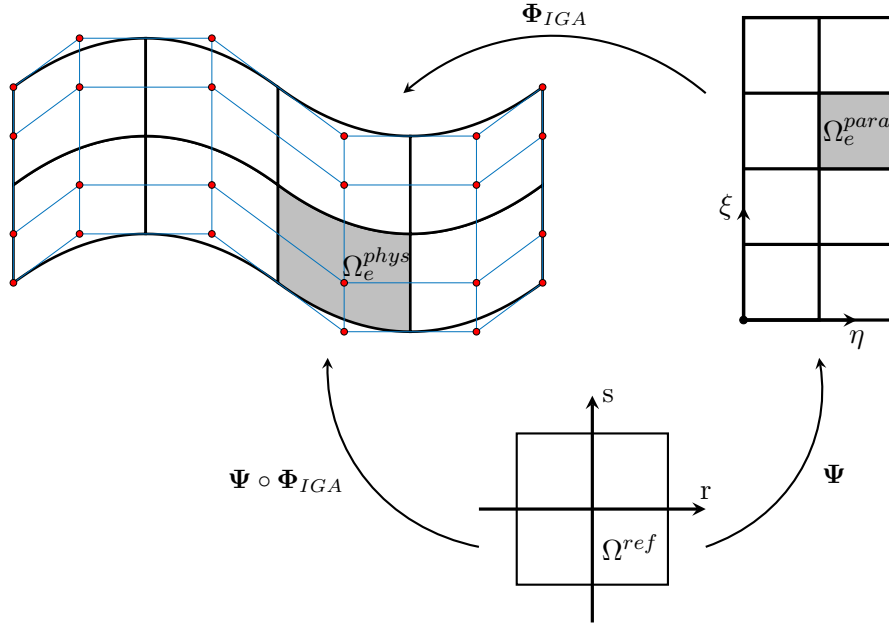


Figure 3.8: Physical space, parametric space and reference element for isogeometric finite elements

Isogeometric Analysis was introduced in 2005 by Hughes et al. [89]. The key idea is to utilize the very same NURBS used for representing geometry in CAD systems also for numerical analysis. Since it was published, several other spline types were included into IGA, e.g., analysis-suitable T-Splines [111,135,163] or locally refined B-Splines [96], but this thesis still focuses on NURBS.

In case of NURBS, two different meshes can be identified, see Fig. 3.8. The control mesh, or control polygon, is spanned by the control points. Although it is not directly involved in the numerical analysis, the geometry is strongly depending on the control polygon. For example, the resulting curve, surface or volume is completely enclosed by the convex hull spanned by the control polygon [125]. The second grid is generated by transferring the multidimensional knot vector from parameter space to the physical domain. This implies the triangulation $\mathcal{T}$ into non-overlapping elements, where the numerical integration is applied on. Thus, the elements Ω_e^{para} can be defined in the parametric domain by

$$\Omega_e^{para} \in \bigotimes_{i=1}^{splinedim} [\theta_{l_i}^i : \theta_{l_i+1}^i], \quad l^i \in [p_i + 1, \dots, n_i], \quad (3.19)$$

with p_i being the polynomial degree and n_i as the number of control points in the parametric direction i . In practice, only non-zero knot spans are taken into account. These elements are mapped to the physical space by using the multivariate tensor product of NURBS as mapping function $\Phi(\boldsymbol{\theta})_{IGA}$:

$$\begin{aligned} \Phi : \Omega_e^{para} &\longrightarrow \Omega_e^{phys}, \\ \boldsymbol{\theta} &\longmapsto \Phi_{IGA}(\boldsymbol{\theta}) := \mathbf{x}_h = \sum_{a=1}^{nn} R_A(\boldsymbol{\theta}) \mathbf{X}_a. \end{aligned} \quad (3.20)$$

Note that in comparison to SFEM, the parametric coordinate $\boldsymbol{\theta}$ is globally defined and all NURBS basis functions are involved in the evaluation of the spline function. However, due to the locality property mentioned in Sec. 3.1, only $p + 1$ basis functions need to be evaluated on each element.

Following the isoparametric concept, the solution space is constructed with the same basis functions R_A of NURBS as the geometry, resulting in the interpolation func-

tion:

$$\mathbf{u}_h(\boldsymbol{\theta}) = \sum_{A=0}^{n_{cp}} R_A(\boldsymbol{\theta}) \mathbf{U}_A, \quad (3.21)$$

while the unknowns $\mathbf{U}_A$ are related to the control points. As in SFEM, the numerical integration is performed on simple reference elements, like the bi-unit square element in case of spline surfaces, cf. Fig. 3.8.

The usage of NURBS basis functions allows to transfer all properties like the polynomial degree and the smoothness, respectively. Of course, an approximation space sufficient for approximating the geometry is usually not large enough for numerical analysis. Refinement strategies like h-, k- and p- refinement are strong tools allowing to decrease the numerical error [44, 89]. This can lead to smaller linear system sizes in comparison to SFEM for a given level of accuracy [22, 45, 90]. A detailed discussion of the mathematical properties of IGA is presented in [8].

The available smooth geometry representation is beneficial for many applications, e.g., for a continuous representation of normal distances in contact mechanics [49, 145] or high continuity between elements in shell analysis [101]. However, the spline is in general not interpolatory, i.e., the nodal basis for geometry and solution is given at the control points. This implies some difficulties during numerical analysis, i.e., if properties of the geometry are needed for numerical analysis at the control points, where the numerical problem is solved. For example, the definition of the normal at control points is not unique, yet needed for shell theory [59] or contact mechanics [116].

Note, the implicit triangulation holds only in case of one-patch NURBS. A discretization by several patches is possible, but needs special handling, e.g., if the triangulation of the subdomains does not conform at the intersections [58]. In this thesis, only one-patch geometries are applied.

3.3.3 NURBS-enhanced finite elements

Both ansatz spaces introduced in the previous section are independent from the spatial dimension in principle. The grid generation for spatial domains with polygons (2D) or polyhedra (3D) of finite size is provided by academic and commercial tools. In contrast, the generation of volume splines is still an ongoing challenge in the IGA community. Hence, despite recent progress in a variety of directions [133, 163], IGA is currently not applicable to problems on complex three-dimensional domains. As an alternative on middle ground between isogeometric analysis and standard finite elements,

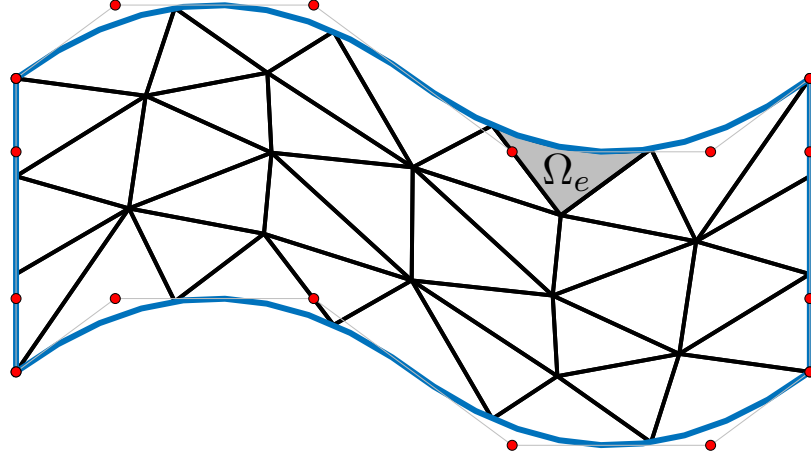


Figure 3.9: Spatial discretization with NEFEM. NURBS is now included in the mapping of all elements in vicinity of the boundaries.

Sevilla, Fernandez-Mendes, and Huerta have proposed the NURBS-enhanced finite element method (NEFEM) [136, 137].

The original formulation of NEFEM

In NEFEM, instead of the full geometry, only its boundary is represented using NURBS. In the interior of the geometry, a standard finite element mesh is utilized, which preserves all advantages of existing meshing algorithms (cf. Fig. 3.9). Note that this approach leads to two different kinds of elements: (1) standard finite elements in the interior and (2) elements with a NURBS edge alongside the boundary. Usually, elements of category (1) will be in the vast majority, keeping the computation very efficient. Through the elements of category (2), the geometry is made available during the process of evaluating the integrals of the appropriate weak form (e.g., of the variational equation (3.69)). The initial integration domain Ω^h is no longer only an approximation of the real domain Ω , cf. sec. 3.2.3. Translated to the numerical implementation, this factor is incorporated through the position of the integration points needed in the finite element method during numerical quadrature: These are determined from the curved NURBS geometry and not from the approximated geometry (cf. Figure 3.10 for a schematic illustration). If we consider boundary integrals — e.g., for the evaluation of the forces the fluid exerts on the structure — the integration points are distributed on the curved geometry.

The unknown function continues to be represented using Lagrangian polynomials. These

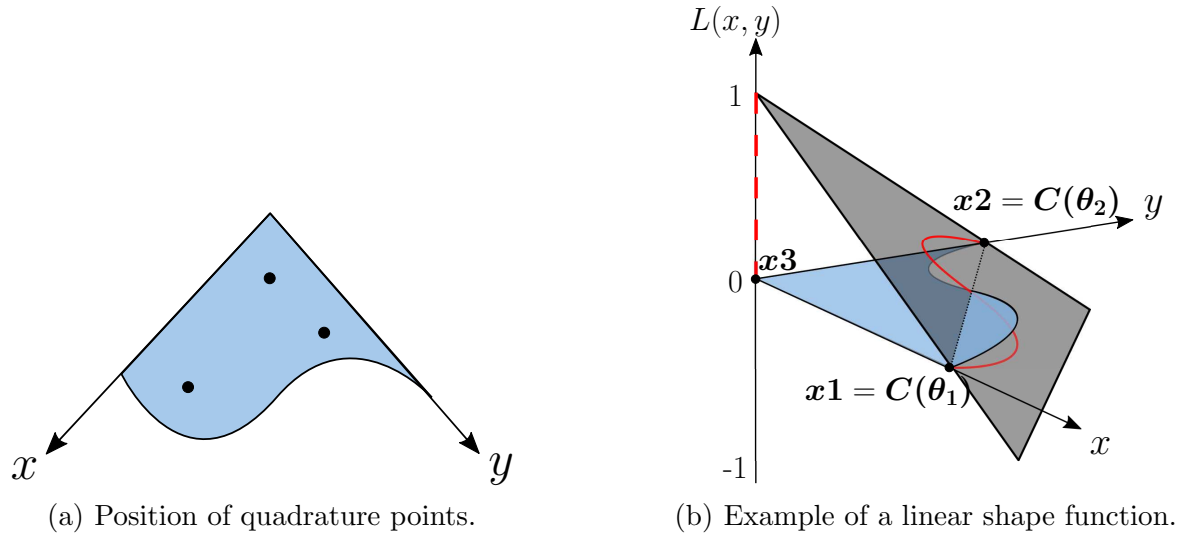


Figure 3.10: Illustration of the principal NEFEM concepts as introduced in [136]. Triangular elements along the boundary of the domain are equipped with one, possibly curved edge, which is represented by a portion of a NURBS curve. This gives access to the geometry. (a) The NEFEM quadrature points x are adapted to the curved triangle shape. (b) Example of a linear shape function in the NEFEM context. Negative values and values larger than 1.0 may occur.

can be of any order, in particular independent of the order of the NURBS basis. In the NEFEM version proposed in [136,137] the choice was made to compute the shape functions in the global coordinates. This variant — termed by the authors as Cartesian FEM — has the advantage that one obtains undistorted polynomial functions in the global elements, an attribute of particular importance when used with polynomials of higher order. The price to pay is an oddity that arises from this representation: negative shape function values as well as values larger than one may occur along the boundary (cf. Figure 3.10).

In terms of implementation, there are two important modifications with respect to a standard finite element method: (1) The definition and evaluation of the shape functions and (2) the placement of the quadrature points. Regarding (1), Sevilla performs all evaluations in the global coordinate system. With respect to (2), the original method uses a bi-unit square reference element; even for triangular elements. This enables the use of tensorized 1D Gauss-Legendre quadrature rules. Furthermore, a clear distinction between the NURBS direction and the standard direction can be made; leading to

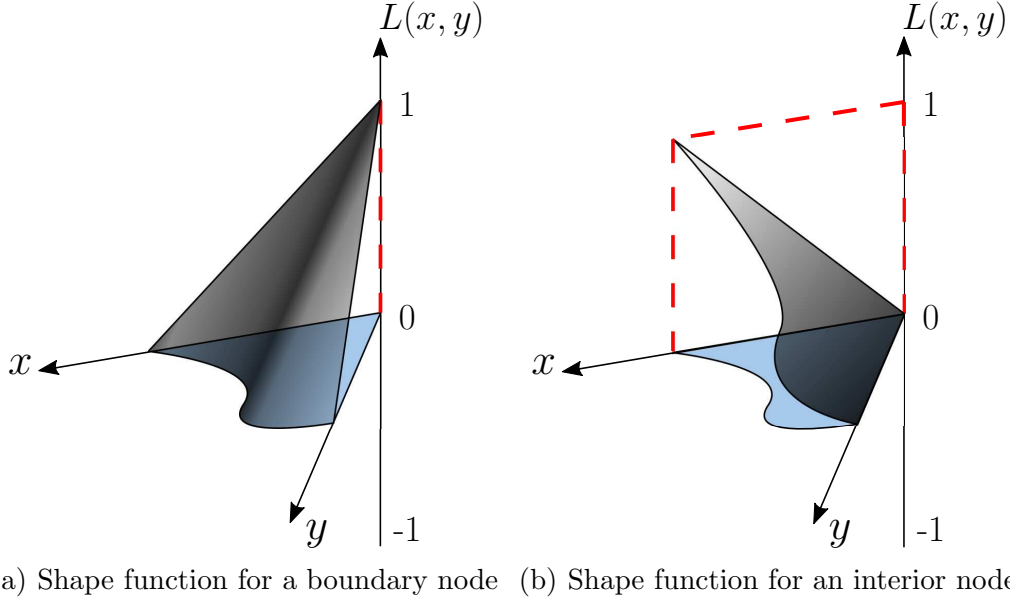


Figure 3.12: Shape functions for elements with curved edges. In (a) a shape function for a boundary node is exemplified. In (b), we see a shape function for the interior node. Note that it is zero along the curved boundary edge.

boundary integrals. In the context of Dirichlet boundaries and boundary integrals, this shape function definition, however, leads to a grave disadvantage: The values along the boundary edge are always affected by nodal values of interior nodes. Since both Dirichlet boundary conditions and the boundary integral connected to the load transfer in FSI are of utter importance to the scenarios considered in this work, it was imperative to modify the shape function definition such that shape functions of inner nodes are zero along the curved edge (cf. Figure 3.12). The shape function definition — inspired by the so-called *p-FEM* in [138] — is performed on the reference element. It is transformed to the global elements using a non-linear mapping Φ , which includes the NURBS definition. This mapping was slightly modified — termed Triangle-Rectangle-Triangle (TRT) mapping —. Both mappings are introduced below.

p-FEM

For p-FEM, the NURBS boundary is associated with the parametric coordinate ξ inside the reference element Ω_{ref} . Therefore, the inner coordinate $\mathbf{x}_3$ maps to $(\hat{\xi} = 0, \hat{\eta} = 1)$.

Based on this, the geometric mapping Φ_{PFEM} reads:

$$\begin{aligned} \Phi_{PFEM} : \Omega^{ref} &\longrightarrow \Omega_e, \\ (\hat{\xi}, \hat{\eta}) &\longmapsto \Phi_{PFEM}(\hat{\xi}, \hat{\eta}) := \frac{1 - \hat{\xi} - \hat{\eta}}{1 - \hat{\xi}} C(\theta(\hat{\xi})) + \frac{\hat{\xi}\hat{\eta}}{1 - \hat{\xi}} \mathbf{x}_1 + \hat{\eta} \mathbf{x}_3. \end{aligned} \quad (3.22)$$

This mapping is linear along the element edges and has a singularity for $\hat{\xi} = 1$.

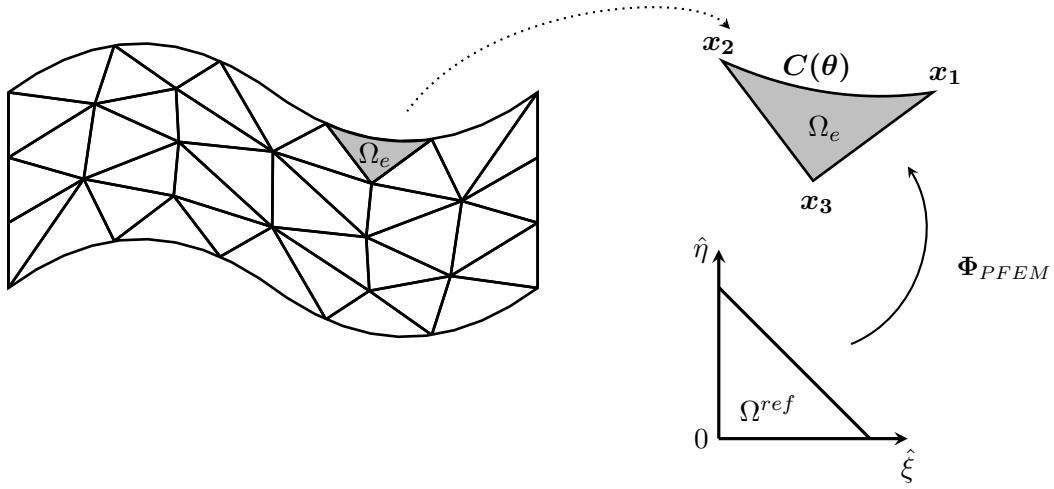


Figure 3.13: The pFEM mapping from the reference triangle to the global triangle with one curved edge.

In [138], no derivation of the p-FEM mapping was given. Furthermore, the author was not able to find any derivation in further literature. Thus, a different triangle-triangle geometric mapping is presented in the following.

Triangle-Rectangle-Triangle geometric mapping

The mapping introduced is based on the idea to construct shape functions by degeneration [79, 164]. With that, a simple mapping from a quadrilateral element to a triangular element can be constructed by coalescing two nodes and adding together their shape functions. The idea is now to use the simple mapping from a reference rectangle onto a reference triangle to apply consecutively the inverse of that mapping and then a mapping between a reference rectangle and the physical element, also used in the original

version of the Cartesian mapping. The derivation of the mapping is illustrated in Figure 3.14. It is used for both the definition of the shape function and the placement of the quadrature points.

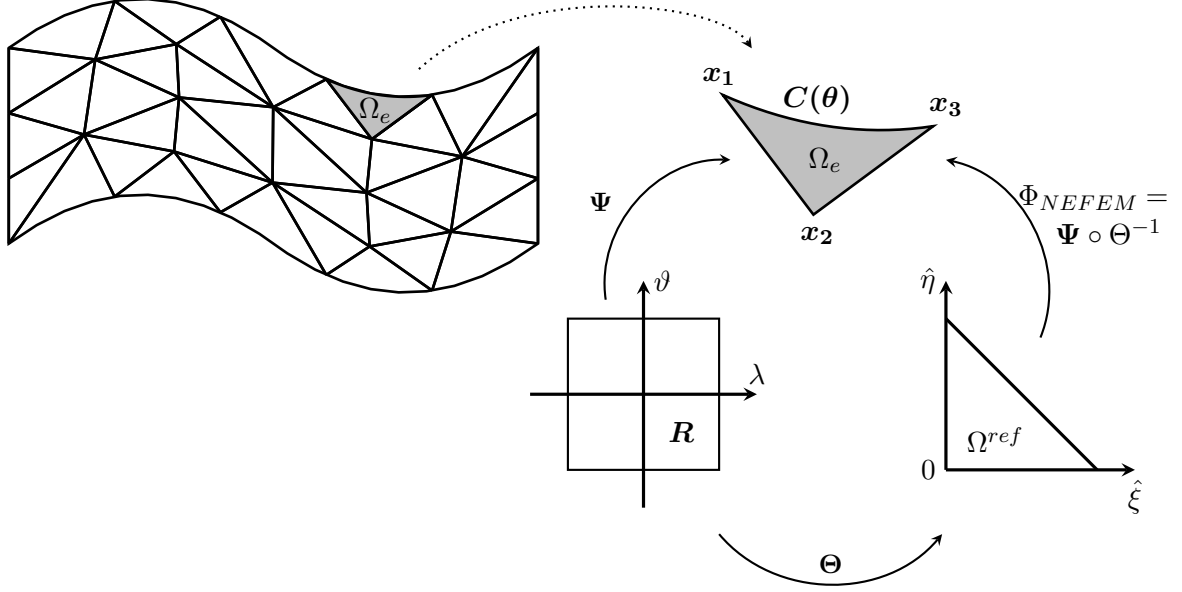


Figure 3.14: The TRT mapping from the reference triangle to the global triangle with one curved edge was derived from the original mapping from the reference bi-unit square to the curved triangle. By incorporating the bi-unit square, it can still be ensured that the NURBS direction and the interior direction are clearly separated, thus leading to straight interior edges even though the boundary edge is curved. Ψ is the mapping utilized in [136], Φ is the new TRT mapping.

In a first step, the mapping Θ from $\mathbf{R}$ onto Ω^{ref} by collapsing two nodes of the quadrilateral is defined as:

$$\begin{aligned} \Theta : \mathbf{R} &\longrightarrow \Omega^{ref} \\ \Theta : \hat{\xi} &= \frac{1}{4} (1 + \lambda) (1 - \vartheta) \\ \hat{\eta} &= \frac{1}{4} (1 - \lambda) (1 - \vartheta). \end{aligned} \tag{3.23}$$

The same strategy is applied in the original mapping used in Cartesian NEFEM between

the reference quadrilateral and the physical element:

$$\begin{aligned}\Psi : \mathbf{R} &\longrightarrow \Omega_e, \\ (\lambda, \vartheta) &\longmapsto \Psi(\lambda, \vartheta) := 0.5(1 - \vartheta) \mathbf{C}(\theta(\lambda)) + 0.5(1 + \vartheta) \mathbf{x}_2,\end{aligned}\tag{3.24}$$

see Eq. (3.26). Here, a linear mapping of the NURBS segment is involved, associating the NURBS boundary of Ω_e to the edge $(1 - \hat{\eta} - \hat{\xi})$ on Ω^{ref} . The inverse mapping Θ^{-1} becomes:

$$\begin{aligned}\Theta^{-1} : \Omega^{ref} &\longrightarrow \mathbf{R}, \\ \Theta^{-1} : \lambda &= 1 - \frac{2\hat{\eta}}{\hat{\xi} + \hat{\eta}}, \\ \vartheta &= 1 - 2(\hat{\xi} + \hat{\eta}).\end{aligned}\tag{3.25}$$

Hence, Θ^{-1} can be combined with Ψ , defined in Eq. (3.26), to define an alternative mapping for Φ_{TRT} :

$$\begin{aligned}\Phi_{TRT} &= \Theta^{-1} \circ \Psi : \Omega^{ref} \longrightarrow \Omega_e \\ (\hat{\eta}, \hat{\xi}) &\longmapsto \Phi_{TRT}(\hat{\eta}, \hat{\xi}) = (1 - \hat{\eta} - \hat{\xi})\mathbf{x}_2 + (\hat{\eta} + \hat{\xi})\mathbf{C}\left(\frac{\Theta_1 \hat{\eta} + \Theta_3 \hat{\xi}}{\hat{\eta} + \hat{\xi}}\right).\end{aligned}\tag{3.26}$$

Here, $\hat{\eta}$ and $\hat{\xi}$ denote the coordinates in the reference triangle. $\mathbf{x}_2$ is the global coordinate of the interior node and $\Theta_{1,3}$ are the NURBS coordinates of the FE boundary nodes.

Figure 3.15 compares the p-FEM mapping from [136] with the mapping in (3.26). It is obvious, that in case of the TRT mapping, the relative distance of an arbitrary point on a straight edge between the boundary and the interior node is kept constant. In case of the pFEM mapping a comparable property cannot be identified. Note that the p-FEM mapping has a singularity at one of the boundary nodes, thus excluding this point in case of a boundary integral. The TRT mapping has its singularity at the interior node. Further comparisons between the p-FEM mapping and the TRT mapping can be found in [76].

As in IGA, NEFEM benefits from the smooth geometry at the boundary. Geometrical properties of the domain, e.g., normals, tangents, the curvature, etc., can be computed exactly. Since linear interpolation functions are still used to construct the solution space,

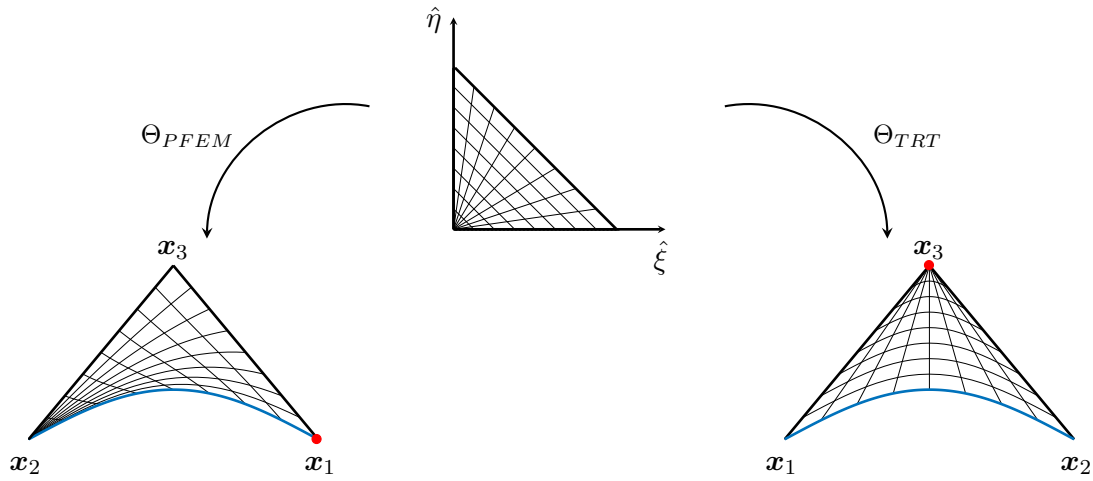


Figure 3.15: Comparison of the p-FEM and the TRT mapping for an element with one circular edge, discontinuities are marked by a red dot.

higher order derivatives of the solution are not available and the convergence order still depends on the applied ansatz space.

Remark on parameterization

NEFEM needs a linear parameterization of a NURBS. Otherwise, the linear mapping along the spline leads to a shifted distribution of quadrature points within the elements, which affects the accuracy of the numerical integration. A similar effect was observed by Cottrell for IGA [44]. Additionally, geometrical discontinuities, i.e., kinks, lead to a discontinuous mapping. This can be counteracted by splitting elements at the kink. Same holds for elements that are enclosing patch boundaries or even connecting points of closed splines.

Remark on other spline types

Since only the geometric information of the NURBS is used for the mapping, the NURBS-enhanced finite element method might be extended to any other spline type, or function providing geometrical information, in terms of a *geometrically enhanced finite element method*.

3.4 Numerical method for the elastodynamic problem

Studying elastodynamic problems, one is usually interested in deformations and internal stresses. Since the movement of particles within a deforming elastic solid is relatively small in comparison to the movement of fluid particles, the Lagrangian viewpoint is commonly used in solid mechanics. Here, the specific elastodynamic problem to solve is achieved by closing the balance laws in material description, cf. Tab. 2.1, with the hyper-elastic material model introduced in Sec. 2.1.4. Since the balance of angular momentum equation (2.32) is implied by the symmetry of the stress tensor (2.39) and in absence of temperature changes, the elastodynamic problem can be solved by discretizing the momentum equation (2.30) only. By formulating the variational approach with respect to a stress-free initial configuration as a total Lagrangian approach, the continuity equation is fulfilled directly. The boundary value problem is completed with the boundary conditions

$$\mathbf{d} = \mathbf{g} \text{ on } (\Gamma_{\mathbf{X}})_g, \quad (3.27a)$$

$$\mathbf{S}\mathbf{N} = \mathbf{h} \text{ on } (\Gamma_{\mathbf{X}})_h, \quad (3.27b)$$

and initial conditions

$$\mathbf{d}(\mathbf{X}, 0) = \mathbf{d}_0, \quad (3.28)$$

$$\frac{\partial \mathbf{d}(\mathbf{X}, 0)}{\partial t} = \dot{\mathbf{d}}_0. \quad (3.29)$$

Here, the boundary $(\Gamma_{\mathbf{X}})_g \subset \Gamma_{\mathbf{X}}$ represents regions on the boundary, where Dirichlet boundary conditions, like local constraints, are applied. The boundary $(\Gamma_{\mathbf{X}})_h \subset \Gamma_{\mathbf{X}}$ stands for the part of the boundary where Neumann boundary conditions, i.e., external forces, are acting.

In the following, the numerical method to solve the resulting elastodynamic problem is introduced. It is a semi-discrete method for a continuum mechanical problem in the Lagrangian viewpoint, following methods of lines.

3.4.1 Variational formulation

The variational formulation is achieved by multiplication of the momentum equation in the Lagrangian viewpoint (2.30) with a weighting function $\mathbf{w} \in \mathcal{V}$ and integration about the material domain $\Omega_{\mathbf{X}}^h$. Applying the divergence theorem and using the boundary conditions (3.27) the variational problem is given by:

$$\int_{\Omega_{\mathbf{X}}} \mathbf{w} \cdot \rho \frac{D^2 \mathbf{d}}{Dt^2} + \nabla_{\mathbf{X}} \mathbf{w} : \mathbf{F} \mathbf{S} + \mathbf{w} \cdot \rho \mathbf{b} \, d\Omega = + \int_{\Gamma_{\mathbf{X}}} \mathbf{w} \cdot \mathbf{h} \, d\Gamma. \quad (3.30)$$

Due to the highest order of the spatial derivatives of both the solution $\mathbf{d}$ and the weighting function $\mathbf{w}$ can be assumed to be part of the Sobolev space $H^1(\Omega_{\mathbf{X}})$. Additionally, the weighting function $\mathbf{w}$ has to satisfy the homogenous boundary condition on $(\Gamma_{\mathbf{X}})_g$. Thus, the test function belongs to the space $\mathcal{V}$:

$$\mathcal{V}_{\mathbf{w}} = \{\mathbf{w} \in H^1(\Omega_{\mathbf{X}}) \mid \mathbf{w} = \mathbf{0} \text{ on } (\Gamma_{\mathbf{X}})_g\}, \quad (3.31)$$

and the unknown solution $\mathbf{d}$ lies in the solutions space $\mathcal{S}$:

$$\mathcal{S}_{\mathbf{d}} = \{\mathbf{d} \in H^1(\Omega_{\mathbf{X}}), t \in [0, T] \mid \mathbf{d} = \mathbf{g} \text{ on } (\Gamma_{\mathbf{X}})_g\}. \quad (3.32)$$

Note that — in contrast to the test function space $\mathcal{V}$ — the solution space $\mathcal{S}$ varies as a function of time. With both spaces the problem can now be stated as:

Given $(\mathbf{d}_0)$, find $\mathbf{d} \in \mathcal{S}_{\mathbf{d}}$ such that Eq. (3.30) is fulfilled $\forall \mathbf{w} \in \mathcal{V}_{\mathbf{w}}$.

The problem is discretized in space by applying the finite dimensional subspaces introduced in Sec. 3.3,

$$\mathcal{S}_{\mathbf{d}}^h = \{\mathbf{d} \mid \mathbf{d} \in H^1(\Omega_{\mathbf{X}}^h), \mathbf{d}|_{\Omega_e} \in \mathcal{I}_0(\Omega^e) t \in [0, T] \forall e \text{ and } \mathbf{d} = \mathbf{g} \text{ on } (\Gamma_{\mathbf{X}}^h)_g\} \quad (3.33)$$

$$\mathcal{V}_{\mathbf{w}}^h = \{\mathbf{w} \in H^1(\Omega_{\mathbf{X}}^h) \mid \mathbf{w}|_{\Omega_e} \in \mathcal{I}_0(\Omega^e) \forall e \text{ and } \mathbf{w} = \mathbf{0} \text{ on } (\Gamma_{\mathbf{X}}^h)_g\} \quad (3.34)$$

Find $\mathbf{d}^h \in \mathcal{S}_{\mathbf{d}}^h$ such that $\forall \mathbf{w}^h \in \mathcal{V}_{\mathbf{w}}^h$,

$$\int_{\Omega_{\mathbf{X}}^h} \mathbf{w}^h \cdot \rho \frac{D^2 \mathbf{d}^h}{Dt^2} + \nabla_{\mathbf{X}} \mathbf{w}^h : \mathbf{F}^h \mathbf{S}^h + \mathbf{w}^h \cdot \rho \mathbf{b} \, d\Omega = \int_{\Gamma_{\mathbf{X}}^h} \mathbf{w}^h \cdot \mathbf{h} \, d\Gamma. \quad (3.35)$$

This results in the following system of equation, which is still continuous in time:

$$\mathbf{M} \frac{D^2 \mathbf{d}^h}{Dt^2} + \mathbf{N}(\mathbf{d}^h) + \mathbf{f}_{body} = \mathbf{f}_{ext}. \quad (3.36)$$

Note that, depending on the specific finite interpolation space $\mathcal{I}_0$, the variables $\mathbf{d}^h$ are stored at the grid or control points. While the mass matrix $\mathbf{M}$ is assumed to be constant, the discrete internal forces $\mathbf{N}(\mathbf{d}^h)$ are nonlinearly depending on the displacements in case of a hyperelastic material. The term $\mathbf{f}_{body}$ includes the discrete body forces and $\mathbf{f}_{ext}$ are the discrete external surfaces forces. To solve the system of equation, it has to be discretized in time and, finally, a solution strategy for the fully discrete nonlinear system is needed.

For all two-dimensional test cases, (3.36) is solved with respect to displacements only. This might lead to numerical instabilities, e.g., in case of large time steps. Unphysical coupling between individual strains produces artificial energy that lead to unrealistically stiff system response, referred to as locking. Especially in case of shell elements these can be critical. Detailed discussions about locking effects in shell elements can be found in [23, 24, 127, 156]. These instabilities can be counteracted, e.g., by a corotational formulation for volumetric elements [28, 61, 64] or improved numerical integration for shell elements [1]. Since the thesis is focusing on the benefits of spline-based coupling methods — especially between NEFEM/SFEM and IGA — locking effects are not further discussed.

3.4.2 Time discretization

Following the method of lines, (3.39) needs to be discretized in time. To simplify notation the superscript h for the discrete solution in space is dropped in this section. An improved convergence is achieved, when the general requirements for a stable time integration scheme are fulfilled. On the one hand, damping of high-frequency responses is desired, which are mainly based on numerical errors. On the other hand dissipation of lower frequencies should be minimized to allow to resolve the physical response. The generalized- α -method applied in the presented thesis, introduced by Chung and Hulbert [39], is an unconditionally stable, second-order accurate algorithm for linear problems, that ensures an optimal combination of high- and low-frequency dissipation. Although the method was originally introduced for linear problems it was studied on

nonlinear problems by Kuhl and Crisfield [104]. The derivation of the method is briefly summarized below.

Introducing a generalized mid-point rule for all kinematic variables,

$$\mathbf{a}_{n+1-\alpha_m} = (1 - \alpha_m) \mathbf{a}_{n+1} + \alpha_m \mathbf{a}_n, \quad (3.37)$$

$$e_{n+1-\alpha_f} = (1 - \alpha_f) e_{n+1} + \alpha_f e_n, \quad e \in \{\mathbf{u}, \mathbf{d}, \mathbf{f}, t\}, \quad (3.38)$$

into the space-discrete Cauchy's equation of motion (3.39) results in

$$\mathbf{M} \mathbf{a}_{n+1-\alpha_m} + \mathbf{N}_{n+1-\alpha_f} + \mathbf{f}_{n+1-\alpha_f, body} = \mathbf{f}_{n+1-\alpha_f, ext}. \quad (3.39)$$

Hence, the equation is verified at an intermediate time increment between t^n and t^{n+1} . Kuhl and Crisfield suggest to compute the time-discrete nonlinear internal forces in the following ways [104]:

$$\mathbf{N}_{n+1-\alpha_f} = (1 - \alpha_f) \mathbf{N}(\mathbf{d}_{n+1}) + \alpha_f \mathbf{N}(\mathbf{d}_n) \quad \text{or} \quad \mathbf{N}_{n+1-\alpha_f} = \mathbf{N}(\mathbf{d}_{n+1-\alpha_f}). \quad (3.40)$$

The number of unknown variables can be reduced by applying the Newmark approximation in time, presented first by Newmark [119, 120], under assumption of linear acceleration:

$$\mathbf{u}_{n+1} = \mathbf{u}_n + (1 - \gamma) \mathbf{a}_n \Delta t + \gamma \mathbf{a}_{n+1} \Delta t, \quad (3.41)$$

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \mathbf{u}_n \Delta t + \left(\frac{1}{2} - \beta\right) \mathbf{a}_n \Delta t^2 + \beta \mathbf{a}_{n+1} \Delta t^2. \quad (3.42)$$

Changing the Newmark parameters γ and β allows for switching between an explicit and implicit time integration, while changing the stability or the rate of convergence. Solving (3.42) for $\mathbf{a}_{n+1}$ and introducing the resulting equation into (3.37) gives:

$$\mathbf{a}^{n+1-\alpha_m} = \frac{(1 - \alpha_m)}{\beta \Delta t^2} (\mathbf{d}^{n+1} - \mathbf{d}^n) - \frac{1 - \alpha_m}{\beta \Delta t} \mathbf{u}^n - \frac{1 - \alpha_m - 2\beta}{2\beta} \mathbf{a}^n. \quad (3.43)$$

That allows to express the acceleration at t^{n+1} by the unknown displacements and kinematic variables at t^n .

By introducing the following constants:

$$c_0 = \frac{(1 - \alpha_m)}{\beta \Delta t^2}, \quad c_1 = \frac{1 - \alpha_m}{\beta \Delta t}, \quad c_2 = \frac{1 - \alpha_m - 2\beta}{2\beta}, \quad (3.44)$$

the time-discrete equation of motion is often written as:

$$\mathbf{M} (c_0 (\mathbf{d}_{n+1} - \mathbf{d}_n) - c_1 \mathbf{u}_n - c_2 \mathbf{a}_n) + \mathbf{N}_{n+1-\alpha_f} - \mathbf{f}_{n+1-\alpha_f, body} = \mathbf{f}_{n+1-\alpha_f, ext} \quad (3.45)$$

In case, the internal forces are linear, Chung and Hulbert derive a second-order accurate scheme [39] by choosing the Newmark constants as

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f \text{ and } \beta = \frac{1}{4} (1 - \alpha_m + \alpha_f)^2, \quad (3.46)$$

which is unconditionally stable if

$$\alpha_m \leq \alpha_f \leq \frac{1}{2}. \quad (3.47)$$

For a user-specified spectral radius ρ_∞ , optimal damping characteristics are achieved by choosing:

$$\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1} \text{ and } \alpha_f = \frac{\rho_\infty}{\rho_\infty + 1}. \quad (3.48)$$

Note, even if the presented method is unconditionally stable in linear cases, numerical dissipation can violate energy conservation. The topic is discussed by Kuhl and Crisfield [104]. The same paper includes suggestions to improve conservation of energy, linear and angular momentum.

After spatial and temporal discretization, the resulting equation still needs to be linearized, as presented in the following section.

3.4.3 Linearization of the nonlinear problem

In the previous section, the time integration method was introduced by leaving the nonlinear internal forces within the left-hand side untouched, so that the residual is still nonlinear in that term. The nonlinear problem can be solved by applying the Newton-Raphson method introduced in Sec. 3.2.2. By defining the left-hand side of (3.45) as $\mathbf{T}(\mathbf{d})$ and its right-hand side as $\mathbf{T}$, the equation can be written in residual form:

$$\mathbf{R}(\mathbf{d}) = \mathbf{T}(\mathbf{d}) - \mathbf{T} = 0. \quad (3.49)$$

To simplify the notation, the current time step $n + 1$ is dropped here. The derivation of the residual can be computed by using the Gateaux derivative

$$DR(\mathbf{d}^k)[\Delta \mathbf{d}] = DN(\mathbf{d}^k)[\Delta \mathbf{d}] + \frac{(1 - \alpha_m)}{\beta \Delta t^2} \mathbf{M} \Delta \mathbf{d}. \quad (3.50)$$

The derivative of the nonlinear internal forces in continuous form leads to the tangential stiffness tensor $\mathbf{K}_t$ [7, 13]:

$$\mathbf{K}_t = DN(\mathbf{d}^k)[\Delta \mathbf{d}] = \int_{\Omega_x} \mathbf{F}^{kT} \nabla_{\mathbf{X}} \mathbf{w} : \mathcal{C} \mathbf{F}^{kT} \nabla_{\mathbf{X}} \Delta \mathbf{d} + \nabla_{\mathbf{X}} \mathbf{w} : \nabla_{\mathbf{X}} \Delta \mathbf{d} \mathbf{S}^k \, d\Omega. \quad (3.51)$$

The first term can be interpreted as the material contribution, the second as the geometric contribution to the nonlinear internal forces. By spatial discretization we get the tangential stiffness matrix $\mathbf{K}^k$. This leads finally to the effective tangential matrix:

$$DR(\mathbf{d}^k)[\Delta \mathbf{d}] \approx \mathbf{K}^k \Delta \mathbf{d} = \left(\mathbf{K}_t^k + \frac{(1 - \alpha_m)}{\beta \Delta t^2} \mathbf{M} \right) \Delta \mathbf{d}. \quad (3.52)$$

It can be used within the iterative solver (3.13). To improve the starting vector $\mathbf{d}^0$, it can be computed by the Newmark predictor (3.42) under assumption of linear acceleration.

Within the presented solution strategy for the nonlinear system of the elastodynamic problem, only the nonlinearity of the left-hand side or stiffness matrix was regarded. However, potential nonlinearities in the right-hand side due to the displacements can strongly influence the convergence order and stability of the overall algorithm [118]. Unfortunately, these nonlinearities appear especially in FSI problems, where the fluid loads are in mutual dependency to the current deformation of the solid. Since these additional nonlinearities introduce a strong dependency between the two single-field problems, it is usually not practical to involve them in partitioned approaches [157], that are applied in this thesis.

3.5 Numerical methods for fluid mechanical problem

While numerical methods for solid mechanics are dominated by finite element methods, the numerical methods for fluid mechanics are dominated by finite volume methods.

Due to the large displacement of particles, an Eulerian viewpoint is preferred, and the derivation of finite volumes is quite natural through an evaluation of Reynolds transport theorem on fixed finite volumes. However, since the finite element method is neither depending on the kinematic viewpoint nor on the constitutive laws it can be also applied to fluid dynamics. In this thesis, the spatial and temporal discretization is conducted by following the space-time discretization. The key idea is to discretize the space-time continuum Q — assembled by all spatial domains for a given time-span — at once. The applied method was derived initially to solve elastodynamic problems [92], transferred later to incompressible Euler and Navier-Stokes equations on rigid domains [139] and extended as the *deformable spatial domain/ stabilized space time method* to deforming domains [146, 147].

Before the method is presented, the boundary value problem for fluid dynamics is completed with the boundary conditions

$$\mathbf{u} = \mathbf{g} \text{ on } (\Gamma_{\mathbf{x},t})_g, \quad (3.53a)$$

$$\mathbf{T}\mathbf{n} = \mathbf{h} \text{ on } (\Gamma_{\mathbf{x},t})_h \quad (3.53b)$$

and the initial condition

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0. \quad (3.54)$$

$$(3.55)$$

As for the elastodynamic problem, the boundaries $(\Gamma_{\mathbf{x},t})_g$ and $(\Gamma_{\mathbf{x},t})_h$ represent again subsets of the boundary $\Gamma_{\mathbf{x}}$, where Dirichlet or Neumann boundary conditions are applied. In case of incompressible fluids, Dirichlet boundary conditions are prescribing the velocities, e.g., values at the inflow or on no-slip walls, and Neumann conditions are prescribed stresses. The new subscript t was added, since it is already assumed that the boundary is allowed to move. Note that with semi-discrete approaches, a time-dependent domain requires special treatment of the balance laws in sense of a Arbitrary-Lagrangian-Eulerian viewpoint. However, in space-time discretizations this is implied; a more detailed discussion about that topic is given in Sec. 3.5.4.

3.5.1 Variational formulation

As for of the elastodynamic problem, the momentum equation is multiplied by a weighting function $\mathbf{v}$ for velocity, and the continuity equation, also called incompressibility

condition, by a weighting function q for pressure. Both equations are added up and integrated over the space-time domain Q with boundary P . By applying the divergence theorem, the space-time variational formulation of the incompressible Navier-Stokes equations can be formulated as follows:

Given $\mathbf{u}_0$, find $\mathbf{u} \in \mathcal{S}_u$ and $p \in \mathcal{S}_p$ such that $\forall \mathbf{v} \in \mathcal{V}_v, \forall q \in \mathcal{V}_p$:

$$\begin{aligned} \int_Q \mathbf{v} \cdot \rho^f \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{f} \right) + D(\mathbf{v}) : \hat{\mathbf{T}}^f(p, \mathbf{u}) \, dQ + \int_Q q \nabla \cdot \mathbf{u} \, dQ \\ = \int_P \mathbf{v} \cdot \mathbf{h} \, dP. \end{aligned} \quad (3.56)$$

The first term together with the right-hand side are the weak form of the momentum equation, the second term is the weak form of the continuity equation. As in the elastodynamic problem, both, the solution $\mathbf{u}$ and the weighting function $\mathbf{v}$ can be assumed to be part of the Sobolev space $H^1(Q)$. Analogously, the test function for velocities belongs to the space $\mathcal{V}_v$:

$$\mathcal{V}_v = \{ \mathbf{v} \in H^1(Q) \mid \mathbf{v} = \mathbf{0} \text{ on } (P)_g \}, \quad (3.57)$$

the unknown velocity $\mathbf{u}$ lies in the solutions space $\mathcal{S}_u$:

$$\mathcal{S}_u = \{ \mathbf{u} \in H^1(Q) \mid \mathbf{u} = \mathbf{g} \text{ on } (P)_g \}, \quad (3.58)$$

and the trial and weighting function space for the pressure can be stated as:

$$\mathcal{S}_p = \mathcal{V}_p = \{ p \in H^1(Q) \}. \quad (3.59)$$

3.5.2 Deforming spatial domain/stabilized space-time method

Instead of discretizing the whole space-time domain at once, we can utilize the property of physical time-dependent problems that information flows in the direction of positive time only [92]. Hence, the time interval $[0, T]$ is subdivided into subintervals $I_n = (t_n, t_{n+1})$, with t_n and t_{n+1} representing an ordered series of time levels $0 = t_0 < t_1 < t_2 < \dots < t_N = T$. With the spatial domain $\Omega_n = \Omega_{t_n}$ and the boundary $\Gamma_n = \Gamma_{t_n}$, the time slab Q_n can be defined, which is bounded by Ω_n, Ω_{n+1} and the surface P_n , based

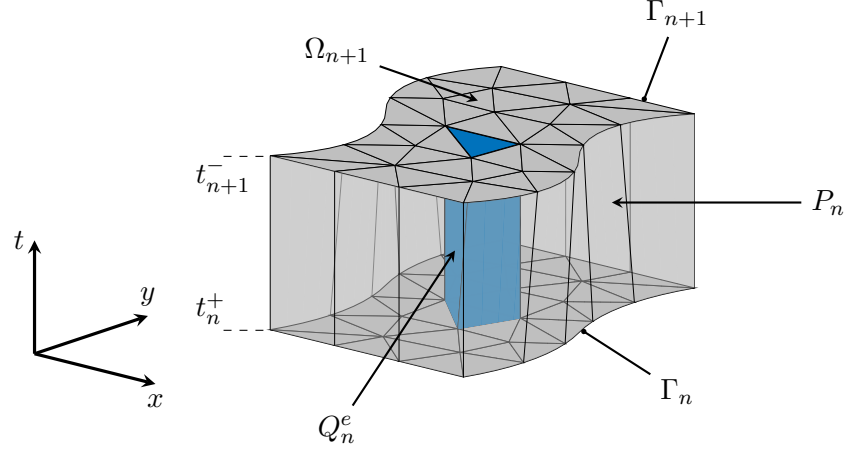


Figure 3.16: Discretized time slab Q_n

on the time varying boundary Γ_t^f for I_n , cf. Fig. 3.16.

Related to that, the solution and weighting function space can be adapted to the discretization in time for each time slab

$$(\mathcal{V}_v^h)_n = \{\mathbf{v}^h \in H^{1h}(Q_n^h) \mid \mathbf{v}^h = \mathbf{0} \text{ on } (P_n^h)_g\}, \quad (3.60)$$

$$(\mathcal{S}_u^h)_n = \{\mathbf{u}^h \in H^{1h}(Q_n^h) \mid \mathbf{u}^h = \mathbf{g} \text{ on } (P_n^h)_g\} \quad (3.61)$$

$$(\mathcal{S}_p^h)_n = (\mathcal{V}_p^h)_n = \{p \in H^{1h}(Q_n^h)\}. \quad (3.62)$$

Note, together with the subdivision into time slabs Q_n the finite dimensional subspace H^{1h} was introduced without any details. A clear definition with respect to the previously introduced finite interpolation spaces will be given in Sec. 3.5.3.

Due to the information flow in positive time direction, the solution is allowed to be computed in the time slabs in a consecutive manner. The problem is solved sequentially for each space-time slab, starting with:

$$(\mathbf{u}^h)_0^+ = \mathbf{u}_0. \quad (3.63)$$

The continuity of the unknown velocity is enforced between the time slabs in a weak sense, following the idea of discontinuous Galerkin in time [92, 98]. In the equations

below, the following notation is used:

$$(\mathbf{u}^h)_n^\pm = \lim_{\epsilon \rightarrow 0} \mathbf{u}(t_n \pm \epsilon), \quad (3.64)$$

$$\int_{Q_n^h} \dots dQ = \int_{I_n} \int_{\Omega_t^h} \dots d\Omega dt, \quad (3.65)$$

$$\int_{P_n^h} \dots dP = \int_{I_n} \int_{\Gamma_t^h} \dots d\Gamma dt. \quad (3.66)$$

A discretization of the solution and weighting function spaces, like in the elastodynamic case, would be possible. But it is well known that the standard Galerkin method produces numerical oscillations in case of convection-dominated flows [57]. Refinement can help to avoid node-to-node oscillations, but decreases efficiency of computations. Instead, stabilization methods are preferred, that add artificial diffusion to Galerkin formulation. For simple one-dimensional problems, optimal upwinding schemes can be developed [88], for complex multidimensional problems this is not possible anymore. Based on the one-dimensional ideas, Streamwise-Upwind/Petrov-Galerkin (SUPG) [30, 31] adds diffusion only in flow direction. In this thesis, the Galerkin/Least-squares method [91] is applied, which adds an element-wise weighted least-squares formulation of the original differential equation. Since this additional term vanishes for the exact solution, the method remains consistent. By introducing the residual forms of the momentum equation

$$\mathcal{R}_{\text{MOM}}(\mathbf{u}^h, p^h) = \rho^f \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f} \right) - \nabla \cdot \hat{\mathbf{T}}^f(p^h, \mathbf{u}^h) \quad (3.67)$$

and the continuity equation

$$\mathcal{R}_{\text{CONT}}(\mathbf{u}^h) = \nabla \cdot \mathbf{u}^h, \quad (3.68)$$

a general formulation [41] of the stabilized space-time formulation of the incompressible Navier-Stokes equations on deforming domains can be stated:

Given $(\mathbf{u}^h)_n^-$, find $\mathbf{u}^h \in (\mathcal{S}_u^h)_n$ and $p^h \in (\mathcal{S}_p^h)_n$ such that $\forall \mathbf{v}^h \in (\mathcal{V}_v^h)_n, \forall q^h \in$

$(\mathbf{v}_p^h)_n$:

$$\begin{aligned}
 \int_{Q_n^h} \mathbf{v}^h \cdot \rho^f \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f} \right) + \mathbf{D}(\mathbf{v}^h) : \widehat{\mathbf{T}}^f(p^h, \mathbf{u}^h) \, dQ + \int_{Q_n^h} q^h \nabla \cdot \mathbf{u}^h \, dQ \\
 + \int_{\Omega_n^h} (\mathbf{v}^h)_n^+ \cdot \rho^f ((\mathbf{u}^h)_n^+ - (\mathbf{u}^h)_n^-) \, d\Omega \\
 + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \tau_1 \mathcal{L}_{\text{MOM}}(\mathbf{v}^h, q^h) \cdot \mathcal{R}_{\text{MOM}}(\mathbf{u}^h, p^h) \, dQ \\
 + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \tau_2 \mathcal{L}_{\text{CONT}}(\mathbf{v}^h) \mathcal{R}_{\text{CONT}}(\mathbf{u}^h) \, dQ \\
 = \int_{P_n^h} \mathbf{v} \cdot \hat{\mathbf{h}} \, dP.
 \end{aligned} \tag{3.69}$$

The first two terms are still the same as in the variational problem (3.56), but the integration is now conducted over the discretized time slab Q_n^h . The third term — called jump term — is introduced by applying discontinuous Galerkin in time [92], as mentioned above. The last two terms on the left-hand side are the stabilization terms in residual form. The first one adds stability to the momentum equation, while the second term covers the velocity field. In both cases the residual form of the momentum equation (3.67) and the incompressibility condition (3.68) ensure consistency. The terms $\mathcal{L}_{\text{MOM}}$ and $\mathcal{L}_{\text{CONT}}$ are called L2 product perturbation of the weighting function and will be defined below. The parameters τ_1 and τ_2

$$\tau_1 = \tau_{\text{MOM}} \frac{1}{\rho^f}, \tag{3.70}$$

$$\tau_2 = \tau_{\text{CONT}} \rho^f, \tag{3.71}$$

are locally defined stabilization parameters. These parameters need to be designed by error estimation, convergence and dimensional analysis [139]. Since they were initially derived on advection-diffusion equations, they are usually split into an intrinsic time scale [92] and the remaining scales. A detailed analysis of stabilization parameters for space-time is given in [139]. The specific perturbation of the weighting function in terms of GLS and the related stabilization parameters used in the presented thesis are introduced

below.

GLS/SUPG stabilization

By adding least-squares forms of residuals to Galerkin methods, the stability is aimed to be enhanced without degrading accuracy [91]. The stabilization is consistent, since the exact solution satisfies the equation. It allows for equal order interpolation functions for velocity and pressure, while the Ladyzhenskaya-Babuška-Brezzi condition is circumvented. Violation of the LBB condition can lead to pressure oscillations in Galerkin methods. This was analyzed for the Navier-Stokes equations involving viscous stress, velocity and pressure primary variables by Behr [21]. The perturbations of the weighting function for GLS are given as:

$$\mathcal{L}_{\text{MOM}}(\mathbf{v}^h, q^h) = \rho^f \left(\frac{\partial \mathbf{v}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{v}^h \right) - \nabla \cdot \hat{\mathbf{T}}^f(q^h, \mathbf{v}^h) \quad (3.72)$$

and

$$\mathcal{L}_{\text{CONT}}(\mathbf{v}^h) = \mathcal{R}_{\text{CONT}}(\mathbf{v}^h) = \nabla \cdot \mathbf{v}^h. \quad (3.73)$$

Details on the method and its parameters τ_{MOM} and τ_{CONT} can be found in [20, 124].

3.5.3 Finite dimensional spaces for space-time

The variational problem in the previous section was introduced without any details on the finite dimensional space $H^{1h}(Q_n^h)$. Taking a look onto a single time slab, discretization can be conducted in several ways. A consequent continuation of the spatial discretization would be a space-time triangulation, e.g., regarding SFEM into n-simplicies [18, 99], which would allow for flexible local space-time refinement and lead to the most efficient realization of space-time finite elements. However, up-to-now there is no commercial grid generation tool supporting space-time discretization in 4D — 3D space plus time — and adaptation is still an ongoing topic. Thus, in today's most common approach each space-time slab is triangulated by exactly one element Q_n^e in time direction by extruding the spatial discretization in the time direction. Hence, the construction of suitable finite dimensional interpolation spaces can be simplified by splitting up in space and time by

the bivariant tensor product of space and time:

$$N_a^\alpha = T^\alpha N_a, \quad (3.74)$$

where T^α represents the interpolation function in time and N_a the interpolation function in space. Hence, T and N can be chosen arbitrary, which allows also to construct function spaces in a similar way $\forall \mathcal{I}_0(\mathbb{R}^{nsd}) \otimes \mathcal{I}_0(\mathbb{R}^1)$

$$(\mathcal{V}_v^h)_n = \{v^h \in H^{1h}(Q_n^h) \mid \forall Q_n^e \mid v|_{\overline{Q_n^e}} \mathcal{I}_0(\mathbb{R}^{nsd}) \otimes \mathcal{I}_0(\mathbb{R}), v^h = \mathbf{0} \text{ on } (P_n^h)_g\}, \quad (3.75)$$

$$(\mathcal{S}_u^h)_n = \{u^h \in H^{1h}(Q_n^h) \mid \forall Q_n^e \mid u|_{\overline{Q_n^e}} \mathcal{I}_0(\mathbb{R}^{nsd}) \otimes \mathcal{I}_0(\mathbb{R}), u^h = \mathbf{g} \text{ on } (P_n^h)_g\}, \quad (3.76)$$

$$(\mathcal{S}_p^h)_n = (\mathcal{V}_p^h)_n = \{p \in H^{1h}(Q_n^e) \mid \forall Q_n^e \mid p|_{\overline{Q_n^e}} \mathcal{I}_0(\mathbb{R}^{nsd}) \otimes \mathcal{I}_0(\mathbb{R})\}. \quad (3.77)$$

Thus, arbitrary combinations of the function spaces are possible, e.g., arbitrary combinations of function spaces introduced in Sec. 3.3. In the presented thesis, linear elements in time are combined with either FEM [92, 146, 147] or NEFEM [143] in space. Theoretical analysis shows that linear finite elements with discontinuous Galerkin in time ensure third order convergence in time [139]. Other combinations like IGA-IGA are presented in [144].

Remark on low order elements

In case of linear space-time Lagrangian finite elements, the residual of the momentum equation after introducing Stokes law for incompressible fluids (2.48) is given as:

$$\mathcal{R}_{\text{MOM}}(\mathbf{u}^h, p^h) = \rho^f \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f} \right) + \nabla p - \bar{\mu} \Delta \mathbf{u}. \quad (3.78)$$

The second order derivative in the viscous term of the residual would disappear in case of linear finite elements, as it appears in the perturbation term. To ensure consistency and enhance the convergence, the term is reconstructed [95].

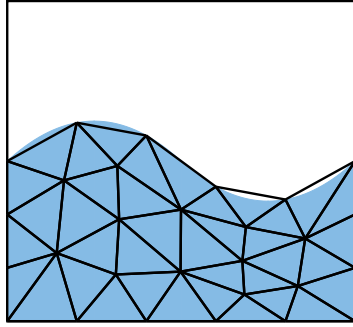


Figure 3.17: Interface tracking

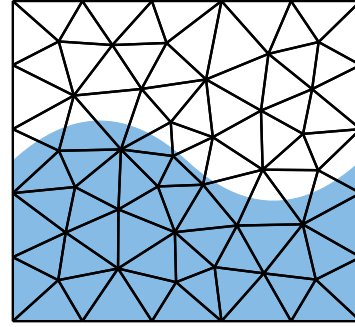


Figure 3.18: Interface capturing

3.5.4 On moving boundaries

The physical problems of fluid-structure interaction and free-surface flows involve moving boundaries of the observed domain. The numerical methods to treat such moving boundaries can be categorized in two types, interface capturing and interface tracking. In interface capturing methods the physical properties of the fluid are determined by the current position of the interface within each cell of a fixed grid, cf. Fig. 3.18. In contrast to that interface tracking utilizes boundary-conforming meshes and the mesh moves alongside the deforming boundary, i.e., the mesh is always adapted to the current position of the boundaries, cf. Fig. 3.17. A detailed overview of both approaches is presented in [62]. Within the presented thesis an interface tracking approach is utilized. Hence, the space-time grid needs to be adapted to the moving boundaries. Literature offers a wide set of methods, like spring analogy [65], methods based on radial basis functions [2], and others. The main aim of grid deformation techniques is to constantly adapt the mesh to the moving domain while keeping the mesh quality of the initial grid during deformation. Therefore, the choice of a method is often problem dependent. Here, the problem is solved with the elastic mesh update method [97] by deforming an artificial elastic solid occupying the fluid domain, which is summarized below. Afterwards, numerical difficulties appearing with the moving grid nodes in computational fluid dynamics are shortly discussed.

Elastic mesh update method

As mentioned above, the method used in the presented thesis is called pseudo-solid approach or elastic mesh update [97]. It considers the computational fluid domain in a separated problem as an elastic solid. Deforming this pseudo elastic solid with

prescribed displacements as Dirichlet boundary conditions at the moving boundaries adapts the filling grid of the computational domain. Since the dynamic behaviour of the pseudo solid is not of interest, time-depending inertia terms are omitted. Similar to the physical elastic solid in Sec. 2.1, the steady equations of motion need to be solved:

$$\nabla \cdot \mathbf{S}^\# = 0 \quad \text{on } \Omega^f. \quad (3.79)$$

The superscript $\#$ is used to distinguish the pseudo elastic problem from the physical one. Analogously to Sec. 2.1.4, Hooke's law for homogeneous and isotropic linear elastic solids can be applied to model the stress tensor

$$\mathbf{S}^\# = \lambda^\# \text{tr}(\mathbf{E}) \mathbf{I} + 2\mu^\# \mathbf{E}, \quad (3.80)$$

that is based on linear strain-displacement relation as in case of the physical problem to model the stress tensor. For further simplification, the Green strain tensor $\mathbf{G}$ is linearized under the assumption of infinitesimal deformation

$$\mathbf{E}(\mathbf{d}) = \frac{1}{2} (\nabla \mathbf{d} + \nabla \mathbf{d}^T). \quad (3.81)$$

The problem can be solved by taking the initial grid as reference in manner of total Lagrangian deformation or, in sense of an incremental Lagrangian approach, by taking the grid of the previous time step as reference. In the latter case, the inner stresses are canceled between the time steps in case of the pseudo elastic approach.

Often, one wants the mesh quality in the vicinity of walls maintained, since the fluid here is very sensitive to perturbations. On the other hand, the coarser parts in the domain play a minor role to the solution and can be heavily deformed. A simple way to obtain that is to connect the stiffness of elements to their size by multiplying it with the inverse of the element Jacobian [148].

Finally, the grid of the upper side of the time slab is adapted by

$$\mathbf{x}_{n+1}^{f,-} = \mathbf{x}_n^{f,+} + \Delta \mathbf{x}^\#, \quad (3.82)$$

where $\Delta \mathbf{x}^\#$ denotes the displacements between the spatial discretization of the upper and lower time slab. Applying the solution strategy to the coupled problem, the pseudo elastic problem is computed based on prescribed displacements by the interface and free-surface, respectively.

Boundary conditions for free-surfaces

If an interface tracking approach is applied to free-surface flows, the influence of the environment to the observed fluid is often neglected. In this case, the dynamic boundary conditions (2.58) on the free-surface Γ^{free} simplifies to

$$\mathbf{T}^f \mathbf{n} = \gamma \kappa \mathbf{n} \text{ on } \Gamma^{free}. \quad (3.83)$$

The kinematic boundary condition in terms of a no-penetration boundary condition is responsible for the mesh movement, i.e., no mass can leave or enter the domain over the interface Γ^{free} :

$$\dot{m} = \int_{\Gamma^{free}} \rho (\mathbf{u} - \mathbf{u}^\#) \mathbf{n} \, d\Gamma \doteq 0. \quad (3.84)$$

Based on this, several kinematic boundary conditions can be derived [16, 19, 62]. One option is to restrict the normal movement of the boundary to the normal fluid velocity, while the tangential velocity is independent:

$$\mathbf{u} \cdot \mathbf{n} = \mathbf{u}^\# \cdot \mathbf{n} \text{ on } \Gamma^{free}. \quad (3.85)$$

Thus, the movement of the boundary is directly depending on the fluid state.

Since equation (3.79) is solved for displacements, displacements as Dirichlet boundary condition on the free surface are required. Hence, the velocities in (3.85) have to be converted. By rewriting the equation to

$$\mathbf{u} \cdot \mathbf{n} = \frac{\partial \mathbf{d}^\#}{\partial t} \cdot \mathbf{n} \text{ on } \Gamma^{free}, \quad (3.86)$$

one way to compute the boundary displacement results in:

$$(\bar{\mathbf{u}} \cdot \bar{\mathbf{n}}) \bar{\mathbf{n}} \Delta t = \Delta \mathbf{x}^\# \text{ on } \Gamma^{free}, \quad (3.87)$$

where $\bar{\mathbf{u}}$ and $\bar{\mathbf{n}}$ denote the velocity and the normal averaged over the time slab $\mathbf{Q}_n$ [168].

Boundary conditions for fluid-structure interaction

In case of FSI, the boundary condition for the mesh deformation, Eq. (3.82), directly depends on the structural solution $\mathbf{d}^s$ on the fluid-structure interaction interface. With

the free-surface boundary conditions, a full set of boundary conditions for the elastic mesh update can be stated as:

$$\Delta \mathbf{x}_n^\# = \begin{cases} \mathbf{d}_{n+1}^s - \mathbf{d}_n^s & \text{on } \Gamma^{fs} \\ (\bar{\mathbf{u}} \cdot \bar{\mathbf{n}}) \bar{\mathbf{n}} \Delta t & \text{on } \Gamma^{free} \\ \mathbf{0} & \text{on } \Gamma^f \setminus (\Gamma^{free} \cup \Gamma^{fs}) \end{cases} \quad (3.88)$$

Remark on conservation laws on moving domains

Within the chapter about continuum mechanics, the Eulerian viewpoint was introduced that is commonly used in fluid mechanics. As quintessence is that the local change of material properties is measured at fixed points in space. In computational fluid dynamics, these fixed points can directly be related to the computational mesh of the fluid domain. However, if an interface tracking approach is applied, the observer domain is changing with time and with it, all grid points are moving. Thus, the Eulerian form is not suitable anymore. This problem is usually handled by the Arbitrary-Lagrangian-Eulerian (ALE) formulation. The ALE version of the generalized balance law in local form is stated as

$$\frac{\partial \hat{\mathbf{a}}}{\partial t} + \nabla_{\mathbf{x}} \cdot (\hat{\mathbf{a}}(\mathbf{u} - \mathbf{u}^\#) - \mathbf{C}_{\mathbf{x}}) - \hat{\mathbf{b}} = \mathbf{0}. \quad (3.89)$$

The only difference from the Eulerian form in Eq. (2.23) is the modified convective term now including the grid velocity $\mathbf{u}^\#$. This equation is usually derived with respect to an additional spatial domain that is fixed in time and by utilizing the geometric conservation law, e.g., in [57, 72]. If finite difference approaches are used for the time-discretization of (3.89) additional conditions to guarantee numerical stability arise, concerning, e.g., the discrete conservation law (DCGL) [66] and the derivation of the grid velocity [72], respectively. Semi-discrete approaches with finite element discretization in space for solving Navier-Stokes equations in ALE formulation are presented, e.g., in [56, 60, 105, 157]. A very detailed discussion of different ALE formulations is presented in [103].

An advantage of the space-time approach is that the space-time formulation implies the ALE formulation. In case of space-time formulations, time-depending boundaries lead to a space-time continuum with $Q \neq \Omega_{\mathbf{x}} \otimes [0, T]$, i.e., the cross-section of the space-time domain is varying in time direction. It has been shown, that the space-time formulation on these space-time domains can be transformed to an ALE formulation on discrete slices in time — where the semi-discrete analysis is applied — by utilizing a space-time Piola transformation [9, 16, 115]. Thus, if a space-time approach is applied, there is no

need for additional methods to fulfill the DGCL, esp., the grid velocity does not have to be computed. Note, if the movement of the boundaries is prescribed, the shape of the whole space-time domain Q would be known. However, in fluid-structure interaction or free-surface flow problems, the position of the boundary is unknown, i.e., an adaption of the space-time domain to the solution is needed. Thus, the term *deformable domain* is used.

4 Numerical methods for the coupled problem

In the previous chapter numerical methods for solving the single-field problems were introduced. As derived in the chapter about continuum mechanics, both single-field problems form a fluid-structure interaction problem when they are acting on domains with a common interface. If the mutual dependency cannot be neglected, the single-field problems cannot be solved independently anymore and the kinematic and dynamic coupling conditions have to be satisfied on the interface, cf. Sec. 2.1.2. In the following sections, the numerical methods to solve the overall problem are introduced. For a start, a short classification of solution methods for coupled problems is given. In the remainder of the chapter, the presented strategy is embedded into this classification. The common classification of solution strategies for coupled problems was presented by Felippa, Park and Farhat [68]. Here, only the relevant parts for fluid-structure interaction are briefly reviewed. In terms of the chosen classification, the fluid-structure interaction problem can be defined from the physical point of view as a surface-coupled two-field problem. From the numerical point of view, a three-field problem needs to be solved, because with the applied interface tracking approach the additional mesh deformation problem appears. Even the movement of the free-surface itself could be regarded as an additional problem.

Similar to the definition of discretization methods, cf. Fig. 3.1, the partitioning can be considered in space and time. While the spatial separation — in case of FSI into two non-overlapping domains — of the fields is specified by *field partitioning*, the temporal partitioning is specified by *splitting*. Additionally, the field partitioning is subdivided into algebraic and differential coupling. In case of *algebraic partitioning* the field equations are discretized first and decomposed afterwards. Typical examples are parallelization strategies for the single-fields. In contrast, the field equations and their domains are decomposed first, e.g., in equations for fluid and solid, and discretized afterwards in sense

of a *differential partitioning*. Since the subdomains are discretized independently, the resulting meshes are usually not matching at the interface.

The solution strategy for coupled problems can be categorized in three main approaches [68]:

Monolithic approaches: The observed system is assembled into one single system of equations, which is solved at once. The assembly can be realized by discretizing the field equations all together, or in case of a differential partitioning, by combining the discretized sub-systems. In the later case, additional equations for spatial and temporal coupling of the sub-system have to be included.

Partitioned approaches: The equations of the involved fields, i.e., fluid and solid, are treated separately, while the relevant data for the coupled problems is exchanged. In case of a surface-coupled problem, information is only exchanged at the common interface. The resulting solution strategies — also called staggered approaches — can be subdivided into implicit and explicit schemes. Implicit schemes iterate between the fields as long as they are not fully converged. Explicit schemes are coupled a certain number of times (often only once per time step). Therefore, the former is often called *strong coupling* while the latter one is called *weak coupling*.

Field elimination: One of the observed fields is reduced, e.g., by model reduction techniques, and applied as boundary condition of the other field.

While field elimination is rarely applicable to FSI problems, the monolithic and partitioned approaches are wide spread. As stated in [68], neither of these is technically superior to the other. On the one hand, monolithic approaches show numerical stability, which is reached with the partitioned approaches for many cases with considerably more effort only. On the other hand, the partitioned approach allows for the reuse of separately developed and maintained solvers, which decreases the effort of developing an overall solver substantially. Because existing and specialized software can be used, the implementation work is eventually restricted to the coupling methods.

Hereafter, the numerical methods applied in this thesis for solving the fluid-structure interaction problem are introduced. Based on a differential partitioning, the aggregated problem is solved with a strong coupling approach with synchronized solvers in time and different methods for the spatial coupling. The single-field problems are solved with the numerical methods introduced in Sec. 3.4 and Sec. 3.5. Although two solvers with different time integration schemes are combined, the solution is computed always for the same time interval. A splitting strategy is not applied. For the spatial coupling a standard

technique is introduced first. Afterwards a new method for the combination of NEFEM and IGA is derived, utilizing the matching interfaces representation, even though the discretization is still non-matching. Eventually, the method allows for a gapless coupling between fluid and structural discretization.

4.1 Temporal coupling between the space-time and semi-discrete approach

As previously stated, the time integration utilized in both solvers is performed with the identical time-step size, i.e., splitting techniques like sub-cycling are not applied. However, due to the different time integration schemes — the generalized- α -scheme on the solid side and space-time finite elements on the fluid side — a special treatment of the coupling conditions is needed.

In Fig. 4.1, the coupling between both solvers is sketched. Each time slab, Q_n , on the fluid side is connected twice with the semi-discrete solver on the structural side. In more detail, each space-time FSI interface $P_n^{fs,f}$ is connected at its edges Γ_n^{fs} and Γ_{n+1}^{fs} with the semi-discrete boundaries, i.e., $\Gamma_{n+1}^{fs,s}$ on the upper edge and $\Gamma_n^{fs,s}$ on the lower one. To simplify the notation, the coupling conditions are discussed in the Eulerian viewpoint and the subscript $\mathbf{x}$ is skipped here. But note that one has to take care of (2.57) to transfer stresses between the different viewpoints. Finally, the coupling conditions introduced in Sec. 2.2 have to be fulfilled on the following boundaries:

$$\Gamma_{n+1}^{fs} = P_n^{fs,f} \cap \Gamma_{n+1}^{fs,s} \quad (4.1a)$$

$$\Gamma_n^{fs} = P_n^{fs,f} \cap \Gamma_n^{fs,s} \quad (4.1b)$$

Hence, the related boundary conditions to be satisfied on each Γ_{n+1}^{fs} can be stated as:

$$\mathbf{x}_{n+1}^f = \mathbf{x}_{n+1}^s \text{ on } \Gamma_{n+1}^{fs}, \quad (4.2)$$

$$\mathbf{u}_{n+1}^f = \mathbf{u}_{n+1}^s \text{ on } \Gamma_{n+1}^{fs}, \quad (4.3)$$

$$\mathbf{T}_{n+1}^f \mathbf{n} = \mathbf{T}_{n+1}^s \mathbf{n} \text{ on } \Gamma_{n+1}^{fs}. \quad (4.4)$$

With these boundary conditions, another issue emerges with the different state variables in both solvers. While the elastodynamic problem is solved for displacements, the fluid

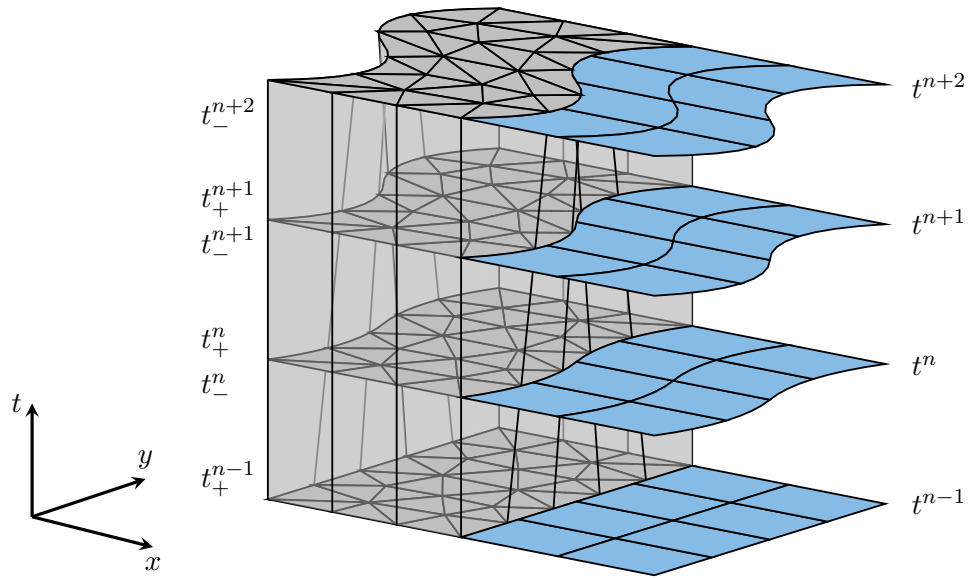


Figure 4.1: Temporal coupling between the space-time slabs in the fluid solver and semi-discrete time steps in the elastodynamic solver.

mechanical problem is solved for velocities. Thus, the kinematic boundary conditions need to be adapted based on the available information. Following the interface tracking approach, the first kinematic condition is satisfied if the computational grid conforms with the current solid state. For the second kinematic coupling condition, several choices are possible. For example, by combining the interface velocity on the fluid side with the Newmark approximation (3.41) on the structural side. In this thesis, a different option is chosen. The velocity on the interface can be computed using a finite-difference approximation based on the grid deformations:

$$\mathbf{u}_{n+1}^{f,-} = \frac{\mathbf{d}_{n+1}^s - \mathbf{d}_n^s}{\Delta t} \text{ on } \Gamma_{n+1}^{fs} \quad (4.5)$$

$$\mathbf{u}_n^{f,+} = \frac{\mathbf{d}_n^s - \mathbf{d}_{n-1}^s}{\Delta t} \text{ on } \Gamma_n^{fs} \quad (4.6)$$

A fully converged solution within the strong coupling approach, subsequently introduced in Sec. 4.3, guarantees energy conservation with respect to the time-integration. However, one has to take care of the *interaction consistency law* [25] to ensure consistency of the boundary conditions in case of a weakly coupled approach.

With equations (4.2)–(4.6), the fluid and solid fields are coupled in time and the mismatch of state variables is eliminated. The following section is now focusing on the spatial coupling.

4.2 Spatial Coupling

As mentioned in the introduction to this chapter, field partitioning leads to non-overlapping domains, and on these, the field equations are independently discretized. With this, non-matching discretizations at the interfaces emerge. Thus, methods for data exchange in terms of spatial coupling are needed. In case of fluid-structure interaction, the information to be exchanged relies on the kinematic and dynamic coupling conditions. More specific, displacements and tractions need to be transferred between both solvers. Literature offers a wide variety of methods to solve this problem. The methods can be categorized by the amount of exchanged information. On the one hand, methods like Nitsche's method [75] or dual mortar method [102] need access to the interpolation space of the interface on both sides. On the other hand, interpolation methods [67], or methods based on radial basis functions [15], only exchange discrete data, which is more suitable for black-box coupling. The aim of the thesis is to keep the solvers as separate as possible

without gaining information of their respective counterpart. Thus, only methods of the latter type are used. The particular methods are introduced in the following. The first presented method is a common interpolation method. Afterwards, it is demonstrated, that combining NEFEM and IGA allows for simplifying the spatial coupling by direct integration. The properties of the introduced methods for spatial coupling are analyzed regarding energy conservation and consistency. The underlying theory of conservation and consistency is presented next.

Since the spatial coupling needs to be computed for each iteration on the discrete interfaces, the notation of (4.1) is simplified in the following for a boundary Γ_{fs} by omitting the time-step index.

4.2.1 Energy conservation and consistency

In the following, energy conservation and consistency is studied with focus on the spatial coupling. Very small time steps are assumed, violation of energy conservation due to time-discretization can be neglected, therefore. Hence, energy conservation over the interface is ensured by [47, 67]

$$\int_{\Gamma^{fs,f}} \mathbf{d}^f \cdot \mathbf{T}^f \mathbf{n}^f d\Gamma^{fs,f} = \int_{\Gamma^{fs,s}} \mathbf{d}^s \cdot \mathbf{T}^s \mathbf{n}^s d\Gamma^{fs,s}. \quad (4.7)$$

To take into account potential differences in the discretization, the interface Γ^{fs} is divided into a fluid side $\Gamma^{fs,f}$ and a structural side $\Gamma^{fs,s}$.

Independently of the applied method, the spatial coupling conditions yield linear relations between the same variables on both sides of the interface. By introducing the transfer matrices $\mathbf{H}_{sf}$ and $\mathbf{H}_{fs}$, these relations can be given as:

$$\mathbf{d}^f = \mathbf{H}_{sf} \mathbf{d}^s, \quad (4.8)$$

$$\mathbf{t}^s = \mathbf{H}_{fs} \mathbf{t}^f. \quad (4.9)$$

$$(4.10)$$

To simplify the notation, the tractions are given by $\mathbf{t} = \mathbf{T}\mathbf{n}$.

Following de Boer et al. [47, 48, 67], a method is conservative if the following relation

between the transfer matrices holds:

$$\mathbf{H}_{fs} = [\mathbf{M}_{ff}\mathbf{H}_{sf}\mathbf{M}_{ss}^{-1}]^T, \quad (4.11)$$

while the matrices $\mathbf{M}_{ff}$ and $\mathbf{M}_{ss}$ depend on the particular discretization method. By utilizing the common form of the interpolation function (3.15), the matrices for the finite element method can be stated as:

$$\mathbf{M}^{ij} = \int_{\Gamma^{fs}} N_i^t(x) N_j^{\mathbf{d}}(x) d\Gamma \quad (4.12)$$

while the superscripts t and $\mathbf{d}$ denote the basis function used to interpolate the tractions and displacements, respectively.

Consistency is satisfied in the different transfer directions, if for constant vectors $\boldsymbol{\beta} = \beta\{\mathbf{1}\}$ the relations

$$\boldsymbol{\beta}^s = \mathbf{H}_{fs}\boldsymbol{\beta}^f \quad (4.13)$$

$$\boldsymbol{\beta}^f = \mathbf{H}_{sf}\boldsymbol{\beta}^s \quad (4.14)$$

hold. In this case, a constant vector $\beta\{\mathbf{1}\}^a$ is transferred into a constant vector $\beta\{\mathbf{1}\}^b$, although the values of the vectors may differ.

All methods studied in [47] are either consistent or conservative. Violating the consistency condition can lead to unphysical oscillations on the surface and failure of the whole solution strategy [47], while non-conservative methods can allow for proper solutions [67]. Together with the methods, analyses of the energy conservation and consistency are presented in the following.

4.2.2 Finite Interpolation Element

The general idea of this method is to transfer the discrete force of every fluid node onto its corresponding base point on the structure, and subsequently use the local non-zero basis functions of the structural discretization to distribute the load to the corresponding nodes of the structure in the sense of a Dirichlet-Neumann coupling, introduced later. In order to transfer the deformation back to the fluid domain, the previously described procedure can be reversed. The method was initially derived without any restriction to the interpolation functions used [67], and can be applied to any kind of finite interpolation function, i.e., based on Lagrangian polynomials or splines [9]. If the interface

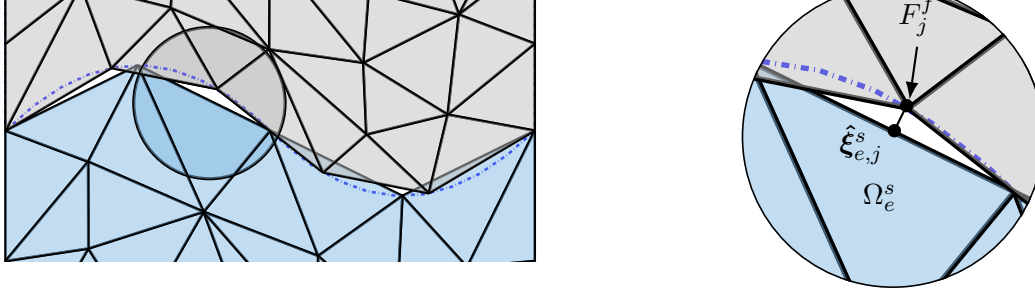


Figure 4.2: Finite Interpolation method for piecewise linear elements. F_i represents a discrete fluid force, $\hat{\xi}_{e,i}^s$ the local parametric coordinate of the closest point along the interface of element Ω_e^s on the structural discretization.

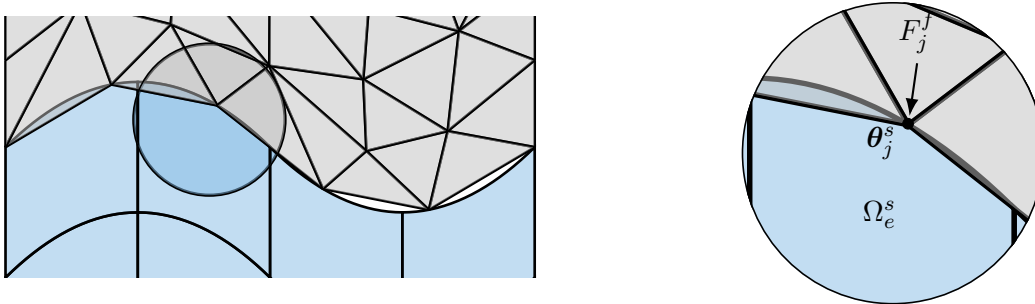


Figure 4.3: Finite Interpolation method for SFEM on fluid side and IGA on structural side. F_i represents a discrete fluid force, θ_i^s the global coordinate of the closed point on the structural spline. Because of the global parametric coordinate, a direct relation to element Ω_e^s is not needed.

approximation is handled separately from the structural discretization, the method is called *finite interpolation elements* [14]. The individual steps are detailed in the following. Differences between the usage of linear Lagrangian or isogeometric finite elements on the solid side are pointed out.

1) *Calculation of discrete forces:*

The method requires discrete forces at each fluid node at the interface Γ_{FS} . One way to compute the forces is to make use of the stress tensor, that is available due to the reconstruction utilized in the stabilization method, cf. Sec. 3.5.3. Therefore, the discrete tractions calculated by the fluid solution can be integrated over the element faces along the interface and distributed to each node. Given the discrete stress $\mathbf{T}_i^f$ at each node on the boundary, the stress distribution is defined as follows:

$$\mathbf{T}(\mathbf{x}) = \sum_i L_i(\mathbf{x}) \mathbf{T}_i, \quad (4.15)$$

where L_i denotes the Lagrangian shape functions. To simplify notation, the global coordinate $\mathbf{x}$ is used to denote the position along the interface for all methods in the following. However, due to mappings introduced together with the finite dimensional interpolation functions in Sec. 3.3, the global coordinate is uniquely related to a local or global parametric coordinate. In order to regain the discrete force at each individual node, the stress has to be multiplied by the corresponding shape function and then integrated, resulting in the following equation

$$\mathbf{F}_i^f = \int_{\Gamma_e^{fs,f}} L_i(\mathbf{x}) \sum_j \left[L_j(\mathbf{x}) \mathbf{T}_j^f \mathbf{n} \right] dr, \quad (4.16)$$

where $\mathbf{n}$ is the normal vector at the interface. In the analysis below, this equation, cf.(4.12), corresponds to

$$\mathbf{F}^f = \mathbf{M}_{ss} \mathbf{t}^f. \quad (4.17)$$

Note that due to the linear approximation of the spatial domain, the normal $\mathbf{n}$ is approximated in case of SFEM. In contrast, that normal can be directly determined based on the underlying spline geometry in case of NEFEM. An alternative method to determine the discrete forces is the consistent boundary flux method [87].

2) *Structural base point computation:*

In the next step, the base point — the closest point — on the structure of each fluid

interface node has to be found. In case of SFEM, the local parametric coordinate within the related element has to be found. In contrast, the global parametric coordinate of the closest point has to be determined for a spline-based representation of the structure. It can, e.g., be calculated by using the second-order algorithm for orthogonal projection presented in [86]. If the geometry is assembled from multiple spline patches, the corresponding patch has to be identified before the parametric coordinate is computed.

3) Transfer of forces onto the structure:

We want to obtain the discrete force as the right hand side of (3.35). Thus, the discrete fluid forces are distributed by multiplying them with the basis functions of the structural discretization evaluated at the determined base point. For each basis function the sum over all basis points — corresponding to the number of fluid points on the fluid interface $nen_{\Gamma_{fs}}^f$ — gives the right hand side contribution for the given node. Since the basis points are defined in local or in global parametric coordinates for SFEM and IGA, respectively, the discrete structural forces are computed by

$$\text{SFEM: } \mathbf{F}_i^s = \sum_j^{nen_{\Gamma_{fs}}^f} L_i^e(\mathbf{x}_j^e) \mathbf{F}_j^f, \quad (4.18a)$$

$$\text{IGA: } \mathbf{F}_i^s = \sum_j^{nm_{\Gamma_{fs}}^f} R_i(\mathbf{x}_j) \mathbf{F}_j^f. \quad (4.18b)$$

Note that since the discrete forces are determined only for element nodes in case of SFEM, the local values have to be summed up in case of neighboring elements. With (4.18a) or (4.18b), respectively, the transfer matrix $\mathbf{H}_{fs}$ is defined. The basis of both finite dimensional ansatz functions fulfill the partition of unity, meaning that

$$\sum_a B_a(\mathbf{x}) = 1 \quad (4.19)$$

holds, while B_a acts again as placeholder for the different basis functions. Thus, the overall force acting on the interface is conserved during the projection.

4) Transfer of deformation onto the fluid:

The computed structural deformation has to be transferred to the fluid in order to account for the mesh deformation. This procedure is straightforward for SFEM and IGA. The elastodynamic problem is formulated at the grid points or the spline control

point. Knowing the parametric coordinates of the basis points, we have to sum up the contributions of all discrete solutions that have a non-zero basis function at that basis point. By distinguishing again between SFEM and IGA, the displacement of one interface point i on the fluid side is given by:

$$\text{SFEM: } \mathbf{d}_i^f = \sum_j^{nen_{\Gamma fs}^s} L_j^e(\mathbf{x}_i^e) \mathbf{d}_i^s, \quad (4.20a)$$

$$\text{IGA: } \mathbf{d}_i^f = \sum_j^{nn_{\Gamma fs}^s} R_j(\mathbf{x}_i) \mathbf{d}_i^s. \quad (4.20b)$$

These equations correspond to the transfer matrix $\mathbf{H}_{sf}$.

Analysis of energy conservation and consistency

Inserting (4.16) into (4.18a) or (4.18b), respectively, and collecting the equations for all nodes on the structural side results in

$$\mathbf{M}_{ss}^T \mathbf{t}_s = \mathbf{H}_{sf}^T \mathbf{M}_{ff}^T \mathbf{t}_f \quad (4.21)$$

by using (4.12) for $\mathbf{M}_{ss}$ and $\mathbf{M}_{ff}$. By resolving the equation for $\mathbf{t}_f$, the following transfer matrix can be identified:

$$\mathbf{H}_{fs} = [\mathbf{M}_{ff} \mathbf{H}_{sf} \mathbf{M}_{ss}^{-1}]^T. \quad (4.22)$$

With (4.11) the method is energy conserving as stated in [67].

With (4.20a) and (4.20b) the structural solution is just evaluated at certain points. With partition of unity the row-sum for the resulting $\mathbf{H}_{sf}$ matrix is equal to one. Thus, the consistency condition for the transfer of displacements is satisfied. But the independency of projection nodes gives, that in general the column sum is $\mathbf{H}_{sf} \neq 1$, and therefore

$$\mathbf{H}_{sf}^T \boldsymbol{\beta} \neq \boldsymbol{\beta} \quad (4.23)$$

The method is globally energy conservative, but not consistent regarding the transfer of pressure.

The method of finite interpolation elements can be applied independently if SFEM or NEFEM is used on the fluid side, since the step 1) would not change. In case of NE-

FEM a slight improvement can be introduced by taking the exact determination of the normal into account. In the following, a simplification of the spatial coupling between NEFEM and IGA is presented, while utilizing the identical representation of the geometry.

4.2.3 Direct transfer between spline-based methods

In case of NEFEM-IGA coupling, the spline representation of the interface is available to both the fluid and structural solver. By this, a perfect compatibility at the fluid-structure interface is guaranteed. This leads to a matching interface representation, cf. Fig. 4.4, but a non-matching discretization, due to the different basis functions, cf. Fig. 4.5.

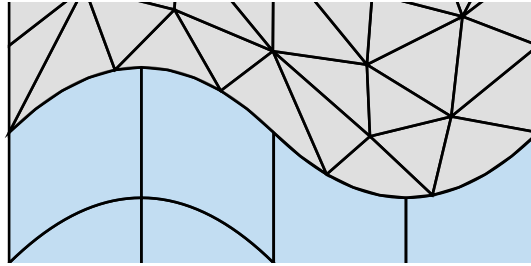


Figure 4.4: Gapless geometry representation of the interface for NEFEM-IGA coupling.

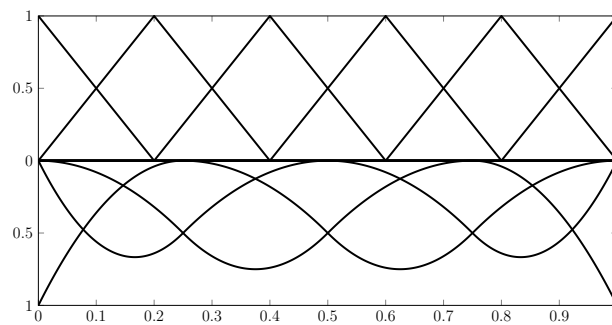


Figure 4.5: Basis functions for NEFEM — on upper half — and NURBS-based isogeometric analysis — on lower half — in parametric space along the interface.

1) *Calculation of discrete forces:*

Since the geometry representation of the coupling interface is identical in both the fluid and structural solver, a direct integration can be applied to get the fluid forces. Since the Galerkin method is applied on the structural side, the right hand side of the structural problem (3.35) can be formulated in case of IGA by using the NURBS basis as a test function as follows:

$$\mathbf{F}_i^s = \int_{\Gamma^{fs}} R_i(\mathbf{x}) (\mathbf{T}^f(\mathbf{x}) \mathbf{n}(\mathbf{x})) d\mathbf{x}. \quad (4.24)$$

Because the NURBS is also available on the fluid side, the integral can be computed by the fluid solver. As previously, the fluid stresses $\mathbf{T}^f$ can be computed from the fluid solution, and the face normals can be determined exactly by evaluating the NURBS. Hence, the boundary integral can be computed piecewise by integration over each fluid grid face:

$$\mathbf{F}_i^s = \sum_e^{n_F} \int_{\Gamma_e^{fs}} R_i(\mathbf{x}) (\mathbf{T}^f(\mathbf{x}) \mathbf{n}(\mathbf{x})) d\mathbf{x}. \quad (4.25)$$

By applying Gaussian integration, cf. Sec. 3.2.1, with n_{GP} as the number of Gauss points, we get

$$\mathbf{F}_i^s = \sum_e^{n_F} \sum_g^{n_{GP}} w(\mathbf{x}_g) R_i(\mathbf{x}_g) (\mathbf{T}^f(\mathbf{x}_g) \mathbf{n}(\mathbf{x}_g)). \quad (4.26)$$

If we insert now the Lagrangian representation of the fluid solution (4.15), the forces acting on the right-hand side of the structural problem can be computed by

$$\mathbf{F}_i^s = \sum_e^{n_F} \sum_g^{n_{GP}} w(\mathbf{x}_g) R_i(\mathbf{x}_g) \left(\sum_j L_j(\mathbf{x}_g) \mathbf{T}_j^f \mathbf{n}(\mathbf{x}_g) \right). \quad (4.27)$$

Note that the direct integration relies on the constraint of matching interface representations. This is not only the case if NEFEM and IGA are combined but also IGA is used on both sides. In the latter case it has to be ensured, that the approximation space of both interpolation functions allows for a matching interface representation; that is not trivial, e.g., in the case of fluid-structure interaction.

Analysis of global energy conservation and consistency

Consider the following approximation for the displacement $\mathbf{d}$ and traction $\mathbf{t} = \mathbf{T}\mathbf{n}$ at the fluid $\Gamma^{f,s,f}$ or structural boundary $\Gamma^{f,s,s}$:

$$\mathbf{t}^f(\mathbf{x}) = \sum_i^{n_f} L_{t_f}^i(\mathbf{x}) \mathbf{t}_i^f \quad (4.28)$$

$$\mathbf{d}^f(\mathbf{x}) = \sum_i^{n_s} R_{d_f}^i(\mathbf{x}) \mathbf{d}_i^f \quad (4.29)$$

$$\mathbf{t}^s(\mathbf{x}) = \sum_i^{n_s} R_{t_s}^i(\mathbf{x}) \mathbf{t}_i^s \quad (4.30)$$

$$\mathbf{d}_s(\mathbf{x}) = \sum_i^{n_s} R_{d_s}^i(\mathbf{x}) \mathbf{d}_i^s \quad (4.31)$$

where n_f and n_s are the number of discrete points on both sides of the interface. Note that the displacement on the fluid side (4.29) is discretized with the NURBS basis functions used on the interface in case of NEFEM. As before, all basis functions are evaluated at the the global coordinate $\mathbf{x}$, which is just a mapping of the parametric coordinate $\boldsymbol{\theta}$. Since on both sides of the interface the displacement is discretized with the NURBS, the transfer matrix simplifies to

$$\mathbf{H}_{sf} = \mathbf{I}. \quad (4.32)$$

Hence, it is obvious that the consistency condition for the displacement is fulfilled. By introducing the matrices

$$\mathbf{M}_{ff}^{ij} = \int_{\Gamma^{fs}} L_{t_f}^i(\mathbf{x}) R_{d_f}^j(\mathbf{x}) \, ds \quad (4.33)$$

$$\mathbf{M}_{ss}^{ij} = \int_{\Gamma^{fs}} R_{t_s}^i(\mathbf{x}) R_{d_s}^j(\mathbf{x}) \, ds \quad (4.34)$$

the direct integration (4.25) can be written as

$$\mathbf{M}_{ss}^T \mathbf{t}^s = \mathbf{M}_{ff}^T \mathbf{t}^f \quad (4.35)$$

which results in the transfer matrix

$$\mathbf{H}_{fs} = (\mathbf{M}_{ff} \mathbf{M}_{ss}^{-1})^T. \quad (4.36)$$

By taking (4.32) into account, this is equivalent to the condition for global energy conservation (4.11). Finally, the consistency for the pressure has to be studied. By defining two constant vectors $\boldsymbol{\beta}_a = \beta \{\mathbf{1}\}^{n_a}$, $a \in \{f, s\}$, equation (4.35) is given by

$$\beta \sum_i^{n_s} \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) R_{t_s}^i(\mathbf{x}) \, ds = \beta \sum_j^{n_s} \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) L_{t_f}^j(\mathbf{x}) \, ds, \quad \forall k = 1, \dots, n_s. \quad (4.37)$$

and can be reformulated by using partition of unity as follows:

$$\beta \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) \underbrace{\sum_i^{n_s} R_{t_s}^i(\mathbf{x}) \, ds}_{=1} = \beta \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) \underbrace{\sum_j^{n_s} L_{t_f}^j(\mathbf{x}) \, ds}_{=1}, \quad \forall k = 1, \dots, n_s. \quad (4.38)$$

This results in

$$\beta \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) \, ds = \beta \int_{\Gamma^{fs}} R_{\mathbf{d}_s}^k(\mathbf{x}) \, ds, \quad \forall k = 1, \dots, n_s. \quad (4.39)$$

Hence, the method is also consistent for pressure.

Note that, although a vector with constant values was used for the pressure, (4.39) still depends on the boundary integral of $R_{\mathbf{d}_s}^k(\mathbf{x})$. Hence, the value of the integral depends on the basis function and its local support. In case of splines the value of (4.39) varies for each k , cf. Fig. 3.5. Thus, a vector $\mathbf{F}$ given by

$$\mathbf{F} = \mathbf{M}_{ss}^T \boldsymbol{\beta}^s = \mathbf{M}_{ff}^T \boldsymbol{\beta}^f \quad (4.40)$$

has in general varying entries.

Summarizing the previous derivations, the direct integration between NEFEM and IGA is globally energy-conserving and consistent for displacement and tractions. In general it can be stated that this correlates directly to the weighted residual methods [36]. For this method in [47] the same statement regarding conservation and consistency was derived under the condition of matching boundaries. Hence, it can be affected by the same problems regarding numerical integration, e.g., if the elements on the fluid

side have a larger support in the physical domain than the elements on the solid side. Refinement by adding additional Gaussian points within the numerical integration can help to counteract the problem [94].

Computation of local stresses

With the previous method, it is obviously possible to compute stresses instead of forces at any control point of a boundary in a energy-conserving way. By using the transfer matrix (4.36), the stresses can be transferred to the control points of the spline by

$$\mathbf{t}^s = (\mathbf{M}_{ff}\mathbf{M}_{ss}^{-1})^T \mathbf{t}^f. \quad (4.41)$$

By having the stresses available at the control points, they can be evaluated at any parametric coordinate on the spline boundary. This might be interesting for the coupling with other methods or in different applications, i.e., if stresses instead of forces are needed.

4.3 Iterative solver

The fluid-structure interaction system is divided by differential partitioning, treating the discretization on both sides independently. In the previous sections, methods for coupling both fields in time and space were discussed. Based on that, both single-field solvers can now be combined to create a fluid-structure interaction solver. Although it would be possible to assemble the single problems as well as the coupling equations separately and combine them afterwards into one single system of equations, the more flexible partitioned approach is followed here. Thus, the single-field problems are solved iteratively in the sense of an implicit staggered approach by exchanging information as boundary conditions at the interface. For solving the single-field problems different solvers are applied, implementing the numerical methods introduced in Ch. 3. To exchange data between both solvers at the interface, the coupling methods introduced in Sec. 4.1 and Sec. 4.2 are utilized. Here, the iterations between both solvers are conducted until convergence, in the sense of a *strongly coupled* approach. Eventually, a fully-converged iterative solution equates to the monolithic solution, see e.g., [126].

If transient problems are solved, the iterative solver is embedded into a time integration scheme. As discussed in 4.1, both single-field solvers compute the same time step, no

splitting scheme is involved. By introducing a operator $\mathbf{H}$ for the iterative solution strategies introduced in Sec. 4.3.2 and Sec. 4.3.3, and an additional operator *predictor* for a predictor to allow for a better initial value of the iterative solver, the algorithm can be stated as given in Alg. 1. To simplify the notation, the unknowns are generalized for the fluid as χ_f and for the structure as χ_s .

Time stepping loop: for $n = 1, \dots, n_{\text{timesteps}}$ do
 | 1) Prediction of structural solution: $\tilde{\chi}_s^{n+1} = \text{predictor}(\chi_s^n)$
 | 2) Call iterative solver: $(\chi_f^{n+1}, \chi_s^{n+1}) = \mathbf{H}(\chi_f^n, \tilde{\chi}_s^{n+1})$
end

Algorithm 1: Time stepping scheme with an embedded iterative solver.

Based on that algorithm, a weakly coupled scheme can be developed by restricting the subiterations of the iterative solver. Together with a suitable predictor, this conforms with a predictor-corrector scheme. Unfortunately, this would not guarantee that the iterative solver fully converges in all time steps, which is not always required, e.g., in problems with compressible fluids. However, the presented thesis is focusing on strongly coupled approaches.

In the following, the implicit staggered approach applied in this thesis is introduced. First, two versions of differential partitioning are described. Finally, the iterative solution strategies are presented, while discussing stability issues and acceleration techniques. During the subsequent discussion, the known values χ_f^n and $\tilde{\chi}_s^{n+1}$ are applied as initial values of the single-field solvers. The superscripts n and $n + 1$ are omitted to simplify notation.

4.3.1 Field partitioning

As mentioned in the introduction, the basis for an iterative solution strategy is field partitioning, dividing the overall system on the interface into two sets of equations and treating them separately. Even the fluid-structure interface can be split into the fluid side $\Gamma_{fs,f}$ and the solid side $\Gamma_{fs,s}$. The single-field problems are reconnected by

prescribing the solution of the particular counterpart as boundary condition on these interfaces. Actually, all combinations between the boundary conditions (3.27) and (3.53) are feasible, although the resulting solvers exhibit differing stability properties. The mathematical foundation of all these methods relies on the field of domain decomposition, see, e.g., [126]. The most intuitive and most common partitioning follows the Dirichlet-Neumann coupling, prescribing the velocity as Dirichlet boundary condition on the fluid side and the tractions in the sense of a Neumann boundary condition on the structural side. Another way is based on a linear combination of the coupling conditions in the sense of Robin boundary conditions. Both strategies are discussed briefly below.

Dirichlet-Neumann coupling

Usually, existing black-box solvers are combined with Dirichlet-Neumann coupling by exchanging just displacements and tractions between solvers. Based on the previously introduced distinction between the two sides of the fluid-structure interface, the applied boundary conditions in the single-field solvers within the iterative solver result in:

$$\text{Dirichlet step: } \mathbf{g}^f = \mathbf{u}^s \text{ on } \Gamma^{fs,f}, \quad (4.42)$$

$$\text{Neumann step: } \mathbf{h}^s = \mathbf{T}^f \mathbf{n} \text{ on } \Gamma^{fs,s}. \quad (4.43)$$

Note that the differences between the Eulerian and Lagrangian viewpoints have to be taken into account, cf. (2.57). However, in case of infinitesimal deformations the linearization in (3.81) is usually applied on the structural side. Then the deformation gradient degenerates to the identity, the difference between Cauchy's stress tensor and second Piola-Kirchhoff stress tensor disappears, and the conversion between the two viewpoints is not needed anymore.

To fulfill the coupling conditions above while combining a fluid solver based on space-time finite elements with a semi-discrete elastodynamic solver, Tezduyar and Sathe [149] suggest to apply the boundary conditions in a weak sense. However, this is possible only if the used software solves for the unknown velocities/displacements and tractions on the boundary. Otherwise, the discrete coupling between field and boundary unknowns is lost. In the presented case, both solvers compute state variables only, while stresses

are evaluated in a postprocessing step. Thus, the coupling conditions have to be applied strongly.

In many cases, the Dirichlet-Neumann coupling is a sufficient method and successfully used in many fields of applications, e.g., aeroelasticity [34, 73, 80, 81, 129]. However, stability problems can occur during fixed-point iterations. Especially applications with incompressible fluids have to be treated carefully, which is discussed in more detail in Sec. 4.3.2. Additionally, if the boundary conditions are prescribed strongly, the method is unable to handle problems in closed domains — sometimes called pure-Dirichlet domains —, e.g., inflating balloons, without any further modification [107]. This can be counteracted by involving Robin boundary conditions into the staggered approach.

Robin-Neumann coupling

As mentioned before, the Dirichlet-Neumann decomposition is not able to solve problems with volume perserving deformations where the fluid domain is completely enclosed by strong Dirichlet boundary conditions. Thus, the coupling conditions need to be modified, e.g., by introducing a volume constraint on the structural side to circumvent problems with mass conservation on the fluid side [107, 154]. Another way involves Robin boundary conditions, i.e., boundary conditions that introduce artificial fluxes on the boundary [3, 74, 121]. Instead of directly utilizing the physical kinematic and dynamic boundary conditions, a linear combination of both, depending on scalar coefficients α_s and α_f , can be applied [3]:

$$\alpha_f \mathbf{u}^f + \mathbf{T}^f \mathbf{n}^f = \alpha_f \frac{\partial \mathbf{d}^s}{\partial t} - \mathbf{T}^s \mathbf{n}^s \text{ on } \Gamma^{fs,f}, \quad (4.44)$$

$$\alpha_s \frac{\partial \mathbf{d}^s}{\partial t} + \mathbf{T}^s \mathbf{n}^s = \alpha_s \mathbf{u}^f - \mathbf{T}^f \mathbf{n}^f \text{ on } \Gamma^{fs,s}, \quad (4.45)$$

with additional restrictions

$$\alpha_f \neq \alpha_s, \quad (4.46)$$

$$\alpha_f, \alpha_s > 0. \quad (4.47)$$

Obviously, these boundary conditions are fulfilled if the physical coupling conditions, cf. Sec. 2.2, are satisfied. Based on that general definition, different coupling schemes can be derived [3], e.g., the previously introduced Dirichlet-Neumann scheme with $\alpha_f \rightarrow \infty$ and $\alpha_s = 0$. In this thesis, a Robin-Neumann scheme is applied in case of enclosed domains, obtained by choosing $\alpha_f > 0$ and $\alpha_s = 0$:

$$\alpha_f \mathbf{u}^f + \mathbf{T}^f \mathbf{n}^f = \alpha_f \frac{\partial \mathbf{d}^s}{\partial t} - \mathbf{T}^s \mathbf{n}^s \text{ on } \Gamma^{f,s,f} \quad (4.48)$$

$$\mathbf{T}^s \mathbf{n}^s = -\mathbf{T}^f \mathbf{n}^f \text{ on } \Gamma^{f,s,s}. \quad (4.49)$$

Instead of applying now the velocity boundary condition directly on the interface, the differences between fluid and structural side are minimized. Even in case of fully enclosed domains, the pure Dirichlet character of the boundary conditions is avoided. During iterations, non-converged predictions of the solid displacement can be counterbalanced by fluid velocity.

Transfer to fluid problem

While the boundary condition applied on the solid side remains unchanged, the Robin boundary condition needs to be integrated into the fluid sub-problem. This is realized by modifying the Robin term for the fluid stresses:

$$\mathbf{T}^f \mathbf{n}^f = \alpha_f (\mathbf{u}^s - \mathbf{u}^f) - \mathbf{T}^s \mathbf{n}^s, \quad (4.50)$$

and applying it as Neumann boundary condition (3.53b) as $\mathbf{h} = \mathbf{T}^f \mathbf{n}^f$. Thus, the Neumann term in the variational formulation (3.56) results in:

$$\int_{P_n} \mathbf{v} \cdot \hat{\mathbf{h}} \, dP = \int_{P_n} \mathbf{v} \cdot \mathbf{T}^f \mathbf{n}^f \, dP = \int_{P_n} \mathbf{v} \cdot (\alpha_f (\mathbf{u}^s - \mathbf{u}^f) + \mathbf{T}^s \mathbf{n}^s) \, dP. \quad (4.51)$$

In the second term of the boundary condition, the unknown velocity $\mathbf{u}^f$ appears. Thus, the boundary integral contributes now to both the right- and left-hand side of this problem. Besides that, the coupling between both solvers is applied in the same fashion as for the Dirichlet-Neumann coupling. But as for the transfer of fluid loads to structural stresses for the Dirichlet-Neumann coupling, the change between different viewpoints

needs also to be considered for the transfer of structural stresses to the fluid stresses here as well. More details about the implementation of the Robin-Neumann coupling into the applied solver framework are given in [100].

Remark on stresses for Robin boundary conditions

The positive feature of the Robin condition is that it avoids a violation of the mass conservation by counterbalancing unexact predictions of the structural displacement. On the other hand, the accuracy of the boundary conditions directly relies on the accuracy of discrete stresses computed on the structural side. As already discussed for the fluid solver in Sec. 3.5.3, stresses cannot be computed directly in case of linear approximations, since their derivatives are vanishing. Alternative methods to the one used on the fluid side are L_2 -Projection, discrete least-squares fitting, or superconvergent patch recovery [78, 123, 165, 166]. In case of NURBS-based isogeometric finite elements with order $p > 1$, the situation is obviously different. Actually, the higher order would allow for a direct computation of stress. However, to guarantee higher-order recovery, the classical FE recovery techniques are discussed in context of IGA in [106].

Note that to simplify the method in the presented thesis Robin-Neumann coupling is only used within a Gauss-Seidel algorithm without relaxation, introduced below in Sec. 4.3.2. This allows to use the reconstructed fluid stresses of the previous iteration in the next iteration on the right-hand side of the fluid problem, instead of reconstructing them again on the structural side.

4.3.2 Gauss-Seidel iteration

Finally, the staggered solution strategy applied in this thesis is presented. Again, the unknowns are generalized for the fluid by χ_f and the structure by χ_s to simplify notation in the following. Additionally, operators for the fluid problem $\mathbf{S}$, the elastodynamic problem $\mathbf{F}$ and for the whole coupled fluid-structure interaction system $\mathbf{H}$ are introduced. Although, the discretization is treated independently and projection might be needed, it is assumed that the projection methods are part of the specific operators here. Same holds for the grid deformation that is assumed to be part of the fluid operator $\mathbf{F}$. Since a surface coupled problem is solved, the operators are dealing only with information — or unknowns — on the interface. The data to be exchanged depends on the

type of differential partitioning in the sense of a Dirichlet-Neumann or Robin-Neumann coupling, as introduced in Sec. 4.3.1. Although the methods are described here for fluid-structure interaction, they can be directly transferred to other coupled problems, e.g., to the coupling between the fluid solver and the deforming domain in case of a free-surface flow problem.

Due to the coupling conditions, a solution of the structural problem χ_s^* can be formulated as a fixed-point iteration, where the fixed point χ_s^* has to be found for every time step $n + 1$:

$$\chi_s^* = \mathbf{H}(\chi_s^*) = \mathbf{S} \circ \mathbf{F}(\chi_s^*). \quad (4.52)$$

Splitting up the FSI operator into its components, one gets the following relation:

$$\begin{aligned} \chi_f^* &= \mathbf{F}(\chi_s^*), \\ \chi_s^* &= \mathbf{S}(\chi_f^*). \end{aligned} \quad (4.53)$$

This can be utilized for a Block-Gauss-Seidel procedure, the classical iteratively staggered solution strategy for coupled problems [29,68,110,157]:

Gauss-Seidel Scheme($\chi^0, k = 0$):
Coupling loop: repeat
 1) Call of fluid solver: $\chi_f^{k+1} = \mathbf{F}(\chi_s^k)$
 2) Call of structural solver: $\tilde{\chi}_s^k = \mathbf{S}(\chi_f^{k+1})$
 3) Update solid solution: $\chi_s^{k+1} = \tilde{\chi}_s^k$
 $k = k + 1$
until $\tilde{\chi}_s^k \approx \chi_s^k$;

Algorithm 2: Pseudo-code for Gauss-Seidel scheme.

The Gauss-Seidel scheme converges linearly. However, its stability depends on several factors, such as time-step sizes, density ratio, and even the structural stiffness as discussed subsequently. The algorithm is the basis for all modifications introduced in the following to increase stability and efficiency. For a start, a simple modification of

the Gauss-Seidel algorithm is introduced that allows for an enhancement of the stability.

Successive over-relaxation

The simplest way to enhance stability is the relaxation between the current und previous solution by the parameter ω . Hence, step 3 of the algorithm 2 can be given as:

$$\chi_s^{k+1} = \omega \tilde{\chi}_s^k + (1 - \omega) \chi_s^k . \quad (4.54)$$

This modification of the Gauss-Seidel procedure corresponds to *Successive Over-Relaxation*, and in the context of fluid-structure interaction, to the iterative staggered scheme for a Dirichlet-Neumann coupling proposed in [110]. The stability analysis in [110] states that the method is conditionally stable and the choice of the relaxation parameter is strongly dependent on the problem. Unfortunately, a unique prediction of the optimal parameter is impossible, especially in transient problems, it can vary in time. Thus, the practical choice is often based on the aim of having a stable solution for all time steps, which is often quite inefficient. Thus, a dynamic relaxation would be preferred. Before introducing more sophisticated methods, the reason for the stability issues in incompressible fluids is shortly discussed in the following.

Statement on added mass

In comparison to monolithic schemes, partitioned algorithms solving the Dirichlet-Neumann problem show instabilities, especially in explicit staggered schemes and when incompressible fluids are involved. Since the interaction consistency law [25] is not satisfied in weakly coupled schemes, errors on the displacement are acting due to the incompressibility directly as pressure reactions on the right-hand side of the structural problem. These instabilities can directly be related to the *added mass effect*. It was verified that the added mass effect depends not only on physical properties like density ratio [35] but also on applied time-integration schemes [71]. Further studies are presented in [42, 117, 152]. In case of incompressible fluids, the weakly coupled schemes are therefore always conditionally stable only. Robin boundary conditions can help to counteract the added mass effect [3, 74, 121]. Banks et al. [4, 5] have demonstrated on simple test cases that

a coupling based on Robin boundary conditions may lead to an unconditionally stable procedure.

In the standard Dirichlet-Neumann coupling the violation of the interaction conservation law can be counteracted by iteratively staggered approaches. Then, the added mass effects are diminishing if the solver is fully converged. Unfortunately, the added mass effect influences the convergence of the iterative process [50, 52, 110]. By convergence analysis, Le Tallec and Mouro [110] have demonstrated that the convergence of the iterative procedure can rely on the same influences as explicit schemes. The stability decreases with decreasing time-step size, structural stiffness, or mass ratio. A convergence with a fixed relaxation parameter is not guaranteed. However, the choice of an optimal relaxation parameter is difficult, especially since it can vary over time and iterations. Therefore the most pessimistic choice is often chosen, leading to inefficient solvers. The efficiency can be improved by more sophisticated methods, following the idea of dynamic updates of the relaxation parameter. These are mainly discussed in the context of interface quasi-Newton schemes, as introduced in the following.

4.3.3 Newton-based schemes

Even if the original Gauss-Seidel procedure allows to solve many coupled problems, the stability problems and efficiency problems with fixed relaxation mentioned before make more elaborate methods necessary. One choice are methods based on Newton's method to solve nonlinear problems, already introduced for the solution of single-field problems in Sec. 3.2.2. The main idea behind these methods is to derive a better prediction for the Gauss-Seidel procedure. The prediction is computed with a Newton step for the solution on the interface. By writing (4.52) in residual form

$$\mathbf{R}(\boldsymbol{\chi}_s^*) = \boldsymbol{\chi}_s^* - \mathbf{H}(\boldsymbol{\chi}_s^*) = \boldsymbol{\chi}_s^* - \mathbf{S} \circ \mathbf{F}(\boldsymbol{\chi}_s^*) = \mathbf{0}. \quad (4.55)$$

a solution with the Newton-Raphson method (3.13) would allow to solve the system by:

$$D\mathbf{R}(\boldsymbol{\chi}_s^k) \Delta \boldsymbol{\chi}_s^k = \mathbf{R}(\boldsymbol{\chi}_s^k), \quad (4.56)$$

$$\boldsymbol{\chi}_s^{k+1} = \boldsymbol{\chi}_s^k + \Delta \boldsymbol{\chi}_s^k. \quad (4.57)$$

Unfortunately, the residual (4.55) depends on the coupled problem $\mathbf{H}$ and any derivation of the Jacobian $D\mathbf{R}(\boldsymbol{\chi}^k)$ or its inverse would need to solve the nonlinear system.

This is contrary to the differential partitioning and impossible to realize if black-box solvers are involved. Hence, the key idea of quasi-Newton methods is to approximate the Jacobian by using results of previous time steps. The approximation is utilized to compute a prediction for the next iteration in the Gauss-Seidel iteration. In case of fluid-structure interaction problems only interface information is included, leading to the term of *interface quasi-Newton methods*. The three variants applied in the presented thesis are discussed briefly in the following. Further variants are discussed in [26, 51, 53–55, 112]

4.3.4 Aitken's method

A very simple method is following the idea of a modified Newton method, where the Jacobian is computed only once per time step, solving the system by

$$D\mathbf{R}(\boldsymbol{\chi}_s^0)\Delta\boldsymbol{\chi}_s^k = \mathbf{R}(\boldsymbol{\chi}_s^k), \quad (4.58)$$

$$\boldsymbol{\chi}_s^{k+1} = \boldsymbol{\chi}_s^k + \Delta\boldsymbol{\chi}_s^k. \quad (4.59)$$

Since this idea can slow down the convergence, Aitken introduced an acceleration parameter ω_k [93, 108],

$$\boldsymbol{\chi}_s^{k+1} = \boldsymbol{\chi}_s^k + \omega_k \Delta\boldsymbol{\chi}_s^k, \quad (4.60)$$

that can be computed in terms of two subsequent residuals [108]

$$\omega_k = -\omega_{k-1} \frac{(\mathbf{R}_{k-1})^T (\mathbf{R}_k - \mathbf{R}_{k-1})}{\|\mathbf{R}_k - \mathbf{R}_{k-1}\|_2^2}. \quad (4.61)$$

The relaxation parameter can be considered as an estimation for $D\mathbf{R}(\boldsymbol{\chi}_s^0)/D\mathbf{R}(\boldsymbol{\chi}_s^k)$, see e.g., [7]. Finally, a starting parameter has to be chosen. Often, the relaxation parameter is bound by an initial value, $\omega_k \geq \omega_0 \forall k$, [108].

Comparing the successive over relaxation in Sec. 4.3.2 with the Aitken method, the only difference is the dynamically adapted relaxation parameter. The Aitken method allows for a significant increase of the efficiency while keeping the effort for implementation rather small. Additionally, the computational costs are limited if the Aitken relaxation is computed with interface data only [108].

Interface quasi-Newton inverse least-squares

While the Aitken's method was initially derived to accelerate the modified Newton method that uses a constant Jacobian, quasi-Newton methods try to approximate the Jacobian in every iteration. In the context of interface quasi-Newton methods, it is convenient to derive the Newton iteration for the inverse form of the residual

$$\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^*) = \tilde{\boldsymbol{\chi}}^* - \mathbf{H}^{-1}(\tilde{\boldsymbol{\chi}}^*), \quad (4.62)$$

where a fixed point of equation (4.52) is also fixed point of the inverted fixed-point equation (4.62). The resulting Newton step is:

$$\boldsymbol{\chi}^{k+1} = \tilde{\boldsymbol{\chi}}^k + D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)^{-1}\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k). \quad (4.63)$$

This would need the evaluation of $D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)$ or its inverse, respectively. As already mentioned, this is contrary to the iterative procedure with black-box solver. Instead, quasi-Newton methods try to approximate the Jacobian or its inverse.

A method presented by Degroote et al. [51] is founded on a Taylor expansion of all solutions of previous iterations with respect to the result of the current iteration k :

$$\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^i) = \tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k) + D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)(\tilde{\boldsymbol{\chi}}^i - \tilde{\boldsymbol{\chi}}^k), \quad \forall i = 1, \dots, k-1. \quad (4.64)$$

Each Taylor expansion of a residual of the previous iteration $\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^i)$ results in an equation to compute the inverse of the Jacobian

$$\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^i) - \tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k) = D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)(\tilde{\boldsymbol{\chi}}^i - \tilde{\boldsymbol{\chi}}^k) \Rightarrow D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)^{-1}\Delta\tilde{\mathbf{R}}_i^k = \Delta\tilde{\boldsymbol{\chi}}_i^k. \quad (4.65)$$

By collecting the data, the *input matrix* $\mathbf{W}_k$ and the *output matrix* $\mathbf{V}_k$ can be constructed [51] as:

$$\mathbf{W}_k = [\Delta\tilde{\boldsymbol{\chi}}_0^k, \Delta\tilde{\boldsymbol{\chi}}_1^k, \dots, \Delta\tilde{\boldsymbol{\chi}}_{k-1}^k] \quad \text{with } \Delta\tilde{\boldsymbol{\chi}}_i^k = \tilde{\boldsymbol{\chi}}_s^k - \tilde{\boldsymbol{\chi}}_s^i, \quad (4.66a)$$

$$\mathbf{V}_k = [\Delta\tilde{\mathbf{R}}_0^k, \Delta\tilde{\mathbf{R}}_1^k, \dots, \Delta\tilde{\mathbf{R}}_{k-1}^k] \quad \text{with } \Delta\tilde{\mathbf{R}}_0^k = \tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k) - \tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^i), \quad (4.66b)$$

with the matrices $\mathbf{W}_k$ and $\mathbf{V}_k$ storing the changes of the structural deformation and the residual, respectively, between the current iteration with respect to all previous

iterations. Since (4.65) holds for each input-output relation, the approximated inverse of the Jacobian $\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \approx D\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k)^{-1}$ for iteration k is determined by the relation

$$\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k = \mathbf{W}_k . \quad (4.67)$$

In practical cases, the number of iterations is much smaller than the number of degrees of freedom in the problem system, even if only the interface is included. Hence, the system of equations (4.67) is underdetermined. A solution for that problem is presented in [51,112]. The minimization in the Frobenius norm $\|\cdot\|_F$ by

$$\|\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}(\tilde{\boldsymbol{\chi}}^k)\|_F \rightarrow \min \quad (4.68)$$

ensures the existence of a unique solution. With the method of Lagrange multipliers and utilizing the constraint (4.67), an update strategy can be derived. An explicit derivation of the approximated Jacobian can be circumvented by using

$$\boldsymbol{\chi}^{k+1} = \tilde{\boldsymbol{\chi}}^k + \mathbf{W}_k \boldsymbol{\alpha} , \quad (4.69)$$

with the vector $\boldsymbol{\alpha} = (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T (-\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}^k))$. Note at this point that the usage of the inverse fixed-point iteration avoids linear dependencies to the initial solution in $\mathbf{W}_k$ [112]. Since a first result is needed, an initial solution step is computed with fixed under-relaxation, while the fixed relaxation factor has to be chosen in a way preventing divergence.

An efficient way to compute $\boldsymbol{\alpha}$ is given via the solution of the least-square problem [112]

$$\min \|\mathbf{V}_k \boldsymbol{\alpha} + \tilde{\mathbf{R}}(\boldsymbol{\chi}^k)\|_2 \quad (4.70)$$

that can be solved, e.g., by QR decomposition. In literature, the method is referred to as ***interface quasi-Newton inverse least squares (IQN-ILS) method*** [51].

By using the relation $\tilde{\mathbf{R}}(\tilde{\boldsymbol{\chi}}_s^k) = \mathbf{R}(\boldsymbol{\chi}_s^k) = \tilde{\boldsymbol{\chi}}_s^k - \boldsymbol{\chi}_s^k =: \mathbf{R}^k$ the quasi-Newton method transferred into algorithm 3.

The convergence can be improved by including information of previous time steps into the input and output matrices, also suggested in [51]. However, there is no mathematical way to predict the optimal number of time steps included into the derivation of

IQN-ILS(χ^0):

1st iteration: $\tilde{\chi}_s^0 = \mathbf{H}(\chi_s^0) = \mathbf{S} \circ \mathbf{F}(\chi_s^0)$

Form residual: $\mathbf{R}^0 = \tilde{\chi}_s^0 - \chi_s^0$

Under-relaxation: $\chi_s^1 = \omega \tilde{\chi}_s^0 + (1 - \omega) \chi_s^0$

$k = 0$ **Coupling loop: repeat**

 Perform one loop:

$$\tilde{\chi}_s^k = \mathbf{H}(\chi_s^k)$$

 Form residual:

$$\mathbf{R}^k = \tilde{\chi}_s^k - \chi_s^k$$

 Build input matrix:

$$\mathbf{W}_k = [\Delta \chi_{k-1}^k, \dots, \Delta \chi_0^k] \quad \text{with } \Delta \chi_i^k = \tilde{\chi}^k - \tilde{\chi}^i$$

 Build output matrix:

$$\mathbf{V}_k = [\Delta \mathbf{R}_{k-1}^k, \dots, \Delta \mathbf{R}_0^k] \quad \text{with } \Delta \mathbf{R}_i^k = \mathbf{R}^k - \mathbf{R}^i$$

 Solve least-squares problem:

$$\min \|\mathbf{V}_k \alpha + \mathbf{R}^k\|_2$$

 Post-processing update:

$$\chi_s^{k+1} = \tilde{\chi}_s^k + \mathbf{W}_k \alpha$$

$$k = k + 1$$

until $\tilde{\chi}_s^k \approx \chi_s^{k+1}$;

Algorithm 3: Interface quasi-Newton inverse least-squares method (IQN-ILS) [51].

the inverse Jacobian. For the most efficient approximated inverse Jacobian, linear dependencies and contradicting information has to be avoided [112]. Instead, by updating the inverse Jacobian with the method presented in the following these problems are diminishing.

Interface quasi-Newton inverse multiple-vector Jacobian method

The method introduced before is using only data of previous iterations for the reconstruction of the Jacobian. Additionally, information of previous time steps can be used to improve the convergence. The method was published first by [26] for block-iterative methods and modified in [112] for the acceleration of the Gauss-Seidel procedure. It combines the interface quasi-Newton method introduced before with an update similar to Broyden's method. The idea of Broyden's method is to minimize the evaluations of the Jacobian by evaluating it only once exactly in the very first iteration as in the modified Newton, but update it in the following iterations by a rank-one update [32]. It was extended by Kvaalen to approximate the inverse of the Jacobian [109]. Instead of an upate between two iterations, the IQN-IMVJ method updates the inverse Jacobian of

the previous time step ${}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}$ with the Jacobian of the current iteration ${}^{n+1}\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}$. The Jacobian of the current time step is approximated as in the IQN-ILS method. Hence, the update can be stated as:

$${}^{n+1}\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = \Delta_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}} + {}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}, \quad (4.71)$$

while $\Delta_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}$ denotes the update. Introducing (4.71) into (4.67) results in:

$$\Delta_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}} \mathbf{V}_k = \mathbf{W}_k - {}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k. \quad (4.72)$$

As in case of the IQN-ILS, the system of equations is underdetermined. Again, by minimizing

$$\|\Delta_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}\|_F \rightarrow \min \quad (4.73)$$

in the Frobenius norm while applying the right-hand side of (4.72) as a constraint, a unique inverse Jacobian update can be computed. Thus, the Jacobian update can be computed by taking all input pairs of the current coupling step into account. As the update of the Jacobian corresponds to the difference of the two Jacobians, the new Jacobian approximation is ensured to be as close as possible to the previous one in a minimal norm sense. In analogy to the IQN-ILS method, Eq. (4.72) yields the update formula for the **interface quasi-Newton inverse multiple-vector Jacobian (IQN-IMVJ) method** [112]:

$${}^{n+1}\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = {}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} + \left(\mathbf{W}_k - {}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T. \quad (4.74)$$

And finally, the interface Quasi-Newton update of the IQN-IMVJ method results in:

$$\begin{aligned} \boldsymbol{\chi}_s^{k+1} &= \tilde{\boldsymbol{\chi}}_s^k - {}^{n+1}\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{R}(\tilde{\boldsymbol{\chi}}_s^k) \\ &= \tilde{\boldsymbol{\chi}}_s^k - \left({}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} + \left(\mathbf{W}_k - {}^n\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T \right) \mathbf{R}(\tilde{\boldsymbol{\chi}}_s^k). \end{aligned} \quad (4.75)$$

During fluid-structure interaction computations, the update strategy is conducted with respect to the Jacobian of the previous time step, cf. Alg. 4. In the first step, the last approximation of the Jacobian in the previous time step is used as an initial guess in the current step. In all subsequent iterations, the inverse Jacobian of the previous iteration is updated with the current Jacobian, which is actually approximated with the input

and output matrices $\mathbf{W}_k$ and $\mathbf{V}_k$. This implies that information of past time steps is included. With this strategy, information of the past is included in the derivation of the inverse Jacobian.

IQN-IMVJ(χ^0):
First iteration: $\tilde{\chi}_s^0 = \mathbf{H}(\chi_s^0)$
Form residual: $\mathbf{R}^0 = \tilde{\chi}_s^0 - \chi_s^0$
if (1st time step) **then**
 Under-relaxtion step: $\chi_s^1 = \omega \tilde{\chi}_s^0 + (1 - \omega) \chi_s^0$
 Initialize previous inverse Jacobian: $\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = \mathbf{0}$
else
 Update previous inverse Jacobian to new one: $\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = {}^{n+1}_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}$
 QN-Update with previous inverse Jacobian: $\chi_s^1 = \tilde{\chi}_s^0 - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{R}^0$
end
k=0
Coupling loop: repeat
 Perform iteration: $\tilde{\chi}_s^k = \mathbf{H}(\chi_s^k)$
 Form residual: $\mathbf{R}^k = \tilde{\chi}_s^k - \chi_s^k$
 Build input matrix: $\mathbf{W}_k = [\Delta \tilde{\chi}_{k-1}^k, \dots, \Delta \tilde{\chi}_0^k]$ with $\Delta \tilde{\chi}_i^k = \tilde{\chi}_s^k - \tilde{\chi}_s^i$
 Build output matrix: $\mathbf{V}_k = [\Delta \mathbf{R}_{k-1}^k, \dots, \Delta \mathbf{R}_0^k]$ with $\Delta \mathbf{R}_i^k = \mathbf{R}^k - \mathbf{R}^i$
 Compute update: ${}^{n+1}_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = {}^n \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} + \left(\mathbf{W}_k - {}^n \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T$
 Post-processing step: $\chi_s^{k+1} = \tilde{\chi}_s^k - {}^{n+1}_k \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{R}^k$
 k = k + 1
until $\tilde{\chi}_s^{k+1} \approx \chi_s^k$;

Algorithm 4: Interface Quasi-Newton inverse multiple vector Jacobian method (IQN-IMVJ) [112].

Interface quasi-Newton inverse multiple-vector least-squares method

The application of the IQN-IMVJ method is quite expensive since the inverse $(\mathbf{V}_k^T \mathbf{V}_k)^{-1}$ has to be computed in every iteration. A slight modification of the method should be presented here, which counteracts this drawback by combining the IQN-IMVJ method with ideas of the IQN-ILS method, resulting in the *interface quasi-Newton inverse multiple-vector least-squares method (IQN-IMVLS)*. Rewriting (4.75) as

$$\boldsymbol{\chi}_s^{k+1} = \tilde{\boldsymbol{\chi}}_s^k + \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}(-\mathbf{R}(\boldsymbol{\chi}_s^k)) + \left(\mathbf{W}_k - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T (-\mathbf{R}(\boldsymbol{\chi}_s^k)) \quad (4.76)$$

allows to introduce the vector $\boldsymbol{\alpha} = (\mathbf{V}_k^T \mathbf{V}_k)^{-1} \mathbf{V}_k^T (-\mathbf{R}(\boldsymbol{\chi}_s^k))$ in analogy to (4.69). With that, the update can be formulated as,

$$\boldsymbol{\chi}_s^{k+1} = \tilde{\boldsymbol{\chi}}_s^k + \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}(-\mathbf{R}(\boldsymbol{\chi}_s^k)) + \left(\mathbf{W}_k - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) \boldsymbol{\alpha}, \quad (4.77)$$

while the vector $\boldsymbol{\alpha}$ can be computed efficiently by solving again the least-squares problem (4.70). Eventually, the Jacobian of the current time step has to be computed explicitly only once as preparation for the next time step, i.e., if the current time step is converged. The resulting algorithm is presented in Alg. 5.

4.3.5 Remark on relaxation with splines

All relaxation variants introduced in previous sections compute the relaxation with respect to the structural deformation. If isogeometric finite elements are used, the relaxation is computed for the displacement of the control point coordinates. However, the relaxation of the control points implies the relaxation of the geometry. Since the fluid grid points are displaced by the evaluation of the current geometry, the relaxation of the spline equals the relaxation of the grid points. But since the number of control points is typically lower in comparison the number of fluid points, this variant is more efficient.

IQN-IMV-LS(χ^0):
 First iteration: $\tilde{\chi}_s^0 = \mathbf{H}(\chi_s^0)$
 Form residual: $\mathbf{R}^0 = \tilde{\chi}_s^0 - \chi_s^0$
if (1st time step) **then**
 Under-relaxtion step: $\chi_s^1 = \omega \tilde{\chi}_s^0 + (1 - \omega) \chi_s^0$
 Initialize previous inverse Jacobian: $\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = \mathbf{0}$
else
 Update previous inverse Jacobian to new one: $\widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = {}_{k+1}^n \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1}$
 QN-Update with previous inverse Jacobian: $\chi_s^1 = \tilde{\chi}_s^0 - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{R}^0$
end
 k=0
Coupling loop: repeat
 Perform iteration: $\tilde{\chi}_s^k = \mathbf{H}(\chi_s^k)$
 Form residual: $\mathbf{R}^k = \tilde{\chi}_s^k - \chi_s^k$
 Build input matrix: $\mathbf{W}_k = [\Delta \tilde{\chi}_{k-1}^k, \dots, \Delta \tilde{\chi}_0^k]$ with $\Delta \tilde{\chi}_i^k = \tilde{\chi}_s^k - \tilde{\chi}_s^i$
 Build output matrix: $\mathbf{V}_k = [\Delta \mathbf{R}_{k-1}^k, \dots, \Delta \mathbf{R}_0^k]$ with $\Delta \mathbf{R}_i^k = \mathbf{R}^k - \mathbf{R}^i$
 Solve LS problem: $\min \|\mathbf{V}_k \alpha + \mathbf{R}^k\|_2$
 Post-processing step: $\chi_s^{k+1} = \tilde{\chi}_s^k + \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} (-\mathbf{R}^k) + \left(\mathbf{W}_k - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) \alpha$
 k = k + 1
until $\tilde{\chi}_s^{k+1} \approx \chi_s^k$,
 Compute update Jacobian:
 ${}_{k+1}^n \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} = \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} + \left(\mathbf{W}_k - \widehat{\mathbf{J}}_{\tilde{\mathbf{R}}}^{-1} \mathbf{V}_k \right) \left(\mathbf{V}_k^T \mathbf{V}_k \right)^{-1} \mathbf{V}_k^T$

Algorithm 5: Interface Quasi-Newton inverse multiple vector least-squares method (IQN-IMVLS).

5 Influence of discretization methods to FSI test case

In order to evaluate the accuracy of the new fluid-structure interaction approach, three different test cases are chosen. The first was proposed by Schäfer et al. [132], the second one by Wall [157] and the third by Turek and Hron [151]. The first two test cases involve steady laminar incompressible channel flow around a rigid and elastic cylinder respectively, while the third test case involves unsteady fluid-induced vibrations of a flexible structure attached to a rigid cylinder [79]. Thereby, the adapted NEFEM method with direct coupling can be tested for simple problems. It is expected that the convergence rate will remain the same as for standard FEM, since the same shape function definition is still utilized for solution space, but the absolute value of the error will decrease. The latter is due to a decrease in geometrical error due to integration over a smooth and — at least in the rigid case — the exact domain.

5.1 Cylinder

5.1.1 Rigid Cylinder

The first test case is used to compare the influence of using NEFEM in contrast to FEM and involves laminar flow around a fixed rigid cylinder. The configuration is proposed in [132], in particular this is the “Test case 2D-1”. A sketch of the computational domain is provided in Figure 5.1.

A parabolic inflow velocity is given at the inlet:

$$\mathbf{u}(0, y, t) = \left(\frac{4Uy(H-y)}{H^2}, 0 \right). \quad (5.1)$$

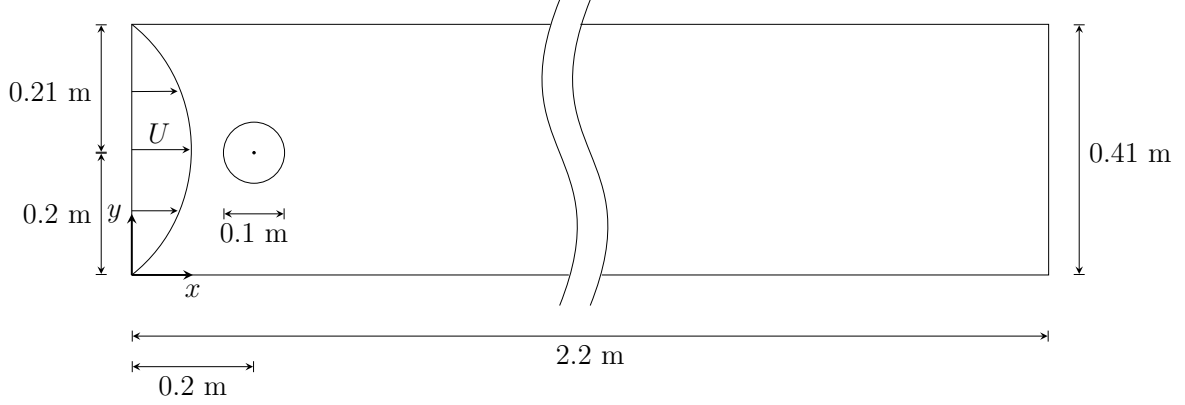


Figure 5.1: Sketch of the rigid cylinder benchmark.

No-slip conditions are applied at the upper and lower wall as well as on the cylinder, and a free-flow boundary condition on the outlet. All parameters used for the simulation can be found in Table 5.1. The Reynolds number is defined as

$$Re = \frac{\rho \bar{U} D}{\mu}, \quad (5.2)$$

where $\bar{U}$ is the average inflow velocity.

parameter	symbol	value
maximum inflow velocity	U	$0.2\text{--}1.5 \text{ m/s}$
dynamic viscosity	μ	0.001 kg/ms
density	ρ	1.0 kg/m^3
cylinder diameter	D	0.1 m
Reynolds number	Re	20

Table 5.1: Rigid cylinder benchmark parameters.

For the given Reynolds number, the flow is steady. Therefore, steady Navier-Stokes equations are solved.

An unstructured grid with 50 boundary elements on the cylinder and 5411 elements is used as a starting point. Four further grids are considered, which are obtained by

a simple h -refinement of the initial grid. This is done by splitting every edge in half. An overview of the utilized grids is given in Table 5.2. In case of NEFEM, the geometry of the cylinder is represented by a second-order NURBS line with 10 control points.

	elements	elements on cylinder wall
Grid 1	5,411	50
Grid 2	13,863	100
Grid 3	55,442	200
Grid 4	221,808	400
Grid 5	887,232	800

Table 5.2: Grids used for the rigid cylinder benchmark simulations.

The reference value that serves as a basis for the comparison between the results for NEFEM and standard FEM is the drag coefficient. In both cases — NEFEM and FEM — the drag coefficient computed on the final grid (NEFEM: $c_d = 5.570928$ and FEM: $c_d = 5.569983$) is in good agreement with the reference value of $c_d = 5.579535$ given by featflow. We therefore consider both implementations to be valid for this test case. In addition, Figure 5.2 gives a convergence plot for the drag obtained with the two methods. The featflow result is employed as a reference result. As expected, both methods feature the same convergence rate. However, one can observe that NEFEM slightly improves the error constant.

5.1.2 Flexible Cylinder

The second test case can be viewed as an extension of the test case in Sec. 5.1.1. Instead of a rigid cylinder, the test case now involves an elastic structure [157]. The dimensions of the domain are detailed in Figure 5.3. The inflow velocity is constant; on the lower and upper wall, a slip condition is applied. A no-slip boundary condition is used on the cylinder. The mass enclosed by the cylinder is neglected. The elastic structure is completely fixed on its rightmost point (on the horizontal symmetry axis). In order to avoid asymmetric results induced by numerical errors, the displacement of the leftmost cylinder point is suppressed in y -direction.

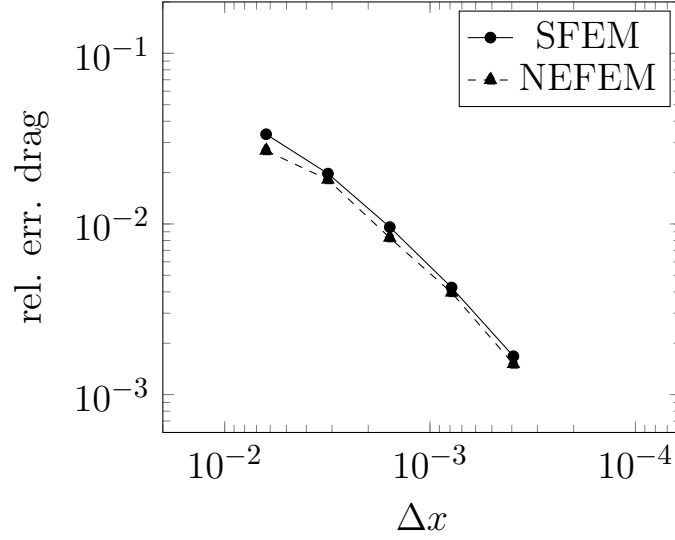


Figure 5.2: Grid convergence for rigid cylinder.

parameter	symbol	value
inflow velocity	U	0.10282 m/s
dynamic viscosity	μ	0.0000182 kg/ms
density (fluid)	ρ^f	1.18 kg/m^3
cylinder diameter	D	0.006 m
Reynolds number	Re	40
Young's modulus	E	100 Pa
Poisson number	ν^s	0.3
density (structure)	ρ^s	1000 kg/m^3

Table 5.3: Flexible cylinder parameters.

The given Reynolds number of 40 leads to a steady solution and a significant deformation of the cylinder. Pressure distribution around the rigid and deformed cylinder is given in Fig. 5.4.

On the fluid side, an unstructured grid with 40 boundary elements on the cylinder and 2352 elements in total is generated initially. Similar to the first test case, six additional grids are created by applying simple h -refinement to the initial grid. An overview of the grids used is given in Table 5.4. The geometry of the structure is defined by a second-order NURBS with 40 elements in circular direction and 5 elements in radial direction. Because the effect of NEFEM on FSI should be isolated, the discretization of the elastic structure remains unchanged within the presented study.

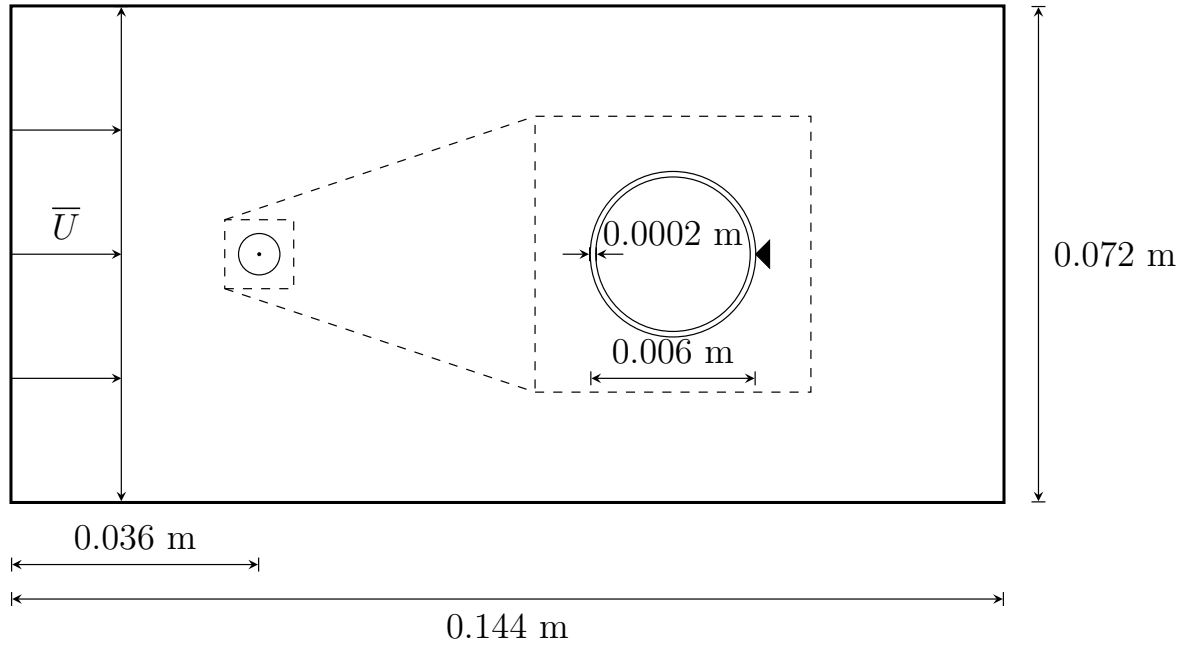


Figure 5.3: Sketch of the flexible cylinder test case.

	elements	elements on cylinder wall
Grid 1	2352	40
Grid 2	9408	80
Grid 3	37632	160
Grid 4	150528	320
Grid 5	602112	640
Grid 6	9633792	2560

Table 5.4: Grids used for the flexible cylinder simulations.

Two reference quantities are used to assess the convergence: (1) drag on the cylinder and (2) the displacement of the leftmost cylinder point. The results are presented as relative results to grid 6 in Fig. 5.5. As in the rigid case, the slopes of the graphs are almost identical, but the relative error of NEFEM is clearly lower. In the presented case, the offset between both graphs is in the order of one refinement level.

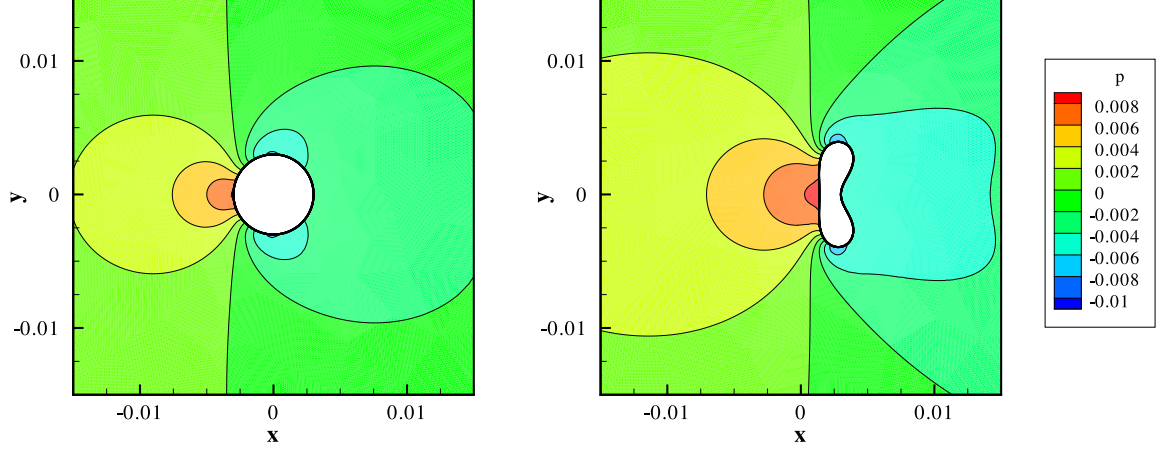


Figure 5.4: Pressure distribution for the undeformed (left) and the deformed (right) cylinder.

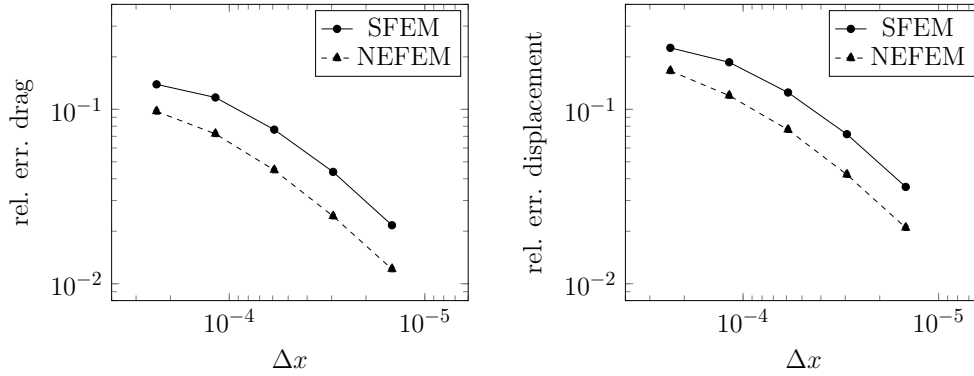


Figure 5.5: Convergence with respect to drag (left) and displacement (right) for the flexible cylinder.

5.1.3 Influence of deformation to the benefit of NEFEM

By comparing the rigid results with the results of the flexible cylinder, we can see that the benefit of using NEFEM instead of SFEM on the fluid side is clearly varying. But as already mentioned, the solution is still approximated with linear Lagrangian finite elements. Hence, any difference results from the different geometrical mapping. It is obvious that in case of a straight interface the differences between NEFEM and SFEM are

vanishing. Thus, in the fluid-structure interaction the absolute difference in the error is expected to be directly related to the deformation. To confirm that, the previous test case is computed by using different Young's moduli and by keeping the flow conditions constant. Hence, with decreasing stiffness the deformation of the cylinder is increasing and with that the maximum curvature, as presented in Fig. 5.6.

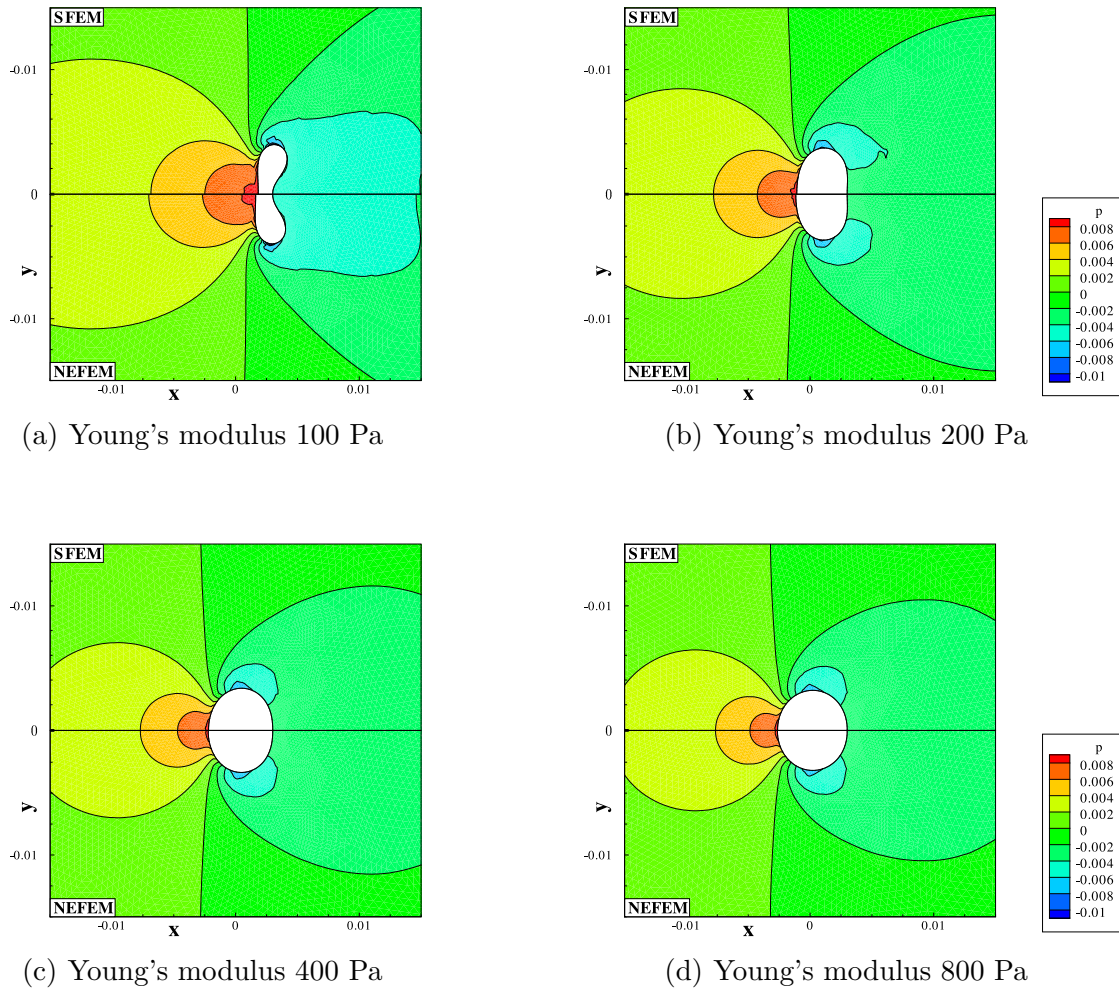


Figure 5.6: Pressure field for different Young's moduli.

In the comparison of the pressure fields between NEFEM and SFEM it can already be stated, that the differences between both methods are reduced with increasing stiffness

and decreasing deformation. Since the initial geometry is cylindrical, the differences converge with increasing Young's modulus against the results of a rigid cylinder. This is confirmed in Fig. 5.7. The absolute difference between the displacements is decreasing, and so is the benefit of using NEFEM. On the other hand, the relative benefit with respect to the displacement is more or less constant.

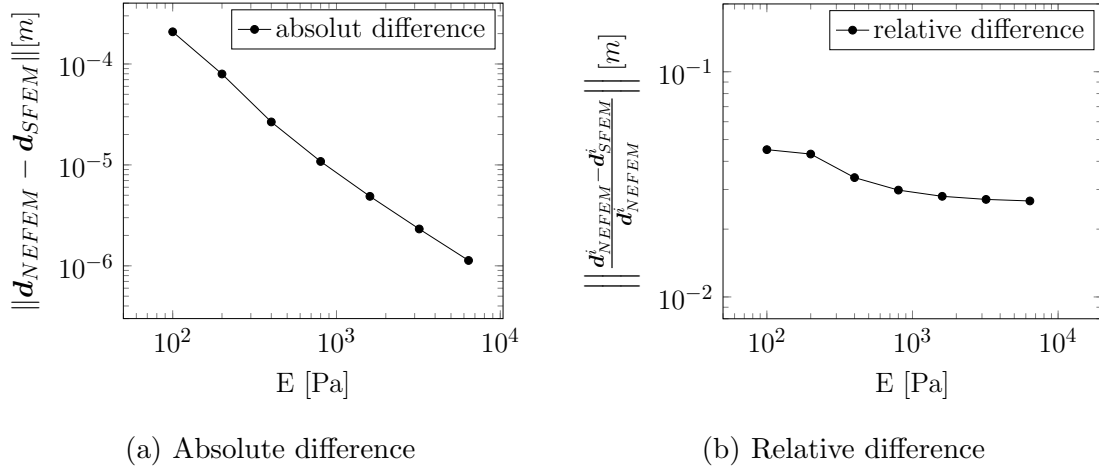
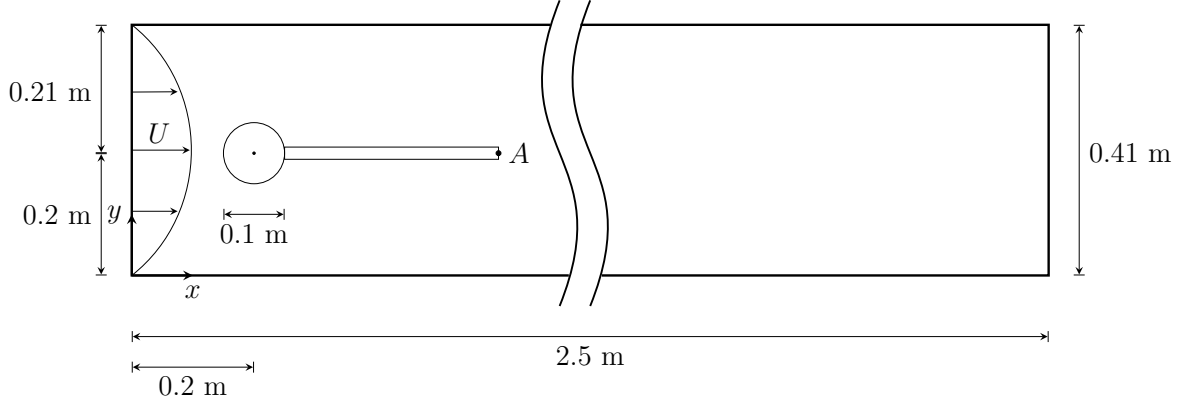


Figure 5.7: Absolute and relative differences in displacement between SFEM and NEFEM for changing Young's moduli.

5.2 2D flexible structure

The third test case is used in order to show the performance of the method for transient FSI problems. The configuration is proposed in [151] and is referred to as *FSI benchmark - FSI2*. A sketch of the computational domain is given in Fig. 5.8. The deformable elastic structure, or "flag", is fixed at the flag-cylinder interface.

Similar to Section 5.1.1, a parabolic velocity profile is set at the inlet. No-slip conditions are applied at the upper and lower wall, as well as on the cylinder and the flag, and a free-flow boundary condition is applied at the outlet. All parameters relevant for the simulations are shown in Table 5.5. For the given Reynolds number of 100, the interaction of the flow and the elastic structure causes self-induced oscillations of the

**Figure 5.8:** Sketch of the moving flag featflow benchmark.

parameter	symbol	value
inflow velocity	U	$1.0\text{--}1.5\text{ m/s}$
dynamic viscosity	μ	1.0 kg/ms
density	ρ^f	1000.0 kg/m^3
cylinder diameter	D	0.1 m
Reynolds number	Re	100
Young's modulus	E	$1.4 \cdot 10^6\text{ Pa}$
Poisson number	ν^s	0.4
density (structure)	ρ^s	10000.0 kg/m^3

Table 5.5: Moving flag featflow benchmark parameters.

	elements in domain	elements on flag
Grid 1	3,572	75
Grid 2	14,288	150
Grid 3	57,152	300
Reference	162,134	584

Table 5.6: Grids used for moving flag simulations.

structure, which result in a quasi-steady periodic wave-like deformation of the flag. The pressure distribution at time $t=13.6\text{ s}$ is given in Fig. 5.9.

The fluid domain is discretized using an unstructured grid with 75 boundary elements on the flag surface and 3572 elements in total. Similarly to the previous test cases, two other grids were created by applying simple h -refinement to the initial grid. As reference for the convergence study the solution computed on a fine grid with 584 elements on the flag surface is used. An overview over the grids used is given in Table 5.5. The structure is discretized using a quadratic NURBS with 60×15 elements. This discretization is kept constant throughout the study. In all computations, a time step of $\Delta t = 0.002\text{ s}$ was applied.

Since flow conditions combined with the unsymmetric configuration lead to flow-induced vibrations of the beam, the movement of point A is plotted in Fig. 5.10 and Fig. 5.11.

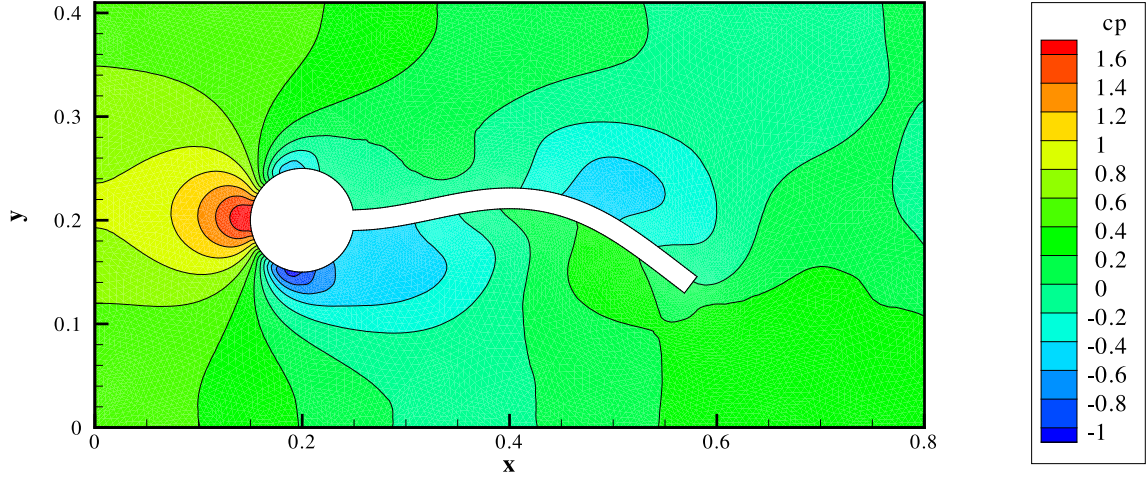


Figure 5.9: Pressure distribution for deformed flag at time $t=13.6s$

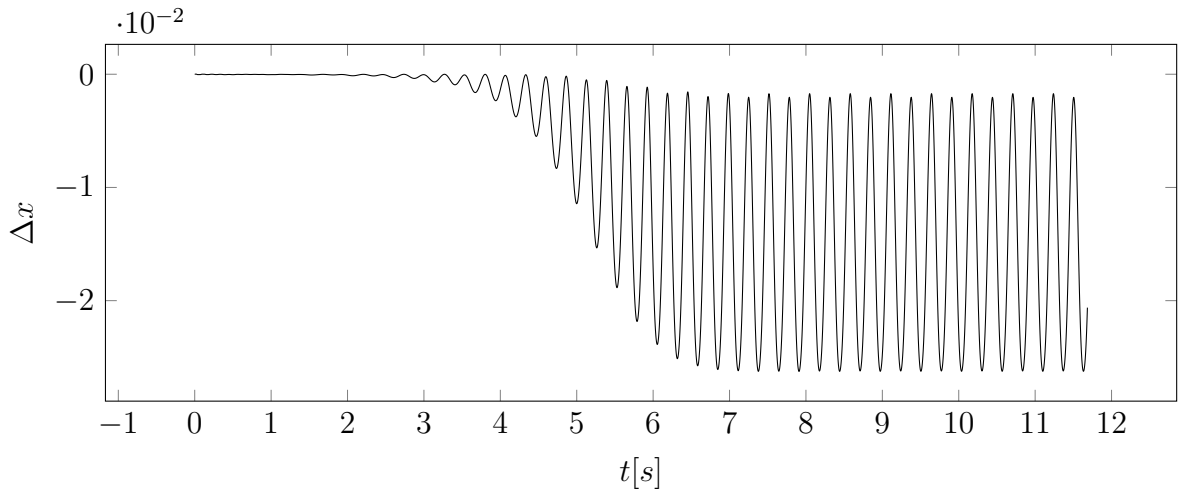


Figure 5.10: Vertical displacement of point A .

After some start up period, the oscillation reaches a stable state. To confirm the observations of the steady cases, the amplitude of the horizontal displacement is compared in a grid convergence study, cf. Fig. 5.12.

The quantity for comparison is the amplitude of the vertical displacement of point A ,

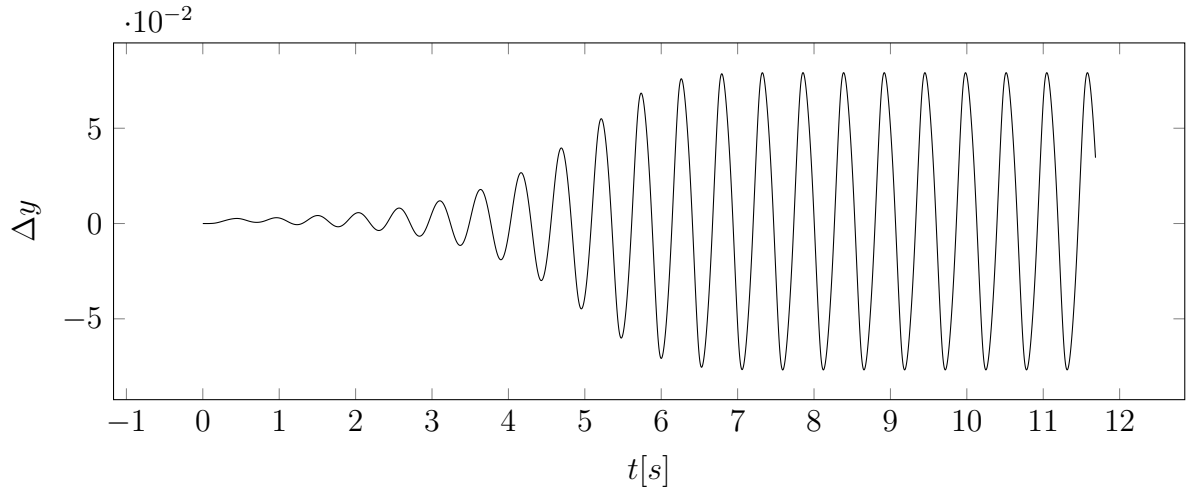


Figure 5.11: Horizontal displacement of point A .

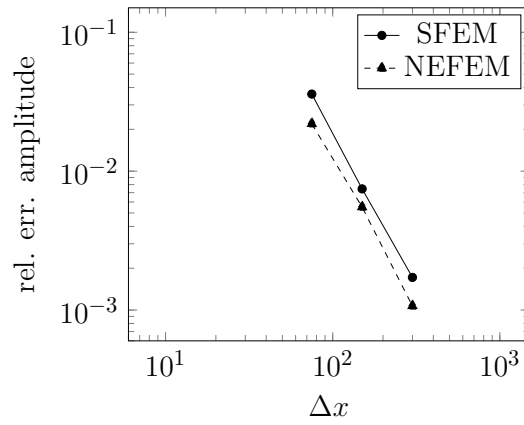


Figure 5.12: Grid convergence of amplitude of vertical displacement at point A .

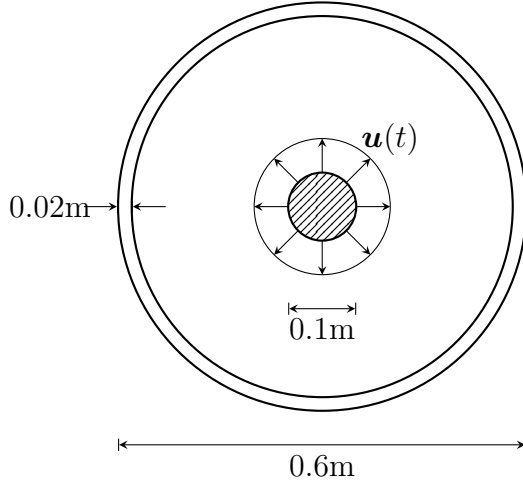
located at the tail of the flag. Fig. 5.12 shows the results as relative error to the reference solution. As in the previous test cases, the offset between both graphs is identical, but again the relative error of NEFEM is clearly lower. However, this can be explained again by the geometry.

6 Challenging problems for partitioned approaches

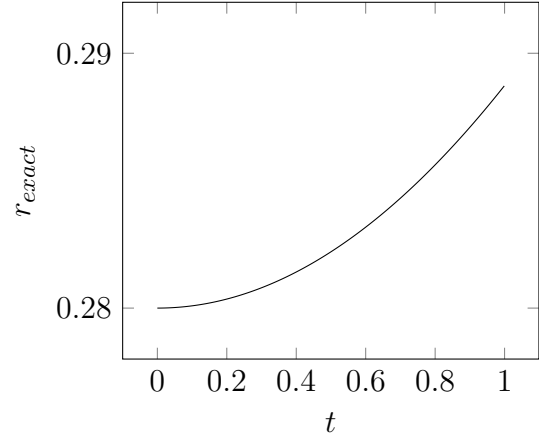
The previous chapter focused on the influence of the new coupling approach between NURBS-enhanced finite elements and NURBS-based isogeometric analysis to the accuracy. For that, problems were chosen that are solveable with partitioned approaches without running into stability problems. For example, all test cases so far exhibited low influence of the added mass effect, i.e., the ratio between the densities on fluid and solid side much larger than one. In the following, test cases are studied that are more difficult to solve with partitioned approaches. The first configuration includes a fluid domain that would be fully enclosed by Dirichlet boundary conditions in case of the standard coupling approach. By utilizing a Robin-Neumann coupling, the differences between NURBS-enhanced finite elements and standard finite elements are pointed out again. The second setup is configured to be unstable. Here, different acceleration methods for partitioned approaches are studied. The main aim of this test case is to confirm the results presented by Degroote and Lindner with isogeometric analysis and to demonstrate the performance of the modified Quasi-Newton method in comparison to the original formulation.

6.1 Balloon

The test case presented is related to the Robin-Neumann coupling discussed in Sec. 4.3.1. The main characteristic of the balloon-type configuration depicted in Fig. 6.1(a) is the fact that the incompressible fluid is completely enclosed by walls, while the remaining boundary on the inner circle is the inflow boundary. The physical properties of both fluid and structure are listed in Tab. 6.1. The linearly increasing inflow velocity leads to a simple blow-up of the domain, cf. Fig. 6.1.



(a) Sketch of the configuration.



(b) Theoretical deformation based on time-dependent inflow.

Figure 6.1: Balloon type test case: circular inflow enclosed by a circular thin structure.

As already discussed in Sec. 4.3.1, the given configuration is impossible to compute with a Dirichlet-Neumann coupling where on all boundaries of the fluid domain a Dirichlet boundary condition would be applied. Due to the incompressibility, every error in the prediction of the current deformation would lead to a violation of the mass conservation on the fluid side. Instead, a Robin-Neumann coupling introduces an artificial flux on the interface during the coupling iterations. This numerically ensures mass conservation, while a fully converged solution still fulfills the physical coupling conditions. The symmetry of the studied configuration would actually allow for a one-dimensional simplification. However, to study the differences between standard finite elements and NURBS-enhanced finite elements the complete configuration is considered. Due to the incompressibility and symmetry the displacement of the fluid-structure interface depends only on the inflow velocity. Especially, the displacement of the interface is independent of the physical properties of both fluid and structure. Thus, the homogeneous deformation of the interface can be computed analytically based on the inflow velocity. Because the domain remains circular, the radius is used as a characteristic variable in the following.

With the inflow $\mathbf{u}(t)$ and the inflow area A_{in} , the exact time-dependent radius can be

parameter	indentifier	value
inflow velocity	$\mathbf{u}(t)$	$0.1 \cdot t \cdot \mathbf{n} \text{ m/s}$
dynamic viscosity	μ	1.0 kg/ms
density (fluid)	ρ^f	1000 kg/m^3
Young's modulus	E	$1.4 \cdot 10^6 \text{ Pa}$
Poisson number	ν^s	0.3
density (structure)	ρ^s	10000 kg/m^3

Table 6.1: Parameters of the balloon test case.

computed under the assumption of exact time integration by

$$r_{exact}(t) = \sqrt{(A_{in} \int_0^t \mathbf{u}(t) dt + 2 \cdot \pi \cdot 0.28^2) \cdot \frac{1}{\pi}}, \quad (6.1)$$

$$r_{polygon}(t) = \frac{\sqrt{(A_{in} \int_0^t \mathbf{u}(t) dt + (2 \cdot \sin(\frac{\pi}{n}) \cdot 0.28^2) \cdot \frac{1}{n} \cdot \tan(\frac{\pi}{n}))}}{\sin(\frac{\pi}{n})}. \quad (6.2)$$

Here, r_{exact} is the exact radius of the two-dimensional balloon, while $r_{polygon}$ corresponds to the radius of the outer circle of a n -polygon. Based on these equations, the minimal error between the exact geometry and its linear approximation can be computed. This error converges against 0 for $n \rightarrow \infty$.

The structure is discretized with 225 quadratic NURBS elements, which allow to represent the circular interface exactly at any time of the transient computation. The fluid domain is discretized with 2400 elements, while both boundaries are equidistantly discretized with 60 elements, which corresponds to a polygon with $n = 60$. To ensure comparability, the inflow boundary is always discretized with standard finite elements.

Since the NURBS-based boundary is included in the numerical integration in case of NURBS-enhanced finite elements, the area of the fluid domain can be computed exactly, only constrained in terms of numerical integration. In contrast, the boundaries are linearly interpolated in case of standard finite elements. Thus, in the presented configuration the approximated domain is slightly smaller than the exact domain. In case of incompressible fluids, this results in a larger deformation for the given inflow condition. This is confirmed by the results presented in Fig. 6.2, where the numerical results computed with standard finite elements and NURBS-enhanced finite elements are presented

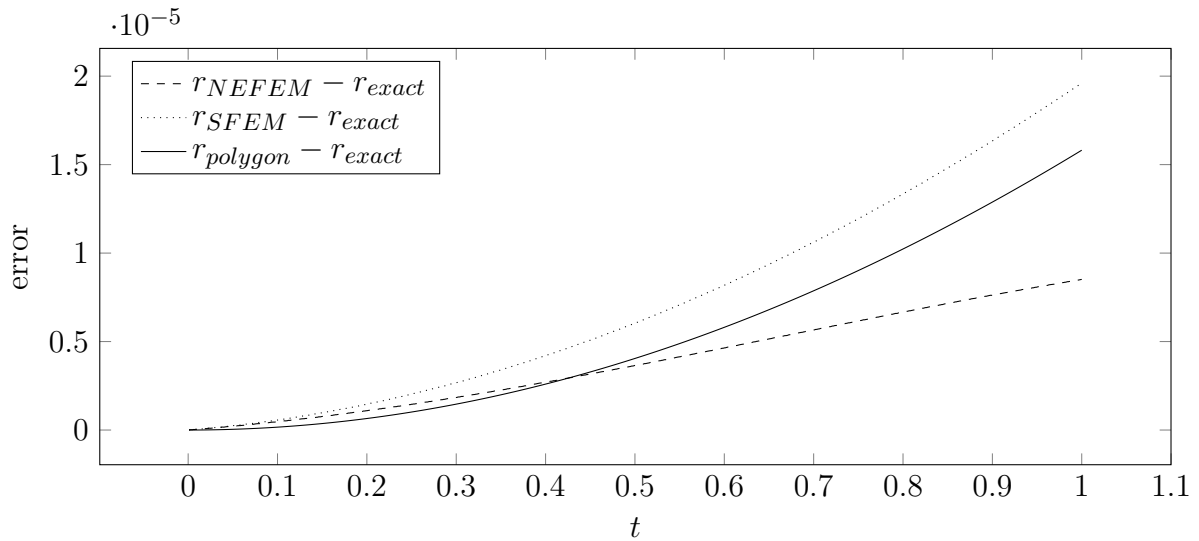


Figure 6.2: Error in radius over time for $\Delta t = 0.001s$.

together with the best numerical error for a polygonal discretization with 60 edges on the boundary. Obviously, the change of the radius is always larger for standard finite elements. Additionally, for the given time-step size of $\Delta t = 0.001s$ the error computed with NURBS-enhanced finite elements is smaller than the best approximation one can get with standard finite elements for almost the whole time span.

However, the transient computation is affected by numerical errors due to time discretization, i.e., by the discrete coupling conditions (4.6). Thus, even the NURBS-enhanced finite elements are not able to compute the exact solution. But it can be expected that the results computed with NURBS-enhanced finite elements converge with decreasing time-step size against the exact solution, while results computed with standard finite elements converge only against the best solution for linear discretization. This assumption is confirmed by the convergence study with respect to the time-step size presented in Fig. 6.3. The order of convergence of NURBS-enhanced finite elements and standard finite elements is almost the same if the respective best solution is taken into account. But with respect to the exact solution, standard finite elements converges against a constant error.

The presented results demonstrate that in this particular case the numerical result of NURBS-enhanced finite elements is depending only on the time discretization, while the results computed with standard finite elements depend additionally on the spatial discretization. Especially, the results of standard finite elements converge against the results of NURBS-enhanced finite elements for $n \rightarrow \infty$. This demonstrates that for closed

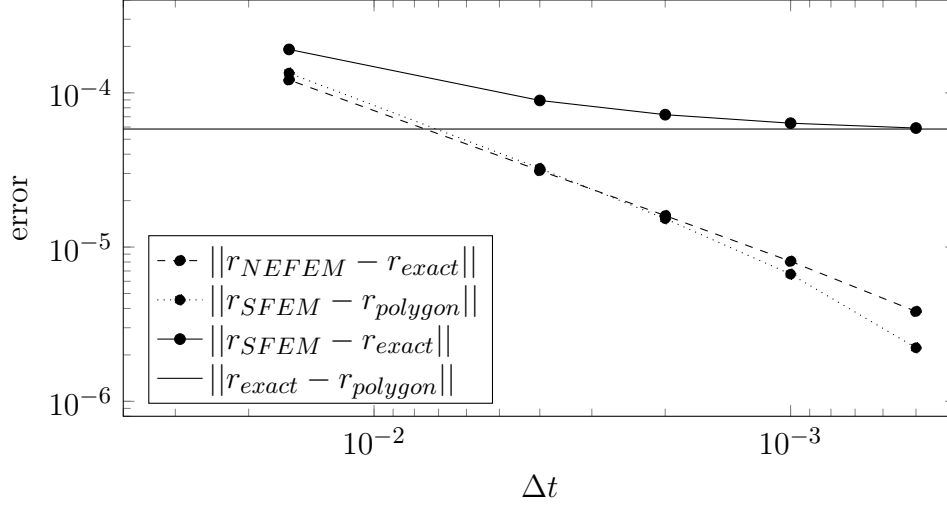


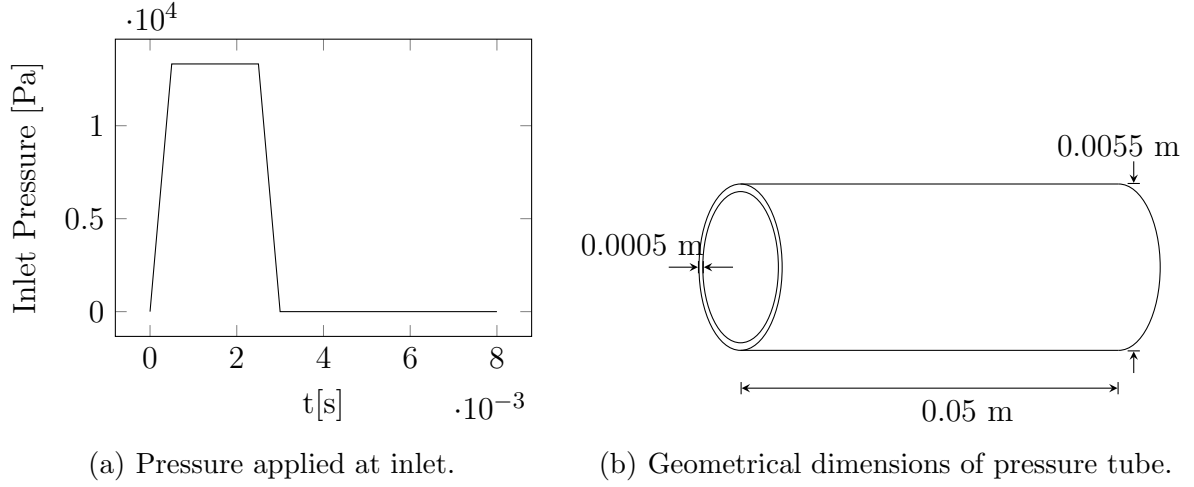
Figure 6.3: Absolute error of the radius at $t=1.0$ s.

domains, the integration of the precise boundary into the finite element discretization of the fluid domain can lead to much more accurate results than standard finite elements are able to compute.

6.2 Elastic pressure tube

With increasing number of degrees of freedom the efficiency of numerical solvers is getting more crucial. Thus, the performance of the iteratively staggered solvers introduced in section 4.3.3 is analyzed in this section for an elastic pressure tube test case, a three-dimensional cylindrical fluid domain bounded by an elastic wall. A pressure wave is travelling through that tube, initiated by a time-dependent pressure jump applied at the inlet. The test case is inspired by [69], but the pressure level was adapted to get a larger displacement. The material properties are given in table 6.2. Note that with a density ratio $\frac{\rho_f}{\rho_s} \approx 0.83$, strong added-mass effects can be expected. The dimensions of the domain are depicted in Fig. 6.4(b). On the wall, a no-slip boundary condition is applied. The time-varying pressure jump used as a boundary condition at the inlet is given in Fig. 6.4(a), linearly increasing to $p_{max} = 13322 \text{ Pa}$ for $0.0s < t < 0.003s$ and linearly decreasing for $0.025s < t < 0.003s$. At the outflow a zero-pressure boundary condition is used. The structure is clamped at both ends.

parameter	symbol	value
inflow velocity	U	$1.0\text{--}1.5\text{ m/s}$
dynamic viscosity	μ	0.001 kg/ms
density (fluid)	ρ^f	1000.0 kg/m^3
Young's modulus	E	$3 \cdot 10^5\text{ Pa}$
Poisson ratio	ν^s	0.3
density (structure)	ρ^s	1200.0 kg/m^3

Table 6.2: Material parameters of the elastic pressure tube test case.**Figure 6.4:** Definitions of the elastic pressure tube test case.

6.2.1 Numerical test

The computations compared in the following are conducted on a fluid grid with approximately 20,000 tetrahedral space-time elements. The geometry of the structure is represented with 320 quadratic NURBS elements, i.e., 40 longitudinal ones and 8 in circumferential direction, resulting in 714 control points. The structural model is based on a non-linear isogeometric shell approximation, presented in [22, 58]. For the comparison of the iterative procedures computations with 80 time steps with $\Delta t = 0.0001$ were conducted. Snapshots of the deformation are presented in Fig. 6.6.

The number of coupling iterations for the iterative solution procedures introduced in Sec. 4.3.3 over the conducted 80 time steps is depicted in Fig. 6.6. The fixed relaxation parameter of $\omega = 0.05$ had to be chosen to avoid divergence of the coupling iterations. With the methods shown here, it can be observed that the more information is included

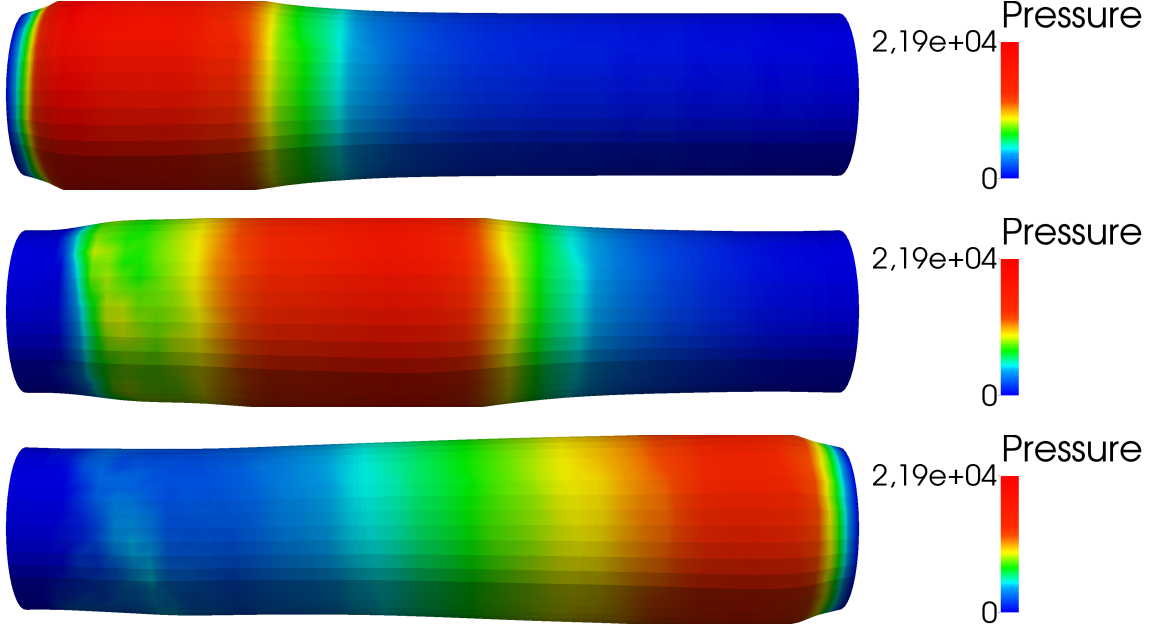


Figure 6.5: Illustration of the pressure peak running through the tube.

into the prediction of the next fixed point, the fewer coupling iterations are required. This holds for comparing Aitken's method with fixed underrelaxation, but also IQN-ILS to Aitken, and IQN-IMVJ to IQN-ILS. The result of the new IQN-IMVLS method is not included in the figure, as it is almost identical to the result of the IQN-IMVJ method. This will be confirmed in the following.

However, to measure the performance of the solvers, a comparison of coupling iterations is not the only criterion. In addition, the computational effort additionally required to compute the prediction is crucial. Thus, the speed-up in computational time is compared for the particular case in 6.3, including also results of IQN-ILS using q input/output matrices of previous time steps. Note that results are strongly depending on the configuration, since the overall speed-up depends on the size of the single-field problems in relation to the size of the interface problem. For the constant relaxation and Aitken's method, the additional computational costs for the prediction of the fixed point iteration are negligible. The relative speed-up in terms of coupling iterations is comparable to the speed-up in terms of computational cost for these methods. This can be expected because the prediction is based on simple mathematical operations. For IQN-ILS, the additional costs are increasing with a growing number of additional time steps that are taken into account. Comparing the different versions of IQN-ILS, the

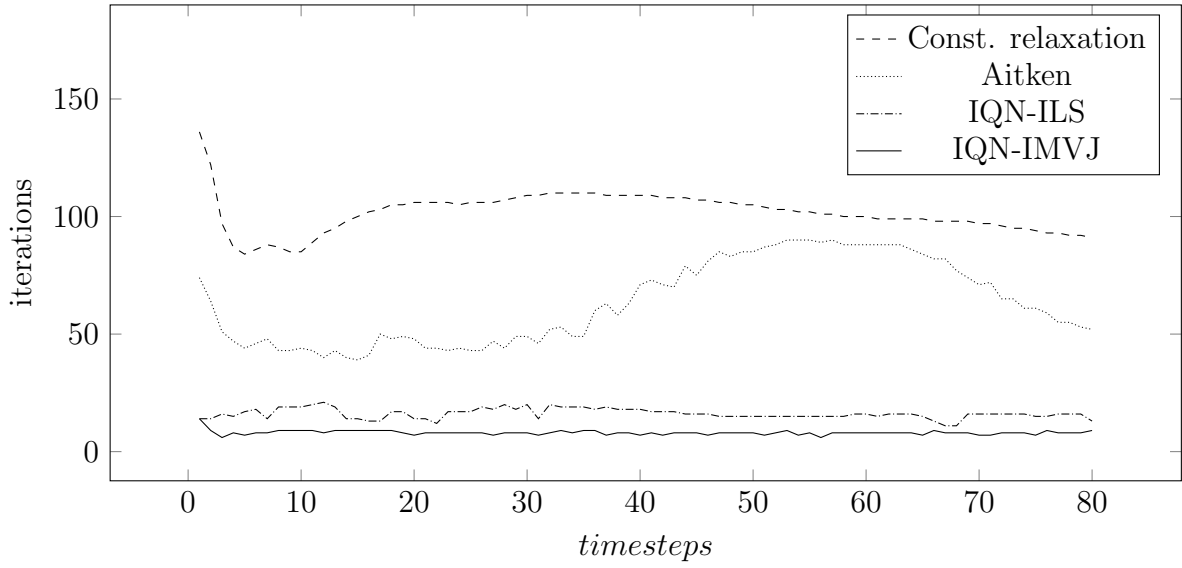


Figure 6.6: Coupling iterations for different types of acceleration.

choice of 2 time steps is optimal. It is well known that this optimum is usually flat but also case-dependent and varying over time [51, 112]. In contrast, the IQN-IMV methods do not require an additional control parameter. A comparable speed-up in terms of coupling iterations is obtained by both inverse multiple-vector methods. However, the speed-up in terms of computational costs is much lower than the one in terms of coupling iterations for IQN-IMVJ. This results from the explicit reconstruction of the update of the approximated inverse Jacobian. By solving just one least-squares problem with IQN-IMVLS, the computational costs can be reduced significantly. More details about the implementation into the partitioned solver and further comparison can be found in [142].

method	average coupling iterations	speed-up in coupling iterations [%]	speed-up in computational time [%]
Underrelaxation $\omega = 0.05$	101.63	100.00	100.00
Aitken	63.35	62.33	68.06
IQN-ILS ($q=0$)	16.01	15.75	18.62
IQN-ILS $q=2$	8.01	7.97	9.34
IQN-ILS $q=5$	8.23	8.09	9.89
IQN-ILS $q=10$	8.44	8.30	10.44
IQN-ILS $q=20$	8.49	8.35	10.83
IQN-IMVJ	8.05	8.00	13.53
IQN-IMVLS	8.14	7.92	10.27

Table 6.3: Speed-up in terms of coupling iterations and computational time. The number of previous time-steps additionally taken into account in case of IQN-ILS are denoted by q .

7 Transfer of advantages of spline-based methods to FSI

In the validation test cases presented in Ch. 5, it was demonstrated that the FSI coupling benefits from the gap-less coupling between NURBS-enhanced finite elements and isogeometric finite elements. This chapter focuses on transferring advantages from isogeometric analysis to fluid-structure interaction problems. While the first case is focusing on fluid-structure interaction with mechanical contact, the second test case demonstrates how the spline-based representation of the structure can simplify the simulation of free-surface flows with flexible structures.

7.1 Falling ring

In all test cases presented so far, the structure was clamped in a way that an interaction with the surrounding walls is avoided. Thus, it was not necessary to include any collision detection into the computational analysis. However, in many technical applications with fluid-structure phenomena, the contact between solids is part of the problem, e.g., in the case of heart valves [85]. Isogeometric analysis is attractive for such contact problems, since with the spline-based representation of the geometry, a continuous description of the distance between potential contact partners is provided. In contact cases, the smooth geometry of the splines can minimize oscillation, e.g., in ironing problems [49, 167]. Hence, the application of the NEFEM-IGA coupling might be beneficial if contact is involved in fluid-structure interaction. This will be studied in the following on a simple test case. As a simplification, the contact formulation is restricted to normal contact, while frictional effects are neglected. Additionally, only one of the contact partners is regarded as elastic. Before the results of the test case are presented, a brief introduction into the applied contact formulation is given.

7.1.1 Contact formulation

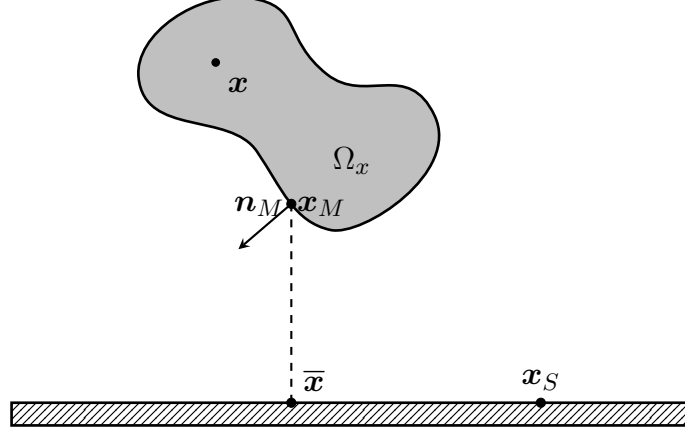


Figure 7.1: Schematic view of contact problem between elastic body and rigid wall.

The contact between two bodies can be detected due to the normal distance g_N . The distance is determined by defining a master body, which is here the elastic body, while the wall is denoted as the slave body. For all points $\mathbf{x}_M$ on the boundary Γ_M of the master body, a projection point $\bar{\mathbf{x}}$ on the boundary Γ_S of the slave body with minimal distance can be determined by orthogonal projection, eventually fulfilling the equation

$$\bar{\mathbf{x}}(\mathbf{x}_M) = \operatorname{argmin}_{\mathbf{x}_S \in \Gamma_S} |\mathbf{x}_S - \mathbf{x}_M|. \quad (7.1)$$

As for the transfer of the nodal forces presented in Sec. 4.2, the orthogonal projection can be computed in case of splines by the projection method suggested in [86]. Finally, the normal distance g_n is given by

$$g_n = (\bar{\mathbf{x}} - \mathbf{x}_M) \cdot \mathbf{n}_M, \quad (7.2)$$

where $\mathbf{n}_M$ denotes the outward-facing normal on the surface of the deformable master object.

The contact problem can be defined based on geometrical constraints. Hence, mutual penetration is prevented by the constraint to the normal distance $g_n \geq 0$. The subset

of Γ_M with $g_n = 0$ is defined as region of active contact. For this subset, additional contact forces need to be included in the structural problem. If frictional effects are neglected, only normal stresses $T_n = \mathbf{n} \cdot \mathbf{T} \mathbf{n}$ contribute to the additional contact force, while tangential stresses $\mathbf{T}_t = \mathbf{T} \mathbf{n} - T_n \mathbf{n}$ are set to zero. This results in the Hertz-Signorini-Moreau contact conditions for frictionless contact, which are introduced as an additional constraint in the structural problem

$$g_n \geq 0, \quad T_n \leq 0, \quad g_n T_n = 0, \quad \mathbf{T}_t = \mathbf{0} \text{ on } \Gamma_C. \quad (7.3)$$

The third condition — the complementary condition — acts as a trigger for contact states [159, 162]. One method to treat the contact condition is the penalty method, where the violation of the no-penetration condition is penalized by:

$$\langle T_n \rangle = \begin{cases} \epsilon g_n, & \text{if } g_n < 0, \\ 0, & \text{else.} \end{cases} \quad (7.4)$$

where ϵ denotes the penalty factor the contact constraints are scaled with. Macaulay brackets guarantee that the contribution is only added for violations of the penetration conditions [116]. A larger penalty parameter enhances the satisfaction of the no-penetration condition, but it simultaneously degrades the condition of the system of equations, which may lead to problems with the linear solver. However, the main advantage of the penalty method is that no additional degrees of freedom are introduced.

Since the normal distance g_n depends on the current position of the contact surface of the deforming object, it depends on the displacement field and, thus, leads to a nonlinear contribution to momentum equation. The weak form of the penalty constraint and its linearization is given, e.g., in [159]. A detailed discussion about isogeometric analysis in contact problems can be found in [116]. Further details of the particular implementation are presented in [77].

7.1.2 Configuration and results

In the presented test case, a flexible structure is immersed in an initially static fluid. Due to gravity the structure moves towards the wall, and after some time, it impacts the rigid wall. As before, the fluid within the ring is not regarded. All geometrical dimensions are

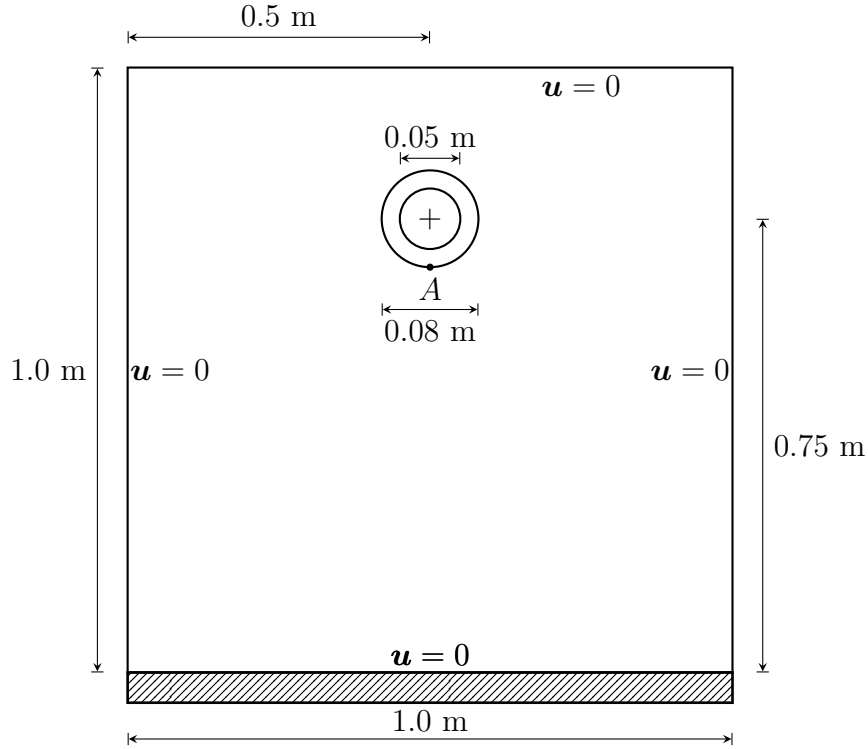


Figure 7.2: Configuration and dimensions for fluid-structure interaction with contact.

given in Fig. 7.1. The physical properties of the elastic structure and the surrounding fluid are listed in Tab. 7.1.

parameter	symbol	value
dynamic viscosity	μ	0.0000182 kg/ms
density (fluid)	ρ^f	1.2041 kg/m^3
Young's modulus	E	$69 \cdot 10^9 \text{ Pa}$
Poisson number	ν^s	0.34
density (structure)	ρ^s	2730 kg/m^3
penalty parameter	ϵ	$1.4 \cdot 10^{10} \text{ Pa}$

Table 7.1: Falling ring test case parameters.

The structure is made of aluminium. The fluid grid in the initial configuration contains approx. 6,000 elements, while the cylinder surface is discretized with 112 elements.

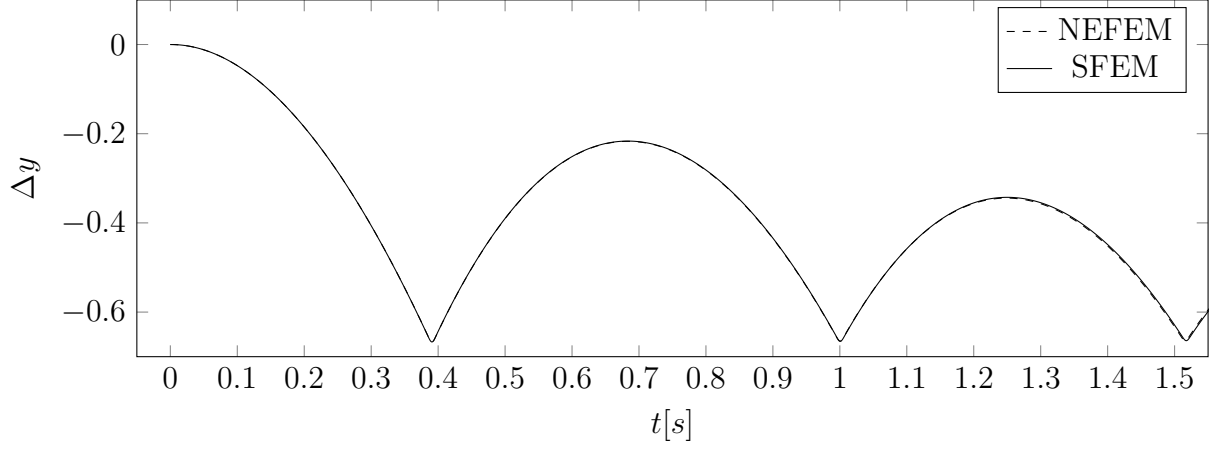


Figure 7.3: Displacement of point A over time.

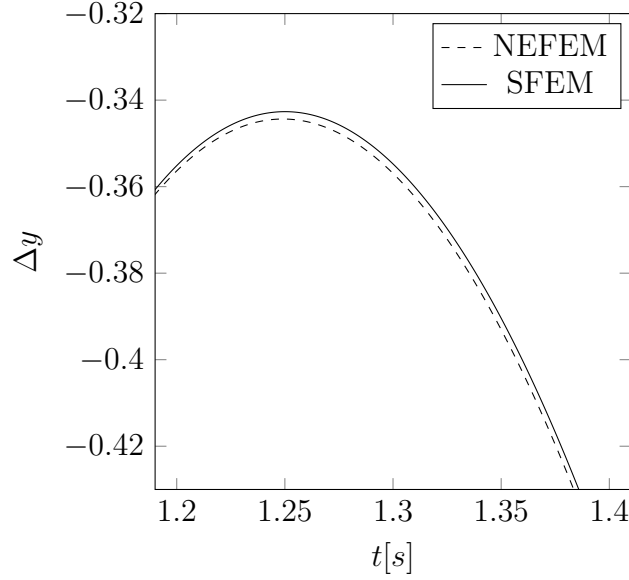


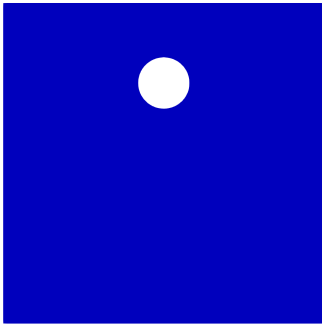
Figure 7.4: Close-up of displacement of point A over time.

Since the test case is computed with the grid deformation technique presented in [77], the number of elements in the domain can slightly vary during the computation. On all walls no-slip boundary conditions are applied. The structure is composed of 225 quadratic NURBS elements. As time step $\Delta t = 0.005 \text{ s}$ is chosen, while the gravity is 9.81 m/s^2 .

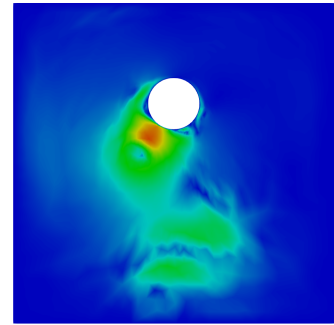
The test case was computed with both NURBS-enhanced and linear Lagrangian finite elements on the fluid side. In Fig. 7.3, the displacement of the falling ring measured

at point A is compared. For both cases the movement is clearly damped by the fluid forces. This can be expected, since energy of the moving ring is transferred to the surrounding fluid due to the no-slip boundary condition. This results in a motion of the fluid presented by snapshots in Fig. 7.5. However, the comparison of the ring motion shows a derivation of both applied methods, which is confirmed by the close-up in Fig. 7.4.

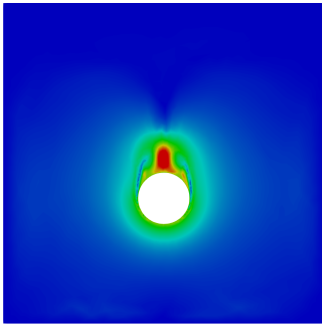
While in the flag test case the excitation continues, here the fluids lead to a continuous damping of the motion and more obvious difference between the displacements. Since there is no additional energy source, the difference increases over the time steps. Thus, the test case demonstrates how a small improvement of accuracy can lead to a clear absolute difference in the results.



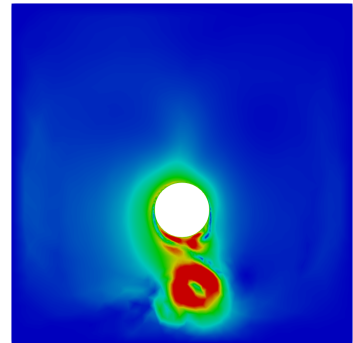
(a) $t = 0.0$ s



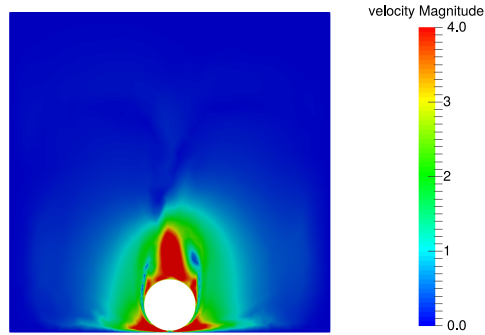
(b) $t = 0.725$ s



(c) $t = 0.275$ s



(d) $t = 0.5$ s



(e) $t = 0.375$ s

Figure 7.5: Snapshots of contact situation.

7.2 Sloshing tank

In many engineering problems, elastic structures are interacting with more than one fluid. Ships and off-shore wind turbines are just two usual examples, where elastic structures are exposed to water and air. Another one are storage tanks, mostly cylindrical structures containing fluids like oil and air. If these tanks are excited, e.g., by seismic loading, the stored fluid starts to slosh, resulting in an additional dynamic loading on the structure. In the worst case, this loading can lead to the failure of the structure and, e.g., to a catastrophic impact on the environment. Typically, the bearing structure of these tanks is using thin-walled structures that can be modeled by shell theory. Since this theory can directly be applied to spline surfaces, isogeometric analysis is very suitable for the structural analysis.

In the following, the boundary conditions for the interaction between a thin-walled structure and a free-surface flow are derived. Since shell theory is applied, the computational domain of the structural problem is reduced to the surface $\Omega_s = \Gamma_s = \mathbf{C}(\theta)$. The elastodynamic problem is solved with NURBS-based isogeometric analysis, the fluid problem with standard finite elements. Further benefits of the spline-based geometry for the solution of the overall problem are derived. Finally, the methodology is demonstrated on an excited storage tank.

7.2.1 On boundary conditions

Often, a two-phase flow can be simplified by neglecting the influence of one of the phases on the other one. In such cases only one fluid is studied, while the two-phase interface is simplified by a free-surface boundary condition, applying (2.58) by cancelling the stresses of the neglected fluid. Additionally, if large fluid domains are studied, capillary effects often only play a minor role. A slip condition is then a proper choice on the walls. The resulting interface conditions on the deforming walls can be stated as:

$$\mathbf{u}_f \cdot \mathbf{n}_s = \mathbf{u}_s \cdot \mathbf{n}_s, \quad (7.5a)$$

$$\mathbf{T}_s \mathbf{n}_s \cdot \mathbf{n}_s = \mathbf{T}_f \mathbf{n}_s \cdot \mathbf{n}_s, \quad (7.5b)$$

$$\mathbf{T}_f \mathbf{n}_s \cdot \mathbf{t}_s = \mathbf{T}_s \mathbf{n}_s \cdot \mathbf{t}_s = 0, \quad (7.5c)$$

$$\mathbf{T}_f \mathbf{n}_s \cdot \mathbf{b}_s = \mathbf{T}_s \mathbf{n}_s \cdot \mathbf{b}_s = 0, \quad (7.5d)$$

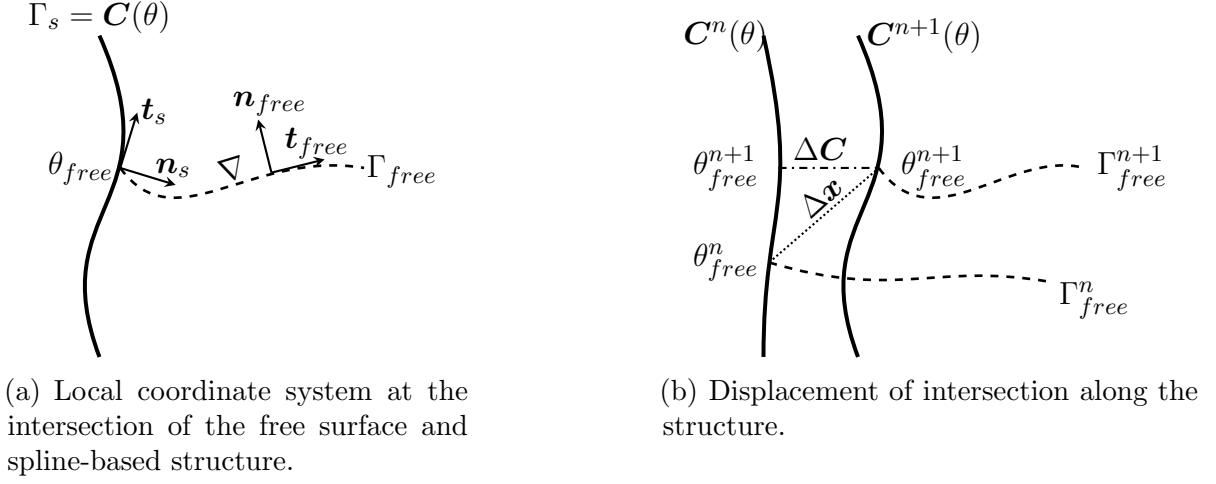


Figure 7.6: Definitions of coordinate systems.

with the normal, tangent and bi-tangent vectors $\mathbf{n}_s$, $\mathbf{t}_s$ and $\mathbf{b}_s$ on the spline-based boundary, cf. Fig. 7.6(a). The first equation represents the modified kinematic coupling condition, which allows for an arbitrary fluid velocity in tangential direction. The following equations (7.5b)-(7.5d) represent the dynamic coupling condition and restrict the effective direction of the stresses in normal direction. Note that in case of fluid-structure interaction with a slip condition the tangential stress vanishes on both sides of the interface.

Since an interface-tracking approach is applied again, the computational grid needs to conform with the moving interfaces of the free surface and the moving walls. For this the grid is deformed with the pseudo-elastic mesh deformation approach. To avoid stretched cells due to the free-surface movement close to the wall, a slip boundary condition is also applied to the pseudo-elastic solid. This choice ensures the first kinematic boundary condition. Although the fluid domain is sliding along the solid interface, the condition

$$\Omega_f \cup \Omega_s = \Gamma_{fs}, \quad (7.6)$$

still needs to be fulfilled, with $\Omega_f = \Omega_{\#}$. The matching of the surface movement can be restricted just to the normal direction:

$$\Delta \mathbf{x}_{\#} \cdot \mathbf{n}_s = \Delta \mathbf{x}_s \cdot \mathbf{n}_s. \quad (7.7)$$

This allows fluid nodes to slide freely along the walls, but still keeps the domains matching at the interface. As for the velocity on the free surface, several options

are possible to compute the normal displacement. Also in this case, a proper choice is

$$\Delta \mathbf{x}_\# = (\Delta \mathbf{x}_s \cdot \mathbf{n}_s) \mathbf{n}_s. \quad (7.8)$$

Finally, both boundary conditions for the grid displacement (3.84) and (7.8) have to be fulfilled, at the intersection between the free-surface boundary Γ_{free} and the deforming wall Γ_{fs} . Since the movement of the free surface is restricted by the wall, the tangential displacement of the mesh along the wall can be stated as

$$\Delta \mathbf{x}_\# \cdot \mathbf{t}_s = \frac{(\mathbf{u}_{free} \cdot \mathbf{n}_{free}) \Delta t}{\cos(\angle \mathbf{t}_s \mathbf{n}_{free})} = \frac{(\mathbf{u}_{free} \cdot \mathbf{n}_{free}) \Delta t}{\mathbf{t}_s \cdot \mathbf{n}_{free}}. \quad (7.9)$$

Note that the equation is only valid if $\mathbf{t}_s \cdot \mathbf{n}_{free} \neq 0$. The complete boundary condition at the intersection is given by the sum of (7.8) and (7.9). Since the effective directions are orthogonal, the application of the boundary in terms of the algorithm sketched below is simplified. Hence, the full set of boundary conditions for the pseudo-elastic mesh approach can be stated as:

$$\Delta \mathbf{x}_\# = \begin{cases} (\Delta \mathbf{x}_s \cdot \mathbf{n}_s) \mathbf{n}_s & \text{on } \Gamma_{fs}, \\ (\mathbf{u}_{free} \cdot \mathbf{n}_{free}) \mathbf{n}_{free} \Delta t & \text{on } \Gamma_{free}, \\ \frac{(\mathbf{u}_{free} \cdot \mathbf{n}_{free}) \Delta t}{\mathbf{t}_s \cdot \mathbf{n}_{free}} \mathbf{t}_s + (\Delta \mathbf{x}_s \cdot \mathbf{n}_s) \mathbf{n}_s & \text{on } (\Gamma_{free} \cap \Gamma_{fs}). \end{cases} \quad (7.10)$$

Note that on the intersection between the free-surface and the moving wall, the mesh nodes are still allowed to move in bi-tangent direction.

To apply the slip condition in both solvers for the fluid problem and the pseudo-elastic problem, an algorithm suggested by Behr in [17] based on [63] is utilized. Similar to [62], the algorithm can be summarized with respect to (7.5a) and (7.7) as:

1. Rotate the equation of the unknown into the local tangential coordinate system.
2. Adapt the equation responsible for the normal and tangential components based on the structural deformation.
3. Compute the remaining components.

Obviously, the interface conditions are strongly depending on the geometry. The derivation of normal and tangent vector is essential. Here, the NURBS-based representation of the geometry can be utilized. With that, the required vectors are continuous functions, at least if the geometry is given by a continuous NURBS. How the tangential system can be constructed is presented in [62].

Both sets of boundary conditions for the fluid and the mesh problem are depending on the structural solution. In case of the mesh problem, the displacement at the interface can be determined by evaluating the spline at subsequent time steps on the parametric coordinate of the current position of the fluid points:

$$(\Delta \mathbf{x}_\# \cdot \mathbf{n}_s) \mathbf{n}_s = ((\mathbf{C}^{n+1}(\theta^{n+1}) - \mathbf{C}^n(\theta^{n+1})) \cdot \mathbf{n}_s) \mathbf{n}_s. \quad (7.11)$$

The parametric coordinate of each fluid point is given by the inverse mapping $\theta^{n+1} = \Phi^{-1}(\mathbf{x}^{n+1})$. But even if the algorithm above utilizes the spline-based geometry to derive the local coordinate system, the solution of the system of equations does not respect the exact geometry. Hence, all displaced coordinates computed by the pseudo-elastic problem are located in the plane spanned by the tangential vectors. In case of curved geometries, the new coordinates do not lie on the NURBS anymore. By a linearization of the spline, the tangential displacement in $\mathbf{x}$ -coordinates can be adjusted [62]:

$$\Delta \mathbf{x}_{k+1}^\# = \mathbf{x}_{k+1}^\# - \mathbf{x}_k^\# \stackrel{!}{=} \mathbf{C}(\theta_{k+1}) - \mathbf{C}(\theta_k) \approx D\mathbf{C}\Delta\theta, \quad (7.12)$$

while $D\mathbf{C}$ denotes the local Jacobian of the NURBS. The update of the parametric coordinates can be compute by $\Delta\theta = D\mathbf{C}^{-1}\Delta\mathbf{x}_{k+1}^\#$, while using the tangential displacement in $\mathbf{x}$ -coordinates as input. The adjusted coordinate can now be evaluated with $\mathbf{x}_{k+1}^\# = \mathbf{C}(\theta_{k+1}) = \mathbf{C}(\theta_k + \Delta\theta)$. Note that the subscript k denotes the iteration in the sense of coupling iterations. The position of the shifted fluid node can now be computed by evaluating the current spline at the new parametric coordinate. While the Jacobian can easily be computed for curves, a derivation of the Jacobian of a NURBS surface embedded in a three-dimensional space is more complex, as presented in [62].

Finally, the normal velocity in (7.5a) has to be determined analogously to (4.6) by the boundary displacement:

$$(\mathbf{u}_{free} \cdot \mathbf{n}_s) \mathbf{n}_s = \left(\frac{\partial \mathbf{d}_s}{\partial t} \cdot \mathbf{n}_s \right) \mathbf{n}_s. \quad (7.13)$$

However, by applying the slip condition for the pseudo-elastic problem, the displacement of the fluid points is not suitable anymore to compute the interface velocity. Since the points are moving freely along the wall, the resulting relative movement can lead to artificial velocity, e.g., when the grid is moving along a curved but currently non-moving structure. Since the local normal vector is not orthogonal to the displacement of the grid points anymore, a non-zero normal velocity would be computed even in that case.

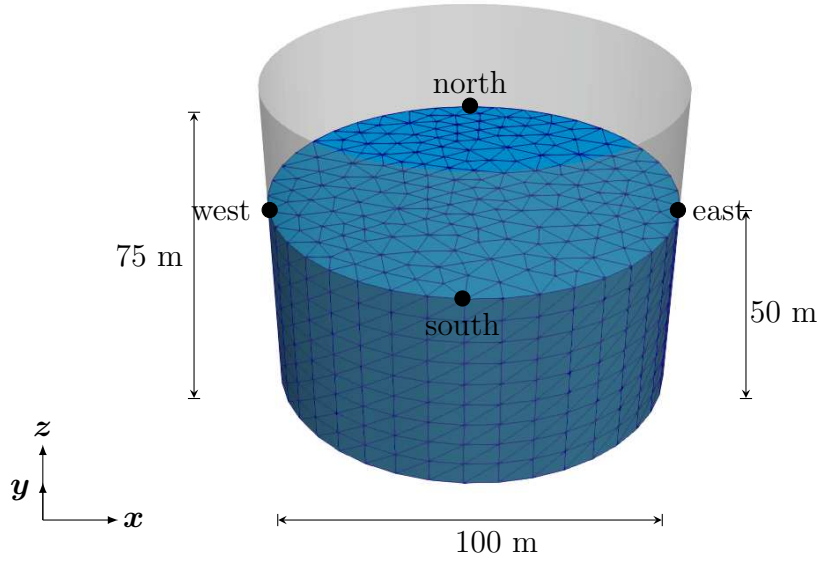


Figure 7.7: Initial configuration of the sloshing tank.

Hence, mass conservation would be violated. Instead, the velocity can be determined by the current movement of the spline and the parametric coordinate of every fluid point. By using the inverse mapping already mentioned above, the velocity at all fluid points on the interface can be computed by:

$$\frac{\partial \mathbf{d}_s}{\partial t} \approx \frac{\mathbf{C}^{n+1}(\theta^{n+1}) - \mathbf{C}^n(\theta^{n+1})}{\Delta t}. \quad (7.14)$$

7.2.2 Configuration and results

The previously introduced boundary conditions are tested on the example of a sloshing tank. The initially steady configuration is excited by a time-dependent movement of the tank bottom. The dimensions of the studied tank are depicted in Fig. 7.7. The material parameters are patterned after an elastic tank structure that carries oil. The particular material parameters are listed in Tab. 7.2.

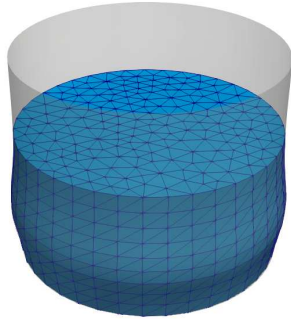
The tank is excited by a prescribed displacement in $\mathbf{x}$ -direction of the tank bottom, following the function $d_x = 10 \cdot \sin(\frac{1}{20} \cdot \pi \cdot t)$. The computation is started, based on a previously computed equilibrium state with respect to gravity. As a further simplification, surface tension is neglected.

parameter	symbol	value
dynamic viscosity	μ	0.4 kg/ms
density (fluid)	ρ^f	900 kg/m^3
Young's modulus	E	$2.15 \cdot 10^9 \text{ Pa}$
Poisson number	ν^s	0.3
density (structure)	ρ^s	7580 kg/m^3
gravity	g	9.81 m/s^2

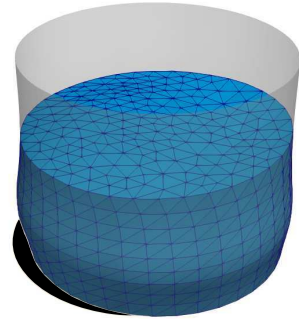
Table 7.2: Parameters of the sloshing tank test case.

The fluid domain is discretized by approximately 8,000 tetrahedral space-time elements. A Reissner-Midlin shell element [58] is utilized for modeling the tank structure, while a coarse discretization of 12 quadratic NURBS elements is applied. The simulation is conducted with a time-step size of $\Delta t = 0.01 \text{ s}$

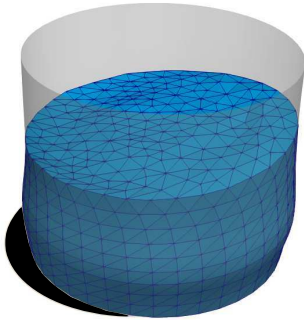
Due to the time-dependent displacement of the tank bottom, the whole configuration starts moving and immediately the fluid begins to slosh. The snapshot series in Fig. 7.8 shows several states during the first period of the excitation. The vertical movement of the free surface is measured for the points *West* and *East* in Fig. 7.12. The points are moving more or less in an opposite way. However, waves on the free surface lead to additional oscillations. With the moving fluid, time-dependent loadings on the structure emerge, resulting in a directly related deformation. The deformation of the structure is measured in x -direction at points *West* and *East*, cf. Fig. 7.9 and Fig. 7.10, and in y -direction at points *East* and *South*, cf. Fig. 7.11. In all figures a direct reaction to the movement of the free surface can be identified. The movement of *East* and *South* is almost symmetric. Since the excitation is acting in x -direction, and due to the symmetry of the configuration, surface waves in y -direction are hardly observed. Thus, the kinematic pressure is mostly acting in x -direction, confirmed by the clearly larger deformation in points *West* and *East*.



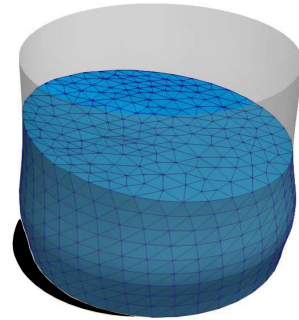
(a) Equilibrium state



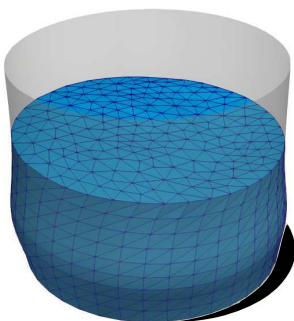
(b) $t = 5s$



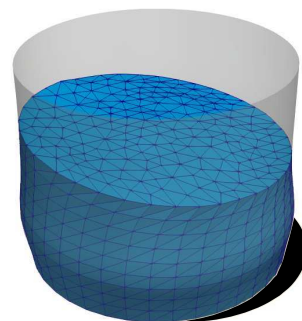
(c) $t = 10s$



(d) $t = 15s$



(e) $t = 25s$



(f) $t = 27.5s$

Figure 7.8: Snapshot series of sloshing tank. The shadow on the bottom plane depicts the initial position.

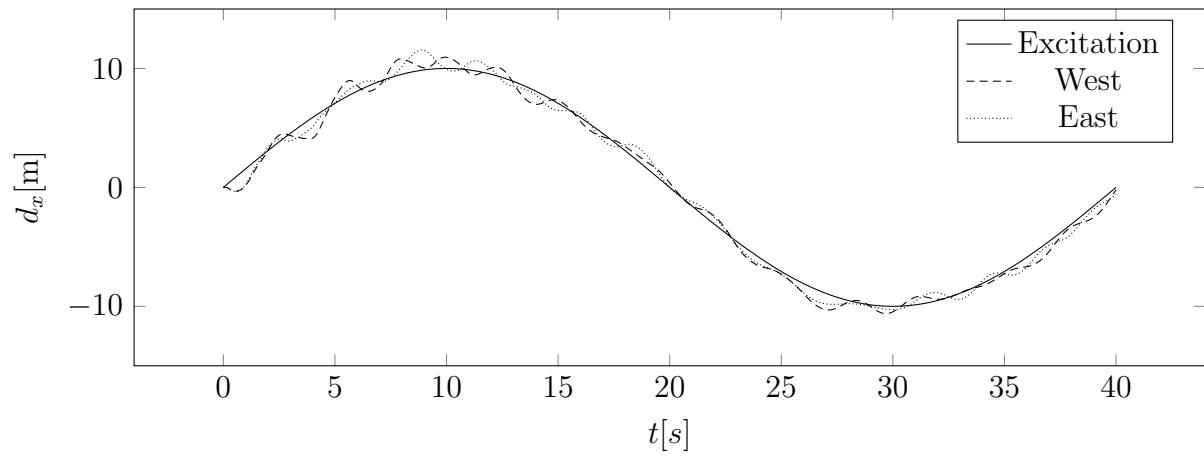


Figure 7.9: Absolute displacement in x -direction.

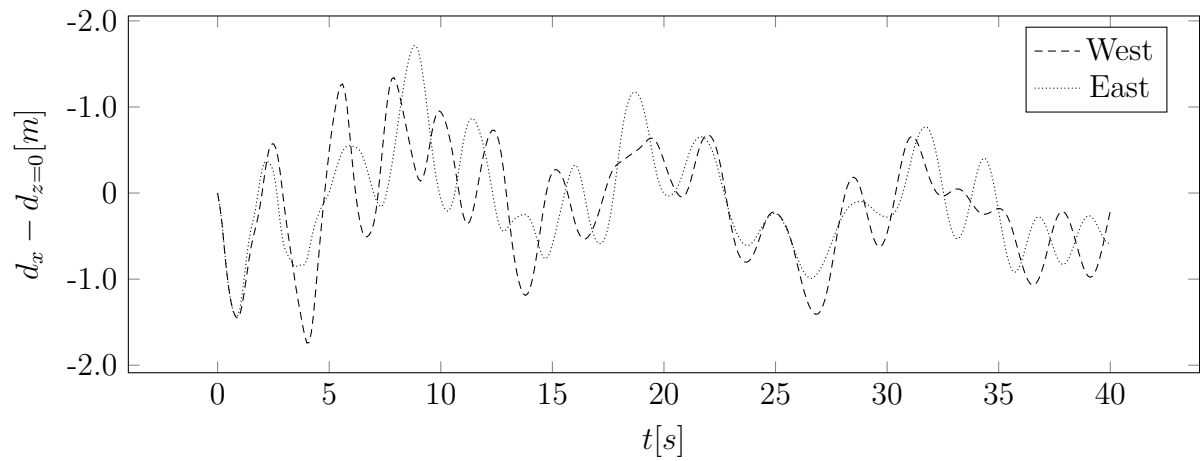


Figure 7.10: Difference of displacement in x -direction with respect to excitation.

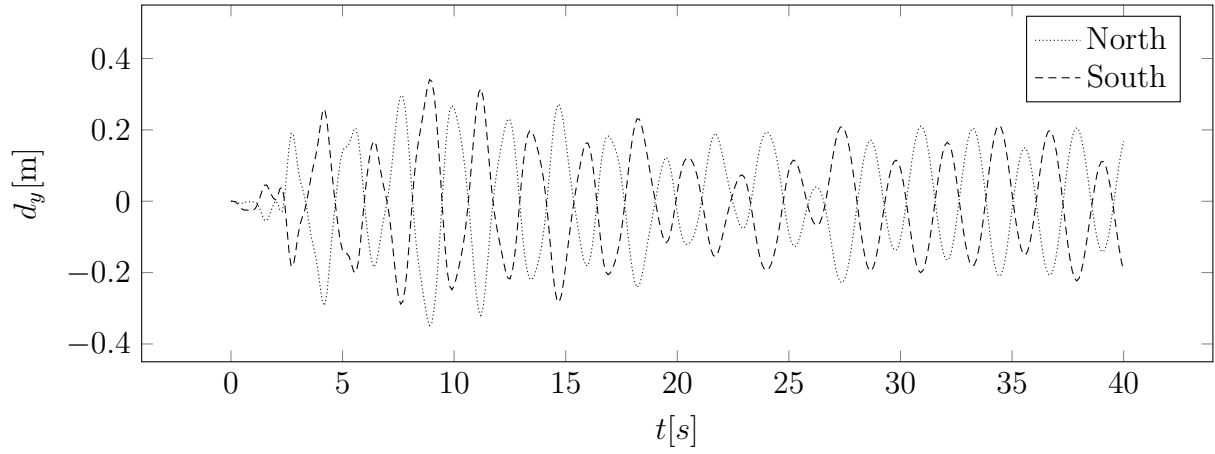


Figure 7.11: Absolute displacement in y -direction.

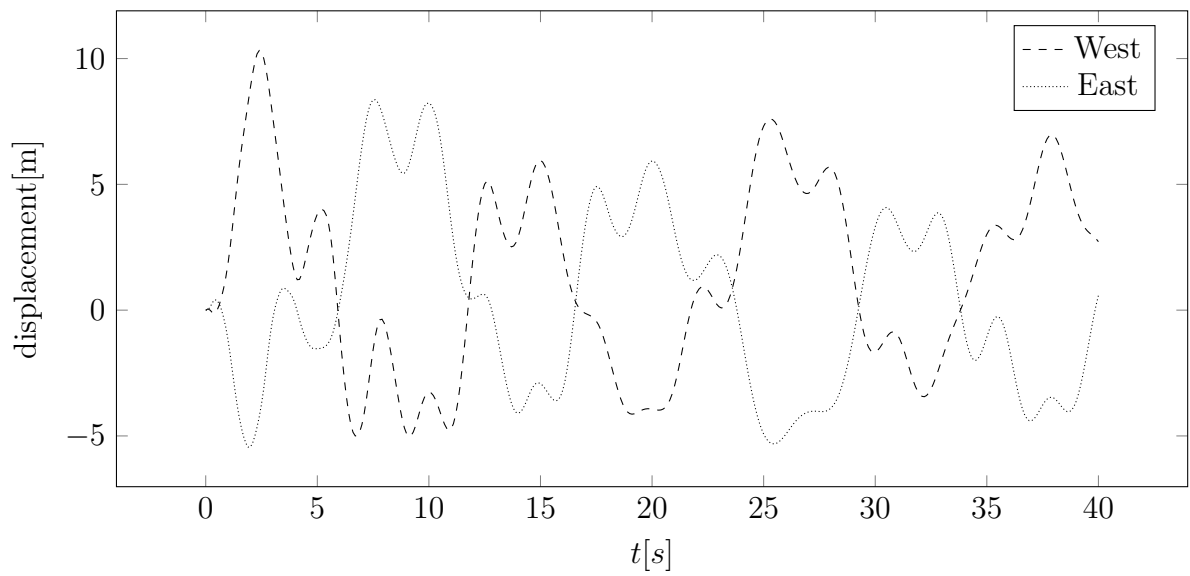


Figure 7.12: Movement of two nodes on free surface.

8 Conclusion and Outlook

8.1 Conclusion

Isogeometric analysis — a powerful spline-based method — is becoming more and more popular. In combination with shell element theory, the application of the method to complex problems of structural mechanics is straightforward. In that case, the required models can be generated using existing CAD systems. However, the generation of volume splines for complex volumetric domains is still difficult. Hence, the application of isogeometric analysis to problems on complex domains is not yet feasible. This would be particularly necessary for flow problems. Instead, the NURBS-enhanced finite element method provides a suitable alternative. The method involves the NURBS-based boundary into numerical integration of the elements in the vicinity of walls, while using standard finite elements in the interior of the domain. In contrast to NURBS-based isogeometric analysis, existing tools can be used for grid generation. The aim of the presented thesis was to identify advantages of spline-based methods for solving fluid-structure interaction problems. In particular, the coupling between NURBS-enhanced finite elements on the fluid side and NURBS-based isogeometric analysis on the solid side was discussed.

First, a new geometrical mapping for NURBS-enhanced finite elements was derived and explained step-by-step. This mapping was implemented into an existing deformable spatial domain/stabilized space-time finite element flow solver. Afterwards the NURBS-enhanced finite elements for solving incompressible flow problems were combined with isogeometric analysis on the structural side for solving fluid-structure interaction problems. Although that coupling can be realized with any solution strategy, it was implemented into a partitioned approach following an iteratively staggered scheme. For that, the space-time finite element flow solver was combined with a semi-discrete structural solver extended by isogeometric analysis. The spatial and temporal coupling between both solvers and their underlying methods were discussed in detail. On the one hand,

the coupling is still a non-matching interface discretization with Lagrangian finite elements on the fluid side and NURBS-based isogeometric analysis on the structural side. But on the other hand, the coupling of both methods leads to a gap-less connection in terms of a matching interface representation due to the utilization of the same NURBS on both sides of the interface. Based on that, simplifications of the spatial coupling could be derived. On the one hand, fluid tractions can be integrated directly into consistent forces on the control points, acting as the right-hand side loads of the structural problem. On the other hand, the displacement of the points on the interface in the fluid grid can be computed by the evaluation of the deformed spline. It was shown, that the method is consistent and globally conservative. Since the implementation of NURBS-enhanced finite elements only involves the geometrical mapping of the elements in the vicinity of the interface, the implementation effort to get a consistent and conservative spatial coupling is reasonable. The same holds for the additional computational effort, as only additional quadrature points need to be included in the enhanced elements during assembly.

With steady and transient two-dimensional numerical cases, the benefit of the coupling, in comparison to the usage of standard finite elements on the fluid side, was verified. Since the fluid equations are still discretized with linear Lagrangian finite elements, the convergence order is unaffected. However, the geometrically exact mapping with respect to the spline boundary can lower the absolute error. In particular, it was shown that the benefit is proportional to the deformation of the observed structure. A remarkable result was presented with a balloon-like test case as an example for closed domains with deformable boundaries. Here, NURBS-enhanced finite elements are able to outperform standard finite elements. Since the geometry is perfectly approximated, the solution is only depending on the discretization error in time. For comparable results, a much finer spatial discretization would be required when using standard finite elements.

The iteratively staggered solution approach was combined with several acceleration methods. As expected, the interface quasi-Newton techniques show for the spline-based methods the same advantages known from literature for standard finite elements. Additionally, a new method that combines ideas of two existing interface quasi-Newton methods was derived. With the new method it was possible to counteract the main drawbacks of these methods regarding efficiency and case dependency.

Finally, a further advantage of a spline-based model on the structural side was introduced in context of fluid-structure interaction involving free-surface flows. The existing meth-

ods for arbitrary rigid wall shapes were extended for moving walls and fluid-structure interaction. In a three-dimensional test case of a sloshing tank, the functionality of the method was demonstrated.

8.2 Outlook

The application of the presented methods might be interesting for further application in more complex problems. First of all, the coupling of the spline-based methods might be beneficial for problems that are more sensitive to an exact representation of the interface, e.g., shock-wall interaction or flow separation in deformable rocket nozzles in compressible regimes. Additionally, it can be expected that the benefit of the coupling between NURBS-enhanced finite elements and isogeometric analysis increases if it is applied to three-dimensional problems. This would also allow to use the coupling for fluid-structure interaction involving free-surface flows, e.g., for the presented sloshing tank. The remarkable results on closed domains should be studied for more complex cases and might be transferred to cases, e.g., involving compressible fluids. In case of the presented balloon test case, the differences were driven by the different approximation of the geometry and the prescribed mass flow. In case of compressible fluids, the additional influence of prescribed energy might be interesting. Eventually, the existing partitioned solver could be integrated into multidisciplinary shape optimization utilizing the parameterized geometry on both sides of the problem.

Furthermore, the presented methods can be extended in several directions. For example, the presented geometrical mapping and the related coupling methods can be extended directly to other spline types, such as locally refined B-Splines. This would allow for transferring advantages of these spline types to fluid-structure interaction problems. As already presented in the thesis, the transfer matrices involved in the spatial coupling allow for a direct evaluation of fluid stresses along the interface. This might be interesting when combining it with isogeometric collocation, but might be also interesting for fluid-fluid coupling with internal interfaces. And finally, the benefit of the modified interface Quasi-Newton method should be confirmed by a detailed analysis of the computational complexity.

Bibliography

- [1] C. Adam, S. Bouabdallah, M. Zarroug, and H. Maitournam. Improved numerical integration for locking treatment in isogeometric structural elements. Part II: Plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 284(Supplement C):106–137, 2015.
- [2] C. Allen and A Rendall. Unified approach to CFD-CSD interpolation and mesh motion using radial basis functions. In *25th AIAA Applied Aerodynamics Conference, Fluid Dynamics and Co-located Conferences*. American Institute of Aeronautics and Astronautics, 2007.
- [3] S. Badia, F. Nobile, and C. Vergara. Fluid–structure partitioned procedures based on Robin transmission conditions. *Journal of Computational Physics*, 227(14):7027–7051, 2008.
- [4] J.D. Banks, W.D. Henshaw, and D.W. Schwendeman. An analysis of a new stable partitioned algorithm for FSI problems. Part II: Incompressible flow and structural shells. *Journal of Computational Physics*, 268:399–416, 2014.
- [5] J.W. Banks, W.D. Henshaw, and D.W. Schwendeman. An analysis of a new stable partitioned algorithm for FSI problems. Part I: Incompressible flow and elastic solids. *Journal of Computational Physics*, 269:108–137, 2014.
- [6] G.K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, 2000.
- [7] K.-J. Bathe. *Finite-Elemente-Methoden*. Springer Berlin Heidelberg, 1989.
- [8] Y. Bazilevs, L. Beira da Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli. Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07):1031–1090, 2006.

- [9] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, and Y. Zhang. Isogeometric fluid-structure interaction: theory, algorithms, and computations. *Computational Mechanics*, 43(1):3–37, 2008.
- [10] Y. Bazilevs, V.M. Calo, Y. Zhang, and T.J.R. Hughes. Isogeometric fluid-structure interaction analysis with applications to arterial blood flow. *Computational Mechanics*, 38(4-5):310–322, 2006.
- [11] Y. Bazilevs, M.-C. Hsu, I. Akkerman, S. Wright, K. Takizawa, B. Henicke, T. Spielman, and T. E. Tezduyar. 3d simulation of wind turbine rotors at full scale. Part I: Geometry modeling and aerodynamics. *International Journal for Numerical Methods in Fluids*, 65(1-3):207–235, 2011.
- [12] Y. Bazilevs, M.-C. Hsu, J. Kiendl, R. Wüchner, and K.-U. Bletzinger. 3d simulation of wind turbine rotors at full scale. Part II: Fluid-structure interaction modeling with composite blades. *International Journal for Numerical Methods in Fluids*, 65(13):236–253, 2011.
- [13] Yuri Bazilevs, Kenji Takizawa, and Tayfun E. Tezduyar. *Computational fluid-structure interaction: methods and applications*. Wiley, Chichester, West Sussex, United Kingdom, 2013.
- [14] A. Beckert. Coupling fluid (CFD) and structural (FE) models using finite interpolation elements. *Aerospace Science and Technology*, 4(1):13–22, 2000.
- [15] A. Beckert and H. Wendland. Multivariate interpolation for fluid-structure-interaction problems using radial basis functions. *Aerospace Science and Technology*, 5(2):125–134, 2001.
- [16] M. Behr. *Stabilized finite element methods for incompressible flows with emphasis on moving boundaries and interfaces*. PhD thesis, University of Minnesota, 1992.
- [17] M. Behr. On the application of slip boundary condition on curved boundaries. *International journal for numerical methods in fluids*, 45(1):43–51, 2004.
- [18] M. Behr. Simplex space-time meshes in finite element simulations. *International Journal for Numerical Methods in Fluids*, 57(9):1421–1434, 2008.
- [19] M. Behr and F. Abraham. Free-surface flow simulations in the presence of inclined walls. *Computer Methods in Applied Mechanics and Engineering*, 191(47):5467–5483, 2002.

-
- [20] M. Behr and T.E. Tezduyar. Finite element solution strategies for large-scale flow simulations. *Computer Methods in Applied Mechanics and Engineering*, 112(1-4):3–24, 1994.
- [21] M.A. Behr, L.P. Franca, and T.E. Tezduyar. Stabilized finite element methods for the velocity-pressure-stress formulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 104(1):31–48, 1993.
- [22] D.J. Benson, Y. Bazilevs, M.C. Hsu, and T.J.R. Hughes. Isogeometric shell analysis: The Reissner-Mindlin shell. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):276–289, 2010.
- [23] M. Bischoff. *Theorie und Numerik einer dreidimensionalen Schalenformulierung*. PhD thesis, Institut für Baustatik, Universität Stuttgart, 1999.
- [24] M. Bischoff, K.-U. Bletzinger, W. Wall, E. Ramm, E. Stein, R. Borst, T.J.R. Hughes, and M. Bischoff. Models and finite elements for thinwalled structures. In *Encyclopedia of computational mechanics*, volume 2. John Wiley, 2004.
- [25] F. J. Blom. A monolithical fluid-structure interaction algorithm applied to the piston problem. *Computer methods in applied mechanics and engineering*, 167(3-4):369–391, 1998.
- [26] A.E.J. Bogaers, S. Kok, B.D. Reddy, and T. Franz. Quasi-Newton methods for implicit black-box FSI coupling. *Computer Methods in Applied Mechanics and Engineering*, 279:113–132, 2014.
- [27] J. Bonet and R.D. Wood. *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 1997.
- [28] A.L. Braun and A.M. Awruch. Geometrically non-linear analysis in elastodynamics using the eight-node Finite Element with one-point quadrature and the generalized-alpha method. *Latin American Journal of Solids and Structures*, 5(1):17–45, 2008.
- [29] C. Braun. *Ein modulares Verfahren für die numerische aeroelastische Analyse von Luftfahrzeugen*. PhD thesis, RWTH Aachen University, Aachen, 2007.
- [30] A N. Brooks and T.J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1-3):199–259, 1982.

- [31] A.N. Brooks and T.J.R. Hughes. Streamline-upwind/Petrov-Galerkin methods for advection dominated flows. In E.D. Norrie, editor, *Proc. 3rd Int. Conf. on Finite Elements in Flow Problems*, volume 2, Canada, 1980.
- [32] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19(92):577, 1965.
- [33] M. Brüderlin. *A procedure for reduced order model based robust aeroelastic control*. PhD thesis, RWTH Aachen University, Aachen. expected in 2018.
- [34] M. Brüderlin, N. Hosters, and M. Behr. Robust active control of a winglet with elastic suspension at transonic flow. *Journal of Guidance, Control, and Dynamics*, 0(0):1–9. reviewed, expected in 2018.
- [35] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4506–4527, 2005.
- [36] J. R. Cebal and R. Lohner. Conservative load projection and tracking for fluid-structure problems. *AIAA Journal*, 35(4):687–692, 1997.
- [37] D. S. Chandrasekharaiah and L. Debnath. *Continuum mechanics*. Elsevier, 2014.
- [38] B. H. Chen. *Numerical analysis of the influence of a winglet actuated control surface by means of direct and reduced-order-model simulation*. PhD thesis, RWTH Aachen University, Aachen, 2014.
- [39] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-alpha method. *Journal of applied mechanics*, 60(2):371–375, 1993.
- [40] P. Ciarlet. *The finite element method for elliptic problems*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 2002.
- [41] R. Codina. Comparison of some finite element methods for solving the diffusion-convection-reaction equation. *Computer Methods in Applied Mechanics and Engineering*, 156(1-4):185–210, 1998.
- [42] C. Conca, A. Osses, and J. Planchard. Added mass and damping in fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 146(3):387–405, 1997.

-
- [43] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs. *Isogeometric analysis: toward integration of CAD and FEA*. Wiley, 2009.
 - [44] J.A. Cottrell, T.J.R. Hughes, and A. Reali. Studies of refinement and continuity in isogeometric structural analysis. *Computer Methods in Applied Mechanics and Engineering*, 196(41):4160–4183, 2007.
 - [45] J.A. Cottrell, A. Reali, Y. Bazilevs, and T.J.R. Hughes. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195(41):5257–5296, 2006.
 - [46] W. Dahmen and A. Reusken. *Numerik für Ingenieure und Naturwissenschaftler*. Springer Berlin Heidelberg, 2008.
 - [47] A. de Boer, A. H. van Zuijlen, and H. Bijl. Comparison of conservative and consistent approaches for the coupling of non-matching meshes. *Computer Methods in Applied Mechanics and Engineering*, 197(49):4284–4297, 2008.
 - [48] A. de Boer, A.H. van Zuijlen, and H. Bijl. Review of coupling methods for non-matching meshes. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1515–1525, 2007.
 - [49] L. De Lorenzis, I. Temizer, P. Wriggers, and G. Zavarise. A large deformation frictional contact formulation using NURBS-based isogeometric analysis. *International Journal for Numerical Methods in Engineering*, (87):1278–1300, 2011.
 - [50] J. Degroote, S. Annerel, and J. Vierendeels. Stability analysis of Gauss-Seidel iterations in a partitioned simulation of fluid-structure interaction. *Computers & Structures*, 88(5):263–271, 2010.
 - [51] J. Degroote, K.-J. Bathe, and J. Vierendeels. Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction. *Computers & Structures*, 87(11-12):793–801, 2009.
 - [52] J. Degroote, P. Bruggeman, R. Haelterman, and J. Vierendeels. Stability of a coupling technique for partitioned solvers in FSI applications. *Computers & Structures*, 86(23):2224–2234, 2008.
 - [53] J. Degroote, A. Swillens, P. Bruggeman, R. Haelterman, P. Segers, and J. Vierendeels. Simulation of fluid-structure interaction with the interface artificial compressibility method. *International Journal for Numerical Methods in Biomedical Engineering*, 26(3-4):276–289, 2010.

- [54] J. Degroote and J. Vierendeels. Multi-solver algorithms for the partitioned simulation of fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 200(25):2195–2210, 2011.
- [55] J. Degroote and J. Vierendeels. Multi-level quasi-Newton coupling algorithms for the partitioned simulation of fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 225-228:14–27, 2012.
- [56] J. Donea, S. Giuliani, and J.-P. Halleux. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Computer methods in applied mechanics and engineering*, 33(1-3):689–723, 1982.
- [57] J. Donea and A. Huerta. *Finite element methods for flow problems*. John Wiley & Sons, 2003.
- [58] W. Dornisch. *Interpolation of rotations and coupling of patches in isogeometric Reissner-Mindlin shell analysis*. PhD thesis, RWTH, Aachen, 2015.
- [59] W. Dornisch, S. Klinkel, and B. Simeon. Isogeometric Reissner-Mindlin shell analysis with exactly calculated director vectors. *Computer Methods in Applied Mechanics and Engineering*, 253:491–504, 2013.
- [60] F. Duarte, R. Gormaz, and S. Natesan. Arbitrary Lagrangian-Eulerian method for Navier-Stokes equations with moving boundaries. *Computer Methods in Applied Mechanics and Engineering*, 193(45-47):4819–4836, 2004.
- [61] L.A. Duarte Filho and A.M. Awruch. Geometrically nonlinear static and dynamic analysis of shells and plates using the eight-node hexahedral element with one-point quadrature. *Finite Elements in Analysis and Design*, 40(11):1297–1315, 2004.
- [62] S. Elgeti and H. Sauerland. Deforming fluid domains within the finite element method: five mesh-based tracking methods in comparison. *Archives of Computational Methods in Engineering*, 23(2):323–361, 2016.
- [63] M. S. Engelman, R. L. Sani, and P. M. Gresho. The implementation of normal and/or tangential boundary conditions in finite element codes for incompressible fluid flow. *International Journal for Numerical Methods in Fluids*, 2(3):225–238, 1982.
- [64] L.F.R. Espath, A.L. Braun, and A.M. Awruch. Isogeometric analysis applied to nonlinear elastodynamics. *Blucher Mechanical Engineering Proceedings*, 1(1):2448–2467, 2012.

-
- [65] C. Farhat, C. Degand, B. Koobus, and M. Lesoinne. An improved method of spring analogy for dynamic unstructured fluid meshes. AIAA Paper 98-2070. American Institute of Aeronautics and Astronautics, 1998.
- [66] C. Farhat, P. Geuzaine, and C. Grandmont. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *Journal of Computational Physics*, 174(2):669–694, 2001.
- [67] C. Farhat, M. Lesoinne, and P. Le Tallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157(1-2):95–114, 1998.
- [68] C. A. Felippa, K. C. Park, and C. Farhat. Partitioned analysis of coupled mechanical systems. *Computer methods in applied mechanics and engineering*, 190(24):3247–3270, 2001.
- [69] M. A. Fernández and M. Moubachir. A Newton method using exact jacobians for solving fluid-structure coupling. *Computers & Structures*, 83(2):127–142, 2005.
- [70] Chair for Computational Analysis of Technical Systems. SplineLib. expected as Open-Source Software in 2018.
- [71] C. Förster, W. A. Wall, and E. Ramm. Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 196(7):1278–1293, 2007.
- [72] Ch. Förster, W. A. Wall, and E. Ramm. On the geometric conservation law in transient flow calculations on deforming domains. *International Journal for Numerical Methods in Fluids*, 50(12):1369–1379, 2006.
- [73] S. Frauholz, N. Hosters, B. Reinartz, and M. Behr. Fluid-structure interaction in the context of a scramjet intake. In *44th AIAA Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, 2014. DOI: 10.2514/6.2014-2449.
- [74] L. Gerardo-Giorda, F. Nobile, and C. Vergara. Analysis and optimization of Robin-Robin partitioned procedures in fluid-structure interaction problems. *SIAM Journal on Numerical Analysis*, 48(6):2091–2116, 2010.

- [75] P. Hansbo, J. Hermansson, and T. Svedberg. Nitsche’s method combined with space-time finite elements for ALE fluid-structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 193(39-41):4195–4206, 2004.
- [76] J. Helmig. A contribution to geometrically exact methods for fluid-structure interaction using the NURBS-enhanced finite element method and isogeometric analysis. Master’s thesis, RWTH Aachen, Aachen, 2015.
- [77] D. Hilger. Extension of a software environment for the simulation of fluid-structure interactions with contact. Master’s thesis, RWTH Aachen, Aachen, 2017.
- [78] E. Hinton and J. S. Campbell. Local and global smoothing of discontinuous finite element functions using a least squares method. *International Journal for Numerical Methods in Engineering*, 8(3):461–480, 1974.
- [79] N. Hosters, J. Helmig, A. Stavrev, M. Behr, and S. Elgeti. Fluid-structure interaction with NURBS-based coupling. *Computer Methods in Applied Mechanics and Engineering*, 2018.
- [80] N. Hosters, M. Klaus, M. Behr, and H.-G. Reimerdes. Application of a partitioned field approach to transient aerothermal problems in rocket nozzles. *Computers & Fluids*, 88:795–803, 2013.
- [81] N. Hosters, M. Klaus, G. Schieffer, M. Behr, and H.-G. Reimerdes. Towards aerothermoelastic simulations of supersonic flow through nozzles. pages 637–654. EDP Sciences, 2013.
- [82] A. Howe. <https://www.howeart.net>, 2017.
- [83] M.-C. Hsu and Y. Bazilevs. Fluid-structure interaction modeling of wind turbines: simulating the full machine. *Computational Mechanics*, pages 1–13, 2012.
- [84] M.-C. Hsu, D. Kamensky, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes. Fluid-structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation. *Computational Mechanics*, 54(4):1055–1071, 2014.
- [85] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, Michael C. H. Wu, Joshua Mineroff, Alessandro Reali, Yuri Bazilevs, and Michael S. Sacks. Dynamic and fluid-structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models. *Computational Mechanics*, 55(6):1211–1225, 2015.

-
- [86] S.-M. Hu and J. Wallner. A second order algorithm for orthogonal projection onto curves and surfaces. *Computer Aided Geometric Design*, 22(3):251–260, 2005.
- [87] T.J.R. Hughes. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2000.
- [88] T.J.R. Hughes and A. Brooks. A multi-dimensional upwind scheme with no crosswind diffusion. In: *Finite Element Methods for Convection Dominated Flows, ASME Winter annual meeting*, 34:19–35, 1979.
- [89] 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(39-41):4135–4195, 2005.
- [90] T.J.R. Hughes, J.A. Evans, and A. Reali. Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems. *Computer Methods in Applied Mechanics and Engineering*, 272(Supplement C):290–320, 2014.
- [91] T.J.R. Hughes, L.P. Franca, and G.M. Hulbert. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73(2):173–189, 1989.
- [92] T.J.R. Hughes and G.M. Hulbert. Space-time finite element methods for elastodynamics: formulations and error estimates. *Computer Methods in Applied Mechanics and Engineering*, 66(3):339–363, 1988.
- [93] B.M. Irons and R.C. Tuck. A version of the Aitken accelerator for computer iteration. *International Journal for Numerical Methods in Engineering*, 1(3):275–277, 1969.
- [94] R.K. Jaiman, X. Jiao, P.H. Geubelle, and E. Loth. Conservative load transfer along curved fluid-solid interface with non-matching meshes. *Journal of Computational Physics*, 218(1):372–397, 2006.
- [95] K. E. Jansen, S. S. Collis, C. Whiting, and F. Shaki. A better consistency for low-order stabilized finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 174(1-2):153–170, 1999.
- [96] K. A. Johannessen, T. Kvamsdal, and T. Dokken. Isogeometric analysis using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, 269:471–514, 2014.

- [97] A. A. Johnson and T. E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer Methods in Applied Mechanics and Engineering*, 119(1):73–94, 1994.
- [98] C. Johnson, U. Nävert, and J. Pitkäranta. Finite element methods for linear hyperbolic problems. *Computer Methods in Applied Mechanics and Engineering*, 45(1-3):285–312, 1984.
- [99] Violeta Karyofylli, Markus Frings, Stefanie Elgeti, and Marek Behr. Simplex space-time meshes in twophase flow simulations. *International Journal for Numerical Methods in Fluids*, 86(3):218–230, 2018.
- [100] F. Key. Stabilization and efficiency enhancement of spline-based numerical methods for fluid- structure Interaction. Master’s thesis, RWTH Aachen, Aachen, 2016.
- [101] J. Kiendl, K.-U. Bletzinger, J. Linhard, and R. Wüchner. Isogeometric shell analysis with Kirchhoff-Love elements. *Computer Methods in Applied Mechanics and Engineering*, 198(49-52):3902–3914, 2009.
- [102] T. Klöppel, A. Popp, U. Küttler, and W.A. Wall. Fluid-structure interaction for non-conforming interfaces based on a dual mortar formulation. *Computer Methods in Applied Mechanics and Engineering*, 200(45-46):3111–3126, 2011.
- [103] E. Knobbe. *On the integration of the arbitrary Lagrangian-Eulerian concept and non-equilibrium thermodynamics*. PhD thesis, TU Delft, Delft University of Technology, Delft, 2011.
- [104] D. Kuhl and M. A. Crisfield. Energy-conserving and decaying algorithms in non-linear structural dynamics. *International journal for numerical methods in engineering*, 45(5):569–599, 1999.
- [105] E. Kuhl, S. Hulshoff, and R. De Borst. An arbitrary Lagrangian Eulerian finite-element approach for fluid-structure interaction phenomena. *International journal for numerical methods in engineering*, 57(1):117–142, 2003.
- [106] M. Kumar, T. Kvamsdal, and K. A. Johannessen. Superconvergent patch recovery and a posteriori error estimation technique in adaptive isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 316(Supplement C):1086–1156, 2017.

-
- [107] U. Küttler, C. Förster, and W.A. Wall. A Solution for the incompressibility dilemma in partitioned fluid-structure interaction with pure Dirichlet fluid domains. *Computational Mechanics*, 38(4-5):417–429, 2006.
 - [108] U. Küttler and W.A. Wall. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43(1):61–72, 2008.
 - [109] E. Kvaalen. A faster Broyden method. *BIT Numerical Mathematics*, 31(2):369–372, 1991.
 - [110] P. Le Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190(24):3039–3067, 2001.
 - [111] X. Li, J. Zheng, T.W. Sederberg, T.J.R. Hughes, and M.A. Scott. On linear independence of T-spline blending functions. *Computer Aided Geometric Design*, 29(1):63–76, 2012.
 - [112] F. Lindner, M. Mehl, K. Scheufele, and B. Uekermann. A comparison of various quasi-Newton schemes for partitioned fluid-structure interaction. In *Proceedings of 6th International Conference on Computational Methods for Coupled Problems in Science and Engineering, Venice*, pages 1–12, 2015.
 - [113] C. C. Long, A. L. Marsden, and Y. Bazilevs. Fluid-structure interaction simulation of pulsatile ventricular assist devices. *Computational Mechanics*, 52(5):971–981, 2013.
 - [114] J.E. Marsden and T.J.R. Hughes. *Mathematical foundations of elasticity*. Dover, New York, 1994.
 - [115] A. Masud and T.J.R. Hughes. A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems. *Computer Methods in Applied Mechanics and Engineering*, 146(1-2):91–126, 1997.
 - [116] M. E. Matzen. *Isogeometrische Modellierung und Diskretisierung von Kontaktproblemen*. PhD thesis, Inst. für Baustatik und Baudynamik der Univ. Stuttgart, Stuttgart, 2015. OCLC: 934169556.
 - [117] D. P. Mok. *Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion*. Number Nr. 36 in Bericht / Institut für Baustatik der Universität Stuttgart. Inst. für Baustatik, Stuttgart, 2001. OCLC: 76350441.

- [118] D.P. Mok, W.A. Wall, M. Bischoff, and E. Ramm. Algorithmic aspects of deformation dependent loads in nonlinear static finite element analysis. *Engineering Computations*, 16(5):601–618, 1999.
- [119] N.M. Newmark. *Computation of dynamic structural response in the range approaching failure*. Department of Civil Engineering, University of Illinois, 1952.
- [120] N.M. Newmark. A method of computation for structural dynamics. *Journal of the engineering mechanics division*, 85(3):67–94, 1959.
- [121] F. Nobile and C. Vergara. An effective fluid-structure interaction formulation for vascular dynamics by generalized Robin conditions. *SIAM Journal on Scientific Computing*, 30(2):731–763, 2008.
- [122] J.T. Oden. *An introduction to mathematical modeling: A course in mechanics*, volume 1. John Wiley & Sons, 2011.
- [123] J.T. Oden and H.J. Brauchli. On the calculation of consistent stress distributions in finite element approximations. *International Journal for Numerical Methods in Engineering*, 3(3):317–325, 1971.
- [124] L. Pauli. *Stabilized finite element methods for computational design of blood-handling devices*. PhD thesis, Verlag Dr. Hut, 2016. DOI: 35355, HT019116406.
- [125] L. Piegl and W. Tiller. *The NURBS Book*. Monographs in Visual Communication. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.
- [126] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations*. Numerical Mathematics and Scientific Computation, Oxford ; New York, new. edition, 1999.
- [127] E. Ramm and W. A. Wall. Shell structuresa sensitive interrelation between physics and numerics. *International journal for numerical methods in engineering*, 60(1):381–427, 2004.
- [128] L. Reimer. *A simulation environment for CFD-based aeroelastic analysis of aircraft*. PhD thesis, RWTH Aachen University, Aachen. expected in 2018.

-
- [129] L. Reimer, C. Braun, G. Wellmer, M. Behr, and J. Ballmann. Development of a modular method for computational aero-structural analysis of aircraft. In *Summary of Flow Modulation and Fluid-Structure Interaction Findings*, Notes on Numerical Fluid Mechanics and Multidisciplinary Design, pages 205–238. Springer, Berlin, Heidelberg, 2010. DOI: 10.1007/978-3-642-04088-7_9.
- [130] R. F. Riesenfeld. *Application of B-spline approximation to geometric problems*. PhD thesis, Ph. D. dissertation, Department of Systems and Information Science, Syracuse University, Syracuse, NY, 1973.
- [131] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.
- [132] M. Schäfer, S. Turek, F. Durst, E. Krause, and R. Rannacher. Benchmark computations of laminar flow around a cylinder. In *Flow simulation with high-performance computers II*, pages 547–566. Springer, 1996.
- [133] D. Schillinger, L. Ded, M.A. Scott, J.A. Evans, M.J. Borden, E. Rank, and T.J.R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-252:116–150, 2012.
- [134] I.J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions. In *IJ Schoenberg Selected Papers*, pages 3–57. Springer, 1988.
- [135] M.A. Scott, X. Li, T.W. Sederberg, and T.J.R. Hughes. Local refinement of analysis-suitable T-splines. *Computer Methods in Applied Mechanics and Engineering*, 213-216:206–222, 2012.
- [136] R. Sevilla, S. Fernández-Méndez, and A. Huerta. NURBS-enhanced finite element method (NEFEM). *International Journal for Numerical Methods in Engineering*, 76(1):56–83, 2008.
- [137] R. Sevilla, S. Fernández-Méndez, and A. Huerta. NURBS-enhanced finite element method (NEFEM): A seamless bridge between CAD and FEM. *Archives of Computational Methods in Engineering*, 18(4):441–484, 2011.

- [138] R. Sevilla Cárdenas. *NURBS-enhanced finite element method (NEFEM)*. Ph.D. Thesis, Universitat Politcnica de Catalunya, 2009.
- [139] F. Shakib. *Finite element analysis of the compressible Euler and Navier-Stokes equations*. PhD thesis, Stanford University, Stanford, CA, USA, 1989.
- [140] R. Siegbert. *Viscoelastic Die Swell as a Design Objective in the Numerical Design of Profile Extrusion Dies*. PhD thesis, RWTH Aachen University, Aachen, 2018.
- [141] E. Sifakis and J. Barbic. FEM simulation of 3d deformable solids: a practitioner’s guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses*, page 20. ACM, 2012.
- [142] T. Spenke. Efficient methods for the simulation of floating elastic structures with thin walls. Master’s thesis, RWTH Aachen, Aachen, 2017.
- [143] A. Stavrev, P. Knechtges, S. Elgeti, and A. Huerta. Space-time NURBS-enhanced finite elements for free-surface flows in 2d: Space-time NURBS-enhanced finite elements for free-surface flows in 2d. *International Journal for Numerical Methods in Fluids*, 81(7):426–450, 2016.
- [144] K. Takizawa and T. E. Tezduyar. Multiscale space-time fluid-structure interaction techniques. *Computational Mechanics*, 48(3):247–267, 2011.
- [145] I. Temizer, P. Wriggers, and T.J.R. Hughes. Contact treatment in isogeometric analysis with NURBS. *Computer Methods in Applied Mechanics and Engineering*, 200(9):1100–1112, 2011.
- [146] T. E. Tezduyar, M. Behr, M. Shakib, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces — the deforming-spatial-domain/space-time procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders. *Computer Methods in Applied Mechanics and Engineering*, 94(3):353–371, 1992.
- [147] T.E. Tezduyar, M. Behr, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces — The deforming-spatial-domain/space-time procedure: I. The concept and the preliminary numerical tests. *Computer Methods in Applied Mechanics and Engineering*, 94(3):339–351, 1992.

-
- [148] T.E. Tezduyar, M. Behr, S. Mittal, and A.A. Johnson. Computation of unsteady incompressible flows with the stabilized finite element methods: Space-time formulations, iterative strategies and massively parallel implementations. *ASME pressure vessels piping div publ pvp.*; *ASME New York.*, 246:7–24, 1992.
- [149] T.E. Tezduyar and S. Sathe. Modelling of fluid–structure interactions with the space–time finite elements: Solution techniques. *International Journal for Numerical Methods in Fluids*, 54(6-8):855–900, 2007.
- [150] T. C.-T. Ting. *Anisotropic Elasticity: Theory and Applications*. Oxford University Press, 1996.
- [151] S. Turek and J. Hron. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In H.-J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction*, volume 53 of *Lecture Notes in Computational Science and Engineering*, pages 371–385. Springer, Berlin, Heidelberg, 2006. DOI: 10.1007/3-540-34596-5_15.
- [152] E.H. van Brummelen. Added mass effects of compressible and oncompressible flows in fluid-structure interaction. *Journal of Applied Mechanics*, 76(2):021206, 2009.
- [153] E.H. van Brummelen, M. Shokrpour-Roudbari, and G.J. van Zwieten. Elasto-capillarity simulations based on the Navier-Stokes-Cahn-Hilliard equations. *arXiv preprint arXiv:1510.02441*, 2015.
- [154] T. M. van Opstal, E. H. van Brummelen, R. de Borst, and M. R. Lewis. A finite-element/boundary-element method for large-displacement fluid-structure interaction. *Computational Mechanics*, 50(6):779–788, 2012.
- [155] K.J. Versprille. *Computer-aided design applications of the rational B-Spline approximation form*. PhD thesis, Syracuse University, 1975.
- [156] W. A. Wall, M. Gee, and E. Ramm. The challenge of a three-dimensional shell formulation-the conditioning problem. In *Proceedings of ECCM*, volume 99, 2000.
- [157] Wolfgang A. Wall. *Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen*. PhD thesis, Institut für Baustatik, Universität Stuttgart, 1999.
- [158] G. Wellmer. *A modular method for the direct coupled aeroelastic simulation of free-flying aircraft*. PhD thesis, RWTH Aachen University, Aachen, 2014.

- [159] P. Wriggers. *Computational contact mechanics*. Springer Science & Business Media, 2006.
- [160] J. Yan, B. Augier, A. Korobenko, J. Czarnowski, G. Kettermann, and Y. Bazilevs. FSI modeling of a propulsion system based on compliant hydrofoils in a tandem configuration. *Computers & Fluids*, 141:201–211, 2016.
- [161] J. Yan, A. Korobenko, X. Deng, and Y. Bazilevs. Computational free-surface fluid-structure interaction with application to floating offshore wind turbines. *Computers & Fluids*, 141:155–174, 2016.
- [162] V.A. Yastrebov. *Numerical Methods in contact mechanics*. John Wiley & Sons, 2013.
- [163] Y. Zhang, W. Wang, and T.J.R. Hughes. Solid T-spline construction from boundary representations for genus-zero geometry. *Computer Methods in Applied Mechanics and Engineering*, 249(Supplement C):185–197, 2012.
- [164] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The finite element method. [Vol. 1]: Its basis and fundamentals*. Elsevier Butterworth-Heinemann, Amsterdam, 6. ed., repr edition, 2010.
- [165] O.C. Zienkiewicz and J.Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357, 1987.
- [166] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33(7):1331–1364, 1992.
- [167] C. Zimmermann and R.A. Sauer. Adaptive local surface refinement based on LR NURBS and its application to contact. *Computational Mechanics*, 2017. arXiv: 1701.08742.
- [168] F. Zwicke, S. Eusterholz, and S. Elgeti. Boundary-conforming free-surface flow computations: Interface tracking for linear, higher-order and isogeometric finite elements. *Computer Methods in Applied Mechanics and Engineering*, 326:175–192, 2017.

This thesis presents the solution of fluid-structure interactions using the same spline description on both sides of the interface. On the structural side, the spline is used in an isogeometric formulation. On the fluid side, the same spline is used in the NURBS-enhanced finite element method. The entire solution methodology of the surface-coupled problem based on an implicit staggered approach is introduced. Consistence and conservation of the spatial coupling are discussed. Numerical advantages of a spline-based coupling are verified by stationary and transient test cases.

ISBN 978-3-8439-3571-5

