Fortschritt-Berichte VDI

VDI

Reihe 18

Mechanik/
Bruchmechanik

M.Sc. Lars Radtke,
Hamburg

Nr. 353

A partitioned solution approach for fluid-structure interaction problems in the arterial system

A partitioned solution approach for fluid-structure interaction problems in the arterial system

Vom Promotionsausschuss der
Technischen Universität Hamburg

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von

Lars Radtke

aus

Itzehoe

2020

Betreuer: Prof. Dr.-Ing. habil. Alexander Düster

Vorsitzender des Promotionsausschusses
Prof. Dr.-Ing. Robert Seifried

Erstgutachter
Prof. Dr.-Ing. Alexander Düster

Zweitgutachter
Prof. Dr. med. Eike Sebastian Debus
Prof. Dr.-Ing. Robert Seifried

Tag der mündlichen Prüfung
11. Juli 2019

Fortschritt-Berichte VDI

Reihe 18

Mechanik/
Bruchmechanik

M.Sc. Lars Radtke,
Hamburg

Nr. 353

A partitioned solution approach for fluid-structure interaction problems in the arterial system

VDI verlag

Radtke, Lars

A partitioned solution approach for fluid-structure interaction problems in the arterial system

Fortschr.-Ber. VDI Reihe 18 Nr. 353. Düsseldorf: VDI Verlag 2020.
298 Seiten, 131 Bilder, 20 Tabellen.
ISBN 978-3-18-335318-7, ISSN 0178-9457,
€ 100,00/VDI-Mitgliederpreis € 90,00.

Keywords: fluid-structure interaction – blood flow – partitioned coupling – high-order finite elements

The present work is concerned with the partitioned solution of the multifield problem arising from a hierarchical modeling approach to cardiovascular fluid-structure interaction. Different strategies to couple the participating field solvers are investigated in detail. This includes staggered and parallel coupling algorithms as well as different methods for convergence acceleration, spatial interpolation and temporal extrapolation of coupling quantities. In the developed modeling and simulation approach, a fully resolved model of a segment of the arterial network is coupled to reduced order models in order to account for the influence of the surrounding.

There is experimental evidence that hemodynamic quantities such as the wall shear stress promote the progression cardiovascular disease. Cardiovascular FSI simulations, that can predict these quantities, are therefore of great interest and can aid in surgical planning and optimization of anastomoses shapes and graft materials.

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

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

Arbeitsgruppe Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik

Als Manuskript gedruckt. Printed in Germany.

ISSN 0178-9457

ISBN 978-3-18-335318-7

Acknowledgements

During my work at the Institute for Ship Structural Design and Analysis at Hamburg University of Technology, from which this thesis has emerged, many great people have helped, inspired and encouraged me in one or the other way. I would like to say "thank you" here - without their input, this output would not exist.

First of all, I would like to express my gratitude to my supervisor Prof. Düster. His expertise in a broad range of topics in the field of numerical computation, calmness in explaining and enthusiasm when discussing new ideas have guided me through my first years in academia. Danke dafür! Having started at the same day in the same office, my colleague Marcel König deserves my special thanks as well. His tenacity and skills, which he was always willing to share, have contributed a lot to this thesis. Vielen Dank! Of course, I would like to thank all of my colleague at the institute for their helpfullness the many fruitful discussions. Danke!

Zooming out, I owe great thanks to my parents Christel and Ronald. They have perhaps once sparked my interest in science and support me unconditionally and without exception in everything I do. Danke für alles! I would also like to thank my friends, many of whom have inspired me technically - oftentimes without an engineering background - and all of whom have encouraged me to stay at the university after may Masters degree. Vielen Dank!

Finally, I would express my deepest thanks to my partner Johanna for her everlasting support, comprehension and love. Ich liebe Dich!

Contents

List of medical terms

abdominal	Refers to the abdomen (stomach)
adventitia	Outermost layer of an arterial wall
anastomosis	Here, connection between bypass graft and artery
aorta	The largest artery in the body
atherosclerosis	Inflammatory disease in arteries
atrium	Smaller chambers of the heart
cardiovascular	Refers to the heart and the circulatory system
coronary	Refers to the heart
diastole	Phase of the cardiac cycle, where blood enters in the left ventricle
distal	Away from the body center, here downstream
dorsal	Refers to the back
endothelium	Layer of cells at the lumen boundary of an artery
erythrocytes	red blood cell
femoral	Refers to the thigh region
iliac	Refers to the intestine region
intima	Innermost layer of an arterial wall
intimal hyperplasia	Abnormal thickening of the intima
leucocytes	White blood cell
media	Middle layer of an arterial wall
proximal	Towards the body center, here upstream
pulmonary	Refers to the lungs
systole	Phase of the cardiac cycle, where blood leaves the left ventricle
thrombocytes	Platelets activating blood clotting
thrombosis	Formation of a blood clot
ventricle	Larger chambers of the heart

Abstract

The present work is concerned with the partitioned solution of the multifield problem arising from a hierarchical modeling approach to cardiovascular fluid-structure interaction. Different strategies to couple the participating field solvers are investigated in detail. This includes staggered and parallel coupling algorithms as well as different methods for convergence acceleration, spatial interpolation and temporal extrapolation of coupling quantities as well as convergence criteria. In the developed modeling and simulation approach, a fully resolved model of a segment of the arterial network is coupled to reduced order models in order to account for the influence of the surrounding. The resulting problem is solved using five specialized field solvers, namely a fluid and a structural solver for the three-dimensional fluid-structure interaction problem, a one-dimensional blood flow solver for the surrounding vessel network, a solver for different types of windkessel models used to obtain physiological boundary conditions at the distal ends of the one- and three-dimensional models, and a solver for an elastic foundation that describes the surrounding tissue. The applicability of the solution approach is demonstrated in terms of several exemplary applications including studies of idealized and patient specific end-to-side anastomoses of bypass grafts. They are known to be prone to the development of intimal hyperplasia, i.e. a thickening of the vessel wall that may lead to occlusions in the anastomosis region. There is experimental evidence that hemodynamic quantities such as the wall shear stress promote the progression of this secondary disease. Cardiovascular FSI simulation are therefore of great interest and can aid in surgical planning and optimization of anastomoses shapes and graft materials.

1 Introduction

Today, numerical simulations play an important role in many fields of engineering and physics. On the application side, they are a versatile tool in the design and optimization process of mechanical, electrical, thermal and other components. On the theory side, they provide deeper insight into complex physical phenomena relying on known, fundamental laws. In both cases simulations may be advantageous over experiments due to lower cost, reproducibility and measurement issues. Of course, for the results to be reliable, a small number of experiments still have to be conducted to provide a validation of the simulation method that can be generalized to the application scenario.

Within the field of computational mechanics, one is interested in the motion of solid bodies or fluids. The framework of continuum mechanics provides the fundamental laws to predict the motion of bodies with complex shape, respectively the motion of fluids in regions with complex shape. Most other descriptions of mechanical phenomena evolve from the continuum mechanics approach due to certain assumptions. Examples are rigid body motion, inviscid or incompressible flows, quasi static or small deformation analyses, and approaches with reduced spatial dimensionality.

The choice of the most suitable description depends on the problem to be solved and sometimes it is necessary to combine different approaches. Fluid-structure interaction (FSI) problems are typical, if not the most common examples for such cases. Due to the different behavior of structures and fluids, different formulations of the laws of continuum mechanics and correspondingly different numerical methods serve to predict their motion. The problem is then regarded as a surface-coupled two-field problem, with fluid and structure each corresponding to one field. The region occupied by the fluid deforms according to the structural deformation, which itself is the result of pressure and shear loads acting on the structure due to the flow. If this interdependency cannot be neglected in order to arrive at a reliable description, the problem is said to be *strongly coupled.* If numerical simulations account for the interdependency in every time instance under

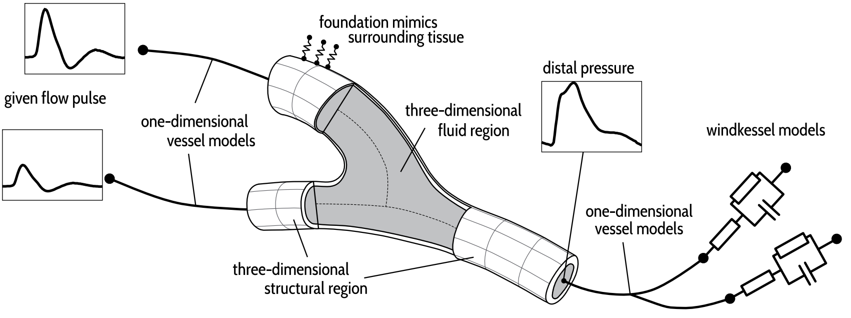

foundation mimics surrounding tissue
given flow pulse
one-dimensional vessel models
three-dimensional fluid region
distal pressure
windkessel models
one-dimensional vessel models
three-dimensional structural region

Figure 1.1: Hierarchical modeling approach to cardiovascular FSI leading to a strongly-coupled multifield problem.

consideration, the solution approach is regarded *implicit*. The FSI in the arterial system is a typical example for a physical phenomenon where such a strong coupling prevails and an implicit solution approach is inevitable in order to obtain realistic results.

Not only different types of matter participating in a problem, but also regions with different accuracy requirements may lead to coupled problems. For example, the mechanical behavior of the entire human body cannot be simulated in a continuum mechanics sense using todays computers. However, a sufficiently small region can be modeled with such a high resolution, while the influence of the surrounding on that region is accounted for by appropriate boundary conditions. In this work, a simulation framework for cardiovascular FSI problems arising from such a hierarchical modeling approach is devised. A small region in the arterial network is modeled using a fully resolved continuum mechanics approach. The upstream and downstream vessel network as well the surrounding tissue are described using reduced models. Figure 1.1 illustrates the different fields or subproblems governing the resulting strongly coupled multifield problem. Particular emphasis is put on the three-dimensional fluid-structure interaction problem, with a focus on efficient coupling procedures and the solution of the structural mechanics subproblem.

Numerical solution approaches for coupled problems can be classified as either *partitioned* or *monolithic* methods. In the monolithic case, all fields are solved simultaneously taking into account the influence of all fields on each other. In consequence, large systems of equations have to be solved but the stability of the simulations is increased compared to the partitioned case. Here, the different fields are solved subsequently, each under the assumption that the other fields are independent of the current one. This requires an iterative solution approach including an exchange of coupling quantities between the fields in order to solve the coupled problem. Further, due to the assumptions of independence within each iteration, stability problems may occur. Often times, partitioned solution approaches are still favorable because of the smaller individual subproblems and because of the greater flexibility. While in the scope of monolithic schemes, each coupled problem demands for its own problem formulation and numerical method, it is possible to reuse specialized existing methods and software when using partitioned schemes. This allows for high-fidelity modeling and simulation approaches within each subproblem, which is valuable especially for arterial FSI. The participating structure (the arterial wall) as well as the fluid (the blood) show complex material behavior or rheology, respectively, which are most often investigated separately within the fields of computational structural dynamics (CSD) and computationally fluid dynamics (CFD). To incorporate state-of-the-art modeling approaches from these two worlds, a partitioned approach constitutes the ideal basis as the simulation software (fluid solvers and structure solvers) can be reused.

Summarizing, a partitioned solution approach is well suited for cardiovascular FSI simulations due to the need for specialized single-field solvers as well as the hierarchic nature of suitable modeling approaches, which demands for a combination of three-dimensional and reduced models as illustrated in Fig. 1.1. Based on these arguments, numerical methods are developed in the scope of this work to realize a corresponding multifield simulation. The methods are used to investigate the blood flow (the hemodynamics) in a connection between a bypass graft and an artery. If an artery is occluded and the blood transport to certain parts of the body is no longer guaranteed, a bypass may need to be implanted. A numerical simulation of the hemodynamics in the distal (downstream) connection is of great interest, due to an increased risk of an anew occlusion in this region. The local flow conditions are assumed to play a key role in the development of such a restenosis, and investigating them can be of help on two different levels. On the one hand, it can provide deeper insight into the process of growths and remodeling of arterial tissue that eventually leads to a restenosis. A comparison of local hemodynamics and clinically observed configurations of higher risk is just one example. On the other hand, knowledge on the hemodynamics is essential in order to optimize anastomoses in terms of shape or graft material.

Some effort has to be made to implement a suitable simulation approach serving this purpose. A major issue is the possible instability of the partitioned solution approach. Different investigations were conducted to find the most suitable coupling algorithm (rules for the data exchange between the single-field solvers) and convergence acceleration scheme (method to modify the result obtained from one solver before passing it on to the other). A modified version of the so called Quasi-Newton-Least-Square algorithm was found superior over other methods. To efficiently solve the structural mechanics subproblem, high-order finite elements are used as the preferred discretization method due to their robustness and convergence characteristics. In order to use the comparably coarse computational meshes that characterize this method for a discretization of anastomoses, curved elements have to be used. To this end, a pipeline for mesh generation was developed, which includes a flexible modeling tool for smooth geometries. The proposed methods allow for an in-depth investigation of the FSI in the arterial system, while being generally applicable on their own. This includes not only the possibility for the methods to be applied to other problems within the field of cardiovascular FSI but also to be used to solve fundamentally different physical phenomena. For example, the partitioned solution approach may also be used for thermo-mechanically or otherwise coupled problems. Likewise, the modeling pipeline may be used to create high-order finite element meshes for a variety of different applications.

Focusing on cardiovascular applications, the next chapter will further motivate the research and give an introduction into vascular substitutes and other medical aspects of the circulatory system. The current state of research in the field of cardiovascular FSI simulations is outlined along with a discussion of the underlying physics and numerical methods. To begin with, this is done from a phenomenological viewpoint, i.e. without any reference to a mathematical formulation. It was decided to separate this more detailed overview of the corresponding interdisciplinary field of research to provide all readers - with or without any background in computational science, engineering or medicine - with adequate information on the topic.

In Chapter 3, the underlying physical phenomena are discussed along with their mathematical formulation. This includes the underlying equations of the three-dimensional FSI problem, which forms the basis for the small region of high resolution, as well as the means to arrive at the reduced models used to describe the surrounding. The formulations are chosen in such a way that they provide a suitable basis for the introduction of the numerical methods. Yet, everything is continuous in this chapter, i.e. no discretization in space or time is performed.

It is in Chapter 4 that the temporal and spatial discretization methods for the subproblems as well as the mentioned modeling pipeline and the partitioned solution approach are explained. Only a few preliminary results are presented here along with the individual methods.

Detailed numerical investigations that serve to prove the applicability of the proposed methods are presented in Chapter 5. The results regarding their stability, efficiency, and accuracy substantiate the preference for the partitioned solution algorithm. Smaller cardiovascular FSI studies are conducted as well, paving the way for applying the partitioned solution approach in the scope of simulations of the FSI in bypass graft anastomoses.

In Chapter 6, exemplary investigations of the hemodynamics in idealized arterial segments and anastomoses with cuffed bypass grafts are presented. A patient-specific simulation of the blood flow in a distal anastomosis of an aorto-femoral bypass and the surrounding vessel network demonstrates the applicability of the developed modeling and simulation approach in the scope of large-scale analyses.

2 Fluid-structure interaction in the arterial system

The flow of blood through the cardiovascular system is a complex physical phenomenon. However, neglecting biochemical effects, the system dynamics can be described very accurately based on a continuum mechanics approach. Therein, the microstructure is not resolved but a continuous material is assumed. This is justified by the large difference in the scales that are identified with the microstructure and the macroscopic effects to be described, i.e. the velocity and the pressure of the blood as well as the deformation of the vessel wall. The macroscopic behavior of the vessel material and the rheology of blood are modeled using constitutive equations or material model. Which type of material model is suitable may be decided based on the microstructure or based on macroscopic experiments. Given suitable constitutive equations, a three-dimensional simulation can be used to predict the state of any particle in a region of interest, whether it belongs to the vessel wall or to the blood that currently occupies the region. The state of a particle in a continuum mechanics is defined by the displacement, velocity, and acceleration as a function of time and space. Given this information, hemodynamic quantities like the wall shear stress, which are known to play an important role in the pathogenesis of cardiovascular diseases, can then be evaluated.

However, regarding the fluid-structure interaction (FSI) in the arterial system, only a small segment can be simulated with such a high level of detail. To describe larger scale parts of the system, reduced models, e.g. one-dimensional models, have to be used. As mentioned in the introduction, cardiovascular studies will likely require combinations of different scales and resolutions. Only in a small region of interest, e.g. the anastomosis region, local feature of the hemodynamics are resolved. The influence of the surrounding system on the hemodynamic in that region is modeled using a coarser resolution that allows for a larger scale. In the corresponding reduced models of the surrounding, space or time dimensions may be dropped in order to reduce computation time.

Before reviewing the corresponding underlying equations in the next chapter, a general overview of the cardiovascular circulation shall be given. This provides the necessary background for an adaption and reduction of the physical laws, so that they can be effectively used in a computer simulation. Detailed information about the cardiovascular system can be found in many textbooks, e.g. [50, 199, 175], on which the following overview is based. In the second part of this chapter, the state-of-the-art in computational modeling of FSI problems is outlined.

2.1 The cardiovascular system

The cardiovascular system of an average male adult contains about five liters of blood. In resting condition, the heart pumps these five liters through the systemic and the pulmonary circulation in about one minute. Figure 2.1 illustrates the different regions of the heart and gives an overview of the flow rates within different parts of the body. Let us consider an exemplary particle in the left ventricle: First, it is ejected into the aorta through the aortic valve. It may then flow along the aorta to the lower parts of the body and return to the heart from the inferior vena cava. It may alternatively flow through one of the branches of the aortic arc to the arms or head and return from the superior vena cava. In both cases, it arrives at the right atrium and is then pumped into the right ventricle. From the right ventricle, it is ejected into the pulmonary artery and then flows through the left or right lung to return to the heart from the pulmonary vein.

In this work, only the systemic arterial system is considered, with a focus on the lower parts of the body. Each beat of the heart initiates a pulse wave that travels through this part of the circulation. With increasing distance from the heart, the pulsatility dies out due to the damping effects of the elastic vessel walls until a nearly stationary flow ensues in the capillaries. These characteristics of the system allow for an isolated modeling approach of arterial hemodynamics. The stationary flow in the capillaries as well as the pulse ejected from the left ventricles provide model boundaries with well-defined conditions. It is only through these conditions that changes within the heart, the pulmonary circulation, and the systemic venous system are accounted for. Despite the independence of the arterial hemodynamics from the details of the hemodynamics in the rest of the cardiovascular system, it

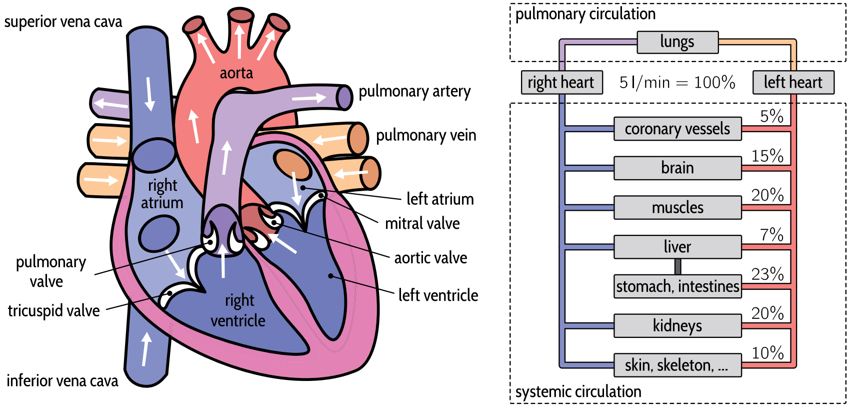


Figure 2.1: Schematic illustrations of the heart (taken from [225] and modified) and the cardiovascular system (percentages according to [175, p. 573]).

constitutes a complex phenomenon for which, so far, no modeling approach has been found that is clearly favorable over others.

2.1.1 Anatomy of the larger arteries

The following remarks can be found in many books on human anatomy or cardiovascular surgery, see e.g. [50, 199]. Considering the systemic circulation in the lower body, the aorta constitutes the common parent vessel of all other vessels. It is divided into five segments. The ascending aorta exits the heart and continues as the aortic arc, from which the vessels supplying the head and the arms branch off. The part passing the heart dorsally (on the backside) is called the descending aorta, followed by the abdominal aorta, which is commonly divided into two parts, namely the suprarenal abdominal aorta (proximal to the bifurcation of the renal arteries) and the abdominal aorta (the part from the renal arteries to the iliac bifurcation, shown in enlargement C in Fig. 2.2). At the iliac bifurcation, the aorta branches into the left and right common iliac artery, which continue in a (topologically) symmetric fashion. At the first larger bifurcation, the common iliac arteries branch into an internal and an external iliac artery. The internal iliac artery supplies the pelvic region, while the (larger) external iliac artery continues in the direction of the upper thigh, where it becomes the common femoral

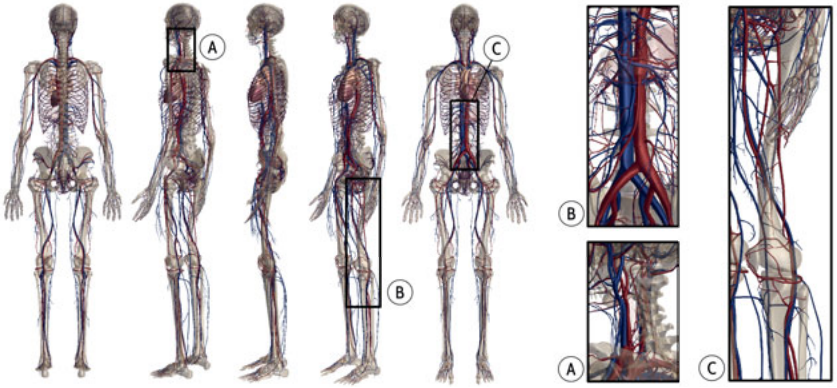


Figure 2.2: Anatomy of the cardiovascular system viewed from the back (left) to the front (right) and enlarged critical regions with respect to the development of atherosclerotic symptoms. The pictures were created using the free browser version of the anatomy software *ZygoteBody* [241].

artery. Still in the upper thigh, different smaller vessels, the largest being the deep femoral artery and the femoral circumflex artery, branch off from the common femoral artery. Below the knee, the common femoral artery continues as the popliteal artery, where this overview of the pathways of the larger arteries concludes. The arterial network branches further into smaller arteries and arterioles, which connect the smaller arteries and the capillaries. While arteries always branch and with a few exceptions (see e.g. the arteries from the intestines to the liver in Fig. 2.1) never join, the capillaries join and increase in diameter and proceed as venules.
As they join together again and increase in diameter, they become veins that lead back to the heart, as can be seen in Fig. 2.2.

The mentioned larger arteries - except the aorta - are classified as muscular arteries. Arteries that are directly connected to the heart (the aorta and the pulmonary artery) and the vessels branching off the aortic arc are classified as elastic arteries [168]. Histologically, any artery consists of three layers, the *tunica intima*, the *tunica media* and the *tunica adventitia*. In this work, only muscular arteries are considered. Figure 2.3 shows a microscopic view of a cut through a muscular artery. According to the thicknesses of the individual layers, it is commonly assumed that the intima has no effect on the mechanical behavior of the artery (see e.g. [101] and the references

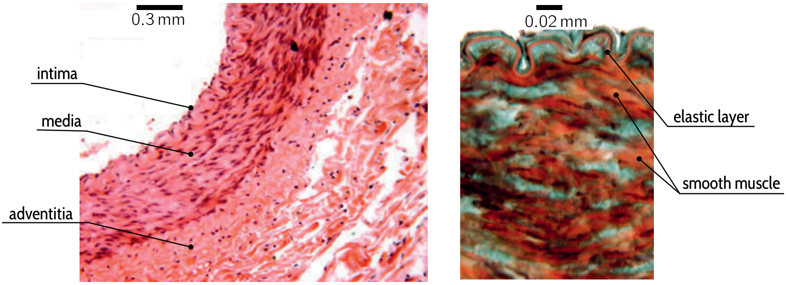


Figure 2.3: Histology of a muscular artery (taken from [157] and relabeled).

cited therein). Of course, from a biochemical or medical point of view it is a very important constituent. At its inner side a single layer of endothelial cells forms the barrier between the blood and the rest of the vessel wall. They play an important role in the growth and remodeling process of arterial tissue as well as the development of cardiovascular diseases as addressed in Section 2.1.3. The intima layer is separated from the media layer by a clearly visible internal elastic layer, see Fig. 2.3. The properties of media and adventitia will be discussed in Section 2.2.2 along with possible constitutive equations, i.e. material models.

2.1.2 Physical characteristics of arterial blood flow

With every beat of the heart, a pulse wave travels through the arterial system. During the opening of the aortic valve and the contraction of the left ventricle, blood is pushed into the aorta. This causes an acceleration of the blood present in the aorta and - due to inertia and viscous effects - an increase of the local pressure which in turn leads to a dilatation of the vessel wall. Thus, the mentioned pulse wave is threefold – consisting of the peaks in the local pressure, the local flow velocity, and the local dilatation, all propagating through the arterial system as waves. They travel with approximately the same velocity, but they can be very different in shape. In the fictive case of a rigid vessel wall and under the assumption that blood is an incompressible fluid, the entire blood in the circulatory system would accelerate at the same time and the increase in pressure would be orders of magnitudes higher. While it is well justified to assume that blood is incompressible, it is the compliance of the vessel wall that cannot be neglected.

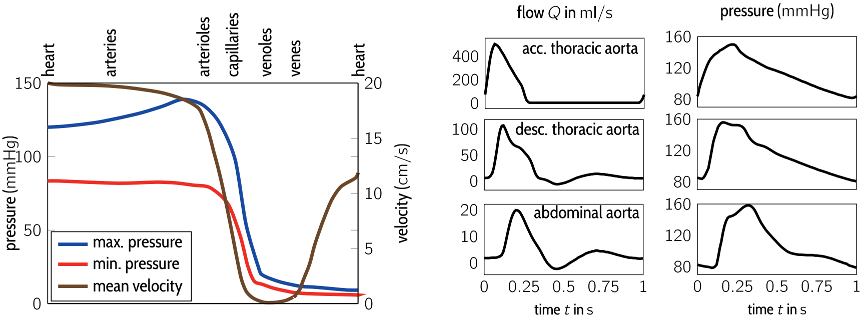


Figure 2.4: Left: Typical changes in pressure and flow velocity with increasing distance from the heart (reconstructed according to information provided in [175, p. 581]). Right: Typical wave forms with increasing distance from the heart (reconstructed from [232]).

In models ignoring this aspect, no wave propagation would be observed at all, or – if taking into account the negligible compressibility of blood – the wave would propagate with the speed of sound ($\approx 1500\,\mathrm{m/s}$ in water). In reality, the compliance leads to a much lower pulse wave velocity of around $5\,\mathrm{m/s}$ in the larger arteries of a healthy human (cf. [175, p. 579]). Further, the wave amplitude dies out with increasing distance from the heart until almost stationary flow conditions prevail in the capillaries. In terms of total resistance due to inertia and viscous effects the larger arteries contribute only a small part. This is especially apparent when looking at the mean pressure, which is almost constant within the larger arteries, as shown in Fig. 2.4. In fact, effects of wave reflections at bifurcations outweigh the effect of a decaying peripheral resistance in the larger arteries such that the peak pressure actually increases, before it decreases rapidly in the arterioles and capillaries. The mean flow velocity is more than one order of magnitude lower ($15\,\mathrm{cm}/sto20\,\mathrm{cm/s}$, cf. [175, p. 581]) than the pulse wave velocity. Blood flow is therefore commonly considered a laminar phenomenon in most parts of the human body. Exceptions are the aortic arc and the carotid bifurcation, where turbulent flow conditions have been observed, see [153, 182]. A detailed explanation of the pulse wave propagation in the circulatory system from a medical point of view is given in [17].

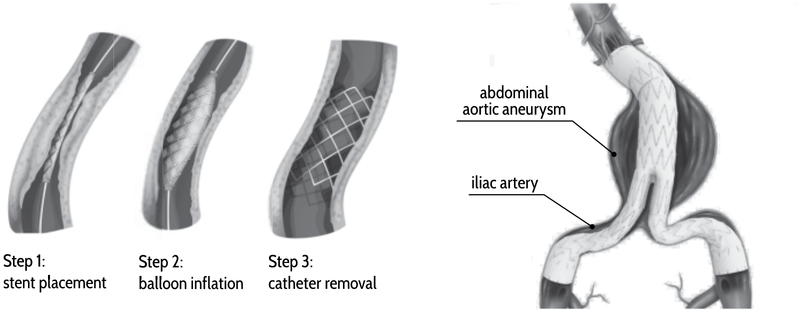


Figure 2.5: Treatment of a stenosis by means of balloon angioplasty (taken from [150]) and a self-extracting stent-graft used to treat an abdominal aortic aneurysm (taken from [173]).

2.1.3 Cardiovascular diseases

Cardiovascular diseases (CVD) are the leading cause of death, not only in industrialized countries, but globally. In 2015, 17.7 million people died due to CVDs (31% of all deaths), three quarters of them in low- or middle income countries [229]. In the European Union, 1.9 million deaths (37.5% of all deaths) were caused by CVDs [70]. Most often, atherosclerosis is identified as the prevailing disease, leading to a failure of arteries due to the built-up of inflammatory lesions. They are also known as atherosclerotic plaques and consist of a lipid core and a fibrous cap that separates the prothrombotic core from the lumen. The risk posed by the plaques is to be seen in a possible occlusion of a vessel. This may happen due to the growth of the plaque itself or (more frequently) due to a rupture of the fibrous cap, leading to the formation of a thrombus [13]. The thrombosis may then lead to an occlusion in the region of the ruptured plaque or, after a detachment of thrombus material, cause an occlusion in smaller, distal vessels.

Narrowed vessel regions apparent in this process are known as *stenoses.* The forming of plaques includes an abnormal proliferation of smooth muscle cells (SMC) into the tunica intima. The corresponding thickening of this layer is called *intimal hyperplasia (IH)* and is addressed in more detail below, as this aspect plays an important role in the healing process after vascular surgery. An overview of the biochemical mechanisms that are involved in this process and a review of recent studies on this subject is given in [185]. Typically, stenoses are found in the carotid artery (cf. Fig. 2.2, enlargement A), the arteries of the upper and lower limb, and in the coronary vessels (vessels supplying the heart). If possible, stenoses are treated using minimal invasive techniques such as the placement of a stent using a catheter. If a stainless steel stent is used, it is advanced to the stenosis region and then dilated using a balloon, as illustrated in Fig. 2.5. (Another type of frequently used stents are self-extracting ones which are made out of shape memory alloys such as Nitinol.)

Another pathological change of the vessel wall is the local enlargement of the lumen, known as an *aneurysm*. If a vessel wall is weakened or damaged, a proliferation of SMC is a necessary response of the vessel wall to restore its strength. However, if the damage is too severe or the proliferation process is disturbed otherwise, the vessel may not withstand the blood pressure and locally increase in diameter while decreasing in wall thickness. The formation of aneurysm and the assessment of the rupture risk are active fields of research, see e.g. [142, 200]. Typically, aneurysms are found along the aorta and in intracranial arteries (arteries in the brain). If possible, aneurysms are treated by endovascular surgery using so called *stent grafts*, i.e. coated stents that provide an artificial vessel wall that shields the aneurysm wall and restores regular flow conditions, see 2.5.

While atherosclerosis is considered a disease with systemic risk factors such as tobacco abuse, unhealthy diet, and low activity, the described symptoms, i.e. stenoses and aneurysms, constitute local changes in the vessel wall. The possible reasons for such a localization are still an open question, but there is no doubt that the local hemodynamics play a key role. Plaques are formed predominantly in regions were the typically laminar flow pattern is disturbed, see [41] and the references cited in [185]. A comprehensive review of recent findings regarding the correlation of hemodynamic factors and the development of atherosclerotic lesions is provided in [159]. Based on this, a novel method for such investigations is presented in [158]. Briefly speaking, the endothelial cells sense - apart from the pressure, being the main load they are subjected to - distributed shear stresses which very much depend on the prevailing flow conditions [3].

These *wall shear stresses (WSS)* are accepted as an important hemodynamic quantity, which has been addressed in many experimental and numerical investigations[1]. Many studies come to the conclusion that low

[1] Almost all corresponding studies concentrate on the WSS and how it influences the growth and remodeling of the vessel wall, see e.g. [41, 83, 201, 185, 200] or how it is influenced by the shape or the material properties of the vessel wall, see e.g. [171, 97] for experimental and [72, 106, 186] for numerical analyses.

and oscillatory WSSs promote intimal hyperplasia and the formation of atherosclerotic plaques. Further, oscillatory WSSs, i.e. WSSs that change their direction significantly over one beat of the heart as well as high WSS gradients (spatial changes) are considered risk factors. Within the field of scientific computing, the underlying *growth and remodeling* has become a subject of increasing interest [106]. Novel simulation approaches also account for the interdependency between the hemodynamics (modeled as a FSI problem) and the biochemical processes in the vessel wall [72]. One of the major challenges when modeling such *fluid-solid-growth problems* are the different time scales on which the FSI problem and the growth and remodeling processes have to be described.

The formation of atherosclerotic plaques is a process that happens in the intimal layer of arteries. While intimal hyperplasia may be considered an important mechanism in the formation of atherosclerotic plaques (cf. [200, p. 381]) it also plays a major role in the failure of vascular repairs. Within a few month after the implantation, stents, stent-grafts and bypass grafts are covered with neointima (a freshly formed intima layer). Neointima hyperplasia is the main cause for a renewed occlusion (restenosis), an issue present especially in the distal connection of bypass grafts and the artery (distal anastomosis), see [41, 83]. The simulation of the hemodynamics in distal anastomoses constitutes the leading application for the development of the numerical methods presented in this work, and it is considered in more detail below. An overview of the recent findings regarding a correlation between the WSS and the development of intimal hyperplasia is given in [41, 83, 201]. Apart from investigations of the flow conditions, these reviews include studies like those in [15], which are concerned with the influence of structural mechanics aspects such as the stress concentration in the vessel wall.

2.1.4 Vascular bypass grafts

If the functionality of an artery is impaired to such an extent that the perfusion of the distal organs is no longer ensured, vascular bypass grafts may be implanted in an open surgery to restore a normal blood flow. Figure 2.6 gives two examples for typical implantation scenarios. Corresponding surgeries are by no means a first choice but a surgery with much larger risks for the patient than a minimally invasive endovascular treatments depicted in Fig. 2.5. Further, neointima hyperplasia and other effects (thrombus

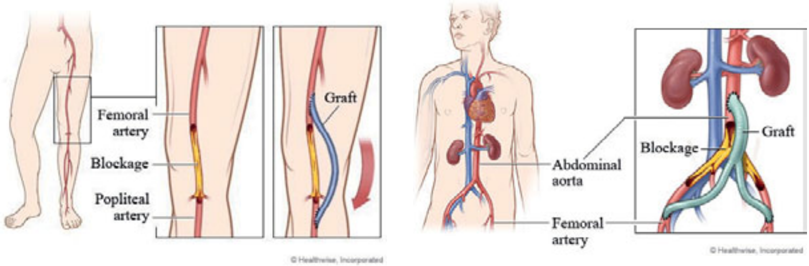


Figure 2.6: Left: Femoropopliteal bypass graft (taken from [96]). Right: Aortibifemoral bypass graft (taken from [95]).

formation, progressive atherosclerosis) may lead to a renewed occlusion as mentioned above. The failure rate strongly depends on the type of vessel that is bypassed, on the patient's medical history, and on the graft material. Clinical studies with patency rates for femoropopliteal bypass grafts have been reported in [49]. In [100], the patency of end-to-side and end-to-end anastomoses is investigated for more than 200 patients. This study includes different anastomoses locations and graft materials. An overview of the corresponding investigations is given in [83, 93]. Artificial graft materials are generally more prone to restenoses than veins, which are used whenever possible. Typical artificial materials for grafts are polytetrafluoroethylene (PTFE) and a fiber material made out of polyethylene terephthalate (PET)[2]. In [170], a comparison of these two standard materials is presented based on a comprehensive literature review.

To counteract the development of intimal hyperplasia due to unnatural hemodynamics in the anastomosis region, different types of end-to-side anastomoses where developed. In [171, 97], experimental investigations were conducted to compare different types, including the ones shown in Fig. 2.7. In the non-standard types, pieces of a vein are used to alter the shape of the anastomosis. In addition to the shape, this also changes the compliance of the anastomosis - since graft materials are generally much stiffer than veins. This *compliance mismatch* influences the growth and remodeling process that takes place after the implantation. However, these effects cannot be considered in experiments like those in [171, 97], where only the shape is reproduced using a silicon mold. Computer simulations

[2] PET graft material is commonly denoted by DuPont's ([63]) brand name Dacron®

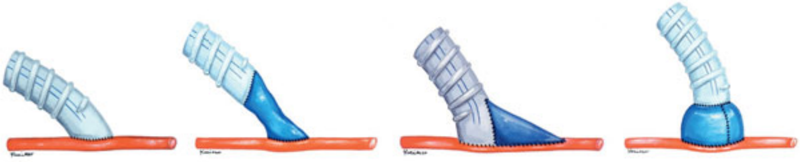


Figure 2.7: Different types of bypass grafts and distal end-to-side anastomoses (taken from [171]). From left to right: Standard bypass and anastomosis, composite bypass graft, Taylor patch, Miller cuff.

on the other hand provide a versatile tool to investigate the influence of different graft materials.

Assuming that the anastomosis shape and the resultant hemodynamics play a more important role than the compliance mismatch, so called *precuffed* artificial grafts were developed, see e.g. [213, 62, 207]. They optimize the shape of the anastomosis similar to the cuffs shown in Fig. 2.7 by an increased diameter at their distal end. Precuffed bypass grafts are considered in more detail in the context of the exemplary study conducted in Chapter 6.

2.2 Computational modeling

The issues related to cardiovascular surgery motivate the development of numerical methods to predict the hemodynamics in certain configurations of stents/stent-grafts and bypass grafts using computer simulations. While simulations can generally be helpful regarding the development of engineering components and the investigation of physical phenomena to avoid the need for a large number of experiments, they are of particular interest in the context of biomechanics. Here, experiments can oftentimes only be performed *in vivo*, i.e. in a clinic on a living patient, because the conditions in the body cannot be emulated in a satisfactory way *in vitro*, e.g. in the laboratory. Investigations that are performed *in silico*, i.e. virtually on a digital computer, may overcome this issue. If the simulation predicts the dynamics of the blood and the vessel wall correctly, the influence of different vessel shapes and materials can be investigated for a variety of conditions such as those during rest and exercise. Over the past decades, the great potential of numerical simulations has led to a rapid growth in the field of computational science with applications pertaining to cardiovascular studies. In addition to three-dimensional continuum mechanics approaches relying on fully resolved models, the mentioned reduced order models are steadily developed further as well. They have been applied for a much longer time since they do not rely on large computational resources. While both classes of models are applicable on their own, it is emphasized that in this work they are used in combination. In the following, an overview of the history and the state-of-the-art of both, reduced and fully resolved cardiovascular modeling and simulation, is outlined. To start with, fully resolved FSI simulations in general are considered.

2.2.1 Fluid-structure interaction

Approaches to solve FSI problems are an active field of research by itself, i.e. without necessarily concerning a specific field of application. Standard text books, e.g. [21, 33, 32], provide an overview of the recent developments. As mentioned in the introduction, solution approaches can be classified as either *monolithic* or *partitioned*. A detailed explanation follows at the beginning of the next chapter. Here, it is sufficient to mention that in monolithic solution approaches, the FSI is described as a single problem using one numerical method. The mechanical behavior of the fluid, the structure and the interaction between the two is predicted simultaneously. Monolithic schemes include space-time methods, which employ a common discretization strategy in space and time. They are used frequently to solve FSI problems (see e.g. [18, 186, 196, 174]) and are characterized by good stability at the cost of a higher implementation effort and non-standard mathematical frameworks.

In partitioned approaches, fluid and structure are treated separately, using individual field solvers. The interaction is accounted for by boundary conditions at the interface between fluid and structure, conveniently called the *wet surface*. In particular, the displacement of the structure at the wet surface is used to define the geometry of the fluid region. In the other direction, the fluid pressure and shear stresses at the wet surface are prescribed as surface loads acting on the structure. Within such coupling procedures, the mentioned coupling quantities have to be exchanged iteratively between the participating field solvers, which is achieved by a coupling software or *coupling manager*. From the coupling manager's point of view, the field solvers constitute black-boxes, i.e. only the respective coupling quantities (displacement and tractions in the FSI case) can be accessed to steer the coupling iterations. This allows to reuse existing software from the worlds of computational fluid and structural dynamics (CFD and CSD), which is not only advantageous in terms of lower implementation effort, but also in terms of specialized numerical methods that can be applied to the respective fields.

However, the iterative solution procedure of partitioned approaches may show a poor convergence behavior and lead to instabilities. The development of convergence acceleration methods that reduce the number of iterations and stabilize the procedure are therefore of great interest. A modified version of the relaxation method from Aitken (1979) [2] was proposed in [107] and was among the first to be used successfully in FSI simulations in many studies, see e.g. [129, 214, 231]. In [144, 81], various convergence acceleration methods were investigated with respect to their applicability for FSI problems. Degroote et al. (2008) [52] proposed a novel acceleration method (the quasi-Newton least squares (QNLS) method mentioned in the introduction), which was found to outperform the comparably simple relaxation methods as shown in [163, 209]. It was developed further in [51, 53, 87] and has also been applied to different applications involving strongly coupled multifield problems, see e.g. [68, 122].

The development of coupling managers is another effort that has to be taken in order to realize partitioned simulations. While many groups make use of the commercial software MPCCI (see [113]), others have implemented their own coupling managers to enhance the flexibility of developing novel coupling algorithms and convergence acceleration methods, see e.g. [209, 31, 220, 143]. Here, the in-house coupling manager *comana* (a C++ framework introduced in [122]) serves as a basis for all coupled simulations.

2.2.2 Arterial hemodynamics

Possibly the first mathematical model of the arterial system that is still frequently used today is the *windkessel model* by Frank (1899) [78] (see [172] for a translation). It relates the flow and pressure in a vessel or vessel network using an analogy to the current and voltage in an electric circuit. In [223] multiple windkessel models were connected to build a model of the larger arteries on a per vessel basis. Today, windkessel models are used as a standard boundary condition in fully resolved simulations to account for the influence of the distal vessel network, see e.g. [203, 188, 19, 128]. Special techniques for this coupling of one- and three-dimensional models are proposed in [217, 216]. Since windkessel models lack any spatial resolution, pulse wave propagation or wave reflection phenomena cannot be described with them. For this type of investigation, one-dimensional models constitute the standard basis. Early applications of these models for arterial blood flow include the studies in [132, 6], where special mathematical formulations are used in order to accurately account for certain features such as the pulse wave velocity or the formation of shock waves. An overview of these simulation approaches is given in [105]. Today, large vessel networks can be modeled using more general one-dimensional models for pulse wave propagation. Dedicated numerical methods have been developed and investigated (see e.g. [179] and the references cited therein) that overcome dispersion and other issues commonly faced when solving wave equations. Like in three-dimensional simulations, windkessel models are typically used as an outflow boundary condition and may also be used as a replacement for parts of a one-dimensional network model, as proposed in [67].

The analytical solution for the velocity profile in a pulsatile flow through a rigid circular tube put forward by Womersley (1955) [228] constitutes another milestone in the history of mathematical models for hemodynamic. Despite the assumptions of a rigid wall and a circular geometry, which render the underlying mathematical equations one-dimensional as well[3], Womersley's formulations are still frequently used today. Given a flow at the inlet of a vessel segment that is to be investigated in a three-dimensional sense, the local velocity needs to be determined. To this end, many studies make use of the velocity profile derived in [228], see e.g. [188, 19, 114, 38]. It should be noted that in [227] corresponding calculations that include the elasticity of the vessel wall were also conducted. However, these are not applied in three-dimensional simulation as frequently.

With increasing computational resources, the fields of CFD and CSD have evolved to a state that allows for three-dimensional simulations of blood flow. In early studies of this kind, like the ones by [193], a rigid wall was assumed, i.e. only a CFD analyses was performed. Also more recently, according analyses are conducted for various investigations of e.g. the formation of stenoses ([237]), optimized bypass grafts ([222, 131]) and anastomoses ([137]), or hemodynamics in aneurysms ([212, 177]). Yet, for many applications, the rigid wall assumption does not lead to satisfactory results,

[3] Here, the only spatial dimension corresponds to the radial directions and not to the axial direction as in models for pulse wave propagation.

as shown in [20, 205, 104]. In view of this, and due to the development of efficient numerical methods in the last decade, blood flow simulations that take the FSI into account have become more and more prominent.

Modern medical imaging techniques such as computed tomography (CT) scanning and magnetic resonance imaging (MRI) allow to reconstruct real vessel geometries and to use them as a basis for a numerical simulations, see e.g. [202]. Torii ([203, 205]), Tezduyar ([194, 195, 196]) and Bazilevs ([19]) were among the first to conduct such *patient specific* simulations that take the FSI into account. While in [203, 205], partitioned solution approaches are employed to solve the FSI problem, monolithic procedures are used in [194, 195, 196, 19]. The latter have been further developed and applied in the scope of cardiovascular FSI simulations in many works, see e.g. those by Takizawa ([188, 190, 187]). An overview of the special techniques developed in this scope is given in [186]. This includes methods for the imposition of Womersley velocity profiles at non-circular inlet boundaries and special techniques to evaluate WSS related quantities, both of which are also applicable when using partitioned solution approaches.

The root of the instability issues associated with partitioned FSI simulation is the *added mass effect*. A detailed mathematical investigation is provided in [77]. The added mass effect is particularly high in cardiovascular studies since the fluid and the structure have almost the same density. Therefore, monolithic solution approaches are favored for cardiovascular FSI problems and biomechanical FSI problems in general in [128]. Partitioned simulations of cardiovascular FSI as conducted in [203, 205, 114], have been limited to simplified material models, which are oftentimes used in combination with a shifted pressure range[4]. However, as shown in [163], novel convergence acceleration methods as the one proposed by [51] can yield stable simulations, also for full pressure ranges and state-of-the-art constitutive equations (material models). Isolated from the question about the most suitable method for the three-dimensional FSI problem, the partitioned solution approach can be used to flexibly couple the FSI problem with reduced order models that account for the surrounding. The practicability of such a coupling was shown in [164]. In the present work, the idea of treating the individual models (fluid, structure, surrounding tissue, distal vessel network) as black-boxes is investigated extensively.

The works cited above are motivated by different application scenarios.

[4] Instead of a blood pressure of $80\,\mathrm{mmHg} to 120\,\mathrm{mmHg}$, only the pulse pressure, i.e. $0\,\mathrm{mmHg} to 40\,\mathrm{mmHg}$, is considered in the simulation

While the hemodynamics in aneurysms are investigated in [202, 203, 205, 195], other studies consider the blood flow in bypass grafts ([14, 114]) or arteries in general ([61, 203]). An overview of the various application scenarios for cardiovascular FSI simulations from a clinical perspective is given in [147, 211].

Constitutive models for soft tissue

A frequently applied constitutive model for soft tissue was proposed by Holzapfel, Gasser and Ogden (2000) [101]. This HGO model takes into account the anisotropy observed in arteries due to their cellular structure. More precisely, the directed collagen fibers in the media and adventitia motivated the development of this transversely isotropic model. Further, it accounts for the stiffening effect of soft tissues at large strain which results in the decreasing compliance of artery with increasing internal pressure. The model has been applied for different kind of applications, such as the simulation of stent-artery contact (see [82]). Different variants of the model have been proposed, including the ones by [235], which take into account the active behavior of the smooth muscle cells in the media. In [176, 236], the numerical solution of structural mechanics problems including the HGO model is emphasized. Less frequently than for single field structural simulations, the HGO model has also been applied to FSI simulations, see [163, 112].

In [204], it was shown that the assumption regarding the vessel's material behavior has a great effect on the results for the WSS and other quantities. Accordingly, it is of great interest to used state-of-the-art material models such as those proposed in [101]. However, the local fiber direction is not accessible with standard, non-invasive medical imaging techniques, which is why these models are not applicable for complicated geometries as faced in patient-specific simulations. Existing isotropic models that can describe the stiffening effect of soft tissue at large strain as well therefore constitute the best alternative. These models include the one proposed by Delfino et al. (1997) [57], who used it to describe the mechanical behavior of a carotid bifurcation. This or equivalent models were used frequently in FSI simulations, see e.g. [189, 238, 108, 104], yet many studies employ simplified material models, see e.g. [203, 205, 146]. Further, isotropic variants of the model proposed by Fung et al. (1979) [80] may be applied. It should be noted, that the model proposed by Delfino is called a Fung-type model as well. However, the only similarity is the exponential relation between stresses and strains, which is also a characteristic of the HGO model. The exponents are different for each model, which motivate the naming convention followed here.

All material models mentioned above were designed to be used along with an incompressibility constraints. However, the experimentally observed compressibility of arteries (see [234]) and the impossibility to use incompressible models in the scope of a finite element method with a pure-displacement formulation motivate a modification of the material laws. While this was already done for the HGO model in [82, 236, 152], compressible variants of the models by Fung and Delfino are derived in the present work (in Section 3.3). To this end, the volumetric isochoric split proposed in [74] is applied. This allows for a solution using a pure-displacement formulation of the high-order finite element method (p-FEM), as mentioned in the introduction. With the p-FEM, undesired locking effects to be expected when using standard low-order methods can be overcome. Locking constitutes a severe underestimation of displacements due to anisotropy, near-incompressibility and/or thin walls, all of which are typical characteristics of arteries. A review of the state-of-the art in the field of the FEM is postponed to Section 4.1, where the underlying mathematical framework is presented.

An issue that was already addressed in [101, 82] are residual strains in arteries. They lead to the effect that an artery wall will spring open if it is cut along the axial direction. Further, an artery segment contracts in the axial direction when explanted. In [195], a novel method for taking into account residual strains was introduced, developed further, and applied to complex geometries in the scope of FSI simulations in [192, 191]. The residual strains in arteries will be addressed again in Chapter5, yet, a detailed investigation is out of the scope of this work.

3 Mechanical modeling of the arterial system

If the phenomenon of blood flow in human arteries is considered as the flow of a viscous fluid through a flexible structure, the problem can be stated as a two-field problem. Further, if the fields (the fluid mechanics field and the structural mechanics field) are modeled in a three-dimensional continuum mechanics sense, the problem is denoted as a fully resolved fluid-structure interaction (FSI) problem here. Both fields are governed by the conservation of mass and momentum. However, due to the different mechanical behavior of fluids and structures, different formulations of these laws are applied. In view of the partitioned solution approach, this leads to separate subproblems for both fields. As outlined in the previous chapter, another reason for a split into subproblems can be the need for different resolutions in different regions of the overall problem domain. The fluid-structure interaction in the cardiovascular system and the mechanical behavior of the entire human body are very typical examples for this. However, there are many application scenarios, in which such a split is useful – for this or another reason.

In view of this, a general formulation for coupled problems is introduced in the first section of this chapter. Without any reference to an underlying application, the difference between monolithic and partitioned solution approaches is illustrated and a common notation for a subproblem is introduced. This level of abstraction will not reappear until the end of Chapter 4, where the partitioned solution approach is discussed in detail. Like the coupling software introduced there, it is designed to be generally applicable to multifield problems.

In the second section, the governing equations of a fully resolved FSI problem are formulated. This includes a general kinematic framework and a formulation of the mentioned conservation equations for different types of computational domains. Only standard constitutive equations are outlined in this section without a reference to a specific application.

Table 3.1: Subproblems governing coupled cardiovascular FSI simulations.

subproblem	state variables	field solver task
fluid	pressure p, velocity $\boldsymbol{v}$, domain displacement $\hat{\boldsymbol{d}}$	Compute the velocity and pressure in a moving region filled with viscous fluid given the displacement of the boundary of that region and the pressure or velocity on the boundary.
structure	displacement $\boldsymbol{d}$	Compute the deformation of a solid body, given tractions or displacements on its boundary. All stresses within the (hyperelastic) body are uniquely defined by its displacement.
1D vessel	pressure $p^{1\mathrm{D}}$, flow $Q^{1\mathrm{D}}$, area $A^{1\mathrm{D}}$	Compute the cross-sectional pressure, flow and area along a vessel, given flow or pressure and the area at both ends.
windkessel	pressure p^{w}, flow Q^{w}	Compute the pressure at the inlet of a vessel network (represented by a windkessel model) given the flow at this point.
foundation	displacement $\boldsymbol{d}^{\mathrm{t}}$	Compute the traction exerted by the surrounding tissue on the arterial wall given the displacement at this point.

In the third section, the application of the partitioned solution approach to cardiovascular FSI problems is taken up again. Different reduced-order models are introduced as further subproblems and their interaction with the three-dimensional FSI problem is discussed. Further, constitutive equations for arteries and blood are introduced.

In summary, all subproblems interacting in the developed simulation approach for cardiovascular FSI are introduced in this chapter. In the next chapter, numerical methods are introduced for each of them. Here, all quantities are to be seen as continuous functions of the time t and (possibly) the current position in space $\boldsymbol{x}$, which are defined for certain parts $\Omega^{(p)}$ of the overall computational domain Ω. Each subproblem or field is identified with one of these parts and different quantities or field variables are used to describe the state of the fluid, respectively the structure, inside $\Omega^{(p)}$. Table 3.1 lists all subproblems and introduces the corresponding state variables.

3.1 Coupled problems

Coupled problems are characterized by the fact that multiple subproblems interact with each other. In order to introduce the indexing in iterative solution procedures and to illustrate the differences between partitioned and monolithic solution approaches, a not further specified coupled problem with n^{p} subproblems is considered. A mathematical formulation for a subproblem p with a state variable $\boldsymbol{u}^{(p)}(\boldsymbol{x}, t)$ as part of a surface coupled problem may be stated as

$$\boldsymbol{D}^{(p)}(\boldsymbol{x}, t, \boldsymbol{u}^{(p)}, \dot{\boldsymbol{u}}^{(p)}, \boldsymbol{u}'^{(p)}, \ddot{\boldsymbol{u}}^{(p)}, \dots) = \mathbf{0} \quad \text{in } \Omega^{(p)} \tag{3.1}$$

$$\boldsymbol{B}^{(p)}(\boldsymbol{x}, t, \boldsymbol{u}^{(p)}, \dot{\boldsymbol{u}}^{(p)}, \boldsymbol{u}'^{(p)}, \ddot{\boldsymbol{u}}^{(p)}, \dots) = \mathbf{0} \quad \text{on } \Gamma_{\mathrm{b}}^{(p)} \tag{3.2}$$

$$\boldsymbol{C}^{(p)}(\boldsymbol{x}, t, \boldsymbol{u}^{(1)}, \dots, \boldsymbol{u}^{(n^{\mathrm{p}})}, \dot{\boldsymbol{u}}^{(1)}, \dots, \dot{\boldsymbol{u}}^{(n^{\mathrm{p}})}, \boldsymbol{u}'^{(1)}, \dots, \boldsymbol{u}'^{(n^{\mathrm{p}})}, \dots) = \mathbf{0} \quad \text{on } \Gamma_{\mathrm{c}}^{(p)}. \tag{3.3}$$

The mathematical equations governing the pth subproblem are represented by (3.1). Boundary conditions which are independent of other subproblems are included in (3.2). The influence of other subproblems is represented by coupling conditions (3.3). Dots (e.g $\dot{\boldsymbol{u}}^{(p)}$) and apostrophes (e.g. $\boldsymbol{u}'^{(p)}$) indicate not further specified temporal and spatial derivatives, respectively. In a volume-coupled problem, the coupling conditions must hold in the domain $\Omega^{(p)}$, rather than on its boundary $\Gamma^{(p)}$. However, the following considerations are independent of the type of coupling. The coupled problem can be stated as follows: *Find* $\boldsymbol{u}^{(p)}$ *such that*

$$\begin{cases} \boldsymbol{D}^{(p)} &= \mathbf{0} \quad \text{in } \Omega^{(p)}, \\ \boldsymbol{B}^{(p)} &= \mathbf{0} \quad \text{on } \Gamma_{\mathrm{b}}^{(p)}, \\ \boldsymbol{C}^{(p)} &= \mathbf{0} \quad \text{on } \Gamma_{\mathrm{c}}^{(p)}. \end{cases} \tag{3.4}$$

at any time $t \in [t_{\mathrm{s}}, t_{\mathrm{e}}]$ *and for all* $p \in [1, 2, \dots, n^{\mathrm{p}}]$

3.1.1 Solution approaches

In view of the numerical methods to be applied to solve the individual subproblems, the state variables $\boldsymbol{u}^{(p)}(\boldsymbol{x}, t)$ will eventually be represented by a finite number of values or *degrees of freedom*. To this end, a corresponding vector of degrees of freedom $\mathbf{u}^{(p)}(t)$ is introduced in the context of the spatial discretization. Throughout this work, discretized quantities are represented by upright letters, while continuous variables or fields are represented using italic letters. The applied numerical methods use a time stepping scheme to compute the state variables at consecutive instances in time $t_0 = t_\mathrm{s}$, $t_1 = t_0 + \Delta t_1$, $t_2 = t_1 + \Delta t_2$, etc., where Δt_k is the *time step size*. Defining $\mathbf{u}_k^{(p)} = \mathbf{u}^{(p)}(t_k)$, the discrete subproblems to be solved for $\mathbf{u}_{k+1}^{(p)}$ in each time step k can be stated as

$$\mathbf{P}_k^{(p)} = \mathbf{D}_k^{(p)}(\mathbf{u}_{k+1}^{(p)}) + \mathbf{B}_k^{(p)}(\mathbf{u}_{k+1}^{(p)}) + \mathbf{C}_k^{(p)}(\mathbf{u}_{k+1}^{(1)}, \ldots, \mathbf{u}_{k+1}^{(n^\mathrm{P})}) = \mathbf{0}. \tag{3.5}$$

As done for the state variables, $\mathbf{D}_k^{(p)}$, $\mathbf{B}_k^{(p)}$ and $\mathbf{C}_k^{(p)}$ were introduced as the discrete counterparts to the continuous equations $\boldsymbol{D}^{(p)}$, $\boldsymbol{B}^{(p)}$ and $\boldsymbol{C}^{(p)}$. They may also depend on the state variables in previous time steps as indicated by the subscript k. The discrete system arising from the application of the respective numerical method is assumed to be closed, i.e. there are as many equations collected in $\mathbf{P}_k^{(p)}$ as there are components in $\mathbf{u}^{(p)}$. It should be noted that – in accordance to $\boldsymbol{B}^{(p)}$ and $\boldsymbol{C}^{(p)}$ being defined only for parts of $\Omega^{(p)}$ or its boundary – $\mathbf{B}_k^{(p)}$ and $\mathbf{C}_k^{(p)}$ will most likely involve only some of the degrees of freedom collected in $\mathbf{u}_{k+1}^{(p)}$. The discrete coupled problem can now be stated as

$$\begin{cases} \mathbf{P}_k^{(1)}(\mathbf{u}_{k+1}^{(1)}, \ldots, \mathbf{u}_{k+1}^{(n^\mathrm{P})}) & = \mathbf{0} \\ \mathbf{P}_k^{(2)}(\mathbf{u}_{k+1}^{(1)}, \ldots, \mathbf{u}_{k+1}^{(n^\mathrm{P})}) & = \mathbf{0} \\ \qquad\qquad\vdots & \quad\vdots \\ \mathbf{P}_k^{(n^\mathrm{P})}(\mathbf{u}_{k+1}^{(1)}, \ldots, \mathbf{u}_{k+1}^{(n^\mathrm{P})}) & = \mathbf{0} \end{cases}. \tag{3.6}$$

It can be generally treated as one nonlinear problem, which may be solved using the Newton-Raphson method. In a monolithic solution approach, the resulting linearized system of equations

$$\begin{bmatrix} \mathrm{D}_{\mathbf{u}^{(1)}}\mathbf{P}_k^{(1)} & \mathrm{D}_{\mathbf{u}^{(2)}}\mathbf{P}_k^{(1)} & \ldots & \mathrm{D}_{\mathbf{u}^{(n^\mathrm{P})}}\mathbf{P}_k^{(1)} \\ \mathrm{D}_{\mathbf{u}^{(1)}}\mathbf{P}_k^{(2)} & \mathrm{D}_{\mathbf{u}^{(2)}}\mathbf{P}_k^{(2)} & \ldots & \mathrm{D}_{\mathbf{u}^{(n^\mathrm{P})}}\mathbf{P}_k^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{D}_{\mathbf{u}^{(1)}}\mathbf{P}_k^{(n^\mathrm{P})} & \mathrm{D}_{\mathbf{u}^{(2)}}\mathbf{P}_k^{(n^\mathrm{P})} & \ldots & \mathrm{D}_{\mathbf{u}^{(n^\mathrm{P})}}\mathbf{P}_k^{(n^\mathrm{P})} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_i^{(1)} \\ \Delta\mathbf{u}_i^{(2)} \\ \vdots \\ \Delta\mathbf{u}_i^{(n^\mathrm{P})} \end{bmatrix} = \begin{bmatrix} -\mathbf{P}_k^{(1)}(\tilde{\mathbf{u}}_i^{(1)}, \ldots, \tilde{\mathbf{u}}_i^{(n^\mathrm{P}}) \\ -\mathbf{P}_k^{(2)}(\tilde{\boldsymbol{u}}_i^{(2)}, \ldots, \tilde{\mathbf{u}}_i^{(n^\mathrm{P}}) \\ \vdots \\ -\mathbf{P}_k^{(n^\mathrm{P})}(\tilde{\boldsymbol{u}}_i^{(n^\mathrm{P})}) \end{bmatrix} \tag{3.7}$$

indeed reflects the basis for the computations performed in each time step. Obviously, all subproblems are solved simultaneously by computing improved approximations $\tilde{\mathbf{u}}_i^{(p)}$ of the new states $\mathbf{u}_{k+1}^{(p)}$. This is done iteratively such that in each iteration i

$$\tilde{\mathbf{u}}_{i+1}^{(p)} = \tilde{\mathbf{u}}_i^{(p)} + \Delta\mathbf{u}_i^{(p)} \tag{3.8}$$

provides an improved approximation. When the Newton-Raphson method converges after n^{n} iterations, the new solution is set to $\mathbf{u}_{k+1}^{(p)} = \tilde{\mathbf{u}}_{n^{\mathrm{n}}}^{(p)}$.

With the introduced formulation, a partitioned solution approach can be interpreted as a *block Gauss-Seidel* method applied to the monolithic system (3.7). To this end, a *coupling iteration* with index j is introduced. Each iteration starts with the first subproblem and $\mathbf{u}_{k+1}^{(1,j)}$ is computed, possibly using Newton-Raphson iterations, i.e.

$$\mathrm{D}_{\mathbf{u}^{(1)}}\mathbf{P}_k^{(1)}\,\Delta\mathbf{u}_i^{(1)} = -\mathbf{P}_k^{(1)}(\tilde{\mathbf{u}}_i^{(1)}, \mathbf{u}_{k+1}^{(2,j)}, \dots, \mathbf{u}_{k+1}^{(n^{\mathrm{P}},j)}), \tag{3.9}$$

$$\tilde{\mathbf{u}}_{i+1}^{(1)} = \tilde{\mathbf{u}}_i^{(1)} + \Delta\mathbf{u}_i^{(1)}. \tag{3.10}$$

When the Newton-Raphson method converges after n^{n} iterations, the new solution is set to $\mathbf{u}_{k+1}^{(1,j)} = \tilde{\mathbf{u}}_{n^{\mathrm{n}}}^{(1)}$. It is noted that $\mathbf{u}_{k+1}^{(1,j)}$ does solve $\mathbf{P}_k^{(1)}(\mathbf{u}_{k+1}^{(1)}, \dots, \mathbf{u}_{k+1}^{(n^{\mathrm{P}})})$ but is not necessarily a solution to the coupled problem. Generally, the process of solving subproblem P in coupling iteration j may be formulated as

$$\mathrm{D}_{\mathbf{u}^{(P)}}\mathbf{P}_k^{(P)}\Delta\mathbf{u}_i^{(P)} = -\mathbf{P}_k^{(P)}(\mathbf{u}_{k+1}^{(1,j+1)}, \dots, \tilde{\mathbf{u}}_i^{(P)}, \dots, \mathbf{u}_{k+1}^{(n^{\mathrm{P}},j)}), \tag{3.11}$$

$$\tilde{\mathbf{u}}_{i+1}^{(P)} = \tilde{\mathbf{u}}_i^{(P)} + \Delta\mathbf{u}_i^{(P)}. \tag{3.12}$$

The subscripts and superscripts were introduced so that any variable can be uniquely defined within the scope of the computations it appears in. Since the variable $\tilde{\mathbf{u}}_i^{(p)}$ never appears outside of the inner solver iterations, the subscripts and superscripts for the time step and the coupling iteration are dropped. Besides this, the following conventions are used throughout this work:

- Intermediate values are marked with a tilde, e.g. $\tilde{\mathbf{u}}_i^{(p)}$. They do not fulfill the respective subproblem.
- A superscript denotes the coupling iteration, e.g. j and indicates that the quantity does not fulfill the coupled problem but only the respective subproblem.
- The time step is indicated by a subscript, e.g. k, which either refers to a certain item from a set of discrete values or an evaluation of a function $f(t)$ at t_k.

In the remainder of this chapter, the superscript j will be dropped for the sake of readability, since all subproblems are introduced separately without a connection to the coupled problem.

3.2 Continuum mechanics

When considering the conservation of a quantity ω, the assumption of a continuum gives rise to so-called *localization arguments.* Due to the fact, that quantities are continuous in space and time, an integral statement may be considered for an arbitrarily small region Ω and for a general conservation law $c(\omega)$

$$\int\limits_{\Omega} c(\omega)\,\mathrm{d}\Omega = 0 \quad \Leftrightarrow \quad c(\omega) = 0 \quad \text{in } \Omega \tag{3.13}$$

holds. The equivalence relation (3.13) is denoted the *spatial* localization argument. A localization argument *with respect to time* can be expressed by

$$\int\limits_{t_a}^{t_b} c(\omega)\,\mathrm{d}t = 0 \quad \Leftrightarrow \quad c(\omega) = 0. \quad \forall t \in [t_{\mathrm{s}}, t_{\mathrm{e}}] \tag{3.14}$$

Different types of regions (referred to as *domains* in the following, as usual) are needed on order to describe fluids and structures. In structural continuum mechanics, as convenient by reasoning, the domain is usually attached to the matter and moves along with it. One could say that the domain is attached to the body under consideration. On the fluid mechanics side, the domain does not follow the matter, but instead reflects an independently moving region, in which the fluid motion is to be described. Fluid particles may enter and leave this type of domains, whose boundaries - in case of FSI - follow the structure.

To formally deal with computations on such moving domains, a general mathematical framework is introduced in the next section. This is done independently of the considered problem to establish general conservation laws in moving domains. In the following sections, the underlying equations for structural and fluid dynamics are introduced by formulating the conservation of mass and momentum based on the general conservation equations for the respective type of domain.

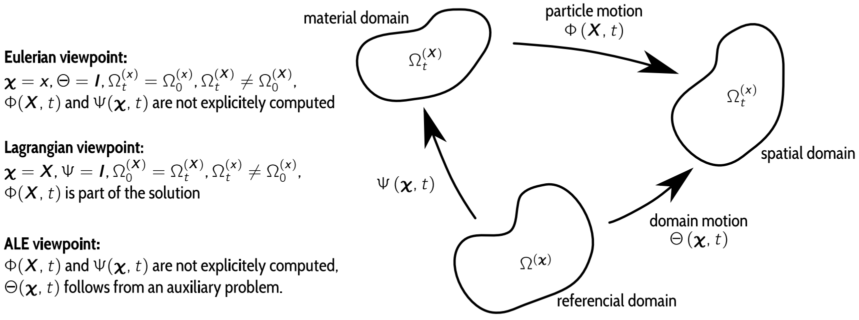

Eulerian viewpoint:
$\boldsymbol{\chi} = \boldsymbol{x}, \Theta = \boldsymbol{I}, \Omega_t^{(x)} = \Omega_0^{(x)}, \Omega_t^{(X)} \neq \Omega_0^{(X)}$,
$\Phi(\boldsymbol{X}, t)$ and $\Psi(\boldsymbol{\chi}, t)$ are not explicitely computed
Lagrangian viewpoint:
$\boldsymbol{\chi} = \boldsymbol{X}, \Psi = \boldsymbol{I}, \Omega_0^{(X)} = \Omega_t^{(X)}, \Omega_t^{(x)} \neq \Omega_0^{(x)}$,
$\Phi(\boldsymbol{X}, t)$ is part of the solution
ALE viewpoint:
$\Phi(\boldsymbol{X}, t)$ and $\Psi(\boldsymbol{\chi}, t)$ are not explicitely computed,
$\Theta(\boldsymbol{\chi}, t)$ follows from an auxiliary problem.
material domain
$\Omega_t^{(X)}$
particle motion
$\Phi(\boldsymbol{X}, t)$
$\Omega_t^{(x)}$
spatial domain
$\Psi(\boldsymbol{\chi}, t)$
domain motion
$\Theta(\boldsymbol{\chi}, t)$
$\Omega^{(\chi)}$
referencial domain

Figure 3.1: Configurations of a moving domain and associated mappings.

3.2.1 Conservation laws on moving domains

The introduction to the kinematics of moving domains given here summarizes the deliberations from [18, 60] using a suitable notation for the later formulation of the numerical methods. A detailed explanation of the basic concepts followed here can be found in standard text books, e.g. [5, 230, 26].

As a general starting point for the description of the physical behavior of continua, a fixed domain $\Omega^{(\chi)} \subset \mathbb{R}^3$ is considered. It represents the observation area or *control volume* that moves and deforms as time evolves in its initial or referential state. While $\Omega^{(\chi)}$ denotes a fixed domain with referential coordinates $\boldsymbol{\chi}$, the current shape of the control volume is represented by the domain $\Omega_t^{(\boldsymbol{x})}$ with spatial coordinates $\boldsymbol{x}$. A space-time mapping

$$\boldsymbol{\Theta}\left(\left[\boldsymbol{\chi}^{\mathrm{T}}, t\right]^{\mathrm{T}}\right) : \Omega^{(\chi)} \times [t_0, t_{\mathrm{end}}] \rightarrow \Omega_t^{(\boldsymbol{x})} \times [t_0, t_{\mathrm{end}}] \tag{3.15}$$

identifies every point $\boldsymbol{\chi}$ in $\Omega^{(\chi)}$ with a point $\boldsymbol{x}$ in $\Omega_t^{(\boldsymbol{x})}$, such that $\boldsymbol{y} = \boldsymbol{\Theta}(\boldsymbol{v})$ holds for the space time coordinates $\boldsymbol{y} = \left[\boldsymbol{x}^{\mathrm{T}}, t\right]^{\mathrm{T}}$ and $\boldsymbol{v} = \left[\boldsymbol{\chi}^{\mathrm{T}}, t\right]^{\mathrm{T}}$. The mapping $\boldsymbol{\Theta}$ can be identified as the *domain motion*, cf. Fig. 3.1. For a formal treatment of derivatives with respect to $\boldsymbol{\chi}$, $\boldsymbol{x}$ and t, the space-time gradient

$$\mathrm{grad}_{\boldsymbol{v}}(\boldsymbol{\Theta}) = \begin{bmatrix} \hat{\boldsymbol{F}} & \hat{\boldsymbol{v}} \\ \boldsymbol{0}^T & 1 \end{bmatrix} \tag{3.16}$$

is established, where

$$\hat{\boldsymbol{F}} = \mathrm{grad}_{\boldsymbol{\chi}}(\boldsymbol{x}) = \mathrm{GRAD}(\boldsymbol{x}) \tag{3.17}$$

denotes a deformation gradient and

$$\hat{\boldsymbol{v}} = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{\chi}} \tag{3.18}$$

denotes the domain velocity.

A third domain $\Omega_t^{(\boldsymbol{X})}$ is introduced, which is called the *material configuration* and coincides with the region that at time $t = 0$ was occupied by all particles in $\Omega_t^{(\boldsymbol{x})}$. Accordingly, a mapping

$$\boldsymbol{\Phi}\left(\left[\boldsymbol{X}^{\mathrm{T}}, t\right]^{\mathrm{T}}\right) : \Omega^{(\boldsymbol{X})} \times [t_0, t_{\mathrm{end}}] \rightarrow \Omega_t^{(\boldsymbol{x})} \times [t_0, t_{\mathrm{end}}] \tag{3.19}$$

relates the initial particle position $\boldsymbol{X}$ with the current one $\boldsymbol{x}$ such that $\boldsymbol{y} = \boldsymbol{\Phi}(\boldsymbol{Y})$ holds with the material space time coordinates $\boldsymbol{Y} = \left[\boldsymbol{X}^{\mathrm{T}}, t\right]^{\mathrm{T}}$. The mapping $\boldsymbol{\Phi}$ can be identified as the *particle motion* and its gradient

$$\mathrm{grad}_{\boldsymbol{Y}}(\boldsymbol{\Phi}) = \begin{bmatrix} \boldsymbol{F} & \boldsymbol{v} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{bmatrix} \tag{3.20}$$

gives rise to the well-known deformation gradient

$$\boldsymbol{F} = \mathrm{grad}_{\boldsymbol{X}}(\boldsymbol{x}) = \mathrm{Grad}(\boldsymbol{x}) \tag{3.21}$$

and the particle velocity

$$\boldsymbol{v} = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{X}} . \tag{3.22}$$

As illustrated in Fig. 3.1, a third mapping that relates the referential coordinates $\boldsymbol{\chi}$ and the material coordinates $\boldsymbol{X}$ follows from the two mappings introduces so far.

$$\boldsymbol{\Psi}\left(\left[\boldsymbol{\chi}^{\mathrm{T}}, t\right]^{\mathrm{T}}\right) : \Omega^{(\boldsymbol{\chi})} \times [t_0, t_{\mathrm{end}}] \rightarrow \Omega_t^{(\boldsymbol{X})} \times [t_0, t_{\mathrm{end}}] \tag{3.23}$$

defines the relation $\boldsymbol{Y} = \boldsymbol{\Psi}(\boldsymbol{\upsilon}) = \boldsymbol{\Phi}^{-1}(\boldsymbol{\Theta}(\boldsymbol{\upsilon}))$ and a deformation gradient

$$\check{\boldsymbol{F}} = \mathrm{GRAD}(\boldsymbol{X}) \tag{3.24}$$

as well as a velocity

$$\check{\boldsymbol{v}} = \left.\frac{\partial \boldsymbol{X}}{\partial t}\right|_{\boldsymbol{\chi}} \tag{3.25}$$

follow from its gradient $\mathrm{grad}_{\boldsymbol{\upsilon}}(\boldsymbol{\Psi})$. According to Fig. 3.1, special choices for the mappings $\boldsymbol{\Theta}$ and $\boldsymbol{\Psi}$ give rise to well-known descriptions of the particle motion $\boldsymbol{\Phi}$.

- If a *Lagrangian* description is employed, the domain motion is chosen to coincide with the particle motion, i.e. $\boldsymbol{\Psi} = \boldsymbol{I}$. This description is commonly used in the field of structural mechanics. Because the spatial domain $\Omega_t^{(\boldsymbol{x})}$ and likewise the mappings $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ are part of the solution to a corresponding structural problem[1] all computations are performed with respect to the configuration $\Omega_0^{(\boldsymbol{X})}$. In the context of structural mechanics,

$$\Omega_0^{\mathrm{s}} = \Omega_0^{(\boldsymbol{X})} = \Omega_t^{(\boldsymbol{X})} = \Omega_0^{(\boldsymbol{x})} = \Omega^{(\boldsymbol{\chi})} \tag{3.26}$$

  denotes the *undeformed reference configuration.* The spatial or *current configuration* is denoted by

$$\Omega_t^{\mathrm{s}} = \Omega_t^{(\boldsymbol{x})}. \tag{3.27}$$

- In an *Eulerian* description, the control volume is independent of the particles. It is fixed in space ($\boldsymbol{\Theta} = \boldsymbol{I}$) and particles enter and leave the fixed domain $\Omega_t^{(\boldsymbol{x})} = \Omega_0^{(\boldsymbol{x})} = \Omega^{(\boldsymbol{\chi})}$ as time evolves. Such a description is commonly used in the field of fluid mechanics. An undeformed configuration $\Omega_t^{(\boldsymbol{X})}$ is not computed, which allows to cope with the large deformations, e.g. vortices in the flow, which are common in fluid dynamics. However, special attention has to be paid when establishing conservation laws within such an Eulerian description. While all quantities are defined in terms of spatial coordinates $\boldsymbol{x}$, the conservation of a quantity must still be formulated with respect to a fixed material point, i.e. a fixed $\boldsymbol{X}$.

- In the context of fluid-structure interaction, the standard Eulerian description is not directly applicable, because flows in moving domains are considered. Therefore, the so called *Arbitrary Lagrangian Eulerian (ALE) description* is used. In the ALE description, the domain motion $\boldsymbol{\Theta}$ and the particle motion $\boldsymbol{\Phi}$ are independent of each other and neither $\Omega_t^{(\boldsymbol{x})}$ nor $\Omega_t^{(\boldsymbol{X})}$ are fixed. As in the Eularian description, special attention has to be paid when establishing conservation laws. Further, an auxiliary problem needs to be formulated, allowing for a computation of the domain motion $\boldsymbol{\Theta}$. In most cases, an *extension operator* is established that allows for a computation of the internal domain displacement $\hat{\boldsymbol{d}} = \boldsymbol{x} - \boldsymbol{\chi}$ from given displacement at the boundary $\partial\Omega_t^{(\boldsymbol{x})}$.

[1] This holds true as long as geometrical nonlinearities are considered

Possible choices for such an extension operator will be given in Section 3.2.3. In the context of fluid mechanics, the moving domain is denoted by

$$\Omega_t^{\mathrm{f}} = \Omega_t^{(\boldsymbol{x})}. \tag{3.28}$$

Material time derivative

The introduced configurations $\Omega^{(\boldsymbol{\chi})}$ (referential), $\Omega^{(\boldsymbol{X})}$ (material), and $\Omega^{(\boldsymbol{x})}$ (spatial) serve as a basis for different formulations of the mentioned conservation laws. For a not further specified quantity ω, its conservation for a certain particle can be described using a function $\omega^{(\boldsymbol{\chi})}(\boldsymbol{\chi}, t)$, $\omega^{(\boldsymbol{x})}(\boldsymbol{x}, t)$ or $\omega^{(\boldsymbol{X})}(\boldsymbol{X}, t)$, depending on the chosen configuration. One can establish

$$\frac{\mathrm{D}\omega}{\mathrm{D}t} = \left.\frac{\partial \omega^{(\boldsymbol{X})}}{\partial t}\right|_{\boldsymbol{X}} = \left.\frac{\partial \omega^{(\boldsymbol{x})}}{\partial t}\right|_{\boldsymbol{x}} + \operatorname{grad}\Big(\omega^{(\boldsymbol{x})}\Big) \cdot \boldsymbol{v} = \left.\frac{\partial \omega^{(\boldsymbol{\chi})}}{\partial t}\right|_{\boldsymbol{\chi}} + \operatorname{GRAD}\Big(\omega^{(\boldsymbol{\chi})}\Big) \cdot \hat{\boldsymbol{v}} \tag{3.29}$$

for the so called *material time derivative* in the different configurations. As done for referential and material coordinates in Eq. (3.17) and (3.21), respectively,

$$\operatorname{grad}(\cdot) = \operatorname{grad}_{\boldsymbol{x}}(\cdot) \tag{3.30}$$

is introduced as the gradient with respect to the spatial coordinates for convenience. In the following $\frac{\mathrm{D}}{\mathrm{D}t}$ generally denotes a derivative with *the particles held fixed*. The superscript indicating the configurations is dropped for the sake of clarity. Since the introduced mappings provide an explicit relation of the different functional forms, e.g. $\omega^{(\boldsymbol{x})}(\boldsymbol{\Phi}(\boldsymbol{x}), t) = \omega^{(\boldsymbol{X})}(\boldsymbol{X}, t)$, it will be clear from the context (differentiation or integration with respect to $\boldsymbol{X}$, $\boldsymbol{x}$ or $\boldsymbol{\chi}$) which functional form is considered. The relations in (3.29) follow directly from a formal application of the space-time gradients and the chain rule, i.e. using the relations

$$\operatorname{grad}_{\boldsymbol{Y}}(\omega) = \operatorname{grad}_{\boldsymbol{Y}}(\boldsymbol{y})^{\mathrm{T}} \operatorname{grad}_{\boldsymbol{y}}(\omega) = \operatorname{grad}_{\boldsymbol{Y}}(\boldsymbol{\upsilon})^{\mathrm{T}} \operatorname{grad}_{\boldsymbol{\upsilon}}(\omega)\,. \tag{3.31}$$

Reynolds transport theorem

Let $\Omega_{\mathrm{L}}^{(\boldsymbol{x})}$ be a Lagrangian spatial domain, i.e. one that is occupied by the same particles for all t. The conservation of the quantity ω associated with the particles is to be formulated. To this end, a source β and a flux vector

$\boldsymbol{\gamma}^{(\boldsymbol{x})}$ are introduced to describe the creation and the transport of ω. This allows to write

$$\frac{\mathrm{D}}{\mathrm{D}t}\int\limits_{\Omega_{\mathrm{L}}^{(\boldsymbol{x})}} \omega \,\mathrm{d}\Omega_{\mathrm{L}}^{(\boldsymbol{x})} = \int\limits_{\partial\Omega_{\mathrm{L}}^{(\boldsymbol{x})}} \boldsymbol{\gamma}^{(\boldsymbol{x})} \cdot \boldsymbol{n} \,\mathrm{d}\partial\Omega_{\mathrm{L}}^{(\boldsymbol{x})} + \int\limits_{\Omega_{\mathrm{L}}^{(\boldsymbol{x})}} \beta \mathrm{d}\Omega_{\mathrm{L}}^{(\boldsymbol{x})} \tag{3.32}$$

for the material time derivative in an integral sense. It should be noted that the flux vector $\boldsymbol{\gamma}^{(\boldsymbol{x})}$ describes a flux of ω from particle to particle only and does not include convective effects. They have to be considered when particularizing the material derivative $\frac{\mathrm{D}}{\mathrm{D}t}$. One arrives at the Reynolds transport theorem

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \left.\frac{\partial \omega}{\partial t}\right|_{\boldsymbol{x}} \mathrm{d}\Omega^{(\boldsymbol{x})} + \int\limits_{\partial\Omega^{(\boldsymbol{x})}} \operatorname{grad}(\omega) \cdot \boldsymbol{v} \,\mathrm{d}\partial\Omega = \int\limits_{\partial\Omega^{(\boldsymbol{x})}} \boldsymbol{\gamma}^{(\boldsymbol{x})} \cdot \boldsymbol{n} \,\mathrm{d}\partial\Omega^{(\boldsymbol{x})} + \int\limits_{\Omega^{(\boldsymbol{x})}} \beta \,\mathrm{d}\Omega^{(\boldsymbol{x})}, \tag{3.33}$$

which seems obvious in consideration of Eq. (3.29). The restriction on the spatial domain to be a Lagrangian one is relaxed in (3.33) due to moving the differentiation inside the integral. With the introduced framework for moving domains, (3.33) can be derived using standard rules for coordinate transformations, see [5, 18]. A powerful technique frequently applied in this context is to first integrate in time, then change (space time) variables using the introduced mappings and, finally, use a localization argument in time. This allows for a transformation of the Reynolds transport theorem from the spatial to the reference and material configuration. To this end, a suitable transformation rule for the flux vector $\boldsymbol{\gamma}^{(\boldsymbol{x})}$ has to be defined. In order to preserve the conservation property of (3.33),

$$\boldsymbol{\gamma}^{(\boldsymbol{X})} = J\,\boldsymbol{F}^{-1}\,\boldsymbol{\gamma}^{(\boldsymbol{x})} \tag{3.34}$$

and

$$\boldsymbol{\gamma}^{(\boldsymbol{\chi})} = J\,\hat{\boldsymbol{F}}^{-1}\,\boldsymbol{\gamma}^{(\boldsymbol{x})} \tag{3.35}$$

the material and referential flux vector, respectively, are introduced. With (3.34) and (3.35), it is ensured that

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \operatorname{div}\Big(\boldsymbol{\gamma}^{(\boldsymbol{x})}\Big)\,\mathrm{d}\Omega^{(\boldsymbol{x})} = \int\limits_{\Omega^{(\boldsymbol{X})}} \operatorname{Div}\Big(\boldsymbol{\gamma}^{(\boldsymbol{X})}\Big)\,\mathrm{d}\Omega^{(\boldsymbol{X})} = \int\limits_{\Omega^{(\boldsymbol{\chi})}} \operatorname{DIV}\Big(\boldsymbol{\gamma}^{(\boldsymbol{\chi})}\Big)\,\mathrm{d}\Omega^{(\boldsymbol{\chi})}, \tag{3.36}$$

see [18] for a proof. Similar to the gradients in Eq. (3.17,3.21,3.30), convenient notations for the divergence with respect to the different coordinates were introduced in (3.36).

General conservation laws

By applying the divergence theorem, (3.33) can be recast into the form

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \left.\frac{\partial \omega}{\partial t}\right|_{\boldsymbol{x}} + \operatorname{div}\Big(\omega \boldsymbol{v} - \boldsymbol{\gamma}^{(\boldsymbol{x})}\Big) - \beta \, \mathrm{d}\Omega^{(\boldsymbol{x})} = 0, \tag{3.37}$$

which represents the general conservation law in the spatial domain. By integrating (3.37) over time before changing variables in the resulting space-time integral (from $\boldsymbol{y}$ to $\boldsymbol{Y}$) and then localizing the new space-time integral in time, one obtains the general conservation law in the material domain

$$\int\limits_{\Omega^{(\boldsymbol{X})}} \left.\frac{\partial \left(J \, \omega\right)}{\partial t}\right|_{\boldsymbol{X}} - \operatorname{Div}\Big(J \, \boldsymbol{F}^{-1} \, \boldsymbol{\gamma}^{(\boldsymbol{x})}\Big) - J \, \beta \, \mathrm{d}\Omega = 0. \tag{3.38}$$

In a corresponding way, a formulation suitable for the ALE description can be established. After some algebra as detailed in [18], one arrives at

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \left.\frac{\partial \omega}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{v} - \hat{\boldsymbol{v}}) \cdot \operatorname{grad}(\omega) + \omega \operatorname{div}(\boldsymbol{v}) - \operatorname{div}\Big(\boldsymbol{\gamma}^{(\boldsymbol{x})}\Big) - \beta \, \mathrm{d}\Omega^{(\boldsymbol{x})} = 0. \tag{3.39}$$

Note that (3.39) represents a mixed formulation of the general conservation law. While time derivatives are taken with respect to the referential time, i.e. with referential coordinates $\boldsymbol{\chi}$ held fixed, spatial gradients are computed with respect to the spatial coordinates $\boldsymbol{x}$. As for a general scalar quantity ω, conservation equations for a vector quantity $\boldsymbol{\omega}$ can be derived. With a corresponding spatial flux tensor $\boldsymbol{\Gamma}^{(\boldsymbol{x})}$ and source vector $\boldsymbol{\beta}$, one obtains

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \left.\frac{\partial \boldsymbol{\omega}}{\partial t}\right|_{\boldsymbol{x}} + \operatorname{div}\Big(\boldsymbol{\omega} \otimes \boldsymbol{v} - \boldsymbol{\Gamma}^{(\boldsymbol{x})}\Big) - \boldsymbol{\beta}\,\mathrm{d}\Omega^{(\boldsymbol{x})} = \mathbf{0}, \tag{3.40}$$

$$\int\limits_{\Omega^{(\boldsymbol{X})}} \left.\frac{\partial\left(J\,\boldsymbol{\omega}\right)}{\partial t}\right|_{\boldsymbol{X}} - \operatorname{Div}\Big(J\,\boldsymbol{\Gamma}^{(\boldsymbol{x})}\boldsymbol{F}^{-\mathrm{T}}\Big) - J\,\boldsymbol{\beta}\,\mathrm{d}\Omega^{(\boldsymbol{X})} = \mathbf{0} \tag{3.41}$$

$$\int\limits_{\Omega^{(\boldsymbol{x})}} \left.\frac{\partial \boldsymbol{\omega}}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{v} - \hat{\boldsymbol{v}}) \cdot \operatorname{grad}(\boldsymbol{\omega}) + \boldsymbol{\omega}\operatorname{div}(\boldsymbol{v}) - \operatorname{div}\Big(\boldsymbol{\Gamma}^{(\boldsymbol{x})}\Big) - \boldsymbol{\beta}\,\mathrm{d}\Omega^{(\boldsymbol{x})} = \mathbf{0} \tag{3.42}$$

as the spatial, material, and mixed form, respectively. From (3.41), the transformation rule for the flux tensor

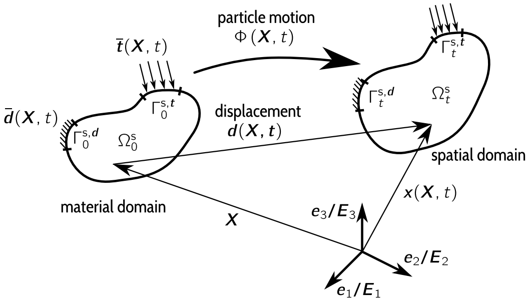
$$\boldsymbol{\Gamma}^{(\boldsymbol{X})} = J\,\boldsymbol{\Gamma}^{(\boldsymbol{x})}\,\boldsymbol{F}^{-\mathrm{T}} \tag{3.43}$$

can be identified.

In the following sections, the conservation for mass and linear momentum will be formulated using (3.37-3.39) and (3.40-3.42) to obtain the governing equations of structural and fluid dynamics. To this end, the density ρ and the linear momentum density $(\rho\,\boldsymbol{v})$ are substituted for the general quantities ω and $\boldsymbol{\omega}$, respectively. Regarding the conservation of mass, no sources or fluxes are present since mass is neither created nor transported from particle to particle. Regarding the conservation of linear momentum, the *Cauchy stress tensor* $\boldsymbol{\sigma}$ can be identified as the spatial flux of linear momentum and sources appear in the form of *volume loads* $\boldsymbol{b}$.

3.2.2 Structural mechanics

The structural mechanics subproblem is based on the formulation of the balance of linear momentum for a Lagrangian control volume. As illustrated in Fig. 3.2, it moves along with the solid body under consideration, which at time $t = 0$ is assumed to be in an undeformed reference configuration. The displacement $\boldsymbol{d}$ is introduced as the primary unknown in structural mechanics. Substituting ρ for ω in (3.38) to obtain the conservation of



Task of a structural field solver
Knowing $d(\boldsymbol{X}, t_k)$*, compute* $d(\boldsymbol{X}, t_{k+1})$ *given surface tractions* $\bar{t}(\boldsymbol{X}, t_{k+1})$ *on* $\Gamma_0^{s,t}$ *and displacments* $\bar{d}(\boldsymbol{X}, t_{k+1})$ *on* $\Gamma_0^{s,d}$.

Discrete structural solve operator
$\mathbf{d}_{k+1} = \mathcal{S}_k(\bar{\mathbf{t}}_{k+1}, \bar{\mathbf{d}}_{k+1})$

Figure 3.2: Structural mechanics kinematics, core quantities and role of a structural field solver in a partitioned simulation.

mass in the material configuration results in

$$\left.\frac{\partial J\,\rho}{\partial t}\right|_{\boldsymbol{X}} = 0. \tag{3.44}$$

The flux and the source term are zero, since mass is not created or transferred from particle to particle. Introducing the density at $t = 0$ as ρ_0,

$$\rho_0 = J\,\rho \tag{3.45}$$

must hold, which can be seen as a kinematic relation rather than a conservation equation. For the case of incompressible materials, it constitutes a constraint (namely $J = 1$) that has to be accounted for by special means, e.g. mixed formulation with additional unknowns, see [230, p. 406]. Here, only nearly incompressible materials are considered. Substituting $(\rho\,\boldsymbol{v})$ for $\boldsymbol{\omega}$ in (3.41) and applying the spatial localization argument, one arrives at the partial differential equation that describes the motion of the body under consideration. Augmented by Dirichlet (given displacement $\bar{\boldsymbol{d}}$) and Neumann boundary conditions (given traction $\bar{\boldsymbol{t}}$) as well as initial conditions, the structural mechanics initial boundary value problem (IBVP) is obtained.

$$\rho_0\,\ddot{\boldsymbol{d}} = \mathrm{Div}(\boldsymbol{P}) + \rho_0\,\boldsymbol{b} \qquad \text{in } \Omega_0^{\mathrm{s}} \tag{3.46}$$

$$\boldsymbol{d} = \bar{\boldsymbol{d}} \qquad \text{on } \Gamma_0^{\mathrm{s},\boldsymbol{d}} \tag{3.47}$$

$$\boldsymbol{P}\,\boldsymbol{N} = \bar{\boldsymbol{t}} \qquad \text{on } \Gamma_0^{\mathrm{s},\boldsymbol{t}} \tag{3.48}$$

$$\boldsymbol{d} = \boldsymbol{d}_0 \qquad \text{in } \Omega_0^{\mathrm{s}}, t = 0 \tag{3.49}$$

$$\boldsymbol{v} = \dot{\boldsymbol{d}} = \boldsymbol{v}_0 \qquad \text{in } \Omega_0^{\mathrm{s}}, t = 0 \tag{3.50}$$

In correspondence to (3.43), the *first Piola-Kirchhoff stress tensor*

$$\boldsymbol{P} = J\,\boldsymbol{\sigma}\,\boldsymbol{F}^{-\mathrm{T}} \tag{3.51}$$

was introduced. Further,

$$\dot{\boldsymbol{d}} = \left.\frac{\partial d}{\partial t}\right|_{\boldsymbol{X}} \quad \text{and} \quad \ddot{\boldsymbol{d}} = \left.\frac{\partial^2 d}{\partial t^2}\right|_{\boldsymbol{X}} \tag{3.52}$$

were introduced as the standard temporal derivatives.

In the context of coupled simulations, the IBVP (3.46–3.50) defines the *structural mechanics subproblem*, which is solved using a structural field solver with the task summarized in Fig. 3.2. Once the problem is discretized using a numerical method, the continuous quantities $\boldsymbol{d}(\boldsymbol{X}, t_k)$, $\bar{\boldsymbol{t}}(\boldsymbol{X}, t_k)$, etc. are represented by the respective degrees of freedom collected in the vectors $\mathbf{d}_k$, $\bar{\mathbf{t}}_k$ etc. This gives rise to the operator formulation introduced in Fig. 3.2. The boundary values $\bar{\mathbf{t}}_k$ and $\bar{\mathbf{d}}_k$ may depend on the state variables of other subproblems. Within the structural subproblem, however, they are treated as given independent quantities, as indicated by the overbar.

Problem (3.46–3.50) must be augmented by a constitutive equation that relates the displacement and stresses such that a closed problem is obtained. To ease to corresponding calculations, the *second Piola-Kirchhoff stress tensor*

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\,\boldsymbol{P} \tag{3.53}$$

is introduced, which is symmetric and defined in the material configuration, as opposed to $\boldsymbol{P}$, see [230, p. 37]. Further, the *right Cauchy-Green deformation tensor*

$$\boldsymbol{C} = \boldsymbol{F}^{\mathrm{T}}\,\boldsymbol{F} \tag{3.54}$$

and the *Green-Lagrange strain tensor*

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{F}^{\mathrm{T}}\,\boldsymbol{F} - \boldsymbol{1}\right) \tag{3.55}$$

are introduced. They relate infinitesimal line elements in the material configuration ($\mathrm{d}\boldsymbol{X}$) and in the current configuration ($\mathrm{d}\boldsymbol{x}$) through

$$\|\mathrm{d}\boldsymbol{x}\|^2 = \mathrm{d}\boldsymbol{X} \cdot \boldsymbol{C}\,\mathrm{d}\boldsymbol{X} \tag{3.56}$$

and

$$\|\mathrm{d}\boldsymbol{x}\|^2 - \|\mathrm{d}\boldsymbol{X}\|^2 = \mathrm{d}\boldsymbol{X} \cdot \boldsymbol{E}\,\mathrm{d}\boldsymbol{X}. \tag{3.57}$$

For a detailed introduction into stress and strain measures, see [230] or [26].

Hyperelastic constitutive equations

Constitutive equations are used to describe the material behavior in solid mechanics. They relate the internal loads within a body to the deformation and, thus, close the structural mechanics IBVP (3.46–3.50). With the aid of experiments, the parameters in the constitutive equation are fitted to give the best possible agreement between simulation and experiment. Throughout this work, hyperelastic constitutive equations are used to describe the behavior of the considered solids. They introduce a dependency between the chosen strain measure and the work conjugated stress measure through a *strain energy density function (SEDF)* W. Choosing the Green-Lagrange strain tensor $\boldsymbol{E}$ as the preferred strain measure (which will always be the case here)

$$\boldsymbol{S} = \frac{\partial W}{\partial \boldsymbol{E}} = 2\,\frac{\partial W}{\partial \boldsymbol{C}} = \boldsymbol{F}^{-1}\,\frac{\partial W}{\partial \boldsymbol{F}} \tag{3.58}$$

gives the second Piola-Kirchhoff stress tensor. This makes it possible to formulate the structural mechanics problem as a minimization problem.

Structural mechanics in terms of energy

The total strain energy stored in an elastic body is given by $E = \int_{\Omega_0^{\mathrm{s}}} W\,\mathrm{d}\Omega_0^{\mathrm{s}}$. The difference between E and the work performed by the external loads (body force $\bar{\boldsymbol{b}}$ and surface traction $\bar{\boldsymbol{t}}$) is given by

$$\Pi = \int\limits_{\Omega_0^{\mathrm{s}}} W - \boldsymbol{d} \cdot \bar{\boldsymbol{b}}\,\mathrm{d}\Omega_0^{\mathrm{s}} - \int\limits_{\Gamma_0^{\mathrm{s}}} \boldsymbol{d} \cdot \bar{\boldsymbol{t}}\,\mathrm{d}\Gamma_0^{\mathrm{s}}. \tag{3.59}$$

The minimum is found by setting the first variation $\delta_{\boldsymbol{d}}\Pi$ to zero, i.e.

$$\delta_{\boldsymbol{d}}\Pi = \int\limits_{\Omega_0^{\mathrm{s}}} \mathrm{D}_{\boldsymbol{E}} W \cdot \delta_{\boldsymbol{d}}\boldsymbol{E} - \delta\boldsymbol{d} \cdot \bar{\boldsymbol{b}}\, \mathrm{d}\Omega_0^{\mathrm{s}} - \int\limits_{\Gamma_0^{\mathrm{s}}} \delta\boldsymbol{d} \cdot \bar{\boldsymbol{t}}\, \mathrm{d}\Gamma_0^{\mathrm{s}} = 0. \tag{3.60}$$

As described in many textbooks, see e.g. [9, 11, 183], the dynamic situation can likewise be found using the *principle of stationary action*[2]. It essentially states that a mechanical system will move from one time instance t_a to another one t_b such that the action

$$H = \int\limits_{t_a}^{t_b} L\, \mathrm{d}t = \int\limits_{t_a}^{t_b} \int\limits_{\Omega_0^{\mathrm{s}}} \frac{1}{2}\rho\dot{\boldsymbol{d}}^2 - W + \boldsymbol{d} \cdot \bar{\boldsymbol{b}}\, \mathrm{d}\Omega_0^{\mathrm{s}} + \int\limits_{\Gamma_0^{\mathrm{s}}} \boldsymbol{d} \cdot \bar{\boldsymbol{t}}\, \mathrm{d}\Gamma_0^{\mathrm{s}}\, \mathrm{d}t \tag{3.61}$$

is minimal, i.e. $\delta_{\boldsymbol{d}}H = 0$. In (3.61), the Lagrangian L is equal to the difference between kinetic and potential energy and if $\delta_{\boldsymbol{d}}H = 0$, it fulfills the *Euler-Lagrange equation*

$$\frac{\partial L}{\partial \boldsymbol{d}} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{d}}} = \boldsymbol{0}, \tag{3.62}$$

which leads back to the PDE (3.46).

Nearly incompressible hyperelasticity

In view of the finite element method (FEM) to be applied for the numerical discretization of the structural mechanics subproblem, it is preferable to stick to the displacement $\boldsymbol{d}$ as only state variable. Such a *pure displacement formulation* limits the choice of suitable SEDFs to those which describe a material behavior that is at least slightly compressible. Material models intended for describing truly incompressible materials can still be used by applying the so called *isochoric-volumetric split* of the SEDF into an isochoric, distortional part W^{dis} and a volumetric part W^{vol} as proposed in [74]. Introducing the isochoric deformation gradient

$$\overline{\boldsymbol{F}} = J^{-\frac{1}{3}}\boldsymbol{F}, \quad \text{with} \quad \det(\overline{\boldsymbol{F}}) = 1, \tag{3.63}$$

and the volumetric deformation gradient

$$\overset{\circ}{\boldsymbol{F}} = J^{\frac{1}{3}}\boldsymbol{1}, \quad \text{with} \quad \det(\overline{\boldsymbol{F}}) = J, \tag{3.64}$$

[2] Note that the principle of stationary action is also known as *Hamilton's principle*, however, as reported in [9, p. 19], only outside of England, especially in Germany and France.

the full deformation gradient is split in a multiplicative way as

$$\boldsymbol{F} = \overline{\boldsymbol{F}}\ \overset{\circ}{\boldsymbol{F}}\,. \tag{3.65}$$

The SEDF is split additively such that

$$W = W^{\mathrm{dis}}(\overline{\boldsymbol{F}}) + W^{\mathrm{vol}}(J). \tag{3.66}$$

Consequently, the second Piola-Kirchhoff stress tensor is also split into two parts and reads

$$\boldsymbol{S} = 2\,\frac{\partial W^{\mathrm{dis}}}{\partial \boldsymbol{C}} + 2\,\frac{\partial W^{\mathrm{vol}}}{\partial \boldsymbol{C}} = \boldsymbol{S}^{\mathrm{dis}} + \boldsymbol{S}^{\mathrm{vol}}. \tag{3.67}$$

In accordance to (3.63), isochoric variants of the right Cauchy-Green deformation tensor and the Green-Lagrange strain tensor can be defined:

$$\overline{\boldsymbol{C}} = \overline{\boldsymbol{F}}^{\mathrm{T}}\overline{\boldsymbol{F}} = J^{-\frac{2}{3}}\boldsymbol{C}, \quad \overline{\boldsymbol{E}} = \frac{1}{2}\left(\overline{\boldsymbol{C}} - \boldsymbol{I}\right). \tag{3.68}$$

For the distortional part W^{dis}, it is common practice to substitute any $\boldsymbol{C}$ or $\boldsymbol{E}$ appearing in a SEDF W^{inc} meant to be used for incompressible material by their isochoric counterparts (see [92]). That is, when constructing a $W^{\mathrm{dis}}(\overline{\boldsymbol{E}})$ from a $W^{\mathrm{inc}}(\boldsymbol{E})$, the functional form for the second Piola-Kirchhoff stress

$$\boldsymbol{S}^{\mathrm{inc}} = 2\,\frac{\partial W^{\mathrm{inc}}}{\partial \boldsymbol{C}} - p\,\boldsymbol{C}^{-1} \tag{3.69}$$

can be reused in the compressible case, where

$$\boldsymbol{S} = \left(\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}}\right)^{\mathrm{T}} \overline{\boldsymbol{S}} + \frac{\partial J}{\partial \boldsymbol{C}}\frac{\partial W^{\mathrm{vol}}}{\partial J} \tag{3.70}$$

is used to compute $\boldsymbol{S}$. In (3.70),

$$\overline{\boldsymbol{S}} = 2\,\frac{\partial W^{\mathrm{dis}}}{\partial \overline{\boldsymbol{C}}} \tag{3.71}$$

has the same form as ${\partial W^{\mathrm{inc}}}/{\partial \boldsymbol{C}}$ and therefore does not need to be derived anew for the compressible case. The terms arising from the application of the chain rule are found to be

$$\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}} = J^{-\frac{2}{3}}\left(\mathcal{I} - \frac{1}{3}\boldsymbol{C} \otimes \boldsymbol{C}^{-1}\right) \tag{3.72}$$

and

$$\frac{\partial J}{\partial \boldsymbol{C}} = \frac{1}{2} J \boldsymbol{C}^{-1}. \tag{3.73}$$

For a comparison with linear elasticity and the linearization of the structural mechanics subproblem in the scope of the finite element method, it is necessary to compute the elasticity tensor

$$\mathcal{C} = \frac{\partial \boldsymbol{S}}{\partial \boldsymbol{E}} = 2\,\frac{\partial \boldsymbol{S}}{\partial \boldsymbol{C}}. \tag{3.74}$$

Like the SEDF and the stresses, it is split into a distortional and a volumetric part as $\mathcal{C} = \mathcal{C}^{\text{dis}} + \mathcal{C}^{\text{vol}}$. For the distortional part,

$$\begin{aligned} \mathcal{C}^{\text{dis}} = {} & \left(\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}}\right)^{\text{T}} \overline{\mathcal{C}} \left(\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}}\right) \\ & - \frac{2}{3}\left(\boldsymbol{S}^{\text{dis}} \otimes \boldsymbol{C}^{-1} + \boldsymbol{C}^{-1} \otimes \boldsymbol{S}^{\text{dis}}\right) \\ & + \frac{4}{3}\left(\overline{\boldsymbol{C}} \cdot \overline{\boldsymbol{S}}\right)\left(\boldsymbol{C}^{-1} \overline{\underline{\otimes}}\, \boldsymbol{C}^{-1} - \frac{1}{3}\boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1}\right) \end{aligned} \tag{3.75}$$

provides a useful relation, where

$$\overline{\mathcal{C}} = 2\,\frac{\partial \overline{\boldsymbol{S}}}{\partial \overline{\boldsymbol{C}}} = 4\,\frac{\partial^2 W^{\text{dis}}}{\partial \overline{\boldsymbol{C}}^2} \tag{3.76}$$

has the same functional form as the elasticity tensor provided for W^{inc}. The dyadic product defined by $\overline{\underline{\otimes}}$ is defined in Section A.1.2. A derivation of (3.75) can be found in [90, p. 82] or [102, p. 255]. Different forms of the fundamental relations given above, which are based on the invariants of $\boldsymbol{C}$ or principle stretches (the squared eigenvalues of $\boldsymbol{C}$) can be found in standard textbooks, e.g. [26, 230, 102].

To obtain physically meaningful results, the volumetric part of the SEDF must ensure that

$$\lim_{J\to\infty} W \to \infty \quad \text{and} \quad \lim_{J\to 0} W \to \infty. \tag{3.77}$$

For a detailed explanation of these and additional constraints that have to be fulfilled to ensure existence and uniqueness of a solution, see [92] or [59]. A commonly applied form of W^{vol} is

$$W_1^{\text{vol}} = \frac{\kappa}{2}\left(1 - J\right)^2. \tag{3.78}$$

However, it is not suitable for problems involving strong compression, as $W^{\mathrm{vol}} \to 1$ for $J \to 0$, which does not ensure (3.77). In [92], the choice

$$W_2^{\mathrm{vol}} = \frac{\kappa}{50} \left(J^5 + J^{-5} - 2 \right) \tag{3.79}$$

is proposed and found to be preferable over (3.78) as it is in line with (3.77). A third possibility proposed in [230] is

$$W_3^{\mathrm{vol}} = \frac{\kappa}{4} \left(J^2 - 1 \right) - \frac{\kappa}{2} \ln \left(J \right), \tag{3.80}$$

which also ensures (3.77). With (3.73), the volumetric parts of the second Piola-Kirchhoff stress are obtained as

$$\boldsymbol{S}_1^{\mathrm{vol}} = \kappa \left(J^2 - J \right) \boldsymbol{C}^{-1}, \tag{3.81}$$

$$\boldsymbol{S}_2^{\mathrm{vol}} = \frac{\kappa}{10} \left(J^5 - J^{-5} \right) \boldsymbol{C}^{-1}, \tag{3.82}$$

$$\boldsymbol{S}_3^{\mathrm{vol}} = \frac{\kappa}{2} \left(J^2 - 1 \right) \boldsymbol{C}^{-1} \tag{3.83}$$

and for the volumetric parts of the elasticity tensors,

$$\mathcal{C}_1^{\mathrm{vol}} = \kappa \left(2\, J^2 - J \right) \boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} - 2\, \kappa \left(J^2 - J \right) \boldsymbol{C}^{-1} \,\overline{\underline{\otimes}}\, \boldsymbol{C}^{-1}, \tag{3.84}$$

$$\mathcal{C}_2^{\mathrm{vol}} = \frac{\kappa}{2} \left(J^5 + J^{-5} \right) \boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} - \frac{\kappa}{5} \left(J^5 - J^{-5} \right) \boldsymbol{C}^{-1} \,\overline{\underline{\otimes}}\, \boldsymbol{C}^{-1}, \tag{3.85}$$

$$\mathcal{C}_3^{\mathrm{vol}} = \kappa\, J^2\, \boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} + \kappa \left(J^2 - 1 \right) \boldsymbol{C}^{-1} \,\overline{\underline{\otimes}}\, \boldsymbol{C}^{-1} \tag{3.86}$$

is obtained.

Well established constitutive equations

A straightforward choice for a constitutive equation in the context of structural mechanics with large displacements is given by the *St.-Venant-Kirchhoff (SVK) model.* It extends the three-dimensional generalization of Hooke's law

$$\boldsymbol{\sigma} = \mathcal{C}^{\mathrm{Hooke}}\, \boldsymbol{\varepsilon} \tag{3.87}$$

from linear elasticity to the geometrically nonlinear regime. While in (3.87) the engineering strain tensor

$$\boldsymbol{\varepsilon} = \operatorname{symg\,rad}(\boldsymbol{d}) \tag{3.88}$$

is related to the Cauchy stress according to the small strain theory, in the SVK model the Green-Lagrange strain $\boldsymbol{E}$ is related to the second Piola-Kirchhoff stress $\boldsymbol{S}$. With a constant elasticity tensor $\mathcal{C} \overset{\text{SVK}}{=} \mathcal{C}^{\text{Hooke}}$,

$$\boldsymbol{S} \overset{\text{SVK}}{=} \mathcal{C}^{\text{Hooke}} \boldsymbol{E} = \lambda \operatorname{tr}(\boldsymbol{E})\, \boldsymbol{I} + 2\mu\, \boldsymbol{E} \tag{3.89}$$

is used, which can be derived from the SEDF

$$W \overset{\text{SVK}}{=} \frac{\lambda}{2}(\operatorname{tr}(\boldsymbol{E}))^2 + \mu \tag{3.90}$$

$$mbsE \cdot \boldsymbol{E}. \tag{3.91}$$

The parameters μ and λ denote the *shear modulus* (or *first Lamé coefficient*) and the *second Lamé coefficient*, respectively. While the SVK model may be used to describe deformations with large rotations, it has a major drawback when considering the compression of a material. Since $W \overset{\text{SVK}}{\to} 0$ as $J \to 0$, (3.77) is violated.

Different formulations of $\mathcal{C}^{\text{Hooke}}$ exist, which are useful when comparing more complex material laws to linear elasticity.

$$\mathcal{C}^{\text{Hooke}} = \lambda\, \mathcal{T} + 2\, \mu\, \mathcal{I}, \tag{3.92}$$

corresponds to the formulation in (3.89).

$$\mathcal{C}^{\text{Hooke}} = \kappa\, \mathcal{T} + 2\, \mu\, \mathcal{D} = \left(\kappa - \frac{2}{3}\, \mu\right) \mathcal{T} + 2\, \mu\, \mathcal{I} \tag{3.93}$$

naturally includes the compression modulus κ and reveals the relation

$$\lambda = \kappa - \frac{2}{3}\, \mu \tag{3.94}$$

considering that

$$\mathcal{D} = \mathcal{I} - \frac{1}{3}\, \mathcal{T}. \tag{3.95}$$

The fourth order tensors $\mathcal{I}$ (the identity tensor), $\mathcal{T}$ (yielding the trace) and $\mathcal{D}$ (yielding the deviatoric part) are defined in Section A.1.3. The introduced parameters from linear elasticity can be related to the well-known Young's modulus E and the Poisson ratio ν by

$$\mu = \frac{E}{2\,(1+\nu)}, \quad \lambda = \frac{E\,\nu}{(1+\nu)\,(1-2\,\nu)}, \quad \kappa = \frac{E}{3\,(1-2\,\nu)}. \tag{3.96}$$

A family of material laws developed to describe the behavior of rubber-like materials is given by the so called *Neo-Hookean* models. In the incompressible case, they are defined by

$$W^{\text{inc}} \overset{\text{NH}}{=} \frac{\mu}{2}\left(\mathrm{I}_{\boldsymbol{C}} - 3\right), \tag{3.97}$$

where $\mathrm{I}_{\boldsymbol{C}} = \operatorname{tr}(\boldsymbol{C})$ denotes the first invariant of the right Cauchy-Green deformation tensor. In the compressible case, (3.97) must be augmented by suitable volumetric terms, which gives rise to different variants of compressible Neo-Hookean models. If not stated otherwise, the variant

$$W \overset{\text{NH}}{=} \frac{\mu}{2}\left(I_{\boldsymbol{C}} - 3\right) - \mu \ln(J) + \frac{\lambda}{2}\left(\ln(J)\right)^2 \tag{3.98}$$

given in [26] is used. Note that this model does not fit into the framework of the isochoric-volumetric split since the first term in (3.98) is not isochoric. The second Piola-Kirchhoff stress is obtained as

$$\boldsymbol{S} \overset{\text{NH}}{=} \mu\left(\boldsymbol{I} - \boldsymbol{C}^{-1}\right) + \lambda \ln(J)\, \boldsymbol{C}^{-1} \tag{3.99}$$

and the elasticity tensor reads

$$\mathcal{C} \overset{\text{NH}}{=} \lambda\, \boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} + 2\left(\mu - \lambda \ln(J)\right) \boldsymbol{C}^{-1} \,\overline{\underline{\otimes}}\, \boldsymbol{C}^{-1}, \tag{3.100}$$

see [26, p. 163]. By comparing the resulting Neo-Hookean elasticity tensor with the one from linear theory, i.e. evaluating $\mathcal{C}$ at $\boldsymbol{d} = \boldsymbol{0}$ and comparing the coefficients, it becomes obvious that parameters λ and μ indeed correspond to the Lamé coefficients from linear elasticity.

Another variant of the Neo-Hookean model that is compatible with the isochoric volumetric split is discussed in [230, p. 46]. It is based on

$$W^{\text{dis}} \overset{\text{NH2}}{=} \frac{\mu}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}} - 3\right) \tag{3.101}$$

in combination with W_3^{vol}. With

$$\overline{\boldsymbol{S}} \overset{\text{NH2}}{=} \mu\, \boldsymbol{1} \tag{3.102}$$

and (3.70) as well as $\boldsymbol{S}_3^{\text{vol}}$ from (3.83), the second Piola-Kirchhoff stress is obtained as

$$\boldsymbol{S} \overset{\text{NH2}}{=} \mu\, J^{\frac{2}{3}}\left(\boldsymbol{1} - \frac{1}{3}\mathrm{I}_{\boldsymbol{C}}\boldsymbol{C}^{-1}\right) + \frac{\kappa}{2}\left(J^2 - 1\right) \boldsymbol{C}^{-1}. \tag{3.103}$$

To arrive at (3.103), it was exploited that

$$\left(\mathcal{I} - \frac{1}{3}\,\boldsymbol{C}^{-1} \otimes \boldsymbol{C}\right)\overline{\boldsymbol{S}} = \overline{\boldsymbol{S}} - \frac{1}{3}\left(\overline{\boldsymbol{S}} \cdot \boldsymbol{C}\right)\boldsymbol{C}^{-1}, \tag{3.104}$$

see [102, p. 230]. The tangent modulus can be derived using (3.75) with $\overline{\mathcal{C}} \overset{\text{NH2}}{=} \mathbf{0} \otimes \mathbf{0}$, however, a direct calculation from (3.103) using (3.58) is less involved. As shown in Section A.2.1, it results in

$$\begin{aligned} \boldsymbol{\mathcal{C}} \overset{\text{NH2}}{=}\, & 2\,\mu\left(\frac{1}{3}\,J^{-\frac{2}{3}}\,\mathrm{I}_{\boldsymbol{C}}\left(\boldsymbol{C}^{-1}\,\overline{\underline{\otimes}}\,\boldsymbol{C}^{-1} + \frac{1}{3}\,\boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1}\right) - \frac{2}{3}\,J^{-\frac{2}{3}}\,\boldsymbol{1} \otimes \boldsymbol{C}^{-1}\right) \\ & + \kappa\,J^2\,\boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} + \kappa\left(J^2 - 1\right)\boldsymbol{C}^{-1}\,\overline{\underline{\otimes}}\,\boldsymbol{C}^{-1}. \end{aligned} \tag{3.105}$$

The relation to the linear elasticity tensor can be investigated by an evaluation of $\mathcal{C}$ for $\boldsymbol{d} = \mathbf{0}$. With $\mathrm{I}_{\boldsymbol{C}}\big|_{\boldsymbol{d}=\mathbf{0}} = 3$, $\boldsymbol{C} = \boldsymbol{1}$ and $J\big|_{\boldsymbol{d}=\mathbf{0}} = 1$,

$$\begin{aligned} \boldsymbol{\mathcal{C}}^{\text{NH2}}\Big|_{\boldsymbol{d}=0} &= 2\,\mu\,\boldsymbol{1}\,\overline{\underline{\otimes}}\,\boldsymbol{1} - \frac{4}{3}\,\mu\,\boldsymbol{1} \otimes \boldsymbol{1} + \frac{2}{3}\,\mu\,\boldsymbol{1} \otimes \boldsymbol{1} + \kappa\,\boldsymbol{1} \otimes \boldsymbol{1} \\ &= \left(\kappa - \frac{2}{3}\,\mu\right)\mathcal{T} + 2\,\mu\,\mathcal{S} \end{aligned} \tag{3.106}$$

is obtained. By comparing (3.106) with (3.93) one finds that κ and μ indeed correspond to the parameters known from linear elasticity. For this comparison with linear elasticity, and for others in the following, it is necessary to exploit the fact that $\mathcal{I}$ and $\mathcal{S}$ can be used interchangeably due to the symmetries of $\boldsymbol{E}$ and $\boldsymbol{S}$.

3.2.3 Fluid mechanics

The fluid mechanics subproblem in the context of FSI is described using the ALE formulation of the conservation of mass and linear momentum. In the present work, as well as in all cited references on cardiovascular FSI, the fluid is assumed to have a (spatially and temporally) constant density, which implies incompressibility (not vice versa). The resulting equations are commonly known as the *incompressible Navier-Stokes equations.*

To start with, the density ρ is substituted for the scalar quantity ω in the general conservation law (3.39) and the linear momentum density $(\rho\,\boldsymbol{v})$ is substituted for the vector quantity $\boldsymbol{\omega}$ in the general conservation law (3.42).

This yields

$$\int\limits_{\Omega_t^{\mathrm{f}}} \left.\frac{\rho}{\partial t}\right|_{\chi} + (\boldsymbol{v}-\hat{\boldsymbol{v}})\cdot \mathrm{grad}(\rho) + \rho\,\mathrm{div}(\boldsymbol{v})\ \mathrm{d}\Omega_t^{\mathrm{f}} = 0, \tag{3.107}$$

$$\int\limits_{\Omega_t^{\mathrm{f}}} \left.\frac{\partial\,(\rho\,\boldsymbol{v})}{\partial t}\right|_{\chi} + (\boldsymbol{v}-\hat{\boldsymbol{v}})\cdot \mathrm{grad}(\rho\,\boldsymbol{v}) + \rho\,\boldsymbol{v}\,\mathrm{div}(\boldsymbol{v}) - \mathrm{div}(\boldsymbol{\sigma}) - \rho\,\boldsymbol{b}\,\Omega_t^{\mathrm{f}} = \boldsymbol{0}. \tag{3.108}$$

As in the structural mechanics case, the flux and source were set to $\boldsymbol{\gamma}^{\boldsymbol{x}} = \mathbf{0}$ and $\beta = 0$ to arrive at (3.107) and, respectively, to $\boldsymbol{\Gamma}^{\boldsymbol{x}} = \boldsymbol{\sigma}$ and $\boldsymbol{\beta} = \boldsymbol{b}$ to arrive at (3.108). By assuming a constant density and localizing Eq. (3.107) and (3.108) in space, one obtains the ALE form of the incompressible Navier-Stokes equations. Augmented by standard boundary conditions, they give rise to the IBVP that represents the fluid mechanics subproblem:

$$\mathrm{div}(\boldsymbol{v}) = 0 \qquad \text{in } \Omega_t^{\mathrm{f}}, \tag{3.109}$$

$$\dot{\boldsymbol{v}} + (\boldsymbol{v}-\hat{\boldsymbol{v}})\cdot\mathrm{grad}(\boldsymbol{v}) - \frac{1}{\rho}\mathrm{div}(\boldsymbol{\sigma}) - \boldsymbol{b} = \boldsymbol{0} \qquad \text{in } \Omega_t^{\mathrm{f}}, \tag{3.110}$$

$$\mathrm{grad}(p)\cdot\boldsymbol{n} = 0 \quad \text{and} \quad \boldsymbol{v} = \bar{\boldsymbol{v}} \qquad \text{on } \Gamma_t^{\boldsymbol{v}}, \tag{3.111}$$

$$\mathrm{grad}(\boldsymbol{v})\ \boldsymbol{n} = \mathbf{0} \quad \text{and} \quad p = \bar{p} \qquad \text{on } \Gamma_t^{p} \tag{3.112}$$

$$\boldsymbol{v} = \boldsymbol{v}_0 \quad \text{and} \quad p = p_0 \qquad \text{in } \Omega_t^{\mathrm{f}}, t = 0 \tag{3.113}$$

The pressure enters the momentum conservation equation (3.110) through the definition of the Cauchy stress tensor

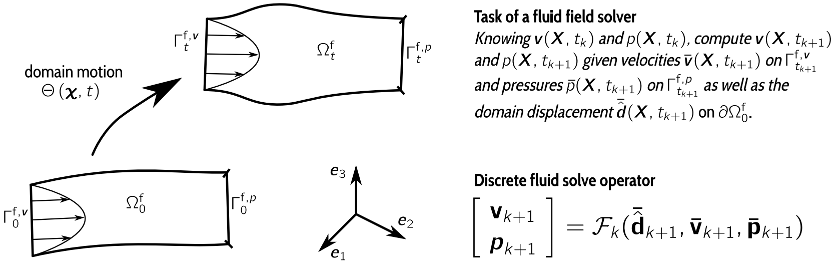
$$\boldsymbol{\sigma} = \boldsymbol{\tau} - p\,\mathbf{1}, \tag{3.114}$$

where $\boldsymbol{\tau}$ denotes the shear stress tensor. It is related to the fluid motion by a constitutive equation, which in the fluid mechanics case is usually based on the concept of viscosity. Constitutive equations are then given in the form

$$\boldsymbol{\tau} = 2\,\eta\,\dot{\boldsymbol{\gamma}} \tag{3.115}$$

where

$$\dot{\boldsymbol{\gamma}} = \frac{1}{2}\left(\mathrm{grad}(\boldsymbol{v}) + (\mathrm{grad}(\boldsymbol{v}))^{\mathrm{T}}\right) = \mathrm{sym}\,\mathrm{grad}\,(\boldsymbol{v}) \tag{3.116}$$



Task of a fluid field solver

Knowing $\mathbf{v}(\boldsymbol{X}, t_k)$ and $p(\boldsymbol{X}, t_k)$, compute $\mathbf{v}(\boldsymbol{X}, t_{k+1})$ and $p(\boldsymbol{X}, t_{k+1})$ given velocities $\bar{\mathbf{v}}(\boldsymbol{X}, t_{k+1})$ on $\Gamma^{f,v}_{t_{k+1}}$ and pressures $\bar{p}(\boldsymbol{X}, t_{k+1})$ on $\Gamma^{f,p}_{t_{k+1}}$ as well as the domain displacement $\bar{\hat{d}}(\boldsymbol{X}, t_{k+1})$ on $\partial\Omega_0^f$.

Discrete fluid solve operator

$$\begin{bmatrix} \mathbf{v}_{k+1} \\ \boldsymbol{p}_{k+1} \end{bmatrix} = \mathcal{F}_k(\bar{\hat{\mathbf{d}}}_{k+1}, \bar{\mathbf{v}}_{k+1}, \bar{\mathbf{p}}_{k+1})$$

Figure 3.3: Fluid dynamic kinematics, core quantities, and role of a fluid field solver in a partitioned simulation.

denotes the shear rate tensor and η denotes the viscosity. *Newtonian fluids* are characterized by a constant viscosity, while *non-Newtonian fluids* may be described using constitutive equations of different forms than that of (3.115). So-called *generalized Newtonian fluids* are described using (3.115), but allowing for a shear rate dependent viscosity. To this end, the viscosity is considered to be a function of the norm of the shear rate $\dot{\gamma}$, i.e.

$$\eta = \eta(\dot{\gamma}) = \eta(\|\dot{\boldsymbol{\gamma}}\|). \tag{3.117}$$

Particular forms of (3.117) used to describe blood are discussed in Section 3.3.

In the context of FSI simulations, the fluid solver is represented using the operator formulation given in Fig. 3.3. Note that the domain displacement at the boundary (represented by $\bar{\hat{\boldsymbol{d}}}$ after the discretization) is considered one of the input quantities due to the fact that the auxiliary problem is usually solved within the fluid solver.

3.2.4 Interface constraints and domain motion

With the fluid and structural mechanics subproblems at hand, an FSI problem can be formulated by additional interface constraints. They must hold on the wet surface, which is denoted by Γ_t^{FSI}. On the fluid side, $\Gamma_t^{\mathrm{FSI}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$, i.e. a velocity boundary condition is applied. On the structure side, the wet surface is considered in the material configuration as $\Gamma_0^{\mathrm{FSI}} \in \Gamma_0^{\mathrm{s},\boldsymbol{t}}$, i.e. a traction boundary condition is applied. Further, the motion of Γ_t^{FSI} must follow the motion of the structure, which reflects a constraint for the domain motion $\boldsymbol{\Theta}(\boldsymbol{\chi}, t)$ or, likewise for the domain displacement, $\hat{\boldsymbol{d}} = \boldsymbol{x} - \boldsymbol{\chi}$. Recalling that $\Omega_0^{(\boldsymbol{X})} = \Omega^{(\boldsymbol{\chi})}$, Γ_0^{FSI} may also denote the wet surface in the reference configuration. The interface constraints can then be stated as:

$$\bar{\boldsymbol{t}} = -p\,\boldsymbol{n} - \boldsymbol{\tau}_{\mathrm{w}} \qquad \text{on } \Gamma_0^{\mathrm{FSI}} \tag{3.118}$$

$$\bar{\boldsymbol{v}} = \dot{\boldsymbol{d}} \qquad \text{on } \Gamma_t^{\mathrm{FSI}} \tag{3.119}$$

$$\hat{\boldsymbol{d}} = \boldsymbol{d} \qquad \text{on } \Gamma_0^{\mathrm{FSI}} \tag{3.120}$$

As mentioned in Section 3.2.1, the motion of the fluid domain, i.e. the displacement $\hat{\boldsymbol{d}}$ inside Ω_t^f, is computed by an auxiliary problem. The auxiliary problem has no physical interpretation, but should meet some requirements:

- The motion defined by $\hat{\boldsymbol{d}}$ must result in a valid mapping $\boldsymbol{\Theta}(\boldsymbol{\chi}, t)$, i.e. $\det\left(\mathrm{grad}_{\boldsymbol{v}}(\boldsymbol{\Theta})\right) > 0$ must hold.
- The solution of the auxiliary problem should be cheap, i.e. computationally inexpensive, considering the fact that no physical consistency is demanded.
- The domain displacement given on Γ_0^{FSI} should be smoothly extended into the inside of the domain such that the cells of the mesh retain a reasonable quality in terms of skewness, aspect ratio and size.

Different approaches are commonly applied to define an auxiliary problem that meets these requirements, some of which are introduced in the following. Instead of applying the equations governing the auxiliary problems directly using the domain displacement $\hat{\boldsymbol{d}}$, it makes sense to use the domain displacement with respect to a reference time t^{r}

$$\hat{d}^*(\boldsymbol{x}) = \hat{d}(\boldsymbol{x}, t) - \hat{d}(\boldsymbol{x}, t^{\mathrm{r}})\,. \tag{3.121}$$

This allows to update t^{r} (e.g. every nth time step) to reset the reference displacement once the total displacement becomes too large. In accordance, reference coordinates

$$\boldsymbol{x}^* = \boldsymbol{\chi} + \boldsymbol{d}^* \tag{3.122}$$

are introduced.

Laplacian smoothing

This approach is implemented in the software package *OpenFOAM* (see [221, 154]), which is applied throughout this work to solve fluid mechanics problems. The displacement is governed by a Laplacian equation and by suitable Dirichlet boundary conditions. The auxiliary problem then reads

$$\begin{aligned}
\operatorname{div}_{\boldsymbol{x}^*}\left(\operatorname{grad}_{\boldsymbol{x}^*}\left(\hat{\boldsymbol{d}}^*\right)\right) &= \boldsymbol{0} && \text{in } \Omega_0^{\mathrm{f}}, && (3.123)\\
\hat{\boldsymbol{d}}^* &= \bar{\hat{\boldsymbol{d}}}^* && \text{on } \Gamma_0^{\mathrm{FSI}}, && (3.124)\\
\hat{\boldsymbol{d}}^* &= \boldsymbol{0} && \text{on } \partial\Omega_0^{\mathrm{f}} \setminus \Gamma_0^{\mathrm{FSI}}. && (3.125)
\end{aligned}$$

While the Laplace equation ensures smoothness, it has to be discretized. In *OpenFOAM*, this is done using the finite volume method. As explained in Section 4.1.2, the discretization procedure may yield displacements $\hat{\boldsymbol{d}}$ that lack the smoothness property and may even yield $\det\left(\operatorname{grad}_{\boldsymbol{v}}(\boldsymbol{\Theta})\right) \leq 0$.

Linear elasticity

In this approach, the fluid domain motion is computed as if Ω_t^{f} was representing an elastic solid. The auxiliary problem then reads

$$\begin{aligned}
\operatorname{div}_{\boldsymbol{x}^*}(\boldsymbol{\sigma}^*) &= \boldsymbol{0} && \text{in } \Omega_t^{\mathrm{f}}, && (3.126)\\
\hat{\boldsymbol{d}}^* &= \bar{\hat{\boldsymbol{d}}}^* && \text{on } \Gamma_t^{\mathrm{FSI}}, && (3.127)\\
\hat{\boldsymbol{d}}^* &= \boldsymbol{0} && \text{on } \partial\Omega_t^{\mathrm{f}} \setminus \Gamma_t^{\mathrm{FSI}}. && (3.128)
\end{aligned}$$

For the associated Cauchy stress tensor,

$$\boldsymbol{\sigma}^* = \mathcal{C}^{\mathrm{Hooke}}\,\boldsymbol{\varepsilon}^* = \mathcal{C}^{\mathrm{Hooke}}\operatorname{symgrad}_{\boldsymbol{x}^*}\left(\hat{\boldsymbol{d}}^*\right) \tag{3.129}$$

is used according to the theory of linear elasticity. In the experimental version of *OpenFOAM* (*FoamExtend*, see [85]) the solution of this auxiliary problem is realized drawing on the finite element method as explained in Section 4.1.2. While this results in better mesh qualities, it is computationally more expensive than using Laplacian smoothing.

Radial basis functions

A promising method for the computation of the domain displacement is to construct a continuous function $\hat{\boldsymbol{d}}^*(\boldsymbol{x}, t)$ that interpolates given discrete values at the boundary collected in $\bar{\hat{\mathbf{d}}}^*$. In general, this will be computationally less expensive than solving a boundary value problem as done in the above approaches. Among other interpolation methods, radial basis functions (RBFs) may be used, see [46, 47, 22]. As this approach is closely related to the computational meshes introduced along with the numerical methods in the next chapter, an explanation is postponed to Section 4.3.5, where different interpolation methods are discussed.

3.3 Mechanical models for the cardiovascular system

So far, the framework for fluid and structural dynamics was described without a connection to a specific application. The focus of this section lies on special formulations that are necessary for the description of cardiovascular FSI. To start with, several material laws for the arterial wall and viscosity models for blood are introduced. Afterwards, the reduced order models used to describe the surrounding tissue and vessel network are explained. Special boundary conditions are needed to couple the three-dimensional FSI problem with the reduced order models. The chapter closes with an explanation of these boundary conditions

3.3.1 Constitutive equations for soft tissue

This section addresses different constitutive equations commonly applied to model arteries and soft tissue in general. Modified versions of the models are proposed, which are applicable in the context of finite element discretization using a pure displacement formulation as mentioned in Section 3.2.2. This assumption of slight compressibility is well justified and has been observed experimentally in [234]. In general, the modeling of soft tissue poses a big challenge due to the complex material behavior that results from the heterogeneous microstructure.

Holzapfel-Gasser-Ogden models

A group of material models often applied to describe the behavior of soft biological tissue is based on the SEDF introduced in [101]. Based on the authors' names, they are referred to as Holzapfel-Gasser-Ogden (HGO) models. The HGO models fall in the class of transversely isotropic materials with two preferred fiber directions. Accordingly, the SEDFs are composed of two different parts as

$$W \overset{\text{HGO}}{=} W^{\text{iso}} + W^{\text{fibre}} \tag{3.130}$$

where W^{iso} denotes the isotropic part that corresponds to the strain energy stored in the matrix material, i.e. the material in which the fibers are embedded. The anisotropic part of the strain energy stored in the fibers is denoted by W^{fibre}, which depends only on strains that occur in the respective preferred fiber directions. While the original HGO model in [101] was indented to be used in the incompressible limit case, different studies have introduced modified versions that extend the model to the compressible regime, see [82, 152, 236]. They are established by applying a possibly incomplete isochoric-volumetric split as explained in the following.

Given the incompressible variant of the HGO model, the question arises where to replace the full deformation tensors ($\boldsymbol{F}$,$\boldsymbol{C}$,$\boldsymbol{E}$) by their isochoric counterparts ($\overline{\boldsymbol{F}}$,$\overline{\boldsymbol{C}}$,$\overline{\boldsymbol{E}}$). If the replacement is done only when establishing W^{iso} and if W^{fibre} is left unchanged, the isochoric-volumetric split is incomplete. In the compressible case, the resulting W^{fibre} is nonzero – not only for distortional but also for purely volumetric deformations. This incomplete split is applied in [152, 236], following the argument that a purely volumetric deformation will actually lead to stresses in the fibers, i.e. an anisotropic stress state. Vice versa, a volumetric stress state will result in anisotropic deformations, e.g. a sphere under hydrostatic loading will deform into an ellipsoid, which conforms to the expected behavior of an anisotropic material. The isotropic part of the SEDF then reads

$$W^{\text{iso}} = W^{\text{dis}} + W^{\text{vol}} \overset{\text{HGO}}{=} \frac{\mu^{\text{HGO}}}{2} \left(\mathrm{I}_{\overline{\boldsymbol{C}}} - 3\right) + \frac{\kappa}{2} \left(1 - J\right)^2 \tag{3.131}$$

where W^{dis} can be identified to be a Neo-Hookean SEDF. For the volumetric part, $W^{\text{vol}} = W_1^{\text{vol}}$ (cf. (3.78)) is chosen despite the known issues for strong compression associated with it. The incomplete split yields an unmodified anisotropic part

$$W^{\text{fibre}} = \frac{k_1}{k_2} \left(e^{k_2 (\mathrm{IV}_{\boldsymbol{C}} - 1)^2} + e^{k_2 (\mathrm{VI}_{\boldsymbol{C}} - 1)^2} \right), \tag{3.132}$$

where the invariants

$$\mathrm{IV}_{\boldsymbol{C}} = \boldsymbol{a}_1 \cdot \boldsymbol{C} \boldsymbol{a}_1, \tag{3.133}$$

$$\mathrm{VI}_{\boldsymbol{C}} = \boldsymbol{a}_2 \cdot \boldsymbol{C} \boldsymbol{a}_2 \tag{3.134}$$

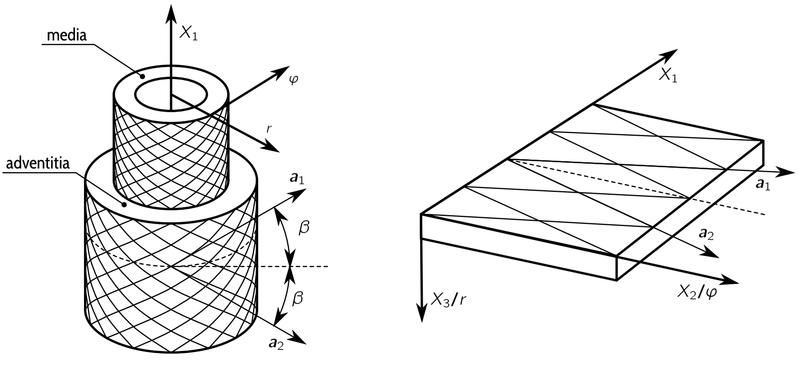


Figure 3.4: Fiber orientation in the HGO material model considering a circular artery with two layers (left) and considering a cut-out of one layer while neglecting the curvature (right).

correspond to the first invariant of $\mathrm{I}_{\boldsymbol{C}}$ in the fiber directions, which are defined by the unit vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$[3]. For arteries, the directions are given by the alignment of the collagen fibers in the media and adventitia. As illustrated in Fig. 3.4, two helically arranged families of fibers can be identified for each layer. If aligned to a Cartesian coordinate system as shown, the fiber directions are given by

$$\boldsymbol{a}_1 = \sin(\beta)\ \boldsymbol{E}_1 + \cos(\beta)\cos(\varphi)\ \boldsymbol{E}_2 + \cos(\beta)\sin(\varphi)\ \boldsymbol{E}_3, \tag{3.135}$$

and

$$\boldsymbol{a}_2 = -\sin(\beta)\ \boldsymbol{E}_1 + \cos(\beta)\cos(\varphi)\ \boldsymbol{E}_2 + \cos(\beta)\sin(\varphi)\ \boldsymbol{E}_3, \tag{3.136}$$

where φ denotes the circumferential angle and β denotes the angle between the fiber direction and the circumferential direction. When describing arteries, the model is intended to be used with individual sets of parameters for the media and the adventitia. As stated in Section 2.2.2, the determination of $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ is an unresolved issue for complex geometries. The second Piola-Kirchhoff stress is found to be

$$\boldsymbol{S} \overset{\mathrm{HGO}}{=} \boldsymbol{S}^{\mathrm{dis}} + \boldsymbol{S}^{\mathrm{fibers}} + \boldsymbol{S}_1^{\mathrm{vol}}, \tag{3.137}$$

[3] The anisotropic counterparts to $\mathrm{II}_{\boldsymbol{C}}$ are denoted by $\mathrm{V}_{\boldsymbol{C}}$ and $\mathrm{VII}_{\boldsymbol{C}}$, which explains the naming convention found in many articles on material models with two fiber directions.

with $\boldsymbol{S}^{\mathrm{dis}}$ as given in (3.103), $\boldsymbol{S}_1^{\mathrm{vol}}$ as given in (3.81) and

$$\boldsymbol{S}^{\mathrm{fibers}} = 2\,k_1 \Big((\mathrm{IV}_{\boldsymbol{C}} - 1)\,\boldsymbol{a}_1 \otimes \boldsymbol{a}_1\, e^{k_2 (\mathrm{IV}_{\boldsymbol{C}}-1)^2} + (\mathrm{VI}_{\boldsymbol{C}} - 1)\,\boldsymbol{a}_2 \otimes \boldsymbol{a}_2\, e^{k_2 (\mathrm{VI}_{\boldsymbol{C}}-1)^2} \Big)\,. \tag{3.138}$$

The tangent modulus can be constructed as

$$\mathcal{C} \overset{\mathrm{HGO}}{=} \mathcal{C}^{\mathrm{iso}} + \mathcal{C}^{\mathrm{fibers}} = \mathcal{C}^{\mathrm{dis}} + \mathcal{C}_1^{\mathrm{vol}} + \mathcal{C}^{\mathrm{fibers}}, \tag{3.139}$$

with $\mathcal{C}_1^{\mathrm{vol}}$ as given in (3.84) and $\mathcal{C}^{\mathrm{dis}}$ as given in (3.105). The fiber part is found to be

$$\begin{aligned} \mathcal{C}^{\mathrm{fibers}} = {} & 4\,k_1\, e^{k_2 (\mathrm{IV}_{\boldsymbol{C}}-1)^2} \Big(1 + 2\,k_2\,(\mathrm{IV}_{\boldsymbol{C}} - 1)^2\Big) (\boldsymbol{a}_1 \otimes \boldsymbol{a}_1) \otimes (\boldsymbol{a}_1 \otimes \boldsymbol{a}_1) \\ & + 4\,k_1\, e^{k_2 (\mathrm{VI}_{\boldsymbol{C}}-1)^2} \Big(1 + 2\,k_2\,(\mathrm{VI}_{\boldsymbol{C}} - 1)^2\Big) (\boldsymbol{a}_2 \otimes \boldsymbol{a}_2) \otimes (\boldsymbol{a}_2 \otimes \boldsymbol{a}_2)\,. \end{aligned} \tag{3.140}$$

Special care has to be taken when investigating the relation between $\mathcal{C}$ and the linear elasticity tensor $\mathcal{C}_{\mathrm{Hooke}}$ for the HGO model. Without the anisotropic part, κ and μ^{HGO} can be easily identified as the bulk modulus and the shear modulus. However, when taking the fiber contribution into account, a distinction has to be made regarding the stiffness in the fiber direction and the stiffness orthogonal to any fiber direction.

Fung model

The material model for arteries introduced in [80] accounts for the typical behavior in a more phenomenological way. Instead of incorporating the fiber directions explicitly into the SEDF, a general form

$$W^{\mathrm{inc}} \overset{\mathrm{Fung}}{=} \frac{c}{2}\left(e^{Q} - 1\right) \quad \text{with} \quad Q = \mathcal{A} \cdot \boldsymbol{E} \otimes \boldsymbol{E} \tag{3.141}$$

is proposed. The tensor of material parameters $\mathcal{A}$ can be constructed such that an anisotropic material behavior is described. An investigation of such an anisotropic variant of Fung's model can be found in [80, 101]. The isotropic variant used here was proposed in [10] and uses

$$Q = \frac{1}{c}\Big(\lambda\,\mathrm{tr}\,(\boldsymbol{E})^2 + 2\,\mu\,\mathrm{tr}\,\big(\boldsymbol{E}^2\big)\Big)\,. \tag{3.142}$$

As can be easily verified, λ and μ correspond to the Lamé constants from linear elasticity. However, this is not the case when using the model in combination with the isochoric volumetric split. The corresponding modified

SEDF then reads

$$W = W^{\mathrm{dis}} + W^{\mathrm{vol}} \overset{\mathrm{Fung}}{=} \frac{c}{2}\left(e^{\overline{Q}} - 1\right) + W^{\mathrm{vol}} \tag{3.143}$$

where

$$\overline{Q} = \frac{1}{c}\left(\lambda^{\mathrm{Fung}}\,\mathrm{tr}\left(\overline{\boldsymbol{E}}\right)^2 + 2\,\mu^{\mathrm{Fung}}\,\mathrm{tr}\left(\overline{\boldsymbol{E}}^2\right)\right). \tag{3.144}$$

The second Piola-Kirchhoff stresses are obtained using (3.70) and

$$\overline{\boldsymbol{S}} = \frac{\partial W^{\mathrm{dis}}}{\partial \overline{\boldsymbol{E}}} \overset{\mathrm{Fung}}{=} e^{\overline{Q}}\left(\lambda_{\mathrm{Fung}}\,\mathrm{tr}\left(\overline{\boldsymbol{E}}\right)\boldsymbol{1} + 2\,\mu_{\mathrm{Fung}}\,\overline{\boldsymbol{E}}\right). \tag{3.145}$$

The elasticity tensor is obtained using (3.75) and

$$\overline{\mathcal{C}} \overset{\mathrm{Fung}}{=} \frac{2}{c}\,e^{-\overline{Q}}\,\overline{\boldsymbol{S}} \otimes \overline{\boldsymbol{S}} + e^{\overline{Q}}\left(\lambda^{\mathrm{Fung}}\,\mathcal{T} + 2\,\mu^{\mathrm{Fung}}\,\mathcal{S}\right). \tag{3.146}$$

Equation (3.146) clearly reveals the relation to linear elasticity (see Eq. (3.92)) in the incompressible case because for $\boldsymbol{d} = \boldsymbol{0}$, $e^{-\overline{Q}} = 1$ and $\overline{\boldsymbol{S}} \overset{\mathrm{Fung}}{=} \boldsymbol{0}$. Comparing the elasticity tensor with that from linear elasticity, λ^{Fung} disappears and μ^{Fung} is identified as the shear modulus μ since

$$\begin{aligned} \mathcal{C}\big|_{\boldsymbol{d}=\boldsymbol{0}} &\overset{\mathrm{Fung}}{=} \left.\left(\left(\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}}\right)^{\mathrm{T}} \overline{\mathcal{C}}\,\frac{\partial \overline{\boldsymbol{C}}}{\partial \boldsymbol{C}}\right)\right|_{\boldsymbol{d}=\boldsymbol{0}} \\ &\overset{\mathrm{Fung}}{=} \left(\mathcal{I} - \frac{1}{3}\mathcal{T}\right)\left(\lambda\,\mathcal{T} + 2\,\mu\,\mathcal{S}\right)\left(\mathcal{I} - \frac{1}{3}\mathcal{T}\right) = 2\,\mu_{\mathrm{Fung}}\left(\mathcal{S} - \frac{1}{3}\mathcal{T}\right). \end{aligned} \tag{3.147}$$

Delfino model

In [57], a material model for arteries was proposed to describe the behavior of a carotid artery. The SEDF proposed for the incompressible limit case considered in [57] reads

$$W^{\mathrm{inc}} \overset{\mathrm{Delfino}}{=} \frac{a}{b}\left(e^{\frac{b}{2}\left(\mathrm{I}_{\boldsymbol{C}}-3\right)} - 1\right). \tag{3.148}$$

Performing the isochoric volumetric split, it is found that

$$W \overset{\mathrm{Delfino}}{=} \frac{a}{b}\left(e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)} - 1\right) + W^{\mathrm{vol}}. \tag{3.149}$$

The stresses are obtained using (3.70) and inserting

$$\overline{\boldsymbol{S}} \overset{\mathrm{Delfino}}{=} a\,e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)}\,\boldsymbol{1} \tag{3.150}$$

leads to

$$\boldsymbol{S}^{\text{dis}} \overset{\text{Delfino}}{=} J^{-\frac{2}{3}} \left(\mathbf{1} - \frac{1}{3}\operatorname{tr}\left(\boldsymbol{C}\right) \otimes \boldsymbol{C}^{-1}\right) a\, e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)}. \tag{3.151}$$

The elasticity tensor is obtained using (3.75) and inserting

$$\overline{\boldsymbol{C}} \overset{\text{Delfino}}{=} \frac{a\,b}{2} e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)}\, \mathcal{T}, \tag{3.152}$$

leads to

$$\begin{aligned} \mathcal{C}^{\text{dis}} \overset{\text{Delfino}}{=} {} & J^{-\frac{4}{3}} \left(\frac{\partial \overline{\boldsymbol{C}}}{\partial C}\right)^{\mathrm{T}} \left(\frac{a\,b}{2} e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)}\, \mathcal{T}\right) \frac{\partial \overline{\boldsymbol{C}}}{\partial C} \\ & - \frac{2}{3}\left(\boldsymbol{S}^{\text{iso}} \otimes \boldsymbol{C}^{-1} + \boldsymbol{C}^{-1} \otimes \boldsymbol{S}^{\text{iso}}\right) \qquad . \\ & + \frac{2}{3} J^{-\frac{4}{3}} \left(\overline{\boldsymbol{C}} \cdot \left(a\, e^{\frac{b}{2}\left(\mathrm{I}_{\overline{\boldsymbol{C}}}-3\right)}\, \mathbf{1}\right)\right) \left(\overline{\boldsymbol{C}}^{-1} \underline{\otimes}\, \overline{\boldsymbol{C}}^{-1} - \frac{1}{3} \overline{\boldsymbol{C}}^{-1} \otimes \overline{\boldsymbol{C}}^{-1}\right) \end{aligned} \tag{3.153}$$

For a comparison with the linear elasticity tensor, it can be easily seen that the term in the second row of (3.153) disappears for $\boldsymbol{d} = \mathbf{0}$, because $\boldsymbol{S}^{\text{iso}}\big|_{\boldsymbol{d}=\mathbf{0}} \overset{\text{Delfino}}{=} \mathbf{0}$, according to (3.151). Here - contrary to all other material models considered – the term in the third row does contribute to the tangent modulus, which is due to the fact that

$$\overline{\boldsymbol{S}}_{\text{Delfino}}\big|_{\boldsymbol{d}=\mathbf{0}} \neq \mathbf{0}. \tag{3.154}$$

This shows that in the incompressible case considered in [57], the material (i.e. the artery) is assumed to be prestressed – a characteristic of the model that is not preserved when performing the isochoric volumetric split, where $\boldsymbol{S}\big|_{\boldsymbol{d}=\mathbf{0}} = \mathbf{0}$ by construction. The issue of prestressing in the context of arteries is addressed in more detail in Section 5.1.1. With

$$\begin{aligned} \left(J^{-\frac{4}{3}} \left(\frac{\partial \overline{\boldsymbol{C}}}{\partial C}\right)^{\mathrm{T}} \overline{\boldsymbol{C}}_{\text{Delfino}} \frac{\partial \overline{\boldsymbol{C}}}{\partial C}\right)\Bigg|_{\boldsymbol{d}=\mathbf{0}} &= \left(\mathcal{I} - \frac{1}{3}\mathcal{T}\right)\left(a\,b\,\mathcal{T}\right)\left(\mathcal{I} - \frac{1}{3}\mathcal{T}\right) \\ &= \left(a\,b\,\mathcal{T} - a\,b\,\mathcal{T}\right)\left(\mathcal{I} - \frac{1}{3}\mathcal{T}\right) = \mathcal{O}, \end{aligned} \tag{3.155}$$

it becomes clear that the term in the third row of (3.153) is actually the only one that contributes to the elasticity tensor in the undeformed state.

This yields

$$
\begin{aligned}
\mathcal{C}^{\text{iso}}\Big|_{\boldsymbol{d}=\mathbf{0}} &= \left(\frac{2}{3}\,J^{-\frac{2}{3}}\left(\overline{\boldsymbol{C}}\cdot\overline{\boldsymbol{S}}_{\text{Delfino}}\right)\left(\overline{\boldsymbol{C}}^{-1}\,\underline{\otimes}\,\overline{\boldsymbol{C}}^{-1}-\frac{1}{3}\overline{\boldsymbol{C}}^{-1}\otimes\overline{\boldsymbol{C}}^{-1}\right)\right)\Bigg|_{\boldsymbol{d}=\mathbf{0}}\\
&= \frac{2}{3}\left(a\,\mathbf{1}\cdot\mathbf{1}\right)\left(\mathcal{S}-\frac{1}{3}\mathcal{T}\right)\\
&= 2\,a\left(\mathcal{S}-\frac{1}{3}\mathcal{T}\right)
\end{aligned}
\tag{3.156}
$$

and the parameter a can be identified as the shear modulus μ.

3.3.2 Constitutive equations for blood

Blood is commonly known to be a *non-Newtonian fluid* or – as explained in Section 3.2.3 – a generalized Newtonian fluid with a shear-rate-dependent viscosity $\eta\left(\dot{\gamma}\right)$. Specific forms of $\eta\left(\dot{\gamma}\right)$ allow for a description of an effect called *shear-thinning*, which is typical for blood due to the presence of red blood cells. With increasing shear rates, the viscosity decreases due to a change of the arrangement of the cells. Of course, according viscosity models describe this effect from the viewpoint of continuum mechanics, i.e. they are only applicable for sufficiently large scales that allow to describe the macroscopic behavior of blood in terms of continuous state variables and properties. On a small scale, the concept of viscosity collapses, meaning that a different modeling approach has to be chosen for simulations of blood flow in capillaries. A very accurate but at the same time computationally very expensive approach is to consider the red blood cells as individual particles as done in [149]. However, corresponding fluid-particle interaction simulations are out of the scope of this work. For the vessel sizes considered here (with a diameter of at least a few millimeters), the continuum mechanics approach is well justified. In fact, non-Newtonian effects are seldom incorporated in comparable cardiovascular FSI simulations since the shear rates are assumed to be in a range were these effects are negligible. Thanks to the possibility of employing well-established viscosity models and reuse tested implementations from the software package *OpenFOAM* ([221, 154]) due to the partitioned solution approach, four commonly applied viscosity models for blood are used to investigate this assumption here. The parameters for all models are collected in Tab. 3.2. In combination with

Table 3.2: Parameters of the viscosity models experimentally obtained for blood.

Model	η_0 (mPa s)	η_∞ (mPa s)	k (Pa s^{n-1})	n (-)	τ_0 (mPa)	Reference
Newtonian	–	4.08	–	–	–	[114]
Newtonian	–	3.5	–	–	–	[166]
Newtonian	–	3.19	–	–	–	[118]
Power	187	3.5	12.171	0.799	–	[118]
Cross	103	5.25	1.15	1.25	–	[1]
Bird-Carreau	56	3.5	11.0	0.357	–	[181, 166]
Herschel-Bulkley	187	–	8.97	0.860	17.5	[212, 118]

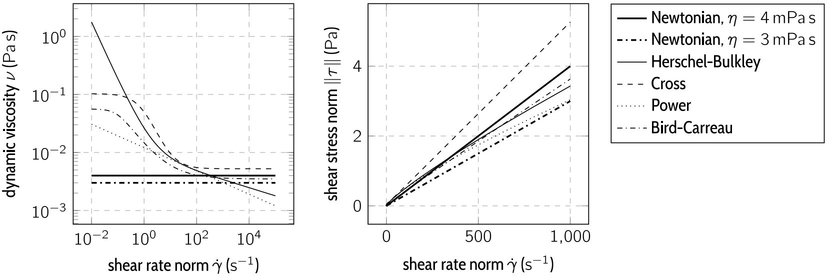


Figure 3.5: Viscosity (left) and norm of the shear stress (right) as a function of the strain rate for the viscosity models under consideration.

Newtonian models, $\nu = \nu_\infty$ is used. In Fig. 3.5, the viscosity and shear stresses are plotted for a selected range of shear rates.

Cross model

This model, proposed by Cross, 1965 [44] uses

$$\eta \overset{\text{Cross}}{=} \eta_\infty + \frac{\eta_0 - \eta_\infty}{1 + (k\dot{\gamma})^n}, \tag{3.157}$$

which results in $\eta = \eta_0$ at $\dot{\gamma} = 0$. With increasing shear rates, the viscosity approaches η_∞, as long as $n > 0$. The model was used in [1] as a viscosity model for simulations of blood flow in a stenosed artery.

Bird-Carreau model

The Bird-Carreau (BC) model goes back to [36] and is used in [181] and [166] to describe the rheology of blood. The relation

$$\eta \overset{\mathrm{BC}}{=} \eta_\infty + (\eta_0 - \eta_\infty)\left(1 + (k\dot{\gamma})^2\right)^{\frac{n-1}{2}} \tag{3.158}$$

results in $\eta = \eta_0$ at $\dot{\gamma} = 0$. With increasing shear rates, the viscosity approaches η_∞, as long as $n < 1$.

Power law

The power law is a comparably simple viscosity model with

$$\eta^{\mathrm{P}} = k\,\dot{\gamma}^{n-1}. \tag{3.159}$$

While (3.159) may be used to describe the shear-thinning effect, the parameters k and n do not provide the necessary flexibility. To this end,

$$\eta \overset{\mathrm{Power}}{=} \begin{cases} \eta_0 & \text{if } \eta^{\mathrm{P}} < \eta_0, \\ \eta_\infty & \text{if } \eta^{\mathrm{P}} > \eta_\infty, \\ \eta^{\mathrm{P}} & \text{else} \end{cases} \tag{3.160}$$

is used with the parameters as given in [118].

Herschel-Bulkley model

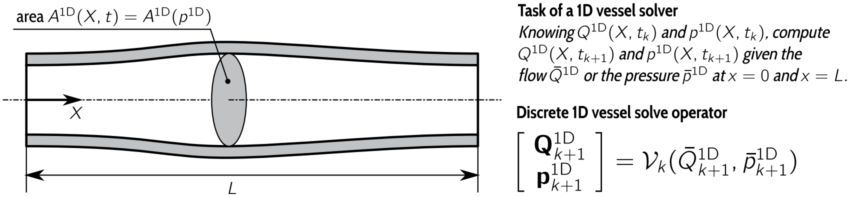
The Herschel-Bulkley (HB) model goes back to [99] and defines the viscosity as

$$\eta \overset{\mathrm{HB}}{=} k\,\dot{\gamma}^{n-1} + \frac{\tau_0}{\dot{\gamma}}. \tag{3.161}$$

In addition to the shear-thinning effect, a yield stress τ_0 is included in the model. In scope of the discretization of the fluid mechanics subproblem, (3.161) cannot be used directly. Instead of prescribing the yield stress τ_0, a penalty viscosity ν_0 is defined. An explanation of this issue is given in [161].

3.3.3 One-dimensional models

The derivation of one-dimensional models for viscous flow in flexible vessels starts with the assumption that its mechanical behavior can be described



Task of a 1D vessel solver

Knowing $Q^{1D}(X, t_k)$ and $p^{1D}(X, t_k)$, compute $Q^{1D}(X, t_{k+1})$ and $p^{1D}(X, t_{k+1})$ given the flow $\bar{Q}^{1D}$ or the pressure $\bar{p}^{1D}$ at $x = 0$ and $x = L$.

Discrete 1D vessel solve operator

$$\begin{bmatrix} \mathbf{Q}^{1D}_{k+1} \\ \mathbf{p}^{1D}_{k+1} \end{bmatrix} = \mathcal{V}_k(\bar{Q}^{1D}_{k+1}, \bar{p}^{1D}_{k+1})$$

Figure 3.6: One-dimensional vessel model and role of a 1D vessel solver in a partitioned simulation.

by average values of the independent mechanical quantities. Considering a single vessel, there is only one spatial coordinate x corresponding to the axial direction. The vessel's geometry is assumed to be circular with a varying cross-sectional area $A^{1D}(x,t)$. At each cross section Γ^{1D}, the behavior of the blood and the vessel wall is described by the flux

$$Q^{1D}(x,t) = A^{1D}\, v^{1D} = \int\limits_{\Gamma^{1D}} v \,\mathrm{d}\Gamma^{1D}, \tag{3.162}$$

and the average pressure

$$p^{1D}(x,t) = \frac{1}{A} \int\limits_{\Gamma^{1D}} p \,\mathrm{d}\Gamma^{1D}. \tag{3.163}$$

In (3.162) v^{1D} denotes the average velocity and v denotes the local velocity in the direction normal to the cross section Γ^{1D}.

Having reduced the structural deformation to a single quantity $A^{1D}(x,t)$, the fluid-structure interaction problem can be stated in a monolithic manner by the one-dimensional Navier-Stokes equations

$$\frac{\partial A^{1D}}{\partial t} + \frac{\partial Q^{1D}}{\partial x} = 0 \tag{3.164}$$

$$\frac{\partial Q^{1D}}{\partial t} + \frac{\partial}{\partial x}\left(\frac{\alpha\, {Q^{1D}}^2}{A^{1D}}\right) + \frac{A^{1D}}{\rho}\,\frac{\partial p^{1D}}{\partial x} + K_R\,\frac{Q^{1D}}{A^{1D}} = 0 \tag{3.165}$$

which as in the three-dimensional case describe the conservation of mass and linear momentum. They can be derived from their three-dimensional counterparts (3.107) and (3.108) by inserting the definitions of the average values or by considering the general conservation laws (3.37) for a one-dimensional control volume with varying cross-sectional area $A^{1D}(x,t)$. A detailed derivation can be found in [76, 179, 160]. In either case, an assumption has to be made on the velocity profile, which affects the final one-dimensional momentum equation (3.165). The parameters α (momentum correction coefficient or Coriolis coefficient) and K_R are introduced to account for this influence. The integration of the quadratic term in the three-dimensional momentum equations to obtain a one-dimensional averaged formulation can then be stated as

$$\int\limits_{\Gamma^{1\mathrm{D}}} v^2 \,\mathrm{d}\Gamma^{1\mathrm{D}} = \alpha \int\limits_{\Gamma^{1\mathrm{D}}} v^{1\mathrm{D}^2} \,\mathrm{d}\Gamma^{1\mathrm{D}} = \alpha\, A^{1\mathrm{D}}\, v^{1\mathrm{D}^2} = \frac{\alpha\, Q^{1\mathrm{D}^2}}{A^{1\mathrm{D}}}. \tag{3.166}$$

For a flat velocity profile, $\alpha = 1$ follows, which is assumed in most studies, see [76]. Similarly, averaging the viscous term in the three-dimensional momentum equation depends on the choice of a velocity profile. Of course, a flat velocity profile will result in zero viscous stresses, i.e. $K_R = 0$. Therefore, in most studies (see e.g. [179, 219]), a quadratic (Poiseuille) velocity profile is assumed for this term – leading to $K_R = 8\,\pi\,\nu$.

A third relation is required to solve an initial boundary value problem for the three unknowns $A^{1\mathrm{D}}$, $Q^{1\mathrm{D}}$, and $p^{1\mathrm{D}}$. An intuitive choice here is to relate the cross-sectional area $A^{1\mathrm{D}}$ and the pressure $p^{1\mathrm{D}}$ by an equation describing the mechanics of the vessel wall. Starting out with a general string model (see [76]) and neglecting inertia and damping terms as well as any spatial displacement gradients, one arrives at

$$p^{1\mathrm{D}} = p^{1\mathrm{D}}_{\mathrm{ext}} + \beta \left(\sqrt{A^{1\mathrm{D}}} - \sqrt{A^{1\mathrm{D}}_0} \right) \quad \text{with} \quad \beta = \frac{\sqrt{\pi}\, h_0\, E}{(1-\nu^2)\, A^{1\mathrm{D}}_0} \tag{3.167}$$

where E, ν, h_0, and $A^{1\mathrm{D}}_0$ denote the vessel's Young's modulus, Poisson ratio, initial wall thickness, and initial cross-sectional area, respectively. The pressure at the outside of the vessel is denoted by $p^{1\mathrm{D}}_{\mathrm{ext}}$. Inserting relation (3.167) into the one-dimensional Navier-Stokes equations (3.164) and (3.165), the problem can be written in standard (conservative) form:

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \frac{\partial \boldsymbol{G}}{\partial x} = \boldsymbol{B} \tag{3.168}$$

with

$$\boldsymbol{Q} = \begin{bmatrix} A^{1\mathrm{D}} \\ Q^{1\mathrm{D}} \end{bmatrix}, \; \boldsymbol{G} = \begin{bmatrix} Q^{1\mathrm{D}} \\ \frac{Q^{1\mathrm{D}^2}}{A^{1\mathrm{D}}} + \frac{\beta}{3\,\rho} A^{1\mathrm{D}^{\frac{3}{2}}} \end{bmatrix}, \; \boldsymbol{B} = \begin{bmatrix} 0 \\ -K_R \frac{Q^{1\mathrm{D}}}{A^{1\mathrm{D}}} \end{bmatrix}. \tag{3.169}$$

As explained in [76], choosing A^{1D} and Q^{1D} as independent state variables is not the only choice, but choosing A^{1D} and v^{1D} is possible as well.

The conservative form gives rise to a characteristic analysis of the problem, which serves as a basis for non-reflecting or partially reflecting boundary conditions at bifurcations and an investigation of wave propagation, as explained in [76, 179]. Here, only a single vessel is considered and standard boundary conditions are used. The area A is assumed to be constant at both ends and, therefore, does not appear in the operator formulation given in Fig. 3.6. Note that the task of the 1D vessel solver allows for a coupling of other models at both ends. Usually, the flow is prescribed at the proximal end ($x = 0$) and the pressure is prescribed at the distal end ($x = L$).

3.3.4 Windkessel models

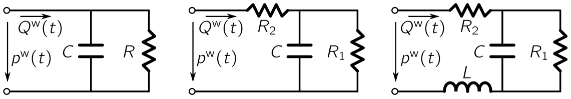
An even more reduced model of the fluid-structure interaction in blood vessels describes the relation of flow and pressure in terms of an analogy to electrical circuits. Within such a model, the flow Q and the pressure p in a vessel correspond to the current and the voltage, respectively. Windkessel models can be classified as zero-dimensional models, as they provide no spatial resolution of their state variables. They are commonly used in cardiovascular FSI simulations to obtain a physiological pressure pulse at the distal end (outlet) of the considered vessel segment. To this end, an implicit boundary condition that computes the pressure in terms of the flow through the respective boundary is implemented in the fluid solver. Alternatively, the windkessel model can be considered as an additional field of the coupled FSI problem.

Considering the types of windkessel models depicted in Fig. 3.7, three types of components are used in this sense.

1. A resistor describes the dissipative viscous effects.
2. A capacitor accounts for the compliance of the vessel wall.
3. An inductivity accounts for inertia effects of the blood.

A detailed analysis of the two-, three- and four-component windkessel models discussed here is given in [217, 216]. Considering the relation between flow and pressure of each model, one finds that

$$Q^{\mathrm{w}} \overset{\text{2c.}}{=} \frac{p^{\mathrm{w}}}{R} + C\,\frac{\mathrm{d}p^{\mathrm{w}}}{\mathrm{d}t} \tag{3.170}$$



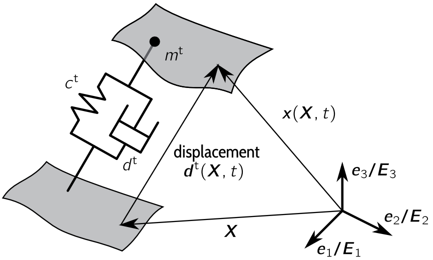
Task of a windkessel solver

Knowing $Q^\mathrm{w}(t_k)$ *and* $p^\mathrm{w}(t_k)$, *compute* $p^\mathrm{w}(t_{k+1})$ *given* $\bar{Q}^\mathrm{w}(t_{k+1})$.

Discrete windkessel solve operator

$$p^\mathrm{w}_{k+1} = \mathcal{W}_k(\bar{Q}^\mathrm{w}_{k+1})$$

Figure 3.7: Two-, three- and four-component windkessel models as electrodynamic circuits and role of a windkessel solver in a partitioned simulation.



Task of a tissue solver

Knowing $d^\mathrm{t}(\boldsymbol{X}, t_k)$, *compute* $\mathbf{t}^\mathrm{t}(\boldsymbol{X}, t_{k+1})$ *given displacements* $d^\mathrm{t}(\boldsymbol{X}, t_{k+1})$.

Discrete tissue solve operator

$$\mathbf{t}^\mathrm{t}_{k+1} = \mathcal{T}_k(\bar{\mathbf{d}}^\mathrm{t}_{k+1})$$

Task of a reduced vessel solver

Knowing $d^\mathrm{t}(\boldsymbol{X}, t_k)$, *compute* $d^\mathrm{t}(\boldsymbol{X}, t_{k+1})$ *given tractions* $\mathbf{t}^\mathrm{t}(\boldsymbol{X}, t_{k+1})$.

Discrete reduced vessel solve operator

$$\mathbf{d}^\mathrm{t}_{k+1} = \mathcal{T}_k^{-1}(\bar{\mathbf{t}}^\mathrm{t}_{k+1})$$

Figure 3.8: Distributed mass-spring-damper system as a reduced model of the surrounding tissue or the vessel itself and role of the respective solvers in a partitioned simulation.

for the two-component model,

$$\left(1 + \frac{R_2}{R_1}\right) Q^\mathrm{w} + C\, R_2 \frac{\mathrm{d}Q^\mathrm{w}}{\mathrm{d}t} \overset{\text{3c.}}{=} \frac{p^\mathrm{w}}{R_1} + C \frac{\mathrm{d}p^\mathrm{w}}{\mathrm{d}t} \tag{3.171}$$

for the three-component model and

$$\left(1 + \frac{R_2}{R_1}\right) Q^\mathrm{w} + \left(R_2\, C \frac{L}{R_1}\right) \frac{\mathrm{d}Q^\mathrm{w}}{\mathrm{d}t} + L\, C \frac{\mathrm{d}^2 Q^\mathrm{w}}{\mathrm{d}t^2} \overset{\text{4c.}}{=} \frac{p^\mathrm{w}}{R_1} + C \frac{\mathrm{d}p^\mathrm{w}}{\mathrm{d}t} \tag{3.172}$$

for the four-component model.

3.3.5 Models for the surrounding tissue

Assuming a traction-free boundary at the outside of the structural model of an artery is common practice in FSI simulations, see e.g. [203, 206, 188]. According to the obvious presence of surrounding tissue in reality, it must be accounted for in some way in the simulation approach. In the studies mentioned above, this is realized by including numerical damping into the problem (mass proportional in [188], through the temporal discretization scheme in [203, 206]). A more powerful approach is to use an elastic or viscoelastic foundation as done in [164, 146]. Of course, shifting the boundary of the structural mechanics problem further away from the outside of the adventitia would be the most accurate way of describing the behavior of the surrounding tissue. However, stopping this process at the most clearly defined boundary – where the skin touches the air and the assumption of a traction-free boundary is well-defined – would yield problem sizes which cannot be handled by today's computers. This motivates the use of reduced models for the surrounding tissue.

The foundation proposed here can be viewed as a mass-spring-damper system that is continuously distributed across a boundary, as illustrated in Fig. 3.8. The governing equation for the displacement $\boldsymbol{d}^{\mathrm{t}}$ is given by

$$m^{\mathrm{t}}\,\ddot{\boldsymbol{d}}^{\mathrm{t}} + d^{\mathrm{t}}\,\dot{\boldsymbol{d}}^{\mathrm{t}} + c^{\mathrm{t}}\,\boldsymbol{d}^{\mathrm{t}} = \bar{\boldsymbol{t}}^{\mathrm{t}}, \tag{3.173}$$

which does not include any interaction between adjacent particles. Still, Eq. (3.173) allows for quite detailed modeling of the surrounding tissue, provided that the inertia parameter m^{t}, the damping parameter d^{t}, and the stiffness parameter c^{t} are considered as spatially variable. Further, nonlinear effects may be included, which is done here for the stiffness parameter. In accordance with the characteristics of soft tissue material,

$$c^{\mathrm{t}} = c_0^{\mathrm{t}} + \frac{c_1^{\mathrm{t}}}{\|\boldsymbol{d}^{\mathrm{t}}\|}\, e^{c_2^{\mathrm{t}}\,\|\boldsymbol{d}^{\mathrm{t}}\|} \tag{3.174}$$

is chosen.

The inertia term in (3.173) can be interpreted as an added mass term in the structural mechanics problem similar to the added mass effect known from standard FSI problems. For cardiovascular FSI problems, it gives rise to another reduced modeling approach. Instead of using the distributed mass-spring-damper system to model the influence of the surrounding tissue on the vessel wall, it can be used as a model of the vessel wall itself. To this end, (3.173) is used to compute the displacement at the wet surface while the tractions are provided from the fluid subproblem as usual. The only difference between the usage as a foundation and the usage as a reduced vessel model is the numerical method applied to discretize (3.173) in space and time as explained in Section 4.1.4. In either case, an operator formulation is possible, as given in Fig. 3.8.

3.3.6 Velocity profiles

The main input to a fully resolved cardiovascular FSI simulation is the flux through the inlet of the fluid domain $\bar{Q}(t)$. From this flux, originating from measurements or a reduced order model, a velocity profile $\bar{\boldsymbol{v}}(\boldsymbol{x},t)$ needs to be computed, which can then be prescribed in terms of a general velocity boundary condition (3.111) of the fluid mechanics subproblem. One obvious constraint for $\bar{\boldsymbol{v}}(\boldsymbol{x},t)$ is that it should result in the given flux, i.e.

$$\bar{Q}(t) = \int\limits_{\Gamma^{\mathrm{f,in}}} \bar{\boldsymbol{v}}(\boldsymbol{x},t)\,\mathrm{d}\Gamma^{\mathrm{F,in}}. \tag{3.175}$$

Assuming a circular inlet with radius R, which is locally parameterized by polar coordinates r and φ, and a velocity profile $\tilde{\bar{\boldsymbol{v}}}(\boldsymbol{x}(r,t)$, which is symmetric about the center of the inlet, the constraint can be formulated as

$$\bar{Q}(t) = \int\limits_{0}^{R}\int\limits_{0}^{2\pi} \tilde{\bar{\boldsymbol{v}}}(r,t)\,r\,\mathrm{d}\varphi\,\mathrm{d}r = 2\,\pi\int\limits_{0}^{R} \tilde{\bar{\boldsymbol{v}}}(r,t)\,r\,\mathrm{d}r. \tag{3.176}$$

This allows to determine the velocity profile based on the theory of flows through circular pipes. The simplest choice results from the theory of inviscid flows and is commonly denoted as a *piston velocity profile* with

$$\tilde{\bar{\boldsymbol{v}}}(r,t) \overset{\text{pist.}}{=} \boldsymbol{v}^{\text{mean}} = \frac{\bar{Q}(t)}{2\,\pi\,R^2}. \tag{3.177}$$

However, with $\bar{\boldsymbol{v}}(R,t) \neq \boldsymbol{0}$, such a velocity boundary condition violates the no-slip boundary condition prescribed at the wet surface which shares all points $\boldsymbol{x}(R,\varphi)$ with the inlet. Prescribing a *Poiseuille velocity profile*, which is based on the theory of stationary viscous flows, circumvents this issue:

$$\tilde{\bar{\boldsymbol{v}}} \overset{\text{Pois.}}{=} 2\,\boldsymbol{v}^{\text{mean}}\left(1-\frac{r^2}{R^2}\right) = \frac{\bar{Q}(t)}{\pi\,R^2}\left(1-\frac{r^2}{R^2}\right). \tag{3.178}$$

While the Poiseuille velocity profile delivers the analytical solution of the three-dimensional Navier-Stokes equation in the case of a circular pipe and a stationary flow, the pulsatile character of blood flow in arteries leads to different velocity profiles. An analytical solution to such pulsatile flows in circular pipes is provided by the *Womersley velocity profile* as derived in

[228]. In [186], the following formulation is proposed:

$$\tilde{\boldsymbol{v}} \overset{\text{Wom.}}{=} A_0 \left(1 - \frac{r^2}{R^2}\right) + \sum_{k=1}^{n} W_k \, A_k \, e^{2\,\pi\,\mathrm{i}\,k\,\frac{t}{T}}, \tag{3.179}$$

with the imaginary number i, the period of one beat of the heart T,

$$W_k = \frac{J_0\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\right) - J_0\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\,\frac{r}{R}\right)}{J_0\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\right) - 1} \tag{3.180}$$

and the Womersley number

$$\alpha = \sqrt{\frac{2\,\pi}{\nu T}}\,R. \tag{3.181}$$

In order to use (3.179), the flow must be represented as a Fourier series

$$\bar{Q}(t) = \sum_{i=1}^{n} B_k \, e^{2\,\pi\,\mathrm{i}\,k\,\frac{t}{T}}. \tag{3.182}$$

According to [186],

$$B_k = 2\,\pi\,R^2 \, \frac{J_0\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\right) - 2\left(\alpha^2\,k\,\mathrm{i}^3\right)^{-\frac{1}{2}}\,J_1\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\right)}{J_0\left(\sqrt{\alpha^2\,k\,\mathrm{i}^3}\right) - 1}\,A_k \tag{3.183}$$

holds for the Fourier coefficients. The introduced velocity profiles are shown in Fig. 3.9 for an exemplary flow and a vessel with a radius of $R = 5\,\mathrm{mm}$. The mean flow rate is $232\,\mathrm{ml/min}$.

Non-circular cross sections

In [188] a mapping technique was introduced that allows for a recalculation of a circular velocity profile such that it can be used for an arbitrarily shaped cross section (since the mapping is not based on any physical arguments, it should be used for nearly circular cross sections only). Denoting by $\boldsymbol{x}_{\mathrm{c}}(t)$ the center of the cross section and by $\boldsymbol{x}_b(\boldsymbol{x}, t)$ the point on the boundary closest to $\boldsymbol{x}$, the radius to be used when evaluating one of the above velocity profiles can be formulated as

$$r(\boldsymbol{x}) = \frac{\|\boldsymbol{x} - \boldsymbol{x}_{\mathrm{c}}\|}{\|\boldsymbol{x} - \boldsymbol{x}_{\mathrm{b}}\| + \|\boldsymbol{x} - \boldsymbol{x}_{\mathrm{c}}\|}\,R, \tag{3.184}$$

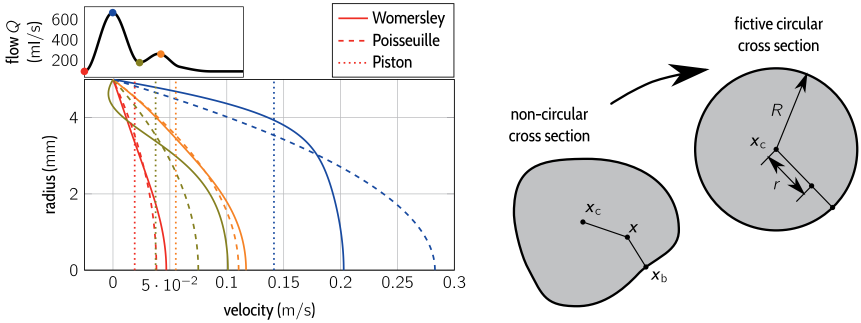


Figure 3.9: Typical velocity profiles prescribed as boundary conditions in cardiovascular FSI simulations and remapping technique for non-circular cross sections.

where

$$R = \sqrt{\int_{\Gamma^{\mathrm{f,in}}} \frac{1}{\pi} \, \mathrm{d}\Gamma^{\mathrm{f,in}}} \tag{3.185}$$

denotes the average radius of the non-circular cross section. The final velocity to be prescribed at the inlet is computed as

$$\bar{\boldsymbol{v}}(\boldsymbol{x},t) = \frac{\bar{Q}(t)}{\int_{\Gamma^{\mathrm{f,in}}} \tilde{\bar{\boldsymbol{v}}}(r,t) \, \mathrm{d}\Gamma^{\mathrm{f,in}}} \tilde{\bar{\boldsymbol{v}}}(r,t) \tag{3.186}$$

which can be seen as a correction that ensures (3.175). A truly circular inlet $\Gamma^{\mathrm{f,in}}$ would yield $\bar{\boldsymbol{v}}(\boldsymbol{x},t) = \tilde{\bar{\boldsymbol{v}}}(r,t)$.

3.3.7 Hemodynamic quantities

As explained in Chapter 2, the wall shear stress (WSS) can be seen as the most important quantity in hemodynamic analyses. It is considered a fluid mechanics quantity here, which is computed from the fluid velocity $\boldsymbol{v}$ or, respectively, the shear rate $\dot{\boldsymbol{\gamma}}$ (see 3.116). With the unit outward normal of the wall $\boldsymbol{n}^{\mathrm{w}}$,

$$\boldsymbol{\tau}^{\mathrm{w}}(\boldsymbol{x},t) = \dot{\boldsymbol{\gamma}}\, \boldsymbol{n}^{\mathrm{w}} = \boldsymbol{t} - \boldsymbol{n}^{\mathrm{w}} \cdot \boldsymbol{t} \tag{3.187}$$

is obtained, which reveals that the WSS is likewise the tangential component of the surface traction exerted by the fluid on the wall.

An open question is, how exactly the WSS influences the remodeling of the vessel wall and how it promotes the development of intima hyperplasia and atherosclerotic plaques. In [83], the following statements are made regarding this issue. As reported in [79], high WSS may injure the endothelial cells and promote the development of lesions. On the other hand, the studies in [35] suggest that low wall shear stress leads to lesions, possibly due to a correspondingly longer residence time of blood particles at the wall. In accordance to this, a safe bandwidth of WSS may exist as suggested in [121]. The hypothesis of such a bandwidth is still accepted today and has been refined by many researchers, see e.g. [142] and the references in [83].

Regardless of whether high or low WSSs classify the local hemodynamics as critical, the question arises whether a low/high mean WSS or a low/high peak WSS is of interest. As stated in [136], low WSSs are to be seen critical only in regions where the maximum WSS over one cardiac cycle is low. To this end, the local maximum wall shear stress

$$\boldsymbol{\tau}^{\mathrm{w}}_{\max}(\boldsymbol{x}) = \max_{t\in[t^*,t^*+T]} \boldsymbol{\tau}^{\mathrm{w}} \tag{3.188}$$

is utilized and critical regions are characterized by a low $\boldsymbol{\tau}^{\mathrm{w}}_{\max}$. Therefore, e.g. when comparing different configurations of bypass graft anastomosis, a quality indicator is the spatial minimum of the temporal maximum WSS, or

$$\boldsymbol{\tau}^{\mathrm{w}}_{\max,\min} = \min_{\boldsymbol{x}\in\Gamma^{\mathrm{w}}} \boldsymbol{\tau}^{\mathrm{w}}_{\max}, \tag{3.189}$$

where Γ^{w} denotes the wall region of interest. Regarding the high wall shear stress hypothesis, peak values are regarded critical, such that

$$\boldsymbol{\tau}^{\mathrm{w}}_{\max,\max} = \max_{\boldsymbol{x}\in\Gamma^{\mathrm{w}}} \boldsymbol{\tau}^{\mathrm{w}}_{\max} \tag{3.190}$$

provides another quality indicator.

As originally proposed in [126], direction changing or oscillatory WSSs may also be used as an indicator for critical regions. As reformulated by [94] and used in many studies of hemodynamics (see e.g. [186, 93, 222, 30]), the oscillating shear index (OSI)

$$\mathrm{OSI} = \frac{1}{2}\left(1 - \frac{\left\|\int_{t^*}^{t^*+T} \boldsymbol{\tau}^{\mathrm{w}}\mathrm{d}t\right\|}{\int_{t^*}^{t^*+T} \left\|\boldsymbol{\tau}^{\mathrm{w}}\right\| \mathrm{d}t}\right) \tag{3.191}$$

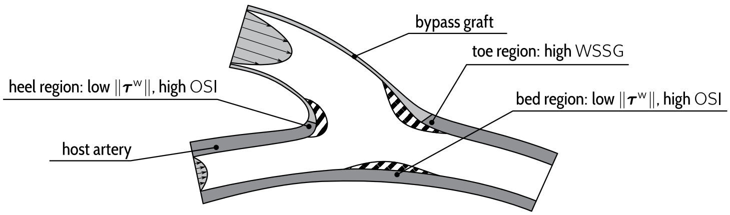

bypass graft
toe region: high WSSG
heel region: low $\|\boldsymbol{\tau}^{\mathrm{w}}\|$, high OSI
bed region: low $\|\boldsymbol{\tau}^{\mathrm{w}}\|$, high OSI
host artery

Figure 3.10: Critical regions identified in distal end-to-side anastomoses (reconstructed from [83]).

provides a measure for how oscillatory the WSSs are.

The introduced indicators for critical regions correspond well with clinically observed locations of intimal hyperplasia and atherosclerotic plaques at inner walls of curved artery segments (see [83, 41]) and at the heel and bed of distal anastomoses, as illustrated in Fig. 3.10. However, intimal hyperplasia also develops in the toe region of anastomoses, where neither low, nor high or oscillatory WSSs prevail. Typically, the WSS gradient is high in this region, which gives rise to another quality indicator, namely the time-averaged WSS gradient. As proposed in [30] it may be computed as

$$\mathrm{WSSG} = \frac{1}{T} \int\limits_{t^*}^{t^*+T} \sqrt{\left(\frac{\partial \tau_1^{\mathrm{w}}}{\partial x_1}\right)^2 + \left(\frac{\partial \tau_2^{\mathrm{w}}}{\partial x_2}\right)^2 + \left(\frac{\partial \tau_2^{\mathrm{w}}}{\partial x_2}\right)^2}\, \mathrm{d}t. \tag{3.192}$$

It should be noted that other definitions of the WSS gradient have been proposed, see e.g. the one in [133], which is based on a split of $\boldsymbol{\tau}^{\mathrm{w}}$ in a part acting in the direction of the mean WSS and another part acting in the orthogonal direction. A discussion on the role of WSSG in hemodynamics analysis in general can be found in [58].

4 Numerical methods

In the previous chapter, several models were proposed for the investigation of the fluid-structure interaction in the arterial system on different scales. The underlying equations of the models have to be solved approximately using numerical methods. In general, the unknown state variables of each model will be approximated using a finite number of unknowns. The state of each discretized model is uniquely defined by these degrees of freedom. The core problem, identified as the three-dimensional fluid-structure interaction problem, is solved by the well-known finite-volume method (applied for the fluid mechanics subproblem (3.109–3.113)) and the finite element method (applied for the structure mechanics subproblem (3.46–3.50)). The focus of the first part (*Space and time discretization*) of this chapter lies on the introduction of these two methods and on the corresponding discretized problems. Further, less complex discretization methods, which are used to solve the equations underlying the windkessel models and the elastic foundation, are introduced.

To provide analysis-suitable models for the three-dimensional FSI problem, a modeling pipeline was developed in the scope of this work. This includes a surface construction scheme that allows for a description of smooth geometries by a small number of parameters. The resulting surface parameterizations are designed in such a way that they can be used directly (without further approximation) in the scope of the high-order finite element discretization. The construction schemes as well as algorithms for the generation of the computational meshes are explained in the second part (*Geometry and mesh generation*) of this chapter.

In the third part (*Partitioned solution approach*), the coupling procedure according to the definition of the coupled problem introduced in Section 3.1 is laid out. This includes different coupling algorithms which refer to the order and type (serial/parallel) of the partitioned solution approach. Each algorithm can then be constructed using common building blocks such as convergence acceleration schemes, predictors, and interpolation methods. The building blocks as well as software tools for the data transfer from one solver to another have been implemented in the C++ library *comana* (see [122]), which will be introduced in the subsequent part (*Coupling software*).

4.1 Space and time discretization

In this section, the solve operators introduced in the previous chapter are filled with life. As done for the continuum mechanics basics, the numerical methods for the fluid and the structural subproblem are introduced here without a reference to a specific application. Due to the formulation of the reduced models as separated subproblems, only standard boundary conditions are needed in order to use the methods in a coupled cardiovascular FSI simulation.

4.1.1 High-order finite elements for structural mechanics

High-order finite elements are used in a variant of the finite-element method often denoted as the p-FEM [184]. In the scope of standard h-FEM, a higher accuracy of the numerical solution is obtained by decreasing the element size h – whereas the p-FEM, the degree p of the polynomials representing the solution within each finite element is increased. This demands for a special treatment of the geometry representation as well as a special construction of the shape functions. The following theory of the FEM can be found in many textbooks, see e.g. [184, 230, 16, 240].

Weak formulation and linearization

Starting with the structural mechanics IBVP (3.46–3.50), the first step is to multiply the PDE (3.46) by a test function $\delta\boldsymbol{d}$ and to integrate the resulting expression over the computational domain $\Omega_0^{(\mathrm{s})}$. The test function $\delta\boldsymbol{d}$ must vanish at the boundary $\Gamma_0^{(\mathrm{s},\boldsymbol{d})}$, where Dirichlet boundary conditions are prescribed. One obtains

$$g = \int\limits_{\Omega_0^{(\mathrm{s})}} \delta\,\boldsymbol{d} \cdot \Big(\rho_0\,\ddot{\boldsymbol{d}} + \mathrm{Div}(\boldsymbol{F}\,\boldsymbol{S}) + \rho_0\,\bar{\boldsymbol{b}}\Big)\;\mathrm{d}\Omega_0^{\mathrm{s}} = 0. \tag{4.1}$$

After some algebra including a partial integration of the divergence term and a replacement of the resulting surface integral by the traction boundary condition (3.48) one arrives at

$$g = \int\limits_{\Omega_0^{\mathrm{s}}} \rho_0\, \delta\boldsymbol{d} \cdot \ddot{\boldsymbol{d}} + \delta\boldsymbol{E} \cdot \boldsymbol{S} + \rho_0\, \delta\boldsymbol{d} \cdot \bar{\boldsymbol{b}}\, \mathrm{d}\Omega_0^{\mathrm{s}} + \int\limits_{\Gamma_0^{\mathrm{s}}} \delta\boldsymbol{d} \cdot \bar{\boldsymbol{t}}\, \mathrm{d}\Gamma_0^{\mathrm{s}} = 0. \tag{4.2}$$

Therein,

$$\delta\boldsymbol{E} = \mathrm{D}_{\boldsymbol{d}}\boldsymbol{E}\, \delta\boldsymbol{d} \tag{4.3}$$

was introduced. The weak formulation (4.2) may also be seen as the result of taking the first variation of the objective function (3.61) of the minimization problem introduced in Section 3.2.2. Then, $\delta\boldsymbol{d}$ can be identified as a displacement variation and $\delta\boldsymbol{E}$ is the variation of the Green-Lagrange strain. In the static case, the method of virtual displacements may likewise lead to (4.2). Here, $\delta\boldsymbol{d}$ is not restricted to any of these interpretations but simply considered a test function. However, the writing from the framework of variational calculus for $\delta\boldsymbol{d}$ and $\delta\boldsymbol{E}$ is borrowed. Coming from the minimization problem, (4.2) is obtained directly ($\boldsymbol{S} \cdot \delta\boldsymbol{E}$ follows directly as the first variation of the strain energy density W, see (3.60)). The more involved derivation starting from (4.1) is summarized in Section A.3.1.

Problem (4.2) will be solved iteratively with the Newton-Raphson method or a similar method. Therefore, the derivative of g in the direction of an increment in the unknown displacement $\Delta\boldsymbol{d}$ is needed. In general, it reads

$$\begin{aligned}\mathrm{D}_{\boldsymbol{d}}\, g \cdot \delta\boldsymbol{d} = &\int\limits_{\Omega_0^{\mathrm{s}}} \delta\boldsymbol{d} \cdot \mathrm{D}_{\boldsymbol{d}}\ddot{\boldsymbol{d}}\, \rho_0\, \Delta\ddot{\boldsymbol{d}}\, \mathrm{d}\Omega_0^{\mathrm{s}} \\ &+ \int\limits_{\Omega_0^{\mathrm{s}}} \delta\boldsymbol{E} \cdot \mathrm{D}_{\boldsymbol{E}}\boldsymbol{S}\, \Delta\boldsymbol{E} + \mathrm{Grad}(\Delta\boldsymbol{d})\ \boldsymbol{S} \cdot \mathrm{Grad}(\delta\boldsymbol{d})\ \mathrm{d}\Omega_0^{\mathrm{s}} \\ &+ \int\limits_{\Omega_0^{\mathrm{s}}} \delta\boldsymbol{d} \cdot \mathrm{D}_{\boldsymbol{d}}\bar{\boldsymbol{b}}\, \Delta\boldsymbol{d}\, \mathrm{d}\Omega_0^{\mathrm{s}} + \int\limits_{\Gamma_0^{\mathrm{s}}} \Delta\boldsymbol{d} \cdot \mathrm{D}_{\boldsymbol{d}}\bar{\boldsymbol{t}}\, \Delta\boldsymbol{d}\, \mathrm{d}\Gamma_0^{\mathrm{s}},\end{aligned} \tag{4.4}$$

where $\Delta\boldsymbol{E} = \mathrm{D}_{\boldsymbol{d}}\boldsymbol{E}\, \Delta\boldsymbol{d}$ denotes the directional derivative of the Green-Lagrange strain tensor. Note that for the dynamic terms in the first row of (4.4), a specific form can not be stated until a temporal discretization scheme is applied. A derivation of the terms in the second row can be found in [230, p. 97]. Allowing only for deformation-independent volume loads $\bar{\boldsymbol{b}}$, as usual, the first term in the third row of (4.4) will disappear. Regarding the second term that contains the surface traction $\bar{\boldsymbol{t}}$, different scenarios are discussed at the end of this section.

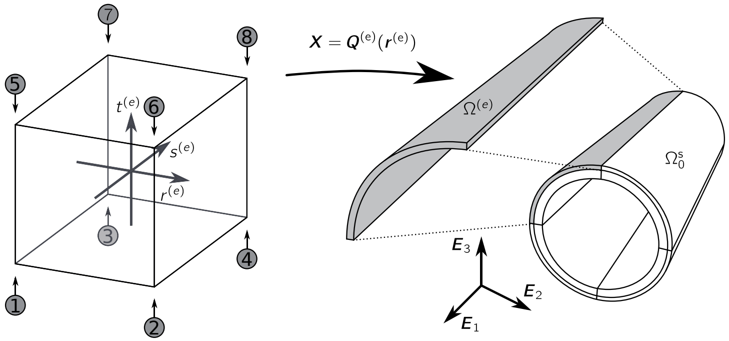


Figure 4.1: Discretization of the computational domain $\Omega^{\}}mrms_0$ with a finite element mesh.

Spatial discretization

Following the standard approach of finite element methods, the domain Ω is partitioned into polygonal non-overlapping subdomains or elements, such that

$$\Omega_0^\mathrm{s} \approx \bigcup_{e=0}^{n^\mathrm{e}} \Omega^{(e)}. \tag{4.5}$$

Depending on the shape of the domain and the type of finite elements used, Ω_0^s can be represented exactly by this partitioning. Generally, however, the finite element mesh is only an approximation of the real geometry. Within each element, the geometry is described using $\hat{n}_e$ shape functions $\hat{N}_i^{(e)}\left(r^{(e)}, s^{(e)}, t^{(e)}\right)$, such that

$$\boldsymbol{X} \approx \sum_{i=0}^{\hat{n}_e} \hat{N}_i^{(e)}\, \boldsymbol{X}_i^{(e)} \qquad\qquad \text{in } \Omega^{(\mathrm{e})}. \tag{4.6}$$

The same approach is used to discretize the unknown displacement

$$\boldsymbol{d} \approx \sum_{i=0}^{n_e} N_i^{(e)}\, \boldsymbol{d}_i^{(e)} = \mathbf{N}^{(e)}\, \mathbf{d}^{(e)} \qquad\qquad \text{in } \Omega^{(e)} \tag{4.7}$$

as well as the test function

$$\delta\boldsymbol{d} \approx \sum_{i=0}^{n_e} N_i^{(e)}\, \delta\boldsymbol{d}_i^{(e)} = \mathbf{N}^{(e)}\, \delta\mathbf{d}^{(e)} \qquad\qquad \text{in } \Omega^{(e)}. \tag{4.8}$$

To ease the notation, the element degree of freedom vector

$$\mathbf{d}^{(e)} = \left[\; \boldsymbol{d}_1^{(e)\mathrm{T}} \;\; \dots \;\; \boldsymbol{d}_{n_e}^{(e)\mathrm{T}} \;\right]^{\mathrm{T}} \tag{4.9}$$

was introduced in (4.7) along with the shape function matrix $\mathbf{N}^{(e)}$ as given in Section A.3.2. The degree of freedom vector for the test function $\delta\mathbf{d}^{(e)}$ is constructed in the same manner as $\mathbf{d}^{(e)}$. The construction of suitable shape functions is addressed at the end of this section. Here, only a few remarks shall be given:

- Shape functions are defined on a local element coordinate system with coordinates $\boldsymbol{r}^{(e)} = \left[\; r^{(e)} \;\; s^{(e)} \;\; t^{(e)} \;\right]^{\mathrm{T}}$ (cf. Fig. 4.1).
- The fact that the same set of shape function N_i is used to discretize the unknown $\boldsymbol{d}$ and the test function $\delta\boldsymbol{d}$ renders the method a *Bubnov-Galerkin method.*
- If the same set of shape functions is used for the geometry and the unknown, i.e. $\hat{N}_i^{(e)} = \hat{N}_i^{(e)}$, the resulting element formulation is called *isoparametric.* This is the case, as long as the two sets of shape functions span the same space, i.e. $\mathrm{span}(\hat{N}_i^{(e)}) = \mathrm{span}(N_i^{(e)})$.
- If $\mathrm{span}(\hat{N}_i^{(e)}) \subset \mathrm{span}(N_i^{(e)})$, the formulation is called *subparametric.* If $\mathrm{span}(N_i^{(e)}) \subset \mathrm{span}(\hat{N}_i^{(e)})$, it is called *superparametric* (given that $\mathrm{span}(N_i^{(e)}) \neq \mathrm{span}(\hat{N}_i^{(e)})$).

Gradients with respect to the material coordinates $\boldsymbol{X}$ are obtained as

$$\mathrm{Grad}(\boldsymbol{d}) \approx \sum_{i=0}^{n_e} \mathrm{Grad}\left(N_i^{(e)}\right) \boldsymbol{d}_i^{(e)} \qquad \text{in } \Omega^{(e)}. \tag{4.10}$$

For the computation of the partial derivatives $\partial N_i^{(e)}/\partial X_j$ involved in $\mathrm{Grad}\left(N_i^{(e)}\right)$, the mapping defined by (4.6) has to be considered. Applying the chain rule, one arrives at

$$\frac{\partial N_i^{(e)}}{\partial X_j} = \frac{\partial r_k^{(e)}}{\partial X_j} \frac{\partial N_i^{(e)}}{\partial r_k^{(e)}}, \tag{4.11}$$

which is amenable to the matrix formulation

$$\mathrm{Grad}\left(N_i^{(e)}\right) = \boldsymbol{J}^{(e)^{-T}} \mathrm{grad}_{\boldsymbol{r}}\left(N_i^{(e)}\right), \tag{4.12}$$

where $\boldsymbol{J}^{(e)}$ denotes the element Jacobian matrix with entries

$$J_{ij}^{(e)} = \frac{\partial X_i}{\partial r_j^{(e)}}. \tag{4.13}$$

With the above concept underlying all finite element methods, the discretized governing equations can be formulated. Starting with Eq. (4.2), the scalar product $\boldsymbol{S} \cdot \delta\boldsymbol{E}$ has to be expressed in a suitable way. To this end, the *Voigt notation* is applied, with

$$\mathbf{E}^{\mathrm{V}} = \begin{bmatrix} E_{11} & E_{22} & E_{33} & 2\,E_{12} & 2\,E_{23} & 2\,E_{13} \end{bmatrix}^{\mathrm{T}} \tag{4.14}$$

and

$$\mathbf{S}^{\mathrm{V}} = \begin{bmatrix} S_{11} & S_{22} & S_{33} & S_{12} & S_{23} & S_{13} \end{bmatrix}^{\mathrm{T}} \tag{4.15}$$

being the Voigt representations of the Green-Lagrange strain and the Second Piola-Kirchhoff stress, respectively. Voigt representations of $\delta\boldsymbol{E}$ and other strain-like quantities are obtained according to (4.14) and stress-like quantities are transformed according to (4.15) in order to preserve the equality of scalar products. Now, $\delta\mathbf{E}^{\mathrm{V}} = \mathbf{B}^{(e)}\,\delta\mathbf{d}^{(e)}$ can be established using the *strain-displacement matrix*

$$\mathbf{B}^{(e)} = \begin{bmatrix} \mathbf{B}_1^{(e)} & \dots & \mathbf{B}_{n_e}^{(e)} \end{bmatrix} \tag{4.16}$$

with $\mathbf{B}_i^{(e)}$ as defined in Section A.3.2. This leads to the discretized weak formulation (4.2)

$$\begin{aligned} g^{(e)} = \mathbf{v}^{(e)\mathrm{T}} & \left[\underbrace{\int_{\Omega_0^{\mathrm{s}}} \mathbf{B}^{(e)\mathrm{T}}\, \mathbf{S}^{\mathrm{V}} \mathrm{d}\Omega_0^{\mathrm{s}}}_{\mathbf{i}^{(e)}} - \underbrace{\int_{\Omega_0^{\mathrm{s}}} \mathbf{N}^{(e)\mathrm{T}}\, \bar{\boldsymbol{b}}\, \mathrm{d}\Omega_0^{\mathrm{s}} + \int_{\Gamma_0^{\mathrm{s}}} \mathbf{N}^{(e)\mathrm{T}}\, \bar{\boldsymbol{t}}\, \mathrm{d}\Gamma_0^{\mathrm{s}}}_{\mathbf{e}^{(e)}} \right] \\ + \mathbf{v}^{(e)\mathrm{T}} & \left[\underbrace{\int_{\Omega_0^{\mathrm{s}}} \mathbf{N}^{(e)\mathrm{T}} \rho_0\, \mathbf{N}^{(e)}\, \mathrm{d}\Omega_0^{\mathrm{s}}}_{\mathbf{M}^{(e)}} \right] \ddot{\mathbf{d}}^{(e)} \end{aligned} \tag{4.17}$$

where the static internal load vector $\mathbf{i}^{(e)}$, the static external load vector $\mathbf{e}^{(e)}$, and the mass matrix $\mathbf{M}^{(e)}$ were introduced. From them, the global system of ordinary differential equations governing the structural mechanics subproblem can be obtained by assembling the element matrices according to their connectivity, i.e. shared degrees of freedom. An algorithm for the assembly process is given in Section A.3.3. The resulting global system reads

$$\mathbf{M}\,\ddot{\mathbf{d}} + \mathbf{D}\,\dot{\mathbf{d}} + \mathbf{i} = \mathbf{e}. \tag{4.18}$$

A damping matrix $\mathbf{D}$ has been introduced in (4.18) for the sake of completeness. In the scope of this work, it was decided to apply either no damping at all or mass proportional damping, where $\mathbf{D} = \zeta_0\,\mathbf{M}$. Before the finite element discretization of the linearized system (4.4) is established, the temporal discretization of Eq. (4.18) is discussed.

Temporal discretization

The generalized-α method proposed in [42] is used to discretize the structural mechanics problem in time. For special choices regarding its parameters, the generalized-α method yields well-known time integration methods such as the Newmark-β method. A detailed analysis of the method is provided in [127]. The basic idea is to evaluate the ODE (4.18) somewhere between time t_k (where the state is already known) and $t_{k+1} = t_k + \Delta t_k$, or, more precisely, at time

$$t_{k+1-\alpha} = (1-\alpha)\;t_{k+1} + \alpha\,t_k. \tag{4.19}$$

This may be written as

$$\mathbf{M}\,\ddot{\mathbf{d}}_{k+\alpha_m} + \mathbf{D}\,\dot{\mathbf{d}}_{k+\alpha_m} + \mathbf{i}_{k+\alpha_f} = \mathbf{e}_{k+\alpha_f}, \tag{4.20}$$

where α_m is used for the inertia (mass) term while α_f is used for the other (force) terms. The acceleration and the velocity are linearly interpolated, i.e.

$$\ddot{\mathbf{d}}_{k+1-\alpha_m} = (1-\alpha_m)\;\ddot{\mathbf{d}}_{k+1} + \alpha_m\,\ddot{\mathbf{d}}_k, \tag{4.21}$$

$$\dot{\mathbf{d}}_{k+1-\alpha_m} = (1-\alpha_f)\;\dot{\mathbf{d}}_{k+1} + \alpha_f\,\dot{\mathbf{d}}_k. \tag{4.22}$$

For the internal and the external load vector,

$$\mathbf{i}_{k+1-\alpha_f} = (1-\alpha_f)\;\mathbf{i}(\mathbf{d}_{k+1}) + \alpha_f\,\mathbf{i}(\mathbf{d}_k), \tag{4.23}$$

$$\mathbf{e}_{k+1-\alpha_f} = (1-\alpha_f)\;\mathbf{e}(\mathbf{d}_{k+1}, t_{k+1}) + \alpha_f\,\mathbf{e}(\mathbf{d}_k, t_k) \tag{4.24}$$

are introduced. Alternative approaches, e.g. $\mathbf{i}_{k+1-\alpha_f} = \mathbf{i}(\mathbf{d}_{k+1-\alpha_f})$ with $\mathbf{d}_{k+1-\alpha_f}$ according to (4.21) and (4.22) are discussed in [127].

Table 4.1: Relation between special choices for the spectral radius ρ_∞ of the generalized-α method and well-known time integration methods.

method	α_m	α_f	β	γ
Newmark-β	0	0	$(1+\rho_\infty)^{-2}$	$\frac{3-\rho_\infty}{2\,\rho_\infty+2}$
HHT-α	0	$\frac{1-\rho_\infty}{1+\rho_\infty}$	$\frac{1}{4}\,(1+\alpha_f)^2$	$\frac{1}{2}+\alpha_f$
WBZ-α	$\frac{1-\rho_\infty}{1+\rho_\infty}$	0	$\frac{1}{4}\,(1-\alpha_m)^2$	$\frac{1}{2}-\alpha_m$
optimal generalized-α	$\frac{2\,\rho_\infty-1}{\rho_\infty+1}$	$\frac{\rho_\infty}{\rho_\infty+1}$	$\frac{1}{4}\,(1-\alpha_m+\alpha_f)^2$	$\frac{1}{2}-\alpha_m+\alpha_f$

Introducing the spectral radius ρ_∞ as a convenient parameter, α_f, α_m as well as β and γ can be computed in accordance to ρ_∞ to yield well-known time integration methods, as summarized in Tab. 4.1. A derivation of the relations is provided in [69]. The optimal parameters given in Tab. 4.1 ensure second order accuracy as well as unconditional stability (for linear problems).

The Newmark approximations

$$\dot{\mathbf{d}}_{k+1} = \left(\frac{\gamma}{\beta\,\Delta t_k}\right)\mathbf{d}_{k+1} - \left(\frac{\gamma}{\beta\,\Delta t_k}\right)\mathbf{d}_k - \left(\frac{\gamma}{\beta}-1\right)\dot{\mathbf{d}}_k - \left(\frac{\gamma\,\Delta t_k}{2\,\beta}-\Delta t_k\right)\ddot{\mathbf{d}}_k, \tag{4.25}$$

$$\ddot{\mathbf{d}}_{k+1} = \left(\frac{1}{\beta\,\Delta t_k^2}\right)\mathbf{d}_{k+1} - \left(\frac{1}{\beta\,\Delta t_k^2}\right)\mathbf{d}_k - \left(\frac{1}{\beta\,\Delta t_k}\right)\dot{\mathbf{d}}_k - \left(\frac{1}{2\,\beta}-1\right)\ddot{\mathbf{d}}_k \tag{4.26}$$

are used to obtain the final system of equations to be solved in each time step. Inserting (4.25) and (4.26) in (4.20) results in the discrete structural subproblem

$$\begin{aligned}\mathbf{P}_k^{\mathrm{s}} = \mathbf{M}\,\left(a_0\,\mathbf{d}_k + a_0\,\dot{\mathbf{d}}_k + a_2\,\ddot{\mathbf{d}}_k\right) + \mathbf{D}\,\left(a_1\,\mathbf{d}_k + a_4\,\dot{\mathbf{d}}_k + a_5\,\ddot{\mathbf{d}}_k\right) + \alpha_f\,\left(\mathbf{i}_k - \mathbf{e}_k\right)\\ + \left(a_0\,\mathbf{M} + a_1\,\mathbf{D}\right)\,\mathbf{d}_{k+1} + (1-\alpha_f)\,\left(\mathbf{i}_{k+1} - \mathbf{e}_{k+1}\right) = \mathbf{0}\end{aligned} \tag{4.27}$$

with

$$\begin{aligned} a_0 &= \frac{1-\alpha_m}{\beta\,\Delta t_k^2}, & a_1 &= \frac{(1-\alpha_f)\,\gamma}{\beta\,\Delta t_k}, & a_2 &= \frac{1-\alpha_m}{\beta\,\Delta t_k},\\ a_3 &= \frac{1-\alpha_m}{2\,\beta}-1, & a_4 &= \frac{(1-\alpha_f)\,\gamma}{\beta}-1, & a_5 &= (1-\alpha_f)\left(\frac{\gamma\,\Delta t_k}{2\,\beta}-\Delta t_k\right). \end{aligned} \tag{4.28}$$

In (4.27), the terms were ordered such that only the second row contains the unknown displacement $\mathbf{d}_{k+1}$. Consequently, these terms are the only ones that contribute to the tangent.

Tangent matrix

As (4.27) was introduced as the discrete counterpart to (4.2),

$$\mathrm{D}_{\mathbf{d}_{k+1}}\mathbf{P}^{\mathrm{s}}_{k} = a_0\,\mathbf{M} + a_1\,\mathbf{D} + (1-\alpha_f)\,(\mathbf{K}_{\mathbf{i}} - \mathbf{K}_{\mathbf{e}}) \tag{4.29}$$

is introduced as the discrete counterpart to (4.4). Therein, the static parts $\mathbf{K}_{\mathbf{i}}$ and $\mathbf{K}_{\mathbf{e}}$ can be formulated on element level before assembling them as explained in Section A.3.3. For the linearization of the internal loads,

$$\mathbf{K}^{(e)}_{\mathbf{i}} = \mathrm{D}_{\mathbf{d}_{k+1}}\mathbf{i}^{(e)} = \int\limits_{\Omega^{\mathrm{s}}_0} \mathbf{B}^{(e)^{\mathrm{T}}}\,\mathbf{C}^{\mathrm{V}}\,\mathbf{B}^{(e)} + \mathbf{G}^{(e)}\,\mathrm{d}\Omega^{\mathrm{s}}_0 \tag{4.30}$$

is obtained directly from (4.4) by inserting the finite element approximation. In (4.30), $\mathbf{C}^{\mathrm{V}}$ denotes the Voigt representation of the elasticity tensor $\mathcal{C} = {}^{\partial \boldsymbol{S}}/_{\partial \boldsymbol{E}}$ (see Section A.3.4 for its construction) and $\mathbf{B}^{(e)}$ comes from the discretization of the Green-Lagrange strain increment $\boldsymbol{\Delta}\mathbf{E}^{\mathrm{V}} = \mathbf{B}^{(e)}\mathbf{d}^{(e)}$. The so-called *geometric stiffness* $\mathbf{G}^{(e)}$ is defined using index notation by

$$G^{(e)}_{ij} = \mathrm{Grad}\Big(N^{(e)}_i\Big)\cdot\boldsymbol{S}\,\mathrm{Grad}\Big(N^{(e)}_j\Big) \tag{4.31}$$

as proposed in [230, p. 129]. The tangent contribution $\mathbf{K}^{(\mathbf{e})}$ from the external loads may take different forms depending on the type of load applied, as explained in the following.

In the simplest scenario, the tractions are *deformation independent* and stated for the reference configuration, i.e. $\bar{\boldsymbol{t}} = \bar{\boldsymbol{t}}^{\mathrm{di}}(\boldsymbol{X},t)$ with

$$\mathrm{D}_{\boldsymbol{d}}\bar{\boldsymbol{t}}^{\mathrm{di}}\,\Delta\boldsymbol{d} = \mathbf{0}. \tag{4.32}$$

So-called *follower loads* are defined for the undeformed reference state and follow the surface they act on. Typically, pressure loads $\bar{\boldsymbol{t}} = \bar{\boldsymbol{t}}^{\mathrm{p}}$ are treated like this. Their contribution in the weak form can be obtained by considering a single face of a finite element, which is parameterized by face coordinates $r^{(e)}_a$ and $r^{(e)}_b$. The relation between the element coordinates and the face coordinates is given in Tab. A.3.3. One obtains

$$\int\limits_{\Gamma^{(e)}_0} \delta\boldsymbol{d}\cdot\bar{\boldsymbol{t}}^{\mathrm{p}}\,\mathrm{d}\Gamma^{(e)}_0 = \int\limits_{\Gamma^{(e)}_t} \delta\boldsymbol{d}\cdot\bar{p}\,\boldsymbol{n}\,\mathrm{d}\Gamma^{(e)}_t = \int\limits_{-1}^{1}\int\limits_{-1}^{1} \delta\boldsymbol{d}\cdot\bar{p}\,\hat{\boldsymbol{n}}\,\mathrm{d}r^{(e)}_a\,\mathrm{d}s^{(e)}_b, \tag{4.33}$$

where $\hat{\boldsymbol{n}}$ denotes the product of the unit normal vector $\boldsymbol{n}$ and the factor originating from the coordinate transformation. Considering the case that $r_a^{(e)} = r^{(e)}$ and $r_b^{(e)} = s^{(e)}$, $\hat{\boldsymbol{n}}$ may be obtained by

$$\hat{\boldsymbol{n}} = \boldsymbol{t}^{(r)} \times \boldsymbol{t}^{(s)}, \tag{4.34}$$

where $\boldsymbol{t}^{(r)}$ and $\boldsymbol{t}^{(s)}$ denote tangent vectors. Introducing a Jacobian matrix $\boldsymbol{j} = \boldsymbol{F}\,\boldsymbol{J}$ that relates the derivatives with respect to the local element coordinates $\boldsymbol{r}$ and the derivatives with respect to the spatial coordinates $\boldsymbol{x}$, the tangent vectors correspond to the respective columns of $\boldsymbol{j}$, i.e.

$$t_i^{(r)} = j_{i1} \quad \text{and} \quad t_i^{(s)} = j_{i2}. \tag{4.35}$$

In the reference configuration, one may write

$$T_i^{(r)} = J_{i1} \quad \text{and} \quad T_i^{(s)} = J_{i2} \tag{4.36}$$

and

$$\hat{\boldsymbol{N}} = \boldsymbol{T}^{(r)} \times \boldsymbol{T}^{(s)}, \tag{4.37}$$

which allows for formulation of the traction vector in the reference configuration as

$$\bar{\boldsymbol{t}}^{\mathrm{p}} = \bar{p}\, J\, \boldsymbol{F}^{-\mathrm{T}}\, \hat{\boldsymbol{N}}. \tag{4.38}$$

The tangent contribution can now be obtained by standard rules, which yield

$$\begin{aligned}\int\limits_{\Gamma_0^{(e)}} \mathrm{D}_{\boldsymbol{d}} \bar{\boldsymbol{t}}^{\mathrm{p}}\, \Delta\boldsymbol{d}\, \mathrm{d}\Gamma_0^{(e)} &= \int\limits_{-1}^{1}\int\limits_{-1}^{1} \left((\mathrm{D}_{\boldsymbol{d}} J)\, \mathbf{F}^{-\mathrm{T}} + J\left(\mathrm{D}_{\boldsymbol{d}} \mathbf{F}^{-\mathrm{T}}\right)\right) \bar{p}\, \hat{\boldsymbol{N}}\, \Delta\boldsymbol{d}\, \mathrm{d}r_a^{(e)}\, \mathrm{d}s_b^{(e)} \\ &= \int\limits_{-1}^{1}\int\limits_{-1}^{1} \left(J\, \mathbf{F}^{-\mathrm{T}} \cdot \mathrm{Grad}(\boldsymbol{d})\ \mathbf{F}^{-\mathrm{T}} - J\, \mathbf{F}^{-\mathrm{T}}\, \mathrm{Grad}(\boldsymbol{d})\ \mathbf{F}^{-\mathrm{T}}\right) \bar{p}\, \hat{\boldsymbol{N}}\, \mathrm{d}r_a^{(e)}\, \mathrm{d}s_b^{(e)}.\end{aligned} \tag{4.39}$$

In partitioned FSI simulations, the traction $\bar{\boldsymbol{t}}^{\mathrm{FSI}}$ on the spatial domain boundary $\Gamma_t^{(\mathrm{s},\boldsymbol{t})}$ is provided, such that a coordinate transformation must be performed, when integrating over $\Gamma_0^{(\mathrm{s},\boldsymbol{t})}$. From

$$\int\limits_{\Gamma_0^{(e)}} \delta\boldsymbol{d} \cdot \bar{\boldsymbol{t}}\, \mathrm{d}\Gamma_0^{(e)} = \int\limits_{\Gamma_t^{(e)}} \delta\boldsymbol{d} \cdot \bar{\boldsymbol{t}}^{\mathrm{FSI}}\, \mathrm{d}\Gamma_t^{(e)} = \int\limits_{-1}^{1}\int\limits_{-1}^{1} \delta\boldsymbol{d} \cdot \bar{\boldsymbol{t}}^{\mathrm{FSI}}\ \|\hat{\boldsymbol{n}}\|\ \mathrm{d}\boldsymbol{r}_a^{(e)}\, \mathrm{d}\boldsymbol{r}_b^{(e)} \tag{4.40}$$

the traction vector in the reference configuration is obtained as

$$\bar{\boldsymbol{t}} = \frac{\|\hat{\boldsymbol{n}}\|}{\|\hat{\boldsymbol{N}}\|}\,\bar{\boldsymbol{t}}^{\mathrm{FSI}}. \tag{4.41}$$

The tangent contribution may be obtained as in the case of follower loads by expressing the normal vectors $\hat{\boldsymbol{n}}$ as $J\,\boldsymbol{F}^{-\mathrm{T}}\hat{\boldsymbol{N}}$ and applying standard rules for differentiation. However, this contribution is independent of rigid body motions and small as long as $J \approx 1$, so that the Newton-Raphson iterations converge well also when neglecting $\mathrm{D}_{\boldsymbol{d}}\bar{\boldsymbol{t}}^{\mathrm{FSI}}$.

Solution procedure

The elaborations from the previous sections allow for a formulation of the solution procedure underlying a nonlinear finite element analysis in terms of a step-by-step procedure.

1. Chose $\mathbf{d}_0$, $\dot{\mathbf{d}}_0$, β, γ, α_f, α_m, Δt.
2. Compute $\mathbf{M}$, $\mathbf{D}$, $\mathbf{i}_0$, $\mathbf{e}_0$.
3. Compute initial acceleration $\ddot{\mathbf{d}}_0 = \mathbf{M}^{-1}\left(\mathbf{e}_0 - \mathbf{D}\,\dot{\mathbf{d}}_0 - \mathbf{i}_0\right)$.
4. Set $\mathbf{K}^{\mathrm{const}} = a_0\,\mathbf{M} + a_1\,\mathbf{D}$.
5. **Start time loop**, set $k = 0$.
6. Predict $\tilde{\mathbf{d}}_0$, e.g. $\tilde{\mathbf{d}}_0 = \mathbf{d}_k$.
7. Set $\mathbf{P}_{k+1}^{\mathrm{const}} = \alpha_f\,(\mathbf{i}_k - \mathbf{e}_k) - \mathbf{M}\left(a_0\,\mathbf{d}_k + a_2\,\dot{\mathbf{d}}_k + a_3\,\ddot{\mathbf{d}}_k\right) - \mathbf{D}\left(a_1\,\mathbf{d}_k + a_4\,\dot{\mathbf{d}}_k + a_5\,\ddot{\mathbf{d}}_k\right)$
8. **Start Newton-Raphson loop**, set $i = 0$.
9. Compute $\tilde{\mathbf{K}}_i = \mathbf{K}_{\mathbf{i}}(\tilde{\mathbf{d}}_i) + \mathbf{K}_{\mathbf{e}}(\tilde{\mathbf{d}}_i)$, $\tilde{\mathbf{i}}_i = \mathbf{i}(\tilde{\mathbf{d}}_i)$, and $\tilde{\mathbf{e}}_i = \mathbf{e}(\tilde{\mathbf{d}}_i)$
10. Set $\tilde{\mathbf{P}}_i = \mathbf{P}_{k+1}^{\mathrm{const}} + \mathbf{K}^{\mathrm{const}}\tilde{\mathbf{d}}_i + (1 - \alpha_f)\left(\tilde{\mathbf{i}}_i - \tilde{\mathbf{e}}_i\right)$,
11. Set $\tilde{\mathbf{K}}_i^{\mathrm{eff}} = \mathbf{K}^{\mathrm{const}} + (1 - \alpha_f)\,\tilde{\mathbf{K}}^i$
12. Solve $\tilde{\mathbf{K}}_i^{\mathrm{eff}}\,\Delta\mathbf{d}_i = -\tilde{\mathbf{P}}_i$ for $\Delta\mathbf{d}_i$.
13. Update solution, $\tilde{\mathbf{d}}_{i+1} = \tilde{\mathbf{d}}_i + \Delta\mathbf{d}_i$.

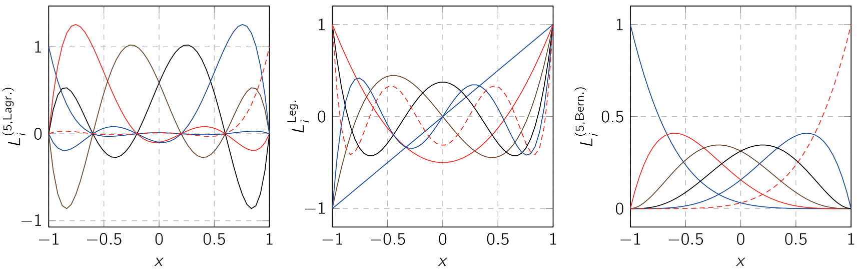


Figure 4.2: Commonly used bases (Lagrange, Legendre, Bernstein) for the space of polynomials $\mathbb{P}^5$.

14. If not converged, increment i and go to Step 9.
15. Set $\mathbf{d}_{k+1} = \tilde{\mathbf{d}}_i$ and compute $\dot{\mathbf{d}}_{k+1}$ and $\ddot{\mathbf{d}}_{k+1}$ acc. to Eqs. (4.22) and (4.21)
16. Increment k and go to Step 6.

Shape functions and ansatz spaces

Polynomial shape functions are usually used in the context of finite element methods. Figure 4.2 shows typical bases for the space of one-dimensional polynomials. While oftentimes Lagrange bases are used to create finite element shape functions (see e.g. [240]), integrated Legendre polynomials are used here. As shown in [239], this leads to a slower increasing condition number of the tangent matrix as the polynomial order is increased, which improves the efficiency of iterative solvers. Mathematically, the bases shown in Fig. 4.2 are equivalent in terms of the space they span. The Legendre polynomials can be constructed with $L_0^{\text{Leg.}} = 1$ and $L_1^{\text{Leg.}} = r$, using the recursion relation

$$L_i^{\text{Leg.}} = \frac{1}{i}\left((2\,i-1)\;L_{i-1}^{\text{Leg.}} - (i-1)\;L_{i-2}^{\text{Leg.}}\right). \tag{4.42}$$

This leads to a recurrence relation for their integrals

$$N_i^{(1D)} = \sqrt{\frac{2i-1}{2}}\int\limits_{-1}^{r} L_i^{\text{Leg.}}\,\mathrm{d}r = \sqrt{\frac{1}{4i-2}}\left(L_i^{\text{Leg.}} - L_{i-2}^{\text{Leg.}}\right) \quad \text{for } i > 2, \tag{4.43}$$

which can readily be used as shape functions for one-dimensional elements. For $i = 1$ and $i = 2$, the well-known nodal shape functions from the h-FEM are used, namely

$$N_1^{(1D)} = \frac{1}{2}(1 - r), \quad N_2^{(1D)} = \frac{1}{2}(1 + r). \tag{4.44}$$

From the one-dimensional basis, three-dimensional finite element bases can be constructed in different ways, leading to different ansatz spaces. In general, the three-dimensional bases may be constructed using

$$N_i^{(e)}(\mathbf{r}^{(e)}) = N_l^{(1\mathrm{D})}(r^{(e)})\, N_m^{(1\mathrm{D})}(s^{(e)})\, N_n^{(1\mathrm{D})}(t^{(e)}). \tag{4.45}$$

Choosing different ranges for the indices l, m, n leads to different ansatz spaces.

Tensor product space (TPS) This ansatz space is based on a simple tensor product structure. All possible combinations of l, m, n are used, i.e.

$$l, m, n \overset{\mathrm{TPS}}{=} 1, \ldots, p + 1. \tag{4.46}$$

The tensor product space constitutes a basis for the full space of three-dimensional polynomials $\mathbb{P}^p \times \mathbb{P}^p \times \mathbb{P}^p$.

Trunk space (TS) Within the tensor product space, different shape functions may be associated with different entities of the hexahedral element. For example, if one and only one of the indices l, m, n is smaller or equal to two, the corresponding $N_i^{(e)}$ is zero on all edges and faces except for one face. The trunk space uses the range

$$l, m, n = 1, \ldots, p + 1 \qquad \text{with } l + m + n \overset{\mathrm{TS}}{=} 1,\ \ldots,\ p, \tag{4.47}$$

which results in a decreased number of the mentioned face shapes functions and the internal shape functions (zero on all faces and all edges).

Serendipity space (SS) By using even less shape functions from the tensor product space, it is possible to establish an ansatz space that contains only nodal end edge shape functions. The corresponding range can be defined as

$$l, m, n = 1, \ldots, p + 1 \quad \text{with } (l - 1)(m - 1) \overset{\mathrm{SS}}{=} 0,\ (m - 1)(n - 1) \overset{\mathrm{SS}}{=} 0, \tag{4.48}$$

which essentially states that two of the indices l, m, n have to be equal to 1.

Quasiregional mapping

In Fig. 4.1, the mapping $\boldsymbol{X} = \boldsymbol{Q}^{(e)}(\boldsymbol{r}^{(e)})$ is introduced. It is constructed using the so-called *quasi regional mapping* concept as explained in detail in [119, 64]. The general idea is to approximate the geometry of $\Omega_0^{(\mathrm{s})}$ using polynomials. If the Lagrange basis

$$L_j^{(p^{\mathrm{g}},\mathrm{Lag.})}(r) = \prod_{\substack{0 \leq i \leq p^{\mathrm{g}} \\ i \neq j}} \frac{r - r_i}{r_j - r_i} \tag{4.49}$$

is used to construct $\hat{N}_i^{(e)}$, this boils down to prescribing the position at a finite number of sampling points, because

$$L_j^{(p^{\mathrm{g}},\mathrm{Lag.})}(r_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{else.} \end{cases} \tag{4.50}$$

To counteract oscillations that usually appear when interpolating real functions with high-order polynomials, optimal interpolation points r_i (approximately computed in [39, 40]) are used. In accordance to the authors' names, they are referred to as *Chen-Babuška points*.

More precisely, the geometry is approximated separately for each entity (face, edge) of the hexahedral element. One obtains

$$\boldsymbol{Q}^{(e)} = \sum_{i=1}^{8} \boldsymbol{n}_i^{(e)} - \sum_{i=1}^{12} \boldsymbol{e}_i^{(e)} + \sum_{i=1}^{6} \boldsymbol{f}_i^{(e)}, \tag{4.51}$$

with the usual nodal mapping

$$\boldsymbol{n}_i^{(e)} = N_l^{(\mathrm{1D})}(r^{(e)})\, N_m^{(\mathrm{1D})}(s^{(e)})\, N_n^{(\mathrm{1D})}(t^{(e)})\, \boldsymbol{X}_i^{(e,\mathrm{n})}, \tag{4.52}$$

the difference between the linear (straight) and the curved edge mapping

$$\boldsymbol{e}_i^{(\mathrm{e})} = \left(\boldsymbol{E}_i^{(e,\mathrm{curv.})}(r_a) - \boldsymbol{E}_i^{(e,\mathrm{lin.})}(r_a)\right) \frac{(1 - r_b)\,(1 - r_c)}{4} \tag{4.53}$$

and the corresponding difference between face mappings

$$\boldsymbol{f}_i^{(e)} = \left(\boldsymbol{F}_i^{(e,\mathrm{curv.})}(r_a, r_b) - \boldsymbol{F}_i^{(e,\mathrm{lin.})}(r_a, r_b)\right) \frac{(1 - r_c)}{2}. \tag{4.54}$$

The relation between the index i and the indices l, m and n in (4.52) as well as the relation between the general face and edge coordinates r_a, r_b, r_c, and the element coordinates $r^{(\mathrm{e})}, s^{(\mathrm{e})}, t^{(\mathrm{e})}$ depends on the numbering of nodes,

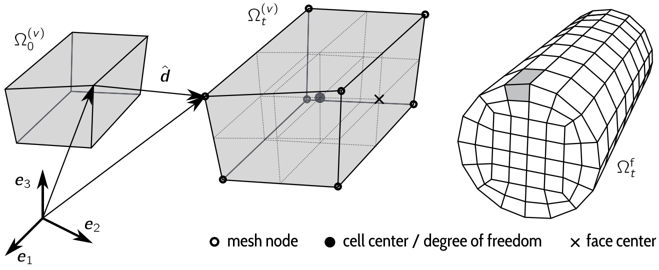


Figure 4.3: Discretization of the computational domain Ω_t^{f} with a finite volume mesh.

edges, and faces. A possible choice is given in Fig. A.3.1, the resulting relations are given in Tab. A.3.1, Tab. A.3.2, Tab. A.3.3. The above considerations involve parameterizations of the edge and face geometry $\boldsymbol{E}_e^{(e)}$ and $\boldsymbol{F}_f^{(\mathrm{e})}$, respectively. They are constructed using Lagrange polynomials, such that

$$\boldsymbol{E}_i^{(e,\mathrm{curv.})} = \sum_{j=0}^{p^{\mathrm{g}}} L_j^{(p^{\mathrm{g}},\mathrm{Lag.})}(r_a^{(e)})\boldsymbol{X}_i^{(e,\mathrm{e})} \tag{4.55}$$

and

$$\boldsymbol{F}_i^{(e,\mathrm{curv.})} = \sum_{i=0}^{p^{\mathrm{g}}}\sum_{j=0}^{p^{\mathrm{g}}} L_i^{(p^{\mathrm{g}},\mathrm{Lag.})}(r_a^{(e)})L_j^{(p^{\mathrm{g}},\mathrm{Lag.})}(r_b^{(e)})\boldsymbol{X}_{ij}^{(e,\mathrm{f})}. \tag{4.56}$$

Due to (4.50), the weights $\boldsymbol{X}_i^{(e,\mathrm{e})}$ and $\boldsymbol{X}_{ij}^{(e,\mathrm{f})}$ can be identified as the points on the geometry that are interpolated.

4.1.2 Finite volumes for fluid mechanics in moving domains

To solve the fluid mechanics subproblem (3.109–3.113), the finite volume method (FVM) is applied. As in the FEM, the discretization process starts with the definition of a computational mesh as illustrated in Fig. 4.3, such that the domain is approximated by the union of n^{v} finite volumes, i.e.

$$\Omega_t^{\mathrm{f}} \approx \bigcup_{v=1}^{n^{\mathrm{v}}} \Omega_t^{(v)}. \tag{4.57}$$

For the FVM considered here, the degrees of freedom correspond to the values of the unknowns at the finite volume's centers. These degrees of freedom for pressure and velocity are collected separately in the global vectors

$$\mathbf{p}_k = \begin{bmatrix} p_k^{(1)} & \dots & p_k^{(n^\mathrm{v})} \end{bmatrix}^\mathrm{T} \tag{4.58}$$

$$\mathbf{v}_k = \begin{bmatrix} \boldsymbol{v}_k^{(1)\mathrm{T}} & \dots & \boldsymbol{v}_k^{(n^\mathrm{v})\mathrm{T}} \end{bmatrix}^\mathrm{T}. \tag{4.59}$$

By rewriting the Navier-Stokes equations apparent in the IBVP (3.109–3.113) as

$$\mathrm{div}(\boldsymbol{v}) = 0, \tag{4.60}$$

$$\dot{\boldsymbol{v}} + \mathrm{div}(\boldsymbol{v} - \hat{\boldsymbol{v}})\,\boldsymbol{v} - \mathrm{div}(\nu\,\mathrm{grad}(\boldsymbol{v})) = \mathrm{div}\Big(\nu\,\mathrm{grad}(\boldsymbol{v})^\mathrm{T}\Big) + \boldsymbol{b} - \frac{1}{\rho}\,\mathrm{grad}(p)\,, \tag{4.61}$$

a discretization is constructed for each term separately. To this end, differentiation schemes known from the finite difference method (FDM) are applied in combination with interpolation schemes, as explained in the following. Further, the non-linearity of the momentum equation (4.61) and the disappearance of the pressure in (4.60) demand for a special solution algorithm, namely a *pressure-correction method*, which will be outlined at the end of this section.

Differentiation and interpolation

The finite volume approach starts by integrating the Navier-Stokes equations over the domain Ω_t^f and splitting the resulting integral into integrals over the individual finite volumes or cells. Considering the time step k, these may then be approximated using a single quadrature point at the cell center. For a general quantity ω,

$$\int\limits_{\Omega_{t_k}^\mathrm{f}} \omega_k \,\mathrm{d}\Omega_t^\mathrm{f} = \sum_{c=1}^{n^\mathrm{v}} \int\limits_{\Omega_{t_k}^{(v)}} \omega_k \,\mathrm{d}\Omega_{t_k}^{(v)} \approx \sum_{v=1}^{n^\mathrm{v}} \omega_k^{(v)} V_k^{(v)}, \tag{4.62}$$

is obtained, where $V_k^{(v)}$ is the cell volume. If the term contains a divergence (cf. (3.109) and the convective and the diffusive term in (4.61)), the volume integral is transformed into a surface integral using the divergence theorem.

The integrals over the cell faces are then again approximated using a single quadrature point at the face center, i.e.

$$\int_{\Omega_{t_k}^{(v)}} \operatorname{div}(\boldsymbol{\omega}_k)\, \mathrm{d}\Omega_t^{(v)} = \sum_{f\in F_v} \int_{\Gamma_{t_k}^{(f)}} \boldsymbol{\omega}_k \cdot \boldsymbol{n}_k^{(f)} \mathrm{d}\Gamma_t^{(f)} \approx \sum_{f\in F_v} \boldsymbol{\omega}_k \cdot \boldsymbol{n}_k^{(f)}\, S_k^{(f)} \tag{4.63}$$

with the face area $S_k^{(f)}$, the face unit normal $\boldsymbol{n}_k^{(f)}$ and F_v, the set of indices of faces adjacent to the volume v. The values at the faces are then represented by interpolating the degrees of freedom of neighboring cells, as detailed in [148]. Finally, spatial derivatives are approximated using differentiation schemes from the FDM, which again involve the degrees of freedom of neighboring cells.

Starting with the continuity equation (4.60) and performing the steps outlined above, one obtains

$$\operatorname{div}(\boldsymbol{v}_k) \to \sum_{f\in F_v} \boldsymbol{v}_k^{(f)} \cdot \boldsymbol{n}_k^{(f)}\, A_k^{(f)} \tag{4.64}$$

for cell c. The same process is applicable to the convective term in the momentum equation (4.61) and yields

$$\operatorname{div}(\boldsymbol{v}_k - \hat{\boldsymbol{v}}_k)\, \boldsymbol{v}_k \to \sum_{f\in F_v} \left(\boldsymbol{v}_k^{(f)} - \hat{\boldsymbol{v}}_k^{(f)}\right) \cdot \boldsymbol{n}_k^{(f)}\, A_k^{(f)}\, \boldsymbol{v}_k^{(f)}. \tag{4.65}$$

The diffusion term on the left-hand side in (4.61) is similarly treated, leading to

$$\operatorname{div}(\nu_k \operatorname{grad}(\boldsymbol{v}_k)) \to \sum_{f\in F_v} \nu_k^{(f)} \operatorname{grad}(\boldsymbol{v})^{(f)} \cdot \boldsymbol{n}_k^{(f)}\, A_k^{(f)}. \tag{4.66}$$

The diffusion term on the right-hand side in (4.61) is augmented by a stabilization term, as explained in [103], and then discretized.

$$\begin{aligned} &\operatorname{div}\left(\nu_k \operatorname{grad}(\boldsymbol{v}_k)^{\mathrm{T}}\right) \\ &\quad\to \operatorname{div}\left(\nu_k \operatorname{grad}(\boldsymbol{v}_k)^{\mathrm{T}} - \frac{1}{3}\,\nu_k \operatorname{tr}\left(\operatorname{grad}(\boldsymbol{v}_k)\right) \mathbf{1}\right) \\ &\quad\to \sum_{f\in F_v} \nu_k^{(f)} \left(\operatorname{grad}(\boldsymbol{v})^{\mathrm{T}(f)} - \frac{1}{3}\operatorname{tr}\left(\operatorname{grad}(\boldsymbol{v}_k)^{\mathrm{T}(f)}\right) \mathbf{1}\right) \cdot \boldsymbol{n}_k^{(f)}\, A_k^{(f)}. \end{aligned} \tag{4.67}$$

Due to the continuity equation and the fact that

$$\operatorname{tr}\left(\operatorname{grad}(\boldsymbol{v}_k)\right) = \operatorname{div}(\boldsymbol{v}_k)\,, \tag{4.68}$$

the stabilization term is non-zero only in the scope of the solution algorithm, where (4.67) is evaluated for a preliminary velocity field that is not divergence-free. The source term is simply represented by the values at the cell centers, giving

$$\boldsymbol{b} \to \boldsymbol{b}_k^{(v)}\, V_k^{(v)}. \tag{4.69}$$

The inertia term is discretized in time using the implicit Euler method, resulting in

$$\dot{\boldsymbol{v}}_k \approx \frac{\boldsymbol{v}_k - \boldsymbol{v}_{k-1}}{\Delta t_{k-1}} \to \left(\frac{\boldsymbol{v}_k^{(v)} - \boldsymbol{v}_{k-1}^{(v)}}{\Delta t_{k-1}} \right) V_k^{(v)}. \tag{4.70}$$

The pressure gradient term is discretized as

$$\operatorname{grad}(p_k) \to \sum_{f \in F_v} p_k^{(f)}\, \boldsymbol{n}_k^{(f)}\, A_k^{(f)}, \tag{4.71}$$

whereas for the terms involving the divergence of a vector field, the volume integral was transformed into a surface integral. In the scope of the pressure corrector method outlined next, the pressure gradient is also needed at the face centers. Its components are then computed using the degrees of freedom of the neighboring cells by

$$\Big((\operatorname{grad}(p_k))^{(f)} \Big)_i \to \frac{p_k^{(A)} - p_k^{(B)}}{d_i^{(AB)}}, \tag{4.72}$$

where $\boldsymbol{d}^{(AB)}$ denotes the distance vector between the centers of neighboring cells A and B. The velocity gradients at the faces in (4.66) and (4.67) are computed in an analogous way. It should be noted that for severely non-orthogonal meshes, a correction method has to be employed when discretizing gradients and convective terms. In *OpenFOAM*, the correction method proposed in [111] is available, however, the pressure correction method outlined in the following implicitly includes such a non-orthogonal corrector. For details on this issue, common to most finite volume methods, see [109].

Pressure correction method

In [148, 71], different pressure correction methods are explained. In general, when using what is called a collocated grid as opposed to a staggered grid, special attention has to be paid when discretizing pressure gradients. An issue known as the checkerboard problem has to be counteracted, which is usually done by applying a so-called *Rhie-Chow interpolation* [167] (cf. [148, p. 585]). In *OpenFOAM*, however, this remedy is not applied explicitly – as explained in [115] – but implicitly included in the solution algorithms. In the present work, the solver *pimpleDyMFoam* is used, which implements a solution algorithm called *PIMPLE*. It constitutes a combination of the well-known algorithms *SIMPLE (semi-implicit method for pressure linked equations)*, see [156, 34] and *PISO (pressure implicit with splitting of operators)* as proposed in [110]. Following [45, 103, 115], the main aspects of the algorithm are explained below.

By applying the introduced discretization schemes, a set of algebraic equations is obtained from the momentum equations (4.61). It may be stated as

$$\mathbf{A}_{k+1}\,\mathbf{u}_{k+1} = \mathbf{s}_{k+1} - \mathbf{G}_{k+1}\,\mathbf{p}_{k+1}, \tag{4.73}$$

where $\mathbf{A}_{k+1}\,\mathbf{u}_{k+1}$ represent all terms from the left hand side of (4.61), $\mathbf{G}_{k+1}\,\mathbf{p}_{k+1}$ denotes the discretized pressure gradient term and $\mathbf{s}_{k+1}$ collects all remaining terms. The discretization matrix $\mathbf{A}$ may be split into its diagonal part with coefficients A_{cc} and its off-diagonal parts with non-zero coefficients A_{cn}. With N_c denoting the set of cell indices representing neighboring cells of cell c, Eq. (4.73) can be written for a single cell as

$$A_{cc}\,\boldsymbol{u}^{(v)}_{k+1} = \left(\mathbf{s}^{(v)}_{k+1} - \sum_{n\in N_c} A_{cn}\,\boldsymbol{u}^{(n)}_{k+1}\right) - \operatorname{grad}(p)^{(v)}_{k+1}\,. \tag{4.74}$$

In the PIMPLE algorithm, improved values for the pressures and the velocities are computed in an iterative manner. Introducing the iteration index i, the first step in every iteration is to compute a preliminary velocity

$$\tilde{\tilde{\boldsymbol{u}}}^{(v)}_{i+1} = \frac{1}{\tilde{A}_{vv}}\left(\tilde{\mathbf{s}}^{(v)} - \sum_{n=1}^{N_n} \tilde{A}_{vn}\,\tilde{\boldsymbol{u}}^{(n)}_{i+1}\right) = \frac{\tilde{\boldsymbol{h}}^{(v)}_{i}}{\tilde{a}_{vv}} \tag{4.75}$$

which does not include a contribution from the pressure. This step is commonly denoted as a *momentum predictor*. In (4.75), the tilde marks quantities that are evaluated for the current trial velocity $\tilde{\boldsymbol{u}}_i$, which is known from the previous iteration. In the first iteration, $\tilde{\boldsymbol{u}}_0 = \boldsymbol{u}_k$ is used. By interpolating $\tilde{\tilde{\boldsymbol{u}}}^{(v)}_{i+1}$ to the face centers of the finite volume mesh, the preliminary face velocity $\tilde{\tilde{\boldsymbol{u}}}^{(f)}_{i+1}$ is obtained and a corresponding preliminary flux

$$\tilde{\tilde{\phi}}^{(f)}_{i+1} = \tilde{\tilde{\boldsymbol{u}}}^{(f)}_{i+1} \cdot \boldsymbol{n}^{(f)}_{k+1} \tag{4.76}$$

can be computed. Taking the divergence of Eq. (4.74), the left hand side is zero due to the continuity equation and it is possible to establish

$$\operatorname{div}\left(\frac{1}{\tilde{A}_{vv}}\,\operatorname{grad}(\tilde{p}_{i+1})^{(v)}\right) = \operatorname{div}\left(\tilde{\tilde{\boldsymbol{u}}}_{i+1}^{(v)}\right). \tag{4.77}$$

Equation (4.77) is called the *pressure correction equation.* The divergence theorem is applied to yield

$$\sum_{f\in F_v}\left(\left(\frac{1}{\tilde{A}_{cc}}\right)^{(f)}\operatorname{grad}(\tilde{p}_{i+1})^{(f)}\right)\cdot\boldsymbol{n}_{k+1}^{(f)} = \sum_{f\in F_v}\tilde{\phi}_{k+1}^{(f)}, \tag{4.78}$$

which can be solved for the pressure $\tilde{p}_{i+1}$ after applying the discretization (4.72). Finally, an improved trial velocity can be computed by the so-called *momentum corrector*

$$\tilde{\boldsymbol{u}}_{i+1}^{(v)} = \tilde{\boldsymbol{u}}_{i+1}^{(v)} - \frac{1}{\tilde{A}_{vv}}\operatorname{grad}(p)^{(v)} \tag{4.79}$$

after applying the discretization (4.71), and a new iteration is started with the momentum predictor (4.75). Once the velocity correction, i.e. $\|\tilde{\mathbf{u}}_{i+1} - \tilde{\mathbf{u}}_i\|$, becomes sufficiently small after N_k iterations, the values are considered converged and the velocity in the current time step is set to $\mathbf{u}_{k+1} = \tilde{\mathbf{u}}_{N_k}$.

4.1.3 Taylor-Galerkin method for one-dimensional blood flow

The considerations in this section follow those in [179] and [89]. Recalling the hyperbolic differential equation (3.168), it can be rewritten as

$$\dot{\boldsymbol{Q}} = \frac{\partial \boldsymbol{Q}}{\partial t} = \boldsymbol{B} - \frac{\partial \boldsymbol{G}}{\partial x} \tag{4.80}$$

and

$$\ddot{\boldsymbol{Q}} = \frac{\partial^2 \boldsymbol{Q}}{\partial t^2} = \frac{\partial \boldsymbol{B}}{\partial \boldsymbol{Q}}\,\frac{\partial \boldsymbol{Q}}{\partial t} - \frac{\partial}{\partial x}\left(\boldsymbol{H}\,\frac{\partial \boldsymbol{Q}}{\partial t}\right) \tag{4.81}$$

can be established with

$$\boldsymbol{H} = \frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}} = \begin{bmatrix} 0 & 1 \\ -\alpha\left(\frac{Q^{1\mathrm{D}}}{A^{1\mathrm{D}}}\right)^2 + \frac{\beta}{2\,\rho\,A_0^{1\mathrm{D}}}\,\sqrt{A^{1\mathrm{D}}} & 2\,\alpha\frac{Q^{1\mathrm{D}}}{A^{1\mathrm{D}}} \end{bmatrix}. \tag{4.82}$$

A second order *Taylor expansion* yields the semi-discrete system

$$
\begin{aligned}
\boldsymbol{Q}_{k+1} &\approx \boldsymbol{Q}_k + \Delta t_k\, \dot{\boldsymbol{Q}} + \frac{\Delta t_k}{2}\, \ddot{\boldsymbol{Q}} \\
&= \boldsymbol{Q}_k - \Delta t_k \left(\boldsymbol{B}_k - \frac{\partial \boldsymbol{G}_k}{\partial x} \right) \\
&+ \frac{\Delta t_k^2}{2} \left(\frac{\partial \boldsymbol{B}_k}{\partial \boldsymbol{U}} \left(\boldsymbol{B}_k - \frac{\partial \boldsymbol{G}_k}{\partial x} \right) - \frac{\partial \left(\boldsymbol{H}_k\, \boldsymbol{B}_k \right)}{\partial x} + \frac{\partial}{\partial x} \left(\boldsymbol{B}_k\, \frac{\partial \boldsymbol{G}_k}{\partial x} \right) \right)
\end{aligned}
\tag{4.83}
$$

which can be rewritten as

$$
\begin{aligned}
\boldsymbol{Q}_{k+1} = \boldsymbol{Q}_k - \Delta t_k\, \frac{\partial}{\partial x} \left(\boldsymbol{G}_k + \frac{\Delta t_k}{2}\, \boldsymbol{H}_k\, \boldsymbol{B}_k \right) \\
- \frac{\Delta t_k^2}{2} \left(\frac{\partial \boldsymbol{B}_k}{\partial \boldsymbol{U}}\, \frac{\partial \boldsymbol{G}_k}{\partial x} - \frac{\partial}{\partial x} \left(\boldsymbol{H}_k\, \frac{\partial \boldsymbol{G}_k}{\partial x} \right) \right) + \Delta t_k \left(\boldsymbol{B}_k + \frac{\Delta t_k}{2}\, \frac{\partial \boldsymbol{B}_k}{\partial \boldsymbol{U}}\, \boldsymbol{B}_k \right) .
\end{aligned}
\tag{4.84}
$$

Equation (4.84) serves as a basis for the *Galerkin method* used to find approximate solutions for $\boldsymbol{Q}_k(x)$. A weak form is constructed by multiplying (4.84) by a test function $\delta\boldsymbol{Q}$ and integrating the result over the domain Ω^{v}, which yields

$$
\begin{aligned}
\int\limits_0^L \left(\boldsymbol{Q}_{k+1} - \boldsymbol{Q}_k \right) \cdot \delta\boldsymbol{Q}\, \mathrm{d}x = \Delta t_k \int\limits_0^L \hat{\boldsymbol{B}}_k \cdot \delta\boldsymbol{Q}\, \mathrm{d}x - \frac{\Delta t_k^2}{2} \int\limits_0^L \frac{\partial \boldsymbol{B}_k}{\partial \boldsymbol{U}}\, \frac{\partial \boldsymbol{G}_k}{\partial x} \cdot \delta\boldsymbol{Q}\, \mathrm{d}x \\
+ \frac{\Delta t_k^2}{2} \int\limits_0^L \boldsymbol{H}_k\, \frac{\partial \boldsymbol{G}_k}{\partial x} \cdot \frac{\partial \delta\boldsymbol{Q}}{\partial x}\, \mathrm{d}x + \Delta t_k \int\limits_0^L \hat{\boldsymbol{G}}_k \cdot \frac{\partial \delta\boldsymbol{Q}}{\partial x}\, \mathrm{d}x
\end{aligned}
\tag{4.85}
$$

with

$$
\begin{aligned}
\hat{\boldsymbol{G}}_k &= \mathbf{G}_k + \frac{\Delta t_k}{2}\, \boldsymbol{H}_k\, \boldsymbol{B}_k, \\
\hat{\boldsymbol{B}}_k &= \mathbf{B}_k + \frac{\Delta t_k}{2}\, \frac{\partial \boldsymbol{B}_k}{\partial \boldsymbol{Q}}\, \boldsymbol{B}_k.
\end{aligned}
\tag{4.86}
$$

The terms in the second row of (4.85) were further integrated by parts to shift the derivatives to the test function as detailed in [89, p. 44]. According to the Galerkin approach, an algebraic system of equations is created from (4.85) by replacing the unknown $\boldsymbol{Q}_k$ as well as the test function $\delta\boldsymbol{Q}_k$ by a weighted sum of shape functions.

Discretized governing equations

As shape functions, the well-known linear shape functions from the finite-element method (see (4.44)) are used. Due to the explicit nature of (4.85), the area A^{1D}_{k+1} and the flux Q^{1D}_{k+1} can be computed separately. Denoting by $\mathbf{A}$ and $\delta\mathbf{A}$, respectively $\mathbf{Q}$ and $\delta\mathbf{Q}$ the corresponding vectors of weights – in this case the nodal values – the spatial discretization of (4.85) yields

$$\delta\mathbf{A}^{\mathrm{T}}\mathbf{M}\,\mathbf{A}_{k+1} = \delta\mathbf{A}^{\mathrm{T}}\,\mathbf{R}^{(A)}_k \tag{4.87}$$

for the first component and

$$\delta\mathbf{Q}^{\mathrm{T}}\mathbf{M}\,\mathbf{Q}_{k+1} = \delta\mathbf{Q}^{\mathrm{T}}\,\mathbf{R}^{(Q)}_k \tag{4.88}$$

for the second component, which can be solved for the new state after eliminating $\delta\mathbf{Q}$ and $\delta\mathbf{A}$. As usual, the global mass matrix $\mathbf{M}$ and the global right hand side vectors $\mathbf{R}^{(Q)}_k$ and $\mathbf{R}^{(A)}_k$ are assembled from their counterparts on element level $\mathbf{M}^{(e)}$ and $\mathbf{R}^{(e)}$. The expressions for these element matrices are given in Appendix A.4. Here, it shall only be noted that the development of efficient numerical discretization schemes for hyperbolic problems like that of one-dimensional viscous flow in a distensible vessel considered here constitutes an active field of research. As pointed out in [179], certain terms of the right hand side vectors are underintegrated or projected onto the finite element space to ensure stability and consistency of the method. However, a detailed analysis of the method is out of the scope of this work, as it will only be used to demonstrate the coupling of a three-dimensional cardiovascular FSI simulation with this reduced order model.

Boundary conditions, typically a given flow Q^{1D} at $x = 0$ and a given pressure p^{1D} at $x = L$ (resulting in a given area A^{1D}), are enforced by moving the corresponding degrees of freedom to the right hand side. In [179], other types of boundary conditions such as those to be prescribed when connecting multiple one-dimensional vessel models to describe a vessel network are discussed in detail. As mentioned in Section 3.3.3, a characteristic analysis of the system (3.168) is necessary to obtain expressions for non-reflecting or partially reflecting boundary conditions in this case.

4.1.4 Solvers for ordinary differential equations

The windkessel model as well as the elastic foundation used to model the surrounding tissue are described by ordinary differential equations (ODEs).

In this section, suitable solution methods and the resulting discretized equations are introduced.

Windkessel models

The windkessel models from Section 3.3.4 are discretized using the backward differences. The temporal derivatives of flow and pressure are accordingly discretized as

$$\left.\frac{\mathrm{d}Q^{\mathrm{w}}}{\mathrm{d}t}\right|_{t=t_{k+1}} \approx \frac{Q^{\mathrm{w}}_{k+1} - Q^{\mathrm{w}}_{k}}{\Delta t_k} \quad \text{and} \quad \left.\frac{\mathrm{d}p^{\mathrm{w}}}{\mathrm{d}t}\right|_{t=t_{k+1}} \approx \frac{p^{\mathrm{w}}_{k+1} - p^{\mathrm{w}}_{k}}{\Delta t_k}. \tag{4.89}$$

Second order derivatives as apparent in (3.172) are discretized as

$$\left.\frac{\mathrm{d}^2 Q^{\mathrm{w}}}{\mathrm{d}t^2}\right|_{t=t_{k+1}} \approx \frac{Q^{\mathrm{w}}_{k+1} - 2\,Q^{\mathrm{w}}_{k} + Q^{\mathrm{w}}_{k-1}}{\Delta t_k\,\Delta t_{k-1}}. \tag{4.90}$$

Inserting this approximation in the ODE (3.170) that governs the two-element windkessel model,

$$p^{\mathrm{w}}_{k+1} \overset{\text{2e}}{=} \frac{\Delta t_k\,R\,Q^{\mathrm{w}}_{k+1} + C\,R\,p^{\mathrm{w}}_{k}}{C\,R + \Delta t_k} \tag{4.91}$$

is obtained. For the three- and four-element windkessel models governed by (3.171) and (3.172),

$$p^{\mathrm{w}}_{k+1} \overset{\text{3e}}{=} \frac{C\,R_1\,p^{\mathrm{w}}_{k} + (R_1 + R_2)\,\Delta t_k\,Q^{\mathrm{w}}_{k+1} + C\,R_1\,R_2\left(Q^{\mathrm{w}}_{k+1} - Q^{\mathrm{w}}_{k}\right)}{C\,R_1 + \Delta t} \tag{4.92}$$

and

$$p^{\mathrm{w}}_{k+1} \overset{\text{4e}}{=} \frac{1}{L\,C\,(R_1\,C + \Delta t_k)}\left((R_1 + R_2)\,L\,C\,\Delta t_k\,Q^{\mathrm{w}}_{k+1}\right. \tag{4.93}$$

$$\left.+R_2\,(Q^{\mathrm{w}}_{k+1} - Q^{\mathrm{w}}_{k}) + R_1\,\Delta t_{k-1}^{-1}\,(Q^{\mathrm{w}}_{k+1} - 2\,Q^{\mathrm{w}}_{k} + Q^{\mathrm{w}}_{k-1})\right) \tag{4.94}$$

are obtained.

Viscoelastic foundation

In the scope of a coupled simulation, the viscoelastic foundation may be used in two different ways as mentioned in Section 3.3.5. If it is used to describe the influence of the surrounding tissue on an artery, it is coupled to a structural mechanics solver that delivers displacements at discrete locations. The foundation solver computes tractions at these locations and the equation to be solved may be written as

$$m^{\mathrm{t}}\,\ddot{\bar{\boldsymbol{d}}}^{\mathrm{t}} + d^{\mathrm{t}}\,\dot{\bar{\boldsymbol{d}}}^{\mathrm{t}} + c^{\mathrm{t}}\,\bar{\boldsymbol{d}}^{\mathrm{t}} = \boldsymbol{t}^{\mathrm{t}}. \tag{4.95}$$

The Newmark approximations (4.25) and (4.26) are used to discretize the temporal derivatives. Setting $\beta = {}^1\!/\!_4$ and $\gamma = {}^1\!/\!_2$ yields the *constant average acceleration method* and

$$\begin{aligned}\boldsymbol{t}^{\mathrm{t}}_{k+1} = m^{\mathrm{t}} &\left(\frac{4}{\Delta t_k^2} \left(\boldsymbol{d}^{\mathrm{t}}_{k+1} - \boldsymbol{d}^{\mathrm{t}}_k - \Delta t_k\, \dot{\boldsymbol{d}}^{\mathrm{t}}_k \right) - \ddot{\boldsymbol{d}}^{\mathrm{t}}_k \right) \\ + d^{\mathrm{t}} &\left(\frac{2}{\Delta t_k} \left(\boldsymbol{d}^{\mathrm{t}}_{k+1} - \boldsymbol{d}^{\mathrm{t}}_k \right) - \dot{\boldsymbol{d}}_k \right) + c^{\mathrm{t}} \boldsymbol{d}^{\mathrm{t}}_{k+1}\end{aligned} \tag{4.96}$$

for the tractions.

If the viscoelastic foundation is used as a reduced structural model, it is coupled to a fluid solver that delivers tractions at discrete locations. Euler methods are then used to compute the displacements at these locations. The equation to be solved for the displacement $\boldsymbol{d}$ at each location may then be written as

$$m^{\mathrm{t}}\,\ddot{\boldsymbol{d}}^{\mathrm{t}} + d^{\mathrm{t}}\,\dot{\boldsymbol{d}}^{\mathrm{t}} + c^{\mathrm{t}}\,\boldsymbol{d}^{\mathrm{t}} = \bar{\boldsymbol{t}}^{\mathrm{t}}. \tag{4.97}$$

Since the Euler methods are intended to be used to solve first order differential equations of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \tag{4.98}$$

a transformation of variables has to be performed. Setting $\mathbf{x} = \begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}\,\mathrm{T}} & \boldsymbol{d}^{\mathrm{t}\,\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$,

$$\begin{bmatrix} \ddot{\boldsymbol{d}}^{\mathrm{t}} \\ \dot{\boldsymbol{d}}^{\mathrm{t}} \end{bmatrix} = \begin{bmatrix} -\frac{d^{\mathrm{t}}}{m^{\mathrm{t}}} \boldsymbol{1} & -\frac{c^{\mathrm{t}}}{m^{\mathrm{t}}} \boldsymbol{1} \\ \boldsymbol{1} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}} \\ \boldsymbol{d}^{\mathrm{t}} \end{bmatrix} + \begin{bmatrix} \frac{1}{m^{\mathrm{t}}} \bar{\boldsymbol{t}}^{\mathrm{t}} \\ \boldsymbol{0} \end{bmatrix} \tag{4.99}$$

is obtained from (4.97), which matches the form of (4.98). Now, the *explicit Euler method* is obtained by considering the equilibrium equations at time t_k and computing the new state at time t_{k+1} by $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t_k\, \mathbf{f}(\mathbf{x}_k)$, which leads to

$$\begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}}_{k+1} \\ \boldsymbol{d}^{\mathrm{t}}_{k+1} \end{bmatrix} = \begin{bmatrix} \left(1 - \frac{\Delta t_k\, d^{\mathrm{t}}}{m^{\mathrm{t}}}\right) \boldsymbol{1} & -\frac{\Delta t_k\, c^{\mathrm{t}}}{m^{\mathrm{t}}} \boldsymbol{1} \\ \Delta t_k\, \boldsymbol{1} & \boldsymbol{1} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}}_k \\ \boldsymbol{d}^{\mathrm{t}}_k \end{bmatrix} + \begin{bmatrix} \frac{1}{m^{\mathrm{t}}} \boldsymbol{t}^{\mathrm{t}}_k \\ \boldsymbol{0} \end{bmatrix}. \tag{4.100}$$

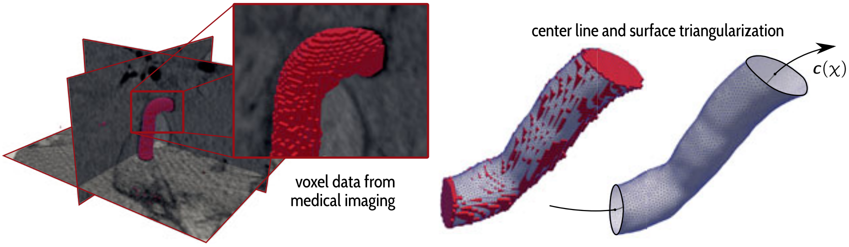


Figure 4.4: Construction of surface triangularizations from medical imaging data.

The *implicit Euler method* is obtained by considering the equilibrium equations at time t_{k+1} and computing the new state by $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t_k \, \mathbf{f}(\mathbf{x}_{k+1})$. This leads to

$$\begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}}_{k+1} \\ \boldsymbol{d}^{\mathrm{t}}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 + \frac{\Delta t_k \, d^{\mathrm{t}}}{m^{\mathrm{t}}} \boldsymbol{1} & \frac{\Delta t_k \, c^{\mathrm{t}}}{m^{\mathrm{t}}} \boldsymbol{1} \\ -\Delta t_k & \boldsymbol{1} \end{bmatrix}^{-1} \left(\begin{bmatrix} \dot{\boldsymbol{d}}^{\mathrm{t}}_{k} \\ \boldsymbol{d}^{\mathrm{t}}_{k} \end{bmatrix} + \begin{bmatrix} \frac{\Delta t_k}{m^{\mathrm{t}}} \boldsymbol{t}^{\mathrm{t}}_{k+1} \\ \boldsymbol{0} \end{bmatrix} \right) . \qquad (4.101)$$

4.2 Geometry and mesh generation

In order to realize simulations of cardiovascular fluid-structure interaction problems, computational meshes are needed for the fluid and the structural subproblem. A popular pipeline for mesh generation starts with medical images, or more precisely, voxel data. This data (a three-dimensional gray scale image) is then segmented, i.e. regions of different tissue are identified on a per-voxel basis. Considering the example depicted in Fig. 4.4, a threshold is defined to extract only those voxels representing the inside of the blood vessel. The resulting voxel geometry must be smoothed to obtain a realistic geometry of a blood vessel. The marching cubes algorithm is a method often applied in this step. From the resulting triangularization of the geometry's boundary, standard techniques can be applied to remesh the surface triangulations and to finally create a tetrahedral mesh. To this end, the free software *VMTK* (The vascular modeling toolkit, [139, 7]) is used, which implements all steps from reading the voxel data to writing out an unstructured mesh with a boundary layer for the fluid subproblem. The triangularization of the wet surface may be extruded to obtain a low-order finite element mesh for the structural mechanics subproblem.

In the context of high-order finite elements, to make use of the surface triangularization, it must be reparameterized. One possibility to carry out the

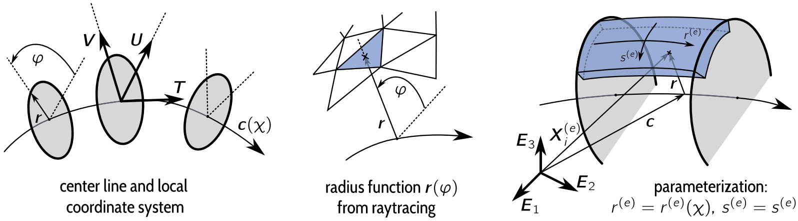

φ
V
U
T
r
c(χ)
center line and local coordinate system
φ
r
radius function r(φ) from raytracing
$r^{(e)}$
$s^{(e)}$
r
c
E_3
$X_i^{(e)}$
E_2
E_1
parameterization: $r^{(e)} = r^{(e)}(\chi)$, $s^{(e)} = s^{(e)}(\varphi)$

Figure 4.5: Parameterization of vessel geometries by a centerline and a radius function.

parameterization is based on a centerline $\boldsymbol{c}(\chi)$ (a curve in three-dimensional space) and a radius function $\boldsymbol{r}(\chi, \varphi)$, as illustrated in Fig. 4.5. In combination with a local coordinate system $(\boldsymbol{T}, \boldsymbol{U}, \boldsymbol{V})$, the centerline and the radius functions uniquely define the vessel surface. Idealized geometries may be defined by choosing a simple radius function, e.g.

$$\boldsymbol{r}(\chi, \varphi) = R\left(\cos(\varphi)\,\boldsymbol{U}(\chi) + \sin(\varphi)\,\boldsymbol{V}(\chi)\right) \tag{4.102}$$

for a vessel with a constant circular cross section with radius R. However, the radius function can also be established from the surface triangularization obtained from the voxel data. To this end, a raytracing algorithm is used to evaluate $\boldsymbol{r}(\chi, \varphi)$. With any definition of the radius functions, the sampling points $\boldsymbol{X}_i^{(e,\mathrm{n})}$, $\boldsymbol{X}_i^{(e,\mathrm{e})}$, $\boldsymbol{X}_i^{(e,\mathrm{f})}$ needed in the scope of the quasi regional mapping technique (see Section 4.1.1) can be obtained using a relation between the local element coordinates, e.g. $r^{(e)}$ and $s^{(e)}$, and the coordinates χ and φ, as shown in Fig. 4.5. This reparameterization as well as the algorithm used to transport the local basis vectors $\boldsymbol{U}$ and $\boldsymbol{V}$ along the curve were implemented in the meshing software *tubeMesh*, which was introduced in [161].

It is generally possible to extend this geometry definition by a centerline and a radius function to cope with bifurcations by introducing individual basis vectors $\boldsymbol{U}_1$, $\boldsymbol{U}_2$, $\boldsymbol{U}_3$, $\boldsymbol{U}_4$ for each quadrant in the local $\boldsymbol{U}$-$\boldsymbol{V}$-plane (see Fig. 4.6). However, the resulting mesh topologies are fixed and may not be suited for arbitrary geometries. For such cases, a more flexible mesh generation pipeline was developed. It starts from a surface description that can be provided by human experts to mimic average or extreme cases that are to be studied more closely. Different from the patient-specific geometries obtained from medical images, a corresponding geometry description should be intuitive and require a limited set of parameters. As

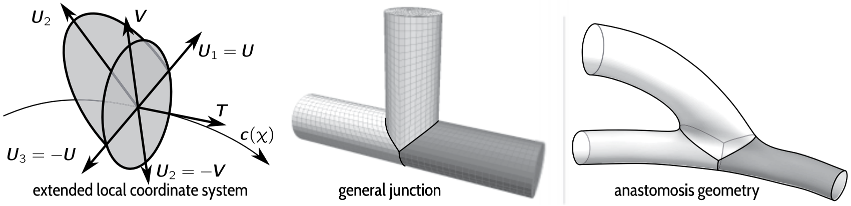


Figure 4.6: Meshing of bifurcations using centerline and radius function in combination with an extended local coordinate system.

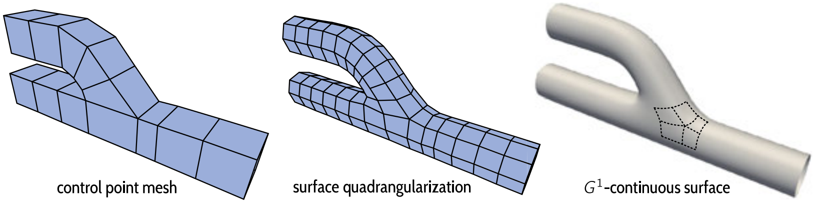


Figure 4.7: Geometry definition using a coarse control point mesh, a subdivision scheme and a construction scheme for smooth quad patches.

shown in Fig. 4.7, the description is based on a coarse input mesh, which is first refined (and smoothed) using a subdivision scheme. The well-known Catmull-Clark subdivision scheme proposed in [37] is used in this step. The resulting quadrangularization is still coarse with respect to a low-order finite element mesh. For high-order finite elements, however, it provides a reasonable element size. To obtain realistic geometries, the quadrangularization is smoothed, which is done using a surface construction scheme that yields G^1-continuous geometries by definition. The main difficulty when deriving according schemes is the handling of irregular nodes, i.e. nodes where fewer or more than four edges join. The introduction of two such construction schemes constitutes the main part of this section. Additional steps on the way from the geometry definition to the computational meshes are summarized afterwards.

4.2.1 G^1-continuous surface construction

The purpose of the construction schemes explained here is to establish a smooth surface parameterization based on a quadrilateral mesh. The topology of the mesh is regarded as fixed. In addition to the nodes of this control point mesh, the normals at the nodes need to be specified. Figure 4.8 illustrates the input and output of the construction scheme for a simple quadrangularization that represents three faces of a cube. In view of the application as a design tool and as the basis for a high-order finite element geometry, certain requirements for the scheme can be formulated.

- The construction scheme should produce smooth geometries for arbitrary mesh topologies. No restriction should be made on the node valence (the number of edges joining at a node).
- The resulting geometry should be *polynomial* within each quadrilateral. This allows to use the geometry to create an isoparametric finite element formulation that does not introduced further discretization errors in terms of the shape of the considered domain.
- The polynomial order of the geometry p^{g} should be as low as possible such that isoparametric or subparametric element formulations can be obtained for low shape function orders p.
- The surface construction scheme should be *tunable*, i.e. it should allow for a modification of the geometry by an adjustment of global and/or local parameters that can be specified in addition to the provided quadrangularization.
- The construction scheme should be *local*, i.e. changing the position of a node should only affect the geometry of the surface close to the node. In the best case, only those quadrilaterals adjacent to the node are influenced. The locality allows for an efficient recomputation when interacting with the mesh in the design process and ensures a certain degree of intuitivity.

Two surface construction schemes are explained that meet these requirements to a reasonable extent. Before going into the details of the schemes, common means regarding the parameterization and regularization of the surface geometry are introduced.

Tensor product quad patches

A common method to parameterize arbitrary surfaces in three dimensions is to divide them into quadrilaterals, which are often referred to as *quad*

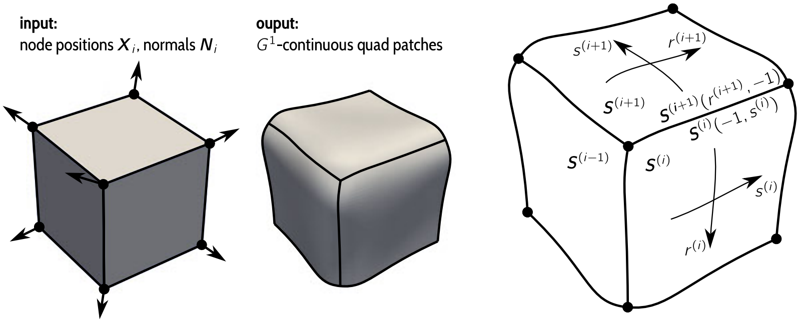

input:
node positions $\boldsymbol{X}_i$, normals $\boldsymbol{N}_i$
ouput:
G^1-continuous quad patches
$s^{(i+1)}$
$r^{(i+1)}$
$\boldsymbol{S}^{(i+1)}$
$\boldsymbol{S}^{(i+1)}(r^{(i+1)}, -1)$
$\boldsymbol{S}^{(i)}(-1, s^{(i)})$
$\boldsymbol{S}^{(i-1)}$
$\boldsymbol{S}^{(i)}$
$s^{(i)}$
$r^{(i)}$

Figure 4.8: Parameterization of a surface by quad patches.

patches. The surface can be described as the union of all patches, i.e.

$$\boldsymbol{S} = \bigcup_{i=1}^{n^{\mathrm{p}}} \boldsymbol{S}^{(i)}. \tag{4.103}$$

A standard parameterization by local coordinates $r^{(i)}$ and $s^{(i)}$ is chosen for each patch $\boldsymbol{S}^{(i)}$. This allows for a straightforward description of the patch's geometry using tensor product-type basis functions, such that on each patch

$$\boldsymbol{S}^{(i)} = \sum_{k=0}^{p^{\mathrm{g}}} \sum_{j=0}^{p^{\mathrm{g}}} L_j(r^{(i)})\, L_k(s^{(i)})\, \boldsymbol{P}_{jk}^{(i)} = \sum_{l=0}^{n} N_l(r^{(i)}, s^{(i)})\, \boldsymbol{P}_l^{(i)}. \tag{4.104}$$

If the one-dimensional basis functions L are chosen to be of polynomial type, then the basis functions N_l form a basis for the space of bilinear surfaces ($p = 1$), biquadratic surfaces ($p = 2$), etc. This means that, if used in combination with the tensor product space introduced in Section 4.1.1 of the same polynomial degree $p = p^{\mathrm{g}}$, the resulting element formulation is isoparametric or subparametric. However, if used in combination with the trunk space or the serendipity space, undesired superparametric element formulations may result even for $p^{\mathrm{g}} < p$.

For neighboring patches, it can occur that $r^{(i)} = s^{(i+1)}$ at the common edge as depicted in Fig. 4.8 and exemplary assumed in the following. Different orientations of the patches will of course lead to different relations, however, any scenario can always be transformed into this one by rotation.

To ease the notation,

$$\boldsymbol{S}_r^{(i)} = \frac{\partial \boldsymbol{S}^{(i)}}{\partial r^{(i)}} \qquad \boldsymbol{S}_s^{(i)} = \frac{\partial \boldsymbol{S}^{(i)}}{\partial s^{(i)}} \tag{4.105}$$

are introduced for the partial derivative of $\boldsymbol{S}(i)$. Higher order derivatives are abbreviated in a corresponding manner.

G^1-continuity constraints

The partitioned surface description $\boldsymbol{S}$ should be *smooth*, which is mathematically best described by the requirement that it is based on a function space of a certain continuity. For example, if one requires that $\boldsymbol{S} \in C^0$, the surface must not have holes. This can be stated as

$$\boldsymbol{S}^i \left(-1, s^{(i)}\right) - \boldsymbol{S}^{(i+1)} \left(r^{(i+1)}, -1\right) = \boldsymbol{0}. \tag{4.106}$$

If $\boldsymbol{S} \in \boldsymbol{C}^1$ is required, the surface must not have kinks, i.e. be smooth as desired. While the requirement that $\boldsymbol{S} \in C^0$ can be directly used in practice, the requirement that $\boldsymbol{S} \in C^1$ is too strict as it is not independent of the parameterization of the individual patches. In the direction of a common edge as the one depicted in Fig. 4.8,

$$\boldsymbol{S}_s^{(i)} \left(-1, s^{(i)}\right) - \boldsymbol{S}_r^{(i+1)} \left(r^{(i)}, -1\right) = \boldsymbol{0} \tag{4.107}$$

can be used, resulting in C^1-continuity in that direction. Requiring the same equality for the derivatives with respect to the patch coordinates that run orthogonal to the common edge would ensure that $\boldsymbol{S} \in C^1$. However, in combination with a low-dimensional basis for the patch descriptions $\boldsymbol{S}^{(i)}$ this requirement is too strong and will not lead to pleasing discretizations. Further, for irregular meshes with node valences other than four, the C^0-constraint along patch boundaries is impossible to fulfill. Instead, the only requirement is that the derivatives with respect to the orthogonal coordinate together with the in-edge derivatives lie in a common plane. This can be expressed as

$$\Phi\, \boldsymbol{S}_r^{(i)} \left(-1, s^{(i)}\right) + \Theta\, \boldsymbol{S}_s^{(i+1)} \left(r^{(i)}, -1\right) + \boldsymbol{S}_s^{(i)} \left(-1, s^{(i)}\right) = 0, \tag{4.108}$$

where Φ and Θ are arbitrary functions of the common edge coordinate $s^{(i)}$ or $r^{(i+1)}$. Then, the resulting surface is G^1-continuous The polynomial degree p^{g} must of course be chosen high enough such that the constraints

(4.106-4.108) can be fulfilled. For a regular quadrangularization, where each internal node is adjacent to exactly four patches, this is possible for $p^{\mathrm{g}} \geq 2$. However, for irregular quadrangularizations, where the valence of at least one node is not equal to four, $p^{\mathrm{g}} = 2$ is not sufficient. The derivation of the minimal p^{g} that allows for the G^1-continuity constraints to be fulfilled has been addressed in many works and, according to [27], has not yet been found for the general case. A detailed explanation is out of the scope of this work and ceded to [27, 180, 24] and the references cited therein.

4.2.2 Polynomial G^1 PN quads

In [155], a construction scheme for G^1-continuous surface is proposed. It uses biquintic polynomial patches, which are termed *polynomial G^1-PN quads* in reference to the general concept of PN patches, see e.g. [218]. Despite its higher order, it was found to be favorable over other polynomial construction schemes such as those proposed in [27, 88] due to a lower implementation effort and a more general applicability[1]. As a basis, the Bernstein polynomials with

$$L_j^{\mathrm{Bern.},p^{\mathrm{g}}}(r) = \frac{p!}{j!\,(p^{\mathrm{g}}-j)!} \left(\frac{r-1}{2}\right)^j \left(\frac{r-1}{2}\right)^{(p^{\mathrm{g}}-j)} \tag{4.109}$$

are used. The patch description may be stated in the form

$$\boldsymbol{S}^{(i)} = \mathbf{N}_{\mathrm{Bern.}}^{(p^{\mathrm{g}})}\, \mathbf{P}^{(i)}\, \mathbf{N}_{\mathrm{Bern.}}^{(p^{\mathrm{g}})}{}^{\mathrm{T}} \tag{4.110}$$

where

$$\mathbf{N}_{\mathrm{Bern.}}^{(p^{\mathrm{g}})} = \left[L_0^{\mathrm{Bern.},p^{\mathrm{g}}}, L_1^{\mathrm{Bern.},p^{\mathrm{g}}}, \ldots, L_{p^{\mathrm{g}}}^{\mathrm{Bern.},p^{\mathrm{g}}}\right] \tag{4.111}$$

is the one-dimensional interpolation matrix for the Bernstein basis of order p^{g} and $\mathbf{P}^{(i)}$ is the matrix or array of control points, such that each entry can be identified with a certain type of control point. From Fig. 4.8 and

[1] A construction scheme for bicubic patches is proposed in [88], and biquartic patches are considered in [27]. In both works, a refinement step precedes the surface construction, because nodes of arbitrary valence can only be handled by the schemes if they are surrounded by regular nodes (with valence four).

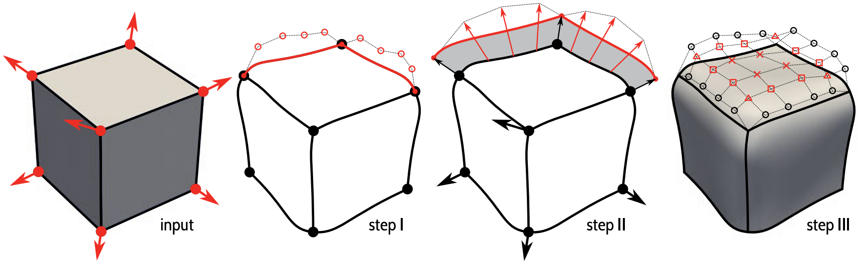


Figure 4.9: Steps in the construction scheme from [155]. From left to right: Input data (nodes and normals), edge curves, edge normal curves, patch interior.

Fig. 4.9, it can be observed that

$$\mathbf{P}^{(i)} = \begin{bmatrix} \boldsymbol{P}_{00}^{(i)} & \boldsymbol{P}_{01}^{(i)} & \dots & \boldsymbol{P}_{0p}^{(i)} \\ \boldsymbol{P}_{10}^{(i)} & \boldsymbol{P}_{11}^{(i)} & \dots & \boldsymbol{P}_{1p}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{p0}^{(i)} & \boldsymbol{P}_{p1}^{(i)} & \dots & \boldsymbol{P}_{pp}^{(i)} \end{bmatrix} \quad \text{corresponds to} \quad \begin{bmatrix} \bullet & \circ & \circ & \circ & \circ & \bullet \\ \circ & \triangle & \square & \square & \triangle & \circ \\ \circ & \square & \times & \times & \square & \circ \\ \circ & \square & \times & \times & \square & \circ \\ \circ & \triangle & \square & \square & \triangle & \circ \\ \bullet & \circ & \circ & \circ & \circ & \bullet \end{bmatrix}, \tag{4.112}$$

such that five different types of control points can be identified. The different steps of the construction scheme are outlined in the following, each step providing a computation rule for certain types of control points. The control points that correspond to the nodes of the quadrangularization as well as the normals at these points serve as an input to the construction scheme (see Figure 4.9). Taking these normals into account as a degree of freedom instead of making them inherent to the scheme is an example for tunability, which was introduced as a desired feature at the beginning of this section. It allows for a prescription of the normal for each node of the mesh independently. Further, there are many different algorithms to compute the normals based on the position of the surrounding nodes. Thus, to keep the number of input parameters small (another desired feature of the scheme) it is possible to automatically compute the normals at some nodes and to prescribe them explicitly at others.

Step I - construction of edge curves Given the normals and nodes of the control point mesh, curves describing the edge of the patches are computed first. After this step, the control points labeled by ∘ in (4.112) are fixed. Denoting the curve corresponding to edge j of patch i by $\boldsymbol{s}_j^{(i)}$ and the corresponding edge coordinate by $r_j^{(i)}$,

$$\begin{aligned} \boldsymbol{s}_1^{(i)} &= \boldsymbol{S}^{(i)}\left(r_1^{(i)}, -1\right), \quad \boldsymbol{s}_2^{(i)} = \boldsymbol{S}^{(i)}\left(r_2^{(i)}, 1\right), \\ \boldsymbol{s}_3^{(i)} &= \boldsymbol{S}^{(i)}\left(-1, r_3^{(i)}\right), \quad \boldsymbol{s}_4^{(i)} = \boldsymbol{S}^{(i)}\left(1, r_4^{(i)}\right). \end{aligned} \tag{4.113}$$

The control points for each $\boldsymbol{s}_j^{(i)}$ are denoted by $\boldsymbol{p}_k^{(i,j)}$, $k = 0, \ldots, 5$, where

$$\boldsymbol{p}_k^{(i,1)} = \boldsymbol{P}_{k0}, \quad \boldsymbol{p}_k^{(i,2)} = \boldsymbol{P}_{k5}, \quad \boldsymbol{p}_k^{(i,3)} = \boldsymbol{P}_{0k} \quad \boldsymbol{p}_k^{(i,4)} = \boldsymbol{P}_{5k}. \tag{4.114}$$

Due to the constraint that the resulting surface must interpolate the input mesh, the control points $\boldsymbol{p}_0^{(i,j)}$ and $\boldsymbol{p}_5^{(i,j)}$ are already fixed. The remaining edge control points are determined such that the boundary curves $\boldsymbol{s}_j^{(i)}$ are orthogonal to the given normal vectors. Denoting by $\boldsymbol{N}_1^{(i,j)}$ and $\boldsymbol{N}_2^{(i,j)}$ the normals at $r_j^{(i)} = -1$ and $r_j^{(i)} = 1$, respectively (i.e. the normals at the control points connected by $\boldsymbol{s}_j^{(i)}$), the orthogonality constraint can be written as

$$\frac{\partial \boldsymbol{s}_j^{(i)}(-1)}{\partial r_j^{(i)}} \cdot \boldsymbol{N}_1^{(i,j)} = 0, \quad \frac{\partial \boldsymbol{s}_j^{(i)}(1)}{\partial r_j^{(i)}} \cdot \boldsymbol{N}_2^{(i,j)} = 0. \tag{4.115}$$

The derivative of a Bernstein polynomial can again be represented as a Bernstein polynomial. In the present case,

$$\frac{\partial \boldsymbol{s}_j^{(i)}}{\partial r_j^{(i)}} = \frac{5}{2} \sum_{k=0}^{4} L_{4,k}^{\text{Bern.}} \left(\boldsymbol{p}_{k+1}^{(i,j)} - \boldsymbol{p}_k^{(i,j)}\right) \tag{4.116}$$

is obtained. Equation (4.115) can be reduced to constraints on the control points, which gives

$$\begin{aligned} \left(\boldsymbol{p}_1^{(i,j)} - \boldsymbol{p}_0^{(i,j)}\right) \cdot \boldsymbol{N}_1^{(i,j)} &= 0, \\ \left(\boldsymbol{p}_5^{(i,j)} - \boldsymbol{p}_4^{(i,j)}\right) \cdot \boldsymbol{N}_2^{(i,j)} &= 0. \end{aligned} \tag{4.117}$$

Restricting the differences between the interior nodes to be orthogonal to the given normal vectors eases the fulfillment of the G^1-constraints in the third step, so

$$\begin{aligned} \left(\boldsymbol{p}_2^{(i,j)} - \boldsymbol{p}_1^{(i,j)}\right) \cdot \boldsymbol{N}_1^{(i,j)} &= 0, \\ \left(\boldsymbol{p}_4^{(i,j)} - \boldsymbol{p}_3^{(i,j)}\right) \cdot \boldsymbol{N}_2^{(i,j)} &= 0 \end{aligned} \tag{4.118}$$

are used as additional constraints. The constraints in (4.117) and (4.118) provide four equations for the twelve unknown components of the curves' interior control points. In order to obtain a unique solution to this underdetermined system, it is augmented by a minimization problem. In [155], the following quantity is minimized:

$$\Pi^{\text{Papazov}} = \sum_{k=0}^{4} \left\| \boldsymbol{p}_{k+1}^{(i,j)} - \boldsymbol{p}_{k}^{(i,j)} \right\|^2 = \sum_{k=0}^{4} \left\| \boldsymbol{\Delta p}_{k}^{(i,j)} \right\|^2 . \tag{4.119}$$

However, to gain some control over the shape, another objective function is proposed here. The goal is to let the norm of the normal vectors $\|\boldsymbol{N}_1^{(i,j)}\|$ and $\|\boldsymbol{N}_2^{(i,j)}\|$ control the curvature of the surface, which can be realized by minimizing

$$\Pi^{\text{curve}} = \left\| \alpha \, \overline{\boldsymbol{\Delta p}}^{(i,j)} - \boldsymbol{\Delta p}_2^{(i,j)} \right\|^2 + \sum_{\substack{i=0 \\ i \neq 2}}^{4} \left\| \overline{\boldsymbol{\Delta p}}^{(i,j)} - \boldsymbol{\Delta p}_k^{(i,j)} \right\|^2 , \tag{4.120}$$

where

$$\overline{\boldsymbol{\Delta p}}^{(i,j)} = \frac{1}{5} \left(\boldsymbol{p}_5^{(i,j)} - \boldsymbol{p}_0^{(i,j)} \right) \tag{4.121}$$

and

$$\alpha = \frac{\left\| \boldsymbol{N}_1^{(i,j)} \right\| + \left\| \boldsymbol{N}_2^{(i,j)} \right\|}{\left\| \overline{\boldsymbol{\Delta p}}^{(i,j)} \right\|} . \tag{4.122}$$

With (4.120), a decrease of the norm of the normals will pull the geometry tighter to the input mesh, as shown in Fig. 4.10. The minimization of either objective function will lead to a linear system, such that it can be used directly to augment the linear system of equations using Lagrange multipliers. A general explanation of how the constraints are augmented by the minimization problem follows at the end of this section.

Step II - construction of edge normal curves In this step, the normal directions along all edges are fixed. While this does not directly yield any surface control points, it provides the basis for their construction in the next step. A boundary normal curve is described by a Bernstein polynomial

$$\boldsymbol{n}_j^{(i)} \left(r_j^{(i)} \right) = \sum_{k=0}^{4} L_{4,k}^{\text{Bern.}} \left(r_j^{(i)} \right) \boldsymbol{q}_k^{(i,j)} \tag{4.123}$$

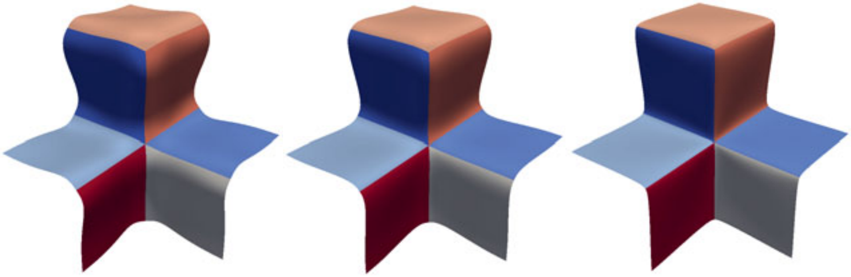


Figure 4.10: Influence of the norm of the given normals when using (4.120) in Step I of the construction scheme for G^1 PN quads from [155].

and should interpolate the normal direction at the nodes. This is ensured by requiring that

$$\boldsymbol{n}_j^{(i)}(-1) = \boldsymbol{q}_0^{(i,j)} = \alpha_0\,\boldsymbol{N}_1^{(i,j)} \quad \text{and} \quad \boldsymbol{n}_j^{(i)}(1) = \boldsymbol{q}_4^{(i,j)} = \alpha_4\,\boldsymbol{N}_2^{(i,j)}. \tag{4.124}$$

As in Step I, additional constraints on the normal curve's interior control points are added to ease the construction of the patch's interior control points in the third step, namely

$$\boldsymbol{q}_1^{(i,j)} = \alpha_1\,\boldsymbol{N}_1^{(i,j)} \quad \text{and} \quad \boldsymbol{q}_3^{(i,j)} = \alpha_3\,\boldsymbol{N}_2^{(i,j)}. \tag{4.125}$$

Since $\boldsymbol{n}_j^{(i)}$ should be orthogonal to $\boldsymbol{s}_j^{(i)}$,

$$\boldsymbol{n}_j^{(i)} \cdot \frac{\partial \boldsymbol{s}_j^{(i)}}{\partial r_j^{(i)}} = \left(\sum_{k=0}^{4} L_{4,k}^{\text{Bern.}}\,\boldsymbol{q}_k^{(i,j)}\right) \cdot \left(5\sum_{k=0}^{4} L_{4,k}^{\text{Bern.}}\left(\boldsymbol{p}_{k+1}^{(i,j)} - \boldsymbol{p}_k^{(i,j)}\right)\right) = 0 \tag{4.126}$$

must hold. After multiplying out and comparing coefficients, this provides five linear equations for the seven unknowns α_0, α_1, α_3, α_4 and the three components of $\boldsymbol{q}_2^{(i,j)}$. To obtain a unique solution, the linear system is again augmented by a minimization problem. To minimize the deviation of the normal curve's norm from those of the corner normals and to minimize the torsion of the normal curve, the objective function

$$\Pi^{\text{Normals}} = \left\|\boldsymbol{q}_2^{(i,j)} - \frac{1}{2}\left(\boldsymbol{N}_1^{(i,j)} + \boldsymbol{N}_2^{(i,j)}\right)\right\|^2 + \sum_{\substack{i=0\\ i\neq 2}}^{4} (\alpha_i - 1)^2 \tag{4.127}$$

is minimized as proposed in [155].

Step III - construction of patch interior The remaining control points of each patch are determined by making the patch's derivatives orthogonal to the normal curves constructed in the previous step. This is achieved by computing the remaining control points such that

$$\begin{aligned} \boldsymbol{n}_1^{(i)}(r^{(i)}) \cdot \boldsymbol{S}_r^{(i)}\left(r^{(i)}, -1\right) = 0, \quad \boldsymbol{n}_2^{(i)}(r^{(i)}) \cdot \boldsymbol{S}_r^{(i)}\left(r^{(i)}, 1\right) = 0, \\ \boldsymbol{n}_3^{(i)}(s^{(i)}) \cdot \boldsymbol{S}_s^{(i)}\left(-1, s^{(i)}\right) = 0, \quad \boldsymbol{n}_4^{(i)}(s^{(i)}) \cdot \boldsymbol{S}_s^{(i)}\left(1, s^{(i)}\right) = 0. \end{aligned} \tag{4.128}$$

Due to the special construction of the edge curves and edge normal curves, the constraints in (4.128) simplify significantly after multiplying out. A system of 32 equations for 36 unknowns (the components of all control points labeled by □ or △ in (4.112)) is obtained as detailed in [155]. The G^1-constraints are independent of the innermost control points (labeled by × in (4.112)), which accordingly can be chosen freely. To obtain a unique solution, the differences between the control points and target points are minimized using the objective function

$$\Pi^{\text{Interior}} = \sum_{\substack{i=1 \\ j=1}}^{4} \left\| \boldsymbol{P}_{ij} - \tilde{\boldsymbol{P}}_{ij} \right\|^2 . \tag{4.129}$$

The target positions $\tilde{\boldsymbol{P}}_{ij}$ are computed by fitting a bicubic patch to the corner positions and normals before elevating its degree to biquintic without changing its shape. This can be done by demanding that

$$\mathbf{N}^{(3)}_{\text{Bern.}}(r^{(i)})\, \mathbf{P}^{(i)}_3\, \mathbf{N}^{(3)}_{\text{Bern.}}(s^{(i)}) = \mathbf{N}^{(5)}_{\text{Bern.}}(r^{(i)})\, \tilde{\mathbf{P}}^{(i)}\, \mathbf{N}^{(5)}_{\text{Bern.}}(s^{(i)}) \tag{4.130}$$

holds at 36 positions $(r^{(i)}, s^{(i)})$. In (4.130), $\mathbf{N}^{(3)}$ and $\mathbf{N}^{(5)}$ are the interpolation matrices for patches of degrees three and five, respectively, as introduced in (4.111). The control point matrices or arrays $\mathbf{P}^{(i)}_3$ and $\tilde{\mathbf{P}}^{(i)}$ are constructed as $\mathbf{P}^{(i)}$, see (4.112).

4.2.3 General polynomial G^1 quads

The modified construction scheme from [155] is local, of acceptably low order (biquintic) and provides smooth surfaces for input meshes with nodes of any valence. However, the scheme is restricted to the biquintic Bernstein basis. In this section, a more general construction scheme is presented, which allows to use any basis for two-dimensional patches (not necessarily tensor product type bases) as long as it leads to a solvable system of equations when formulating the general G^1-constraints. The $\hat{n}$ control points (or basis function weights) of patch i are collected in the vector

$$\mathbf{p}^{(i)} = \left[\begin{array}{ccc} \boldsymbol{P}_1^{(i)\mathrm{T}} & \dots & \boldsymbol{P}_n^{(i)\mathrm{T}} \end{array} \right]^{\mathrm{T}} \tag{4.131}$$

in a one-dimensional fashion as opposed to the arrangement in the array $\mathbf{P}^{(i)}$, cf. (4.112). An interpolation matrix $\hat{\mathbf{N}}$ constructed like the one from the FEM (see Section A.3.2) is used such that

$$\boldsymbol{S}^{(i)} = \hat{\mathbf{N}}\,\mathbf{p}^{(i)}. \tag{4.132}$$

Further, a gradient matrix

$$\mathbf{G}_r\,(r,s) = \left[\begin{array}{ccccccc} \frac{\partial N_1}{\partial r} & 0 & 0 & & \frac{\partial N_n}{\partial r} & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial r} & 0 & \dots & 0 & \frac{\partial N_n}{\partial r} & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial r} & & 0 & 0 & \frac{\partial N_n}{\partial r} \end{array} \right] \tag{4.133}$$

is introduced to establish

$$\boldsymbol{S}_r^{(i)} = \frac{\partial \boldsymbol{S}^{(i)}}{\partial r} = \mathbf{G}_r\,\mathbf{p}^{(i)}. \tag{4.134}$$

A gradient interpolation matrix $\mathbf{G}_s$ enables a corresponding formulation for the s-coordinate.

The fulfillment of the continuity constraints introduced at the beginning of this section can now be achieved by a combination of the following discrete constraints.

- The *node interpolation constraints* can be written as

$$\mathbf{N}\left(r_k^{(i)}, s_k^{(i)}\right)\,\mathbf{p}^{(i)} = \boldsymbol{X}_k^{(i)}, \tag{4.135}$$

where $\boldsymbol{X}_k^{(i)}$ denotes the position of the kth node of patch i and $(r_k^{(i)}, s_k^{(i)})$ denotes its position in the patch's local coordinate system.

- Considering two patches i and $i+1$ with a common edge as in the example shown in Fig. 4.8, the *edge continuity constraint* (4.106) can be discretized as

$$\mathbf{N}\,(r_k, 1)\;\mathbf{p}^{(i)} - \mathbf{N}\,(1, r_k)\;\mathbf{p}^{(i+1)} = \boldsymbol{0}, \tag{4.136}$$

with r_k being the kth value of the common edge coordinate $r = s^{(i)} = r^{(i+1)}$ where the constraint is prescribed.

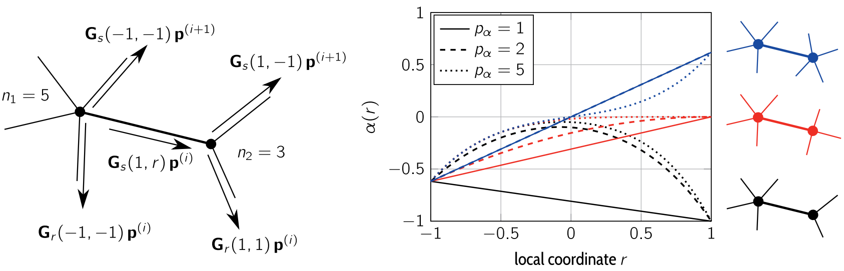


Figure 4.11: Left: Relation between the derivatives in orthogonal and in tangential direction of an exemplary edge. Right: Blending function α in the G^1 continuity constraints for different exponents p_α and different edge configurations.

- The C^1*-continuity constraints in edge direction* as described by (4.107) are discretized as

$$\mathbf{G}_s\left(-1, r_k\right)\,\mathbf{p}^{(i)} - \mathbf{G}_r\left(r_k, -1\right)\,\mathbf{p}^{(i+1)} = \boldsymbol{0}. \tag{4.137}$$

- The G^1*-continuity constraints* as described by (4.108) are discretized as

$$\mathbf{G}_r\left(-1, r_k\right)\,\mathbf{p}^{(i)} + \mathbf{G}_s\left(r_k, -1\right)\,\mathbf{p}^{(i+1)} - \alpha(r_k)\,\mathbf{G}_s\left(-1, r_k\right)\,\mathbf{p}^{(i)} = \boldsymbol{0}, \tag{4.138}$$

  which is obtained by setting $\alpha = \Phi^{-1} = \Theta^{-1}$. In order to preserve the linearity of the surface construction scheme, α must be specified explicitly. With the valences n_1 and n_2 of the two nodes adjacent to the respective edge, $\alpha(-1) = 2\,\cos\left({}^{2\,\pi}\!/_{n_1}\right)$ and $\alpha(1) = 2\,\cos\left({}^{2\,\pi}\!/_{n_2}\right)$ are suitable boundary values (see Fig. 4.11). Introducing an additional parameter p_α,

$$\alpha\left(r\right) = 2\,\cos\left(\frac{2\,\pi}{n_1}\right)\left(\frac{1}{2} - \frac{r}{2}\right)^{p_\alpha} + 2\,\cos\left(\frac{2\,\pi}{n_2}\right)\left(\frac{1}{2} + \frac{r}{2}\right)^{p_\alpha} \tag{4.139}$$

  is used here with $p_\alpha = 2$ if not stated otherwise. For $p_\alpha = 1$, (4.139) corresponds to a linear interpolation of the boundary values.

The constraints may then be assembled into a global constraint matrix

$$\mathbf{C}\,\mathbf{p} = \mathbf{c} \tag{4.140}$$

as summarized in Section A.3.3 for the FEM.

The system is solvable, if p^{g} is chosen high enough and the introduced constraints are combined in a suitable way and prescribed at a suitable number of points. It is still unlikely that the constraints yield a unique solution for the control points $\boldsymbol{P}_{jk}^{(i)}$. Instead, (4.140) can be expected to be an underdetermined system. To obtain a unique solution, it is augmented by a minimization problem. This can be interpreted as a regularization of the surface. The objective functions to be minimized are also called *fairness criteria* and are introduced below. They are discretized in such a way that the minimization problem can be stated as

$$||\mathbf{Q}\,\mathbf{p} - \mathbf{q}||^2 \rightarrow \min. \tag{4.141}$$

Now, standard procedures to solve constraint minimization problems (CMP) can be applied. The method of Lagrange multipliers is used here, where the CMP is formulated as

$$(\mathbf{Q}\,\mathbf{p} - \mathbf{q})^{\mathrm{T}}\,(\mathbf{Q}\,\mathbf{p} - \mathbf{q}) + \boldsymbol{\lambda}\,(\mathbf{C}\,\mathbf{p} - \mathbf{c}) \rightarrow \min. \tag{4.142}$$

Setting the derivative with respect to $\mathbf{p}$ and $\boldsymbol{\lambda}$ equal to zero leads to the linear system

$$\begin{bmatrix} \mathbf{Q}^{\mathrm{T}}\mathbf{Q} & \mathbf{C}^{\mathrm{T}} \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 2\,\mathbf{Q}^{\mathrm{T}}\mathbf{q} \\ \mathbf{c} \end{bmatrix}. \tag{4.143}$$

Using this general approach, constraints can be combined with objective functions in a very flexible way, yet the locality property is lost and one system of equations has to be solved for the degrees of freedom of all patches. In comparison with the systems to be solved for each patch in the construction scheme from [155] this system is large. However, in view of the application of the construction schemes within the mesh generation process, the additionally required computational effort is justified. Since the number of patches is at least as high as the number of finite elements in the resulting mesh, a system that is at least as large has to be solved repeatedly during the simulation. Compared to the overall computational effort of the simulation, the effort for mesh generation is therefore still negligible. In most cases, the number of patches is even low enough to allow for a live computation of the geometry, i.e. as the user interacts with the input mesh.

Regularization and fairness criteria Energy potentials are a common mean to regularize surface geometries. They usually contain derivatives of $\boldsymbol{S}$ and may be constructed from the general potentials given in [224], e.g.

$$\Pi_1 = \sum_{i=1}^{n^{\mathrm{p}}} \int_{-1}^{1} \int_{-1}^{1} \left\| \boldsymbol{S}_r^{(i)} \right\|^2 + \left\| \boldsymbol{S}_s^{(i)} \right\|^2 \mathrm{d}r^{(i)} \, \mathrm{d}s^{(i)} \tag{4.144}$$

$$\Pi_2 = \sum_{i=1}^{n^{\mathrm{p}}} \int_{-1}^{1} \int_{-1}^{1} \left\| \boldsymbol{S}_{rr}^{(i)} \right\|^2 + 2\,\alpha_{rs} \left\| \boldsymbol{S}_{rs}^{(i)} \right\|^2 + \left\| \boldsymbol{S}_{ss}^{(i)} \right\|^2 \mathrm{d}r^{(i)} \, \mathrm{d}s^{(i)}, \tag{4.145}$$

with $\alpha_{rs} = 1$ by default. A so-called *thin plate energy functional* as used in [27] is given by

$$\Pi_{\mathrm{tp}} = \alpha_{\mathrm{tp}} \, \Pi_1 + \Pi_2, \tag{4.146}$$

where α_{p} is introduced as a control parameter. The larger α_{tp}, the tighter the surface is pulled around the control point mesh. As in [224], potentials including higher order derivatives are also used here.

$$\Pi_3 = \sum_{i=1}^{n^{\mathrm{p}}} \int_{-1}^{1} \int_{-1}^{1} \left\| \boldsymbol{S}_{rrr}^{(i)} \right\|^2 + 3\,\alpha_{rrs} \left(\left\| \boldsymbol{S}_{rrs}^{(i)} \right\|^2 + \left\| \boldsymbol{S}_{rss}^{(i)} \right\|^2 \right) + \left\| \boldsymbol{S}_{sss}^{(i)} \right\|^2 \tag{4.147}$$

and

$$\begin{aligned} \Pi_4 = \sum_{i=1}^{n^{\mathrm{p}}} \int_{-1}^{1} \int_{-1}^{1} & \left\| \boldsymbol{S}_{rrrr}^{(i)} \right\|^2 + 4\,\alpha_{rrrs} \left(\left\| \boldsymbol{S}_{rrrs}^{(i)} \right\|^2 + \left\| \boldsymbol{S}_{rsss}^{(i)} \right\|^2 \right) \\ & +6\,\alpha_{rrss} \left\| \boldsymbol{S}_{rrss}^{(i)} \right\|^2 + \left\| \boldsymbol{S}_{ssss}^{(i)} \right\|^2 \mathrm{d}r^{(i)} \, \mathrm{d}s^{(i)} \end{aligned} \tag{4.148}$$

were found to be especially useful. The control parameters are set to $\alpha_{rrrs} = \alpha_{rrss} = \alpha_{rsss} = 1$ by default. Further, integrals of the deviation of $\boldsymbol{S}$ from a given target geometry $\tilde{\boldsymbol{S}}$, namely

$$\Pi_{\tilde{\boldsymbol{S}}} = \sum_{i=1}^{n^{\mathrm{p}}} \int_{-1}^{1} \int_{-1}^{1} \left\| \boldsymbol{S}^{(i)} - \tilde{\boldsymbol{S}}^{(i)} \right\|^2 \mathrm{d}r^{(i)} \, \mathrm{d}s^{(i)} \tag{4.149}$$

are commonly used. The target geometry has to be provided on a per-patch basis and may be constructed from the quadrangularization using less complex parameterizations such as bilinear or bicubic patches.

All objective functions can be discretized in a straightforward manner using a quadrature rule. In accordance with the polynomial basis, the

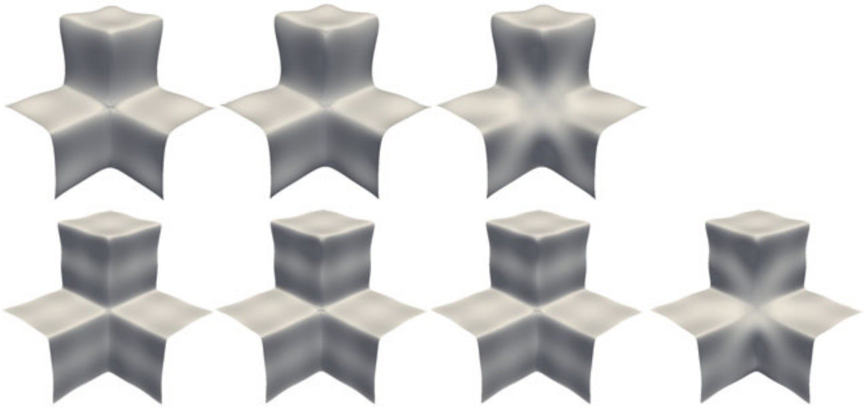


Figure 4.12: Influence of the number of points at which the G^1-continuity constraints (4.138) are prescribed. Top: Polynomial patch of order $p^{\mathrm{g}} = 4$ (2–5 points). Bottom: Polynomial patch of order $p^{\mathrm{g}} = 5$ (2–6 points).

Gauss-Legendre quadrature is chosen here. The matrix $\mathbf{Q}$, or more precisely the products $\left(\mathbf{Q}^{\mathrm{T}}\,\mathbf{Q}\right)$ and $\left(\mathbf{Q}^{\mathrm{T}}\,\mathbf{q}\right)$, are assembled from the counterparts on patch level. Considering (4.144),

$$\left(\mathbf{Q}^{\mathrm{T}}\,\mathbf{Q}\right)^{(i)} \approx \sum_{k=1}^{n^{\mathrm{qp}}} w_k \left(\mathbf{G}_r^{\mathrm{T}}\left(r_k, s_k\right)\,\mathbf{G}_r\left(r_k, s_k\right) + \mathbf{G}_s^{\mathrm{T}}\left(r_k, s_k\right)\,\mathbf{G}_s\left(r_k, s_k\right)\right) \tag{4.150}$$

and

$$\left(\mathbf{Q}^{\mathrm{T}}\,\mathbf{q}\right)^{(i)} = \mathbf{0} \tag{4.151}$$

are obtained, where $\left(r_k, s_k\right)$ and w_k are the quadrature coordinates and weights, respectively. The other objective functions are discretized in a corresponding way using higher order gradient matrices, e.g. $\mathbf{G}_{rrss}$. It should be noted that constraints, which are written as is $\mathbf{C}^{(i)}\,\mathbf{p}^{(i)} = \mathbf{c}^{(i)}$ on patch level, may also be used to construct a potential with $\mathbf{Q}^{(i)} = \mathbf{C}^{(i)}$ and $\mathbf{q}^{(i)} = \mathbf{c}^{(i)}$. This allows, e.g., to drop the requirement that the nodes of the control point mesh are interpolated, which may yield a more regular surface.

Constraint combination In order to construct a G^1-continuous surface from a given control point mesh, the first step is to choose a basis. As a first example, a polynomial tensor-product basis of degree $p^{\mathrm{g}} = 4$ is considered. This gives 25 degrees of freedom for each patch in each space dimension. Without a loss of generality, the Lagrange polynomials are used, such that the degrees of freedom correspond to points on the surface. Prescribing the node interpolation constraint (4.135) fixes four of them. The edge continuity constraint (4.136) is prescribed at $p^{\mathrm{g}} - 1 = 3$ points. This is the number of points that ensures that the common edges of two neighboring patches have the same geometry. (Considering the Lagrange basis, the edge geometry is defined by the 5 interpolated points on that edge, two of which are already fixed due to the node interpolation constraint.) The C^1-continuity constraint (4.137) is therefore not needed. The G^1-continuity constraints (4.138) are prescribed at $p^{\mathrm{g}} + 1 = 5$ points, due to the fact that the derivative in the direction orthogonal to the edge, e.g. $\boldsymbol{S}_r^{(i)}(-1, s^{(i)})$, depends on 5 degrees of freedom. For arbitrary mesh topologies, i.e. node valences, this may lead to redundant constraints because some of these degrees of freedom are already constrained by the edge continuity constraint. However, the resulting global system is still solvable, as long as the prescribed constraints do not contradict each other. In general, a solver that reveals the solvability should be used.

Whether the constraints were prescribed at enough points may be checked using a visualization software. The shading algorithms in computer graphics libraries such as *OpenGL* (see [117]), which constitute the core of visualization software such as *ParaView* (see [12]) are very sensible to sharp edges, such that a faulty combination of constraints can be detected easily. In Fig. 4.12, this is shown for the example of the number of points at which the G^1-continuity was prescribed. According to the thoughts above, a continuous surface is obtained when prescribing the constraint at $p^{\mathrm{g}} + 1$ points.

The examples in Fig. 4.12 were created using the thin plate energy functional (4.146) with $\alpha_{\mathrm{tp}} = 1$. In Fig. 4.13, the influence of α_{tp} is illustrated. As mentioned above, the higher α_{tp}, the tighter the surface is pulled around the control points. The parameter may therefore be associated with the inverse of the thickness of a thin plate or membrane spanned around the control points. Further, the influence of the parameters α_{rs}, i.e. the factor in front of the cross derivative in (4.145), is shown in Fig. 4.13. While α_{tp} and α_{rs} seem to allow for a flexible tuning and a great variety of surfaces, tube-like geometries require the functionals including higher derivatives. As

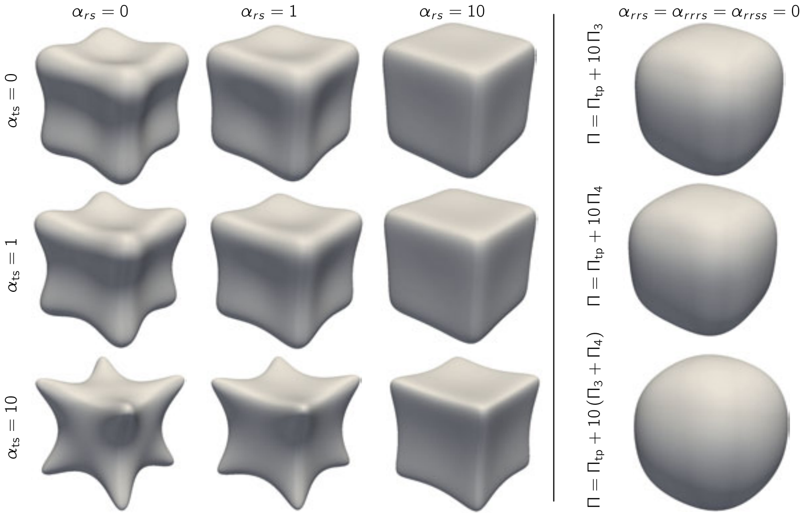

$\alpha_{rs} = 0$
$\alpha_{rs} = 1$
$\alpha_{rs} = 10$
$\alpha_{rrs} = \alpha_{rrrs} = \alpha_{rrss} = 0$
$\alpha_{\mathrm{ts}} = 0$
$\alpha_{\mathrm{ts}} = 1$
$\alpha_{\mathrm{ts}} = 10$
$\Pi = \Pi_{\mathrm{tp}} + 10\,\Pi_3$
$\Pi = \Pi_{\mathrm{tp}} + 10\,\Pi_4$
$\Pi = \Pi_{\mathrm{tp}} + 10\,(\Pi_3 + \Pi_4)$

Figure 4.13: Left: Influence of the parameters α_{ts} and α_{rs}. Right: Influence of functionals with higher order derivatives.

apparent in Fig. 4.13, for the cube geometry chosen as an example, they make the surface bulge out. For tube-like surfaces, which are defined using a coarse mesh with possibly only four control points per cross section, this is a desired feature. It allows to tune the construction scheme such that a corresponding cross section (polygonal, e.g. quadrangular in the control point mesh) becomes circular.

4.3 Partitioned solution approach

In the partitioned solution approach considered here, the solvers for the subproblems act as black-boxes, i.e. no other information than the solution of the respective subproblem can be obtained. Their inner structure, e.g. the underlying system of equations or a derivative with respect to the coupling quantity, is hidden from the outside. This motivates the operator formulation introduced in Chapter 3 for each subproblem, respectively each field solver. To start with, the partitioned solution approach is introduced for the example of a two-field fluid-structure interaction problem. As introduced in Section 3.1, k denotes the current time step. The state of all subproblems at time t_k is known and the solution state at t_{k+1} is to be computed. Recalling the general fluid solve operator

$$\begin{bmatrix} \mathbf{v}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix} = \mathcal{F}_k \circ \begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{v}}_{k+1} \\ \bar{\mathbf{p}}_{k+1} \end{bmatrix}, \tag{4.152}$$

and the structural solve operator

$$\mathbf{d}_{k+1} = \mathcal{S}_k \circ \bar{\mathbf{t}}_{k+1}, \tag{4.153}$$

the question arises, how to realize the coupling, i.e. a fulfillment of the interface constraints (3.118-3.120). Before introducing different coupling algorithms to achieve this in Section 4.3.1, some preliminary considerations have to be taken. At first, it should be recalled that the hat in $\bar{\hat{\mathbf{d}}}_{k+1}$ indicates that it is the discretized displacement of the fluid domain, not the discretized displacement of the structure. Bars $(\bar{\cdot})$ denote prescribed quantities in the respective subproblem. In order to realize an exchange of the coupling quantities, the domain displacement on the coupling interface (discretized according to the fluid mesh and collected in the vector $\bar{\hat{\mathbf{d}}}_{k+1}$) has to be computed from the displacement provided by the structural solver (discretized according to the structural mesh and collected in the vector $\mathbf{d}_{k+1}$). A mapping operator is introduced for this purpose such that

$$\bar{\hat{\mathbf{d}}}_{k+1} = \mathcal{M}_{\mathcal{S}}^{\mathcal{F}} \circ \mathbf{d}_{k+1}. \tag{4.154}$$

A corresponding operator is introduced to compute the tractions (discretized according to the structural mesh and collected in the vector $\bar{\mathbf{t}}_k$) from the velocities and pressures (discretized according to the fluid mesh and collected in the vectors $\mathbf{p}_k$ and $\mathbf{v}_k$). This involves a computation according to Eq. (3.118) and a mapping from one discretization to another, which may be written as

$$\bar{\mathbf{t}}_{k+1} = \mathcal{M}_{\mathcal{F}}^{\mathcal{S}} \circ \mathbf{t}_k = \mathcal{M}_{\mathcal{F}}^{\mathcal{S}} \circ \mathcal{E}_{\boldsymbol{t}} \circ \begin{bmatrix} \mathbf{v}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix}. \tag{4.155}$$

Therein, the evaluation operator $\mathcal{E}_{\boldsymbol{t}}$ yields the tractions

$$\boldsymbol{t}_{k+1} = \mathcal{E}_{\boldsymbol{t}} \begin{bmatrix} \mathbf{v}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix}, \tag{4.156}$$

which are discretized according to the fluid mesh. Accordingly it constitutes an evaluation of Eq. (3.118) on a per-point basis. It should be noted that the mapping and evaluation operators are independent of k, while the solver operators yield different results for the same input depending of the current state.

For the present case of a two field FSI problem, modified fluid and structural solve operators are introduced to ease the notation. The mapping and evaluation operators may be combined with the solve operators to yield

$$\bar{\hat{\mathbf{d}}}_{k+1} = \mathcal{M}_{\mathcal{S}}^{\mathcal{F}} \circ \mathcal{S}_k \circ \bar{\mathbf{t}}_{k+1} = \hat{\mathcal{S}}_k \circ \bar{\mathbf{t}}_{k+1} \tag{4.157}$$

and

$$\bar{\mathbf{t}}_{k+1} = \mathcal{M}_{\mathcal{F}}^{\mathcal{S}} \circ \mathcal{E}_{\boldsymbol{t}} \circ \mathcal{F}_k \circ \begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{v}}_{k+1} \\ \bar{\mathbf{p}}_{k+1} \end{bmatrix} = \hat{\mathcal{F}}_k \circ \bar{\hat{\mathbf{d}}}_{k+1}. \tag{4.158}$$

4.3.1 Coupling algorithms

A coupling algorithm here denotes the scheme according to which the subproblems in a multifield problem are coupled. With the above operators, an FSI problem can be formulated in terms of the fixed point-condition (FPC)

$$\bar{\hat{\mathbf{d}}}_{k+1} = \hat{\mathcal{S}}_k \circ \hat{\mathcal{F}}_k \circ \bar{\hat{\mathbf{d}}}_{k+1}. \tag{4.159}$$

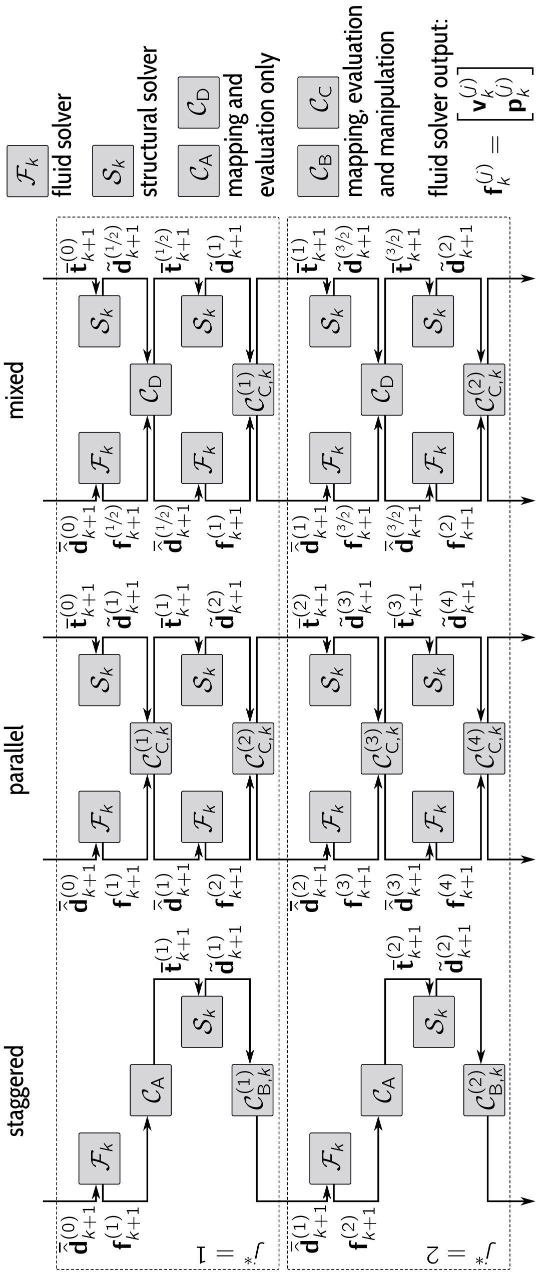

staggered
parallel
mixed
$\mathcal{F}_k$ fluid solver
$\mathcal{S}_k$ structural solver
$\mathcal{C}_\mathrm{A}$ $\mathcal{C}_\mathrm{D}$ mapping and evaluation only
$\mathcal{C}_\mathrm{B}$ $\mathcal{C}_\mathrm{C}$ mapping, evaluation and manipulation
fluid solver output:
$\mathbf{f}_k^{(j)} = \begin{bmatrix} \mathbf{v}_k^{(j)} \\ \mathbf{p}_k^{(j)} \end{bmatrix}$
$J^* = 1$
$J^* = 2$
$\mathcal{C}_{\mathrm{B},k}^{(1)}$
$\mathcal{C}_{\mathrm{B},k}^{(2)}$
$\mathcal{C}_{\mathrm{C},k}^{(1)}$
$\mathcal{C}_{\mathrm{C},k}^{(2)}$
$\mathcal{C}_{\mathrm{C},k}^{(3)}$
$\mathcal{C}_{\mathrm{C},k}^{(4)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(0)}$
$\mathbf{f}_{k+1}^{(1)}$
$\bar{\mathbf{t}}_{k+1}^{(1)}$
$\tilde{\mathbf{d}}_{k+1}^{(1)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(1)}$
$\mathbf{f}_{k+1}^{(2)}$
$\bar{\mathbf{t}}_{k+1}^{(2)}$
$\tilde{\mathbf{d}}_{k+1}^{(2)}$
$\bar{\mathbf{t}}_{k+1}^{(0)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(2)}$
$\mathbf{f}_{k+1}^{(3)}$
$\bar{\mathbf{t}}_{k+1}^{(3)}$
$\tilde{\mathbf{d}}_{k+1}^{(3)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(3)}$
$\mathbf{f}_{k+1}^{(4)}$
$\tilde{\mathbf{d}}_{k+1}^{(4)}$
$\mathbf{f}_{k+1}^{(1/2)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(1/2)}$
$\tilde{\mathbf{d}}_{k+1}^{(1/2)}$
$\bar{\mathbf{t}}_{k+1}^{(1/2)}$
$\mathbf{f}_{k+1}^{(3/2)}$
$\bar{\hat{\mathbf{d}}}_{k+1}^{(3/2)}$
$\tilde{\mathbf{d}}_{k+1}^{(3/2)}$
$\bar{\mathbf{t}}_{k+1}^{(3/2)}$

Figure 4.14: Illustration of selected coupling algorithms. For each algorithm, the operators in the middle column are evaluated by the coupling manager, while the left and the right column are executed by the fluid and the structural solver, respectively.

A straightforward coupling algorithm may be based on the fixed-point iteration (FPI)

$$\bar{\hat{\mathbf{d}}}_{k+1}^{(j+1)} \overset{\text{FPI}}{=} \hat{\mathcal{S}}_k \circ \hat{\mathcal{F}}_k \circ \bar{\hat{\mathbf{d}}}_{k+1}^{(j)}. \tag{4.160}$$

However, such a *standard fixed-point iteration* is unlikely to converge in the general case, or, in other words, a corresponding partitioned solution approach is likely to suffer from instabilities. To improve the stability, convergence acceleration methods are used, which are generally represented by the operator $\mathcal{A}_k^{(j)}$ here. A *modified fixed-point iteration* may then be stated as

$$\bar{\hat{\mathbf{d}}}_{k+1}^{(j+1)} = \mathcal{A}_k^{(j)} \circ \tilde{\bar{\hat{\mathbf{d}}}}_{k+1}^{(j+1)} = \mathcal{A}_k^{(j)} \circ \hat{\mathcal{S}}_k \circ \hat{\mathcal{F}}_k \circ \bar{\hat{\mathbf{d}}}_{k+1}^{(j)}. \tag{4.161}$$

It constitutes the basis for a *staggered* coupling algorithm as illustrated in Fig. 4.14. The additional operators are found to be

$$\mathcal{C}_{\mathrm{A}} = \mathcal{M}_{\mathcal{F}}^{\mathcal{S}} \circ \mathcal{E}_{\boldsymbol{t}} \tag{4.162}$$

and

$$\mathcal{C}_{\mathrm{B},k}^{(j)} = \mathcal{A}_k^{(j)} \circ \mathcal{M}_{\mathcal{S}}^{\mathcal{F}}. \tag{4.163}$$

An alternative variant of the staggered coupling algorithms may be based on the FPC

$$\bar{\mathbf{t}}_{k+1} = \hat{\mathcal{F}}_k \circ \hat{\mathcal{S}}_k \circ \bar{\mathbf{t}}_{k+1}. \tag{4.164}$$

The *parallel* coupling algorithm from Fig. 4.14 is based on the FPC

$$\begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{t}}_{k+1} \end{bmatrix} = \begin{bmatrix} 0 & \hat{\mathcal{S}}_k \\ \hat{\mathcal{F}}_k & 0 \end{bmatrix} \circ \begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{t}}_{k+1} \end{bmatrix}, \tag{4.165}$$

which constitutes an alternative way of formulating the FSI problem. The additional operator $\mathcal{C}_{\mathrm{C},k}^{(j)}$ can again be identified as a combination of mappings, evaluations, and accelerations, which may be defined as

$$\mathcal{C}_{\mathrm{C},k}^{(j)} = \mathcal{A}_l^{(j)} \circ \begin{bmatrix} \mathcal{M}_{\mathcal{S}}^{\mathcal{F}} & 0 \quad 0 \\ 0 & \mathcal{C}_{\mathrm{A}} \end{bmatrix}. \tag{4.166}$$

Given that both solvers take the same time to solve the respective subproblem, one iteration of the parallel algorithm takes only half the time that a staggered iteration takes. To compare the efficiency, the counter j^* is introduced in Fig. 4.14. It counts the number of consecutive solver calls rather than being a zero-based fixed-point iteration index. While $j^* = {}^{j}\!/_{2} + 1$ in the parallel case, the staggered case implies $j^* = j + 1$, which likewise holds for the *mixed* coupling algorithm. The latter constitutes a combination of the parallel and the staggered algorithm and is based on the FPC

$$\begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{t}}_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{\mathcal{S}}_k \circ \hat{\mathcal{F}}_k & 0 \\ 0 & \hat{\mathcal{F}}_k \circ \hat{\mathcal{S}}_k \end{bmatrix} \circ \begin{bmatrix} \bar{\hat{\mathbf{d}}}_{k+1} \\ \bar{\mathbf{t}}_{k+1} \end{bmatrix}. \tag{4.167}$$

The iterative procedure underlying all classes of coupling algorithms introduced above may be stated in terms of a generalized solution vector $\mathbf{s}_k$ and a combined solve operator $\mathcal{X}$ as a modified FPI

$$\tilde{\mathbf{s}}_{k+1}^{(i+1)} = \mathcal{X}_k \circ \mathbf{s}_{k+1}^{(i)}, \tag{4.168}$$

where $\mathcal{X}_k$ constitutes a combination of several operators representing field solvers, evaluations, and mappings. A general coupling algorithm can then be implemented as depicted in Algorithm 1. Therein, the residual $\mathbf{r}_k^{(j)}$ and the prediction operator $\mathcal{P}_k$ were introduced. In the following, specific forms of the introduced data processing blocks, namely convergence acceleration methods (Section 4.3.2), predictors (Section 4.3.3), and convergence criteria (Section 4.3.4) are reviewed. Mapping schemes are considered in Section 4.3.5.

Algorithm 1 General coupling algorithm used to solve a fixed-point condition $\mathbf{s}_{k+1} = \mathcal{X}_k \circ \mathbf{s}_{k+1}$.

```
for $k = 0 \dots n_{\mathrm{ts}} - 1$ do                                    ▷ Start of the time loop
    $\mathbf{s}_{k+1}^{(0)} = \mathcal{P}_k \circ \mathbf{s}_k$                                        ▷ Predict solution
    for $j = 0 \dots n_{\max} - 1$ do                                  ▷ Start of the coupling loop
        $\tilde{\mathbf{s}}_{k+1}^{(j+1)} = \mathcal{X}_k \circ \mathbf{s}_{k+1}^{(j)}$                           ▷ Call all participating solvers
        $\mathbf{r}_k^{(j)} = \tilde{\mathbf{s}}_{k+1}^{(j+1)} - \mathbf{s}_k^{(j)}$
        if CONVERGED($\mathbf{r}_k^{(i)}$) then                                ▷ Check convergence
            $\mathbf{s}_{k+1} = \tilde{\mathbf{s}}_{k+1}^{(j+1)}$
            break                                      ▷ Proceed with the next time step
        else
            $\mathbf{s}_{k+1}^{(j+1)} = \mathcal{A}_k^{(j)} \circ \tilde{\mathbf{s}}_{k+1}^{(j+1)}$                      ▷ Accelerate convergence
        end if
    end for                                               ▷ End of the coupling loop
end for                                                   ▷ End of the time loop
```

4.3.2 Convergence acceleration

The purpose of a convergence acceleration method is to provide an improved solution $\mathbf{s}_{k+1}^{(j+1)}$ based on solutions from previous iterations and time steps. This may be formulated in terms of a solution increment $\Delta\mathbf{s}_{k+1}^{(j)}$ as

$$\mathbf{s}_{k+1}^{(j+1)} = \mathcal{A}_k^{(j)} \circ \tilde{\mathbf{s}}_{k+1}^{(j+1)} = \mathbf{s}_{k+1}^{(j)} + \Delta\mathbf{s}_{k+1}^{(j)}. \tag{4.169}$$

It should be noted that the operator for the acceleration method is equipped with an iteration index, since it generally does not yield the same output when evaluated for the same input in different iterations. This is not the case for the solve operators, which only depend on the current time step k.

One of the fastest methods to obtain a solution to the FPC $\boldsymbol{s}_{k+1} = \mathcal{X} \circ \boldsymbol{s}_{k+1}$ is the Newton-Raphson (NR) method, which was introduced in Section 3.1 for monolithic solution approaches and in Section 3.2.2 for the solution of the structural mechanics subproblem. Transforming the fixed-point condition into the root-finding problem

$$\mathbf{r}_k^{(j)} = \tilde{\mathbf{s}}_{k+1}^{(j+1)} - \mathbf{s}_k^{(j)} = \mathcal{X}_k \circ \mathbf{s}_{k+1}^{(j)} - \mathbf{s}_{k+1}^{(j)} = \mathbf{0}, \tag{4.170}$$

the solution increment can be computed as

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\text{NR}}{=} -\mathbf{J}_j^{-1}\,\mathbf{r}_k^{(j)}, \tag{4.171}$$

where

$$\mathbf{J}_j = \mathrm{D}_{\mathbf{s}}\mathbf{r}_{k+1}^{(j)}\Big|_{\mathbf{s}_{k+1}=\mathbf{s}_{k+1}^{(j)}} \tag{4.172}$$

denotes the Jacobian or tangent matrix of the root-finding problem. However, in view of the black-box requirement on the field solvers, a tangent matrix is not accessible. Correspondingly, the goal of a convergence acceleration method can be seen in computing a reasonable approximation of $\mathbf{J}_j$ solely from previous solution vectors $\mathbf{s}_{k+1}^{(j)}$, $\mathbf{s}_{k+1}^{(j-1)}$, etc. Corresponding methods are termed *quasi-Newton methods*.

For the standard FPI,

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\text{FPI}}{=} \mathbf{r}_{k+1}^{(j)} \tag{4.173}$$

is obtained, i.e. the Jacobian matrix is approximated as $\mathbf{J}_j \approx -\mathbf{1}$. A stabilization of the standard FPI may be achieved by taking only a fraction

Table 4.2: Quantities in a coupling algorithm and their computation throughout the iterations.

	solver output	residual	updated solution	solution increment
j	$\tilde{\mathbf{s}}_{k+1}^{(j+1)} = \mathcal{X}_k \circ \mathbf{s}_{k+1}^{(j)}$	$\mathbf{r}_{k+1}^{(j)} = \tilde{\mathbf{s}}_{k+1}^{(j+1)} - \mathbf{s}_{k+1}^{(j)}$	$\mathbf{s}_{k+1}^{(j+1)} = \mathcal{A}_k^{(j)} \circ \tilde{\mathbf{s}}_{k+1}^{(j+1)}$	$\Delta\mathbf{s}_{k+1}^{(j)} = \mathbf{s}_{k+1}^{(j+1)} - \mathbf{s}_{k+1}^{(j)}$
0	$\tilde{\mathbf{s}}_{k+1}^{(1)}$	$\mathbf{r}_{k+1}^{(0)}$	$\mathbf{s}_{k+1}^{(1)}$	$\Delta\mathbf{s}_{k+1}^{(0)}$
1	$\tilde{\mathbf{s}}_{k+1}^{(2)}$	$\mathbf{r}_{k+1}^{(1)}$	$\mathbf{s}_{k+1}^{(2)}$	$\Delta\mathbf{s}_{k+1}^{(1)}$

of the residual as the solution increment, which leads to the *relaxed fixed-point iteration (RFPI)*

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\text{RFPI}}{=} \omega_j \, \mathbf{r}_{k+1}^{(j)}. \tag{4.174}$$

The relaxation factor ω_j is determined according to one of the relaxation methods introduced below. Relaxation methods may be combined with methods that compute an improved direction of the solution increment $\tilde{\Delta\mathbf{s}}_{k+1}^{(j)}$. The increment of a general modified FPI can then be stated as

$$\Delta\mathbf{s}_{k+1}^{(j)} = \omega_j \, \tilde{\Delta\mathbf{s}}_{k+1}^{(j)}. \tag{4.175}$$

Setting $\omega_j = 1$ and $\tilde{\Delta\mathbf{s}}_{k+1}^{(j)} = \mathbf{r}_{k+1}^{(j)}$ leads back to the increment of the standard FPI (4.173).

For a better overview regarding the explanation of relaxation methods and the following explanation of methods to compute an improved $\tilde{\Delta\mathbf{s}}_{k+1}^{(j)}$, Table 4.2 lists the quantities computed in the first iterations of every time step.

Relaxation methods

In the simplest case, the relaxation parameter is taken to be constant. As usual, this method is called *static relaxation (SR)* with

$$\omega_j \overset{\text{SR}}{=} \omega. \tag{4.176}$$

In Aitken (1950) [2], an acceleration method was introduced, in which the relaxation parameter is computed according to the current convergence behavior. While formulated for scalar sequences, [138] proposed several ways to use the ideas from [2] for vector sequences. A variant of this *Aitken relaxation (AR)* also mentioned in [130] uses

$$\omega_j \overset{\text{AR}}{=} \begin{cases} -\frac{\left(\mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)}\right)^{\mathrm{T}} \mathbf{r}_{k+1}^{(j-1)}}{\left\|\mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)}\right\|^2} & \text{for } j = 1, 4, 7, \dots, \\ \omega & \text{else.} \end{cases} \tag{4.177}$$

as an update rule for the relaxation factor. Accordingly, the actual Aitken relaxation step can only be applied in every third iteration. In intermediate iterations, a constant relaxation parameter ω is used to obtain two *pre-relaxed residuals* $\mathbf{r}^{(j)}$ and $\mathbf{r}^{(j-1)}$ (cf. [130]).

In Irons and Tuck (1969) [107], an improved Aitken relaxation is suggested. Here, it is denoted as *Irons-Tuck relaxation (ITR)*, and it allows for an update of the relaxation factor in every iteration using the recurrence relation

$$\omega_{j+1} \overset{\text{ITR}}{=} -\omega_j \frac{\left(\mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)}\right)^{\mathrm{T}} \mathbf{r}_{k+1}^{(j-1)}}{\left\|\mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)}\right\|^2}. \tag{4.178}$$

The method was successfully used to accelerate partitioned FSI simulations in many works, see e.g. [129, 163, 162, 122, 81], where it is simply called *Aitken relaxation.*

Another relaxation method, investigated in [144], is denoted as *backtrack relaxation (BTR).* Therein, the relaxation factor ω_j is successively reduced by a constant factor $\alpha < 1$, i.e.

$$\omega_j \overset{\text{BTR}}{=} \alpha\, \omega_{j-1}. \tag{4.179}$$

In the first iteration the relaxation factor is set to $\omega_0 = \omega_{\max}$. If ω falls below a given $\omega_{\min}$, no further reductions are performed in succeeding iterations.

Broyden method

Among the methods to compute $\tilde{\Delta}\mathbf{s}_{k+1}^{(j)}$ that were investigated in [144] is the *Broyden method (BM)*, originally proposed in [29]. As explained in detail in [116], several variants of this method have been developed. Here, the one given in algorithmic form in [144] is used and reformulated in terms of the recurrence relation

$$\tilde{\Delta}\mathbf{s}_{k+1}^{(j)} \overset{\text{BM}}{=} \left(\mathbf{p}^{(j)} - (1-\omega_j)\,\tilde{\mathbf{r}}^{(j-1)}\right)\left(1 - \frac{\omega_j\, \tilde{\mathbf{r}}^{(j-1)} \cdot \mathbf{p}^{(j)}}{\|\tilde{\mathbf{r}}^{(j-1)}\|^2}\right)^{-1}. \tag{4.180}$$

In each coupling iteration,

$$\mathbf{p}^{(j)} = \sum_{l=0}^{j-1} \tilde{\mathbf{p}}_l^{(j)} \tag{4.181}$$

has to be computed using $\tilde{\mathbf{p}}_0^{(j)} = -\mathbf{r}_k^{(j)}$ and

$$\tilde{\mathbf{p}}_l^{(j)} = \frac{\mathbf{s}_{k+1}^{(l)} \cdot \tilde{\mathbf{p}}_{l-1}^{(j)}}{\|\mathbf{s}_{k+1}^{(l)}\|^2} \left(\frac{\omega_l}{\omega_{l+1}} \mathbf{s}_{k+1}^{(l+1)} + (\omega_l - 1)\, \mathbf{s}_{k+1}^{(l)} \right) \quad \text{for } k = 1, \ldots, j-1. \tag{4.182}$$

Linear line extrapolation

A method called *linear line extrapolation (LLE)* is proposed in [233]. It was investigated in [144], but it was used with $\omega_j = 1$ instead of combining the approach with a relaxation method. The increment direction is computed as

$$\tilde{\Delta}\mathbf{s}_{k+1}^{(j)} \overset{\text{LLE}}{=} \alpha_j \mathbf{s}_{k+1}^{(j)} + (1 - \alpha_j)\, \mathbf{s}_{k+1}^{(j-1)} - \beta \left(\alpha_j \mathbf{r}_{k+1}^{(j)} + (1 - \alpha_j)\, \mathbf{r}_{k+1}^{(j-1)} \right). \tag{4.183}$$

The parameters α_j are determined according to the minimization problem

$$\alpha_j = \text{argmin} \|\alpha_j \mathbf{r}_{k+1}^{(j)} + (1 - \alpha_j)\, \mathbf{r}_{k+1}^{(j-1)}\|. \tag{4.184}$$

This yields

$$\alpha_j = -\frac{\left(\mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)} \right)^{\mathrm{T}} \mathbf{r}_{k+1}^{(j-1)}}{\left\| \mathbf{r}_{k+1}^{(j)} - \mathbf{r}_{k+1}^{(j-1)} \right\|^2}, \tag{4.185}$$

which can be identified to be the formula used to compute the relaxation parameters in the Aitken relaxation method.

Quasi-Newton least-squares method

A comparably new method originally termed *interface-quasi-Newton with inverse Jacobian from a least-squares model* is introduced here as the *quasi-Newton least-squares (QNLS) method.* The method is based on the ideas in [215] and was first used in the context of partitioned FSI simulations in [51]. Since then, it has been refined, investigated and applied by the same author (see [53, 56, 55]) as well as other research groups (see e.g [87, 122, 163, 162, 164]).

The QNLS method provides an approximation of the inverse of the Jacobian in a least-squares sense. Starting with the idea that the residual in the next iteration $\mathbf{r}_{k+1}^{(j+1)}$ should be zero in an optimal case, one can try to derive a solution increment $\Delta\mathbf{s}_{k+1}^{(j)}$ that minimizes $\mathbf{r}_{k+1}^{(j+1)}$. Introducing residual differences $\Delta\mathbf{r}_{k+1}^{(j,i)}$, the unknown difference

$$\Delta\mathbf{r}_{k+1}^{(j,j+1)} = \mathbf{r}_{k+1}^{(j+1)} - \mathbf{r}_{k+1}^{(j)} \tag{4.186}$$

is approximated by a linear combination of known residual differences, i.e.

$$\boldsymbol{r}_{k+1}^{(j+1)} = \boldsymbol{r}_{k+1}^{(j)} + \Delta\boldsymbol{r}_{k+1}^{(j,j+1)} = \boldsymbol{r}_{k+1}^{(j)} + \sum_{i=0}^{j-1} \alpha_i^{(k)} \Delta\boldsymbol{r}_{k+1}^{(j,i)}. \tag{4.187}$$

Different choices are possible regarding the definition of $\Delta\mathbf{r}_{k+1}^{(j,i)}$. Here,

$$\Delta\mathbf{r}_{k+1}^{(j,i)} = \mathbf{r}_{k+1}^{(i)} - \mathbf{r}_{k+1}^{(j)} \qquad \text{for } i = 0, \ldots, j-1 \tag{4.188}$$

is used as originally proposed in [51] [2]. Regardless of this choice, the vector of coefficients

$$\boldsymbol{\alpha}^{(j)} = \begin{bmatrix} \alpha_0^{(j)} & \ldots & \alpha_{j-1}^{(j)} \end{bmatrix}^{\mathrm{T}} \tag{4.189}$$

is determined by solving the minimization problem

$$\boldsymbol{\alpha}^{(j)} = \operatorname{argmin} \left\| \mathbf{r}_{k+1}^{(j)} + \sum_{i=0}^{j-1} \alpha_i^{(j)} \Delta\mathbf{r}_{k+1}^{(j,i)} \right\|^2 \approx \operatorname{argmin} \|\mathbf{r}_{k+1}^{(j+1)}\|^2. \tag{4.190}$$

The coefficients can then be used to compute the solution increment or difference as a linear combination of known differences

$$\Delta\tilde{\mathbf{s}}_{k+1}^{(j,i)} = \tilde{\mathbf{s}}_{k+1}^{(i)} - \tilde{\mathbf{s}}_{k+1}^{(j+1)} \qquad \text{for } i = 1, \ldots, j. \tag{4.191}$$

Using (4.188) one finds that

$$\Delta\mathbf{r}_{k+1}^{(j,j+1)} = \mathbf{r}_{k+1}^{(j+1)} - \mathbf{r}_{k+1}^{(j)} = \tilde{\mathbf{s}}_{k+1}^{(j+2)} - \mathbf{s}_{k+1}^{(j+1)} - \tilde{\mathbf{s}}_{k+1}^{(j+1)} + \mathbf{s}_{k+1}^{(j)} = \Delta\tilde{\mathbf{s}}_{k+1}^{(j,j+1)} - \Delta\mathbf{s}_{k+1}^{(j)}, \tag{4.192}$$

which gives rise to a formula for the QNLS solution increment

$$\Delta\boldsymbol{s}_{k+1}^{(j)} \overset{\text{QNLS}}{=} \sum_{i=0}^{j-1} \alpha_i^{(j)} \Delta\tilde{\mathbf{s}}_{k+1}^{(j,i+1)} - \Delta\boldsymbol{r}_{k+1}^{(j,j+1)} = \sum_{i=0}^{j-1} \alpha_i^{(j)} \Delta\tilde{\mathbf{s}}_{k+1}^{(j,i+1)} + \boldsymbol{r}_{k+1}^{(j)} \tag{4.193}$$

[2] According to (4.188), the new set of residual differences has to be computed in each coupling iteration. Using the alternative definition $\Delta\mathbf{r}_{k+1}^{(j,i)} = \mathbf{r}_{k+1}^{(i+1)} - \mathbf{r}_{k+1}^{(i)}$, only one new difference has to be computed in every coupling iteration. The formulation of the method presented here is valid as long as (4.186) holds.

The linear combinations (4.187) and (4.193) render each set of vectors $\Delta\mathbf{r}_{k+1}^{(j,i)}$ and $\Delta\tilde{\mathbf{s}}_{k+1}^{(j,i)}$ a basis spanning a vector space. Of course, starting with another definition for the residual differences in (4.188), one can derive a corresponding formula for the solution increment like (4.193). As long as the chosen basis spans the same vector space, such a formulation will be mathematically equivalent.

The question yet to be answered is how to calculate the coefficients $\alpha_i^{(j)}$. The approach is similar in the different variants of the QNLS method, yet, this is the step with room for improvements. In the original variant from [51], a standard **QR**-decomposition is used to solve the least-squares problem (4.190). Introducing the matrix

$$\mathbf{V}_k^{(j)} = \left[\ \Delta\mathbf{r}_{k+1}^{(j,j-1)}\ \ \Delta\mathbf{r}_{k+1}^{(j,j-2)}\ \ \dots\ \ \Delta\mathbf{r}_{k+1}^{(j,1)}\ \ \Delta\mathbf{r}_{k+1}^{(j,0)}\ \right], \tag{4.194}$$

the unknown residual difference is found to be $\Delta\mathbf{r}_{k+1}^{(j,j+1)} = \mathbf{V}_k^{(j)}\,\boldsymbol{\alpha}^{(j)}$ and

$$\boldsymbol{\alpha}_i = \left(\mathbf{V}_k^{(j)^{\mathrm{T}}}\,\mathbf{V}_k^{(j)}\right)^{-1}\mathbf{V}_k^{(j)^{\mathrm{T}}}\,\Delta\mathbf{r}_{k+1}^{(j,j+1)} = -\left(\mathbf{V}_k^{(j)^{\mathrm{T}}}\,\mathbf{V}_k^{(j)}\right)^{-1}\mathbf{V}_k^{(j)^{\mathrm{T}}}\,\Delta\mathbf{r}_{k+1}^{(j,j+1)}\boldsymbol{r}_{k+1}^{(j)} \tag{4.195}$$

can be used to compute the coefficient vector $\boldsymbol{\alpha}_i$. However, using (4.195) involves the expensive calculation of the inverse of $\left(\mathbf{V}_k^{(j)^{\mathrm{T}}}\,\mathbf{V}_k^{(j)}\right)$ and further does not reveal anything regarding the quality of the chosen basis vectors $\Delta\mathbf{r}_{k+1}^{(j,i)}$. Therefore, a **QR**-decomposition $\mathbf{V}_k^{(j)} = \mathbf{Q}^{(j)}\,\mathbf{R}^{(j)}$ is preferred over (4.195) and used to obtain $\boldsymbol{\alpha}^{(j)}$ from

$$\mathbf{R}^{(j)}\,\boldsymbol{\alpha}^{(j)} = -\mathbf{Q}^{(j)^{\mathrm{T}}}\,\boldsymbol{r}_{k+1}^{(j)}. \tag{4.196}$$

Finally, Eq. (4.193) can be reformulated. Introducing the matrix

$$\mathbf{W}_k^{(j)} = \left[\ \Delta\tilde{\mathbf{s}}_{k+1}^{(j,j)}\ \ \Delta\tilde{\mathbf{s}}_{k+1}^{(j,j-1)}\ \ \dots\ \ \Delta\tilde{\mathbf{s}}_{k+1}^{(j,2)}\ \ \Delta\tilde{\mathbf{s}}_{k+1}^{(j,1)}\ \right], \tag{4.197}$$

the solution increment is found to be

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\mathrm{QNLS}}{=} \mathbf{W}_k^{(j)}\,\boldsymbol{\alpha}^{(j)} + \boldsymbol{r}_{k+1}^{(j)}. \tag{4.198}$$

This is the basic variant of the QNLS method proposed in [51]. In the following, different means to improve the stability and efficiency of the method are introduced.

Data reusing Along with the basic variant, a straightforward improvement of the method is proposed in [51]. Depending on the problem characteristics, the data from previous time steps may be reused in the current one. Here, the variant is consequently denoted as *multi-time-step quasi-Newton least-squares method (MTQNLS)*. Introducing the matrices

$$\hat{\mathbf{W}}_k^{(j)} = \left[\ \mathbf{W}_k^{(j)} \quad \mathbf{W}_{k-1}^{(N_{k-1})} \quad \dots \quad \mathbf{W}_{k-\hat{n}}^{(N_{k-\hat{n}})}\ \right] \tag{4.199}$$

and

$$\hat{\mathbf{V}}_k^{(j)} = \left[\ \mathbf{V}_k^{(j)} \quad \mathbf{V}_{k-1}^{(N_{k-1})} \quad \dots \quad \mathbf{V}_{k-\hat{n}}^{(N_{k-\hat{n}})}\ \right], \tag{4.200}$$

where N_k denotes the number of iterations performed in time step k and $\hat{n}$ is the number of reused time steps, the solution increment is computed as

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\text{MTQNLS}}{=} \hat{\mathbf{W}}_k^{(j)}\, \hat{\alpha}^{(j)} + \boldsymbol{r}_{k+1}^{(j)}. \tag{4.201}$$

The coefficient vector $\hat{\alpha}^{(j)}$ again follows from a $\mathbf{QR}$-decomposition of $\hat{\mathbf{V}}_k^{(j)}$ as in the standard variant. A major drawback of the method is that the optimal number of reused time steps $\hat{n}$ is problem-dependent. Too large values will lead to instabilities, while too small values will not improve the efficiency significantly.

Resetting In order to address the problem-dependency of the parameter $\hat{n}$, a modification of the MTQNLS method was proposed in [163]. Therein, the matrices $\hat{\mathbf{W}}_k^{(j)}$ and $\hat{\mathbf{V}}_k^{(j)}$ are reset whenever the number of iterations in a time step exceeds a given maximum. This allows to use the method also for problems, where the optimal $\hat{n}$ changes over the course of the simulation. This is the case in cardiovascular FSI simulations, where the inflow boundary conditions demand for a small number of reused time steps during the systolic flow pulse while a larger number may be used in the diastolic phase with a low flow rate and less sudden changes. This variant of the method is denoted here as *quasi-Newton least-squares method with resetting (QNLSR)*.

Filtering Another possibility to circumvent the issue of a problem-dependent $\hat{n}$ was proposed in [87]. As suggested therein, the method is denoted here as *filtered quasi-Newton least-squares method (FQNLS)*. Following the observation that the instabilities due to a too large $\hat{n}$ are the result of a linear dependency between the columns (basis vectors) of the matrix $\hat{\mathbf{V}}_k^{(j)}$, they may be filtered before using them to compute the solution increment. To this end, the **R**-matrix of the **QR**-decomposition is utilized. Due to the fact that the diagonal elements of **R** can be used as an indicator for linear dependency of the columns of the matrix $\hat{\mathbf{V}}_k^{(j)}$ (cf. [87]), they can serve as the basis for the filtering – as follows:

1. Start with the standard matrices $\tilde{\mathbf{V}}_k^{(j)} = \hat{\mathbf{V}}_k^{(j)}$ and $\tilde{\mathbf{W}}_k^{(j)} = \hat{\mathbf{W}}_k^{(j)}$ from the QNLSMT method.
2. Compute the **QR**-decomposition of $\tilde{\mathbf{V}}_k^{(j)}$.
3. If the norm of a diagonal entry R_{ii} is smaller than an absolute tolerance $e_{\text{tol}}^{\text{QNLS}}$ or ${R_{ii}}/{\|\mathbf{R}\|}$ is smaller than a relative tolerance $e_{\text{rel}}^{\text{QNLS}}$, remove the ith column of $\tilde{\mathbf{V}}_k^{(j)}$ and $\tilde{\mathbf{W}}_k^{(j)}$ and go back to Step 2.

Once all diagonal entries have been found to be sufficiently large, coefficients $\tilde{\boldsymbol{\alpha}}^{(j)}$ are determined as usual and the solution increment is computed as

$$\Delta\mathbf{s}_{k+1}^{(j)} \overset{\text{FQNLS}}{=} \tilde{\mathbf{W}}_k^{(j)}\,\tilde{\boldsymbol{\alpha}}^{(j)} + \boldsymbol{r}_{k+1}^{(j)}. \tag{4.202}$$

The precise procedure of the different QNLS methods is best explained in terms of the pseudo code shown in Alg. 2. As can be seen, in the first iterations of the first time step (and in the first iteration of any time step without data reuse), a static relaxation is performed. The procedure in Alg. 2 may be considered a member function of a class, which further provides members for the data of previous iterations and time steps, such that any previously computed residual $\mathbf{r}_k^{(j)}$ or solution $\mathbf{s}_k^{(j)}$, respectively $\tilde{\mathbf{s}}_k^{(j)}$, is accessible in any call. Of course, in an actual implementation, only data from $\hat{n}$ previous time steps should be stored.

Special choices of the parameters lead to the different variants of the QNLS method introduced above. With $n_{\text{rst}} = \infty$, no resetting is performed. If further $\hat{n} > 0$, $e_{\text{rel}}^{\text{QLNS}} > 0$ and $e_{\text{abs}}^{\text{QLNS}} > 0$, the FQNLS method is obtained, while $\hat{n} > 0$ with $e_{\text{rel}}^{\text{QLNS}} = 0$ or $e_{\text{abs}}^{\text{QLNS}} = 0$ yields the QNLSMT method. The standard variant is obtained with $\hat{n} = 0$ and $e_{\text{rel}}^{\text{QLNS}} = 0$ or $e_{\text{abs}}^{\text{QLNS}} = 0$.

4.3.3 Predictors

When considering dynamic problems, the iterative procedure to be accelerated by the convergence acceleration method needs to be executed for each time step. In each time step, an appropriate initial value $\mathbf{s}_{k+1}^{(0)}$ has to

Algorithm 2 Implementation of different variants of the quasi-Newton least-squares method.

```
procedure COMPUTEMODIFIEDSOLUTION($\tilde{\mathbf{s}}^{(j+1)}_{k+1}$, $k$, $j$)    ▷ Corresp. to $\mathbf{s}^{(j+1)}_{k+1} = \mathcal{A}^{j}_{k} \circ \tilde{\mathbf{s}}^{(j+1)}_{k+1}$
    $\mathbf{r}^{(j)}_{k+1} \leftarrow \tilde{\mathbf{s}}^{(j+1)}_{k+1} - \mathbf{s}^{(j)}_{k+1}$

    if $j = 0$ then    ▷ Resetting
        $j_{\min} = 0$
    else if $j = n_{\text{rst}}$ then
        $j_{\min} = j$
        $k_{\min} = k$
    end if

    if $j - j_{\min} = 0$ and ( $k_{\min} = k$ or $\hat{n} = 0$ ) then
        return $\omega\, \mathbf{r}^{(j+1)}_{k+1}$    ▷ Static relaxation
    end if

    $\mathbf{V}^{(j+1)}_{k} \leftarrow [\,]$, $\mathbf{W}^{(j+1)}_{k} \leftarrow [\,]$
    for $i \leftarrow i_{\min} \dots j-1$ do
        $\Delta\mathbf{r}^{(j,i)}_{k+1} \leftarrow \mathbf{r}^{(i)}_{k+1} - \mathbf{r}^{(j)}_{k+1}$    ▷ Differences acc. to [51]
        $\Delta\tilde{\mathbf{s}}^{(j,i+1)}_{k+1} \leftarrow \tilde{\mathbf{s}}^{(i+1)}_{k+1} - \tilde{\mathbf{s}}^{(j+1)}_{k+1}$
        $\mathbf{V}^{(j+1)}_{k} \leftarrow \begin{bmatrix} \mathbf{V}^{(j+1)}_{k} & \Delta\mathbf{r}^{(j,i)}_{k+1} \end{bmatrix}$
        $\mathbf{W}^{(j+1)}_{k} \leftarrow \begin{bmatrix} \mathbf{W}^{(j+1)}_{k} & \Delta\tilde{\mathbf{s}}^{(j,i+1)}_{k+1} \end{bmatrix}$
    end for

    $\hat{\mathbf{V}}^{(j+1)}_{k} \leftarrow \mathbf{V}^{(j+1)}_{k}$, $\hat{\mathbf{W}}^{(j+1)}_{k} \leftarrow \mathbf{W}^{(j+1)}_{k}$    ▷ Data reusing
    for $i \leftarrow$ MAX($k - \hat{n}$, $k_{\min}$) $\dots k-1$ do
        $\hat{\mathbf{V}}^{(j+1)}_{k} \leftarrow \begin{bmatrix} \hat{\mathbf{V}}^{(j+1)}_{k} & \mathbf{V}^{(N_i)}_{i} \end{bmatrix}$
        $\hat{\mathbf{W}}^{(j+1)}_{k} \leftarrow \begin{bmatrix} \hat{\mathbf{W}}^{(j+1)}_{k} & \mathbf{W}^{(N_i)}_{i} \end{bmatrix}$
    end for

    $\tilde{\mathbf{V}}^{(j+1)}_{k} \leftarrow \hat{\mathbf{V}}^{(j+1)}_{k}$, $\tilde{\mathbf{W}}^{(j+1)}_{k} \leftarrow \hat{\mathbf{W}}^{(j+1)}_{k}$    ▷ Filtering acc. to [87]
    $\mathbf{Q}, \mathbf{R} \leftarrow$ QRDECOMPOSITION($\tilde{\mathbf{V}}^{(j+1)}_{k}$)
    $i \leftarrow$ INDEXOFSMALLESTDIAGONALELEMENT($\mathbf{R}$)
    while $\|R_{ii}\| < e^{\text{QNLS}}_{\text{abs}}$ or $\|R_{ii}\|/\|\mathbf{R}\| < e^{\text{QNLS}}_{\text{rel}}$ do
        $\tilde{\mathbf{V}}^{(j+1)}_{k} \leftarrow$ REMOVECOLUM($i$, $\hat{\mathbf{V}}^{(j+1)}_{k}$)
        $\tilde{\mathbf{W}}^{(j+1)}_{k} \leftarrow$ REMOVECOLUM($i$, $\hat{\mathbf{W}}^{(j+1)}_{k}$)
        $\mathbf{Q}, \mathbf{R} \leftarrow$ QRDECOMPOSITION($\tilde{\mathbf{V}}^{(j+1)}_{k}$)
        $i \leftarrow$ INDEXOFSMALLESTDIAGONALELEMENT($\mathbf{R}$)
    end while

    $\tilde{\alpha}^{(j)} = -\mathbf{R}^{(j)^{-1}}\, \mathbf{Q}^{(j)^{\mathrm{T}}}\, \boldsymbol{r}^{(j)}_{k+1}$    ▷ Quasi-Newton update
    return $\mathbf{s}^{(j)}_{k+1} \tilde{\mathbf{W}}^{(j)}_{k}\, \tilde{\alpha}^{(j)} + \boldsymbol{r}^{(j)}_{k+1}$
end procedure
```

be computed, preferably taking into account the solution of previous time steps to reduce the number of iterations. The predictors presented in the following can all be interpreted as extrapolations of the solution at previous time steps.

Polynomial extrapolation

The trivial case of taking the last converged solution to be the prediction for the new time step can be interpreted as a *constant extrapolation (CE)*. We obtain

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{CE}}{=} \mathbf{s}_k \tag{4.203}$$

as the corresponding prediction rule. Computing a linear function $\mathbf{f}(t) = \mathbf{a}_1^{\mathrm{LE}}\, t + \mathbf{a}_0^{\mathrm{LE}}$ that passes through $\mathbf{s}_k$ and $\boldsymbol{s}_{k-1}$ and setting $\mathbf{s}_{k+1}^{(0)} = \mathbf{f}(t_{k+1})$ gives a *linear extrapolation (LE)*, which in closed form reads

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{LE}}{=} \frac{\mathbf{s}_k - \mathbf{s}_{k-1}}{\Delta t_{k-1}}\, \Delta t_k + \mathbf{s}_k. \tag{4.204}$$

For a constant time step size,

$$\mathbf{s}_{k+1}^{(9)} \overset{\mathrm{LE}}{=} 2\,\mathbf{s}_k - \boldsymbol{s}_{k-1} \quad \text{if } \Delta t_k = \text{const.} \tag{4.205}$$

is obtained.

A *quadratic extrapolation (QE)* is obtained by evaluating a parabola that passes through the last three known solutions. To compute $\mathbf{s}_{k+1}^{(0)}$, the system

$$\begin{bmatrix} 1 & t_k & t_k^2 \\ 1 & t_{k-1} & t_{k-1}^2 \\ 1 & t_{k-2} & t_{k-2}^2 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0^{\mathrm{QE}} & \mathbf{a}_1^{\mathrm{QE}} & \mathbf{a}_2^{\mathrm{QE}} \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} \mathbf{s}_k & \mathbf{s}_{k-1} & \mathbf{s}_{k-2} \end{bmatrix}^{\mathrm{T}} \tag{4.206}$$

is solved for the coefficients $\mathbf{a}_i^{\mathrm{QE}}$ and then

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{QE}}{=} \mathbf{a}_2^{\mathrm{QE}}\, t_{k+1}^2 + \mathbf{a}_1^{\mathrm{QE}}\, t_{k+1} + \mathbf{a}_0^{\mathrm{QE}} \tag{4.207}$$

is used as a prediction. A closed form for the general case is not convenient. A constant time step size, however, simplifies the formula significantly and leads to

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{QE}}{=} 3\,\mathbf{s}_k - 3\,\mathbf{s}_{k-1} + \boldsymbol{s}_{k-2} \quad \text{if } \Delta t_k = \text{const.} \tag{4.208}$$

Note that in the first time steps of a simulation, a constant extrapolation is the only choice. The extrapolation order can then be increased in every time step until the desired order is reached. While it is possible to formulate extrapolations of higher orders in a similar manner to the quadratic case, they are not considered here as they were found to cause stability problems rather than accelerating the convergence process. A geometrical interpretation of the prediction methods is given in Fig. 4.15.

Tangent extrapolation

A method used in [51, 53, 56] is introduced here as tangent extrapolation (TE). The prediction for the case of a constant time step size given therein reads

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{TE}}{=} \frac{5}{2}\,\mathbf{s}_k - 2\,\mathbf{s}_{k-1} + \frac{1}{2}\,\mathbf{s}_{k-2} \quad \text{if } \Delta t_k = \text{const.} \tag{4.209}$$

A derivation may be based on the following steps (cf. Fig. 4.15):

1. Construct a parabola as the QE case using (4.206).
2. Construct a tangent of the parabola at t_k.
3. Evaluate the tangent at t_{k+1}.

The tangent is found to be

$$\mathbf{t}(t) = \mathbf{s}_k + \left(2\,\mathbf{a}_2^{\mathrm{QE}}\,t_k + \mathbf{a}_1^{\mathrm{QE}}\right)(t - t_k)\,, \tag{4.210}$$

such that for the general case,

$$\mathbf{s}_{k+1}^{(0)} \overset{\mathrm{TE}}{=} \mathbf{t}(t_{k+1}) = \mathbf{s}_k + \left(2\,\mathbf{a}_2^{\mathrm{QE}}\,t_k + \mathbf{a}_1^{\mathrm{QE}}\right)\Delta t_k \tag{4.211}$$

yields the prediction. As for higher order polynomial predictors, this method cannot be applied until $k \geq 2$ and a constant, respectively a linear extrapolation has to be performed in the first two time steps.

4.3.4 Convergence criteria

To ensure a converged solution in the current time step before proceeding with the next one, the difference between the coupling quantities in succeeding iterations is used as an error measure. However, not only the introduced residual has to be taken into account but also the change of other coupling

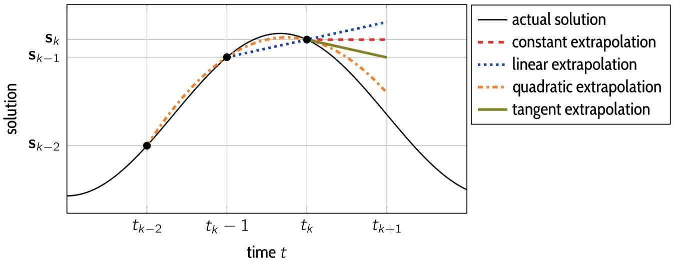


Figure 4.15: Prediction of the solution according to the different extrapolation methods.

quantities which, according to Fig. 4.14, are only passed from one field solver to another by interpolation procedures without further modification. The reason for this is the fact that, generally, the solvers only yield an approximate solution to the nonlinear problems they are designed to solve. Taking this into account, the operators introduced for the field solvers and likewise the combined solve operator in the fixed-point iteration (4.168) are not the same in every iteration. For the example of the two-field FSI problem, this means that a vanishing residual $\mathbf{r}_{k+1}^{(j)} = \mathcal{S}_k \circ \mathcal{F}_k \circ \mathbf{d}_{k+1}^{(j)} - \mathbf{d}_{k+1}^{(j)}$ does not exactly imply a vanishing residual

$$\hat{\mathbf{r}}_i^{(j)} = \mathbf{t}_{k+1}^{(j+1)} - \mathbf{t}_{k+1}^{(j)}. \tag{4.212}$$

In any partitioned multifield simulation, it is therefore necessary to check the residuals for all coupling quantities. In terms of the generic solution vector $\mathbf{s}_k^{(j)}$, a relative convergence criterion

$$\frac{\left\|\tilde{\mathbf{s}}_{k+1}^{(j+1)} - \mathbf{s}_{k+1}^{(j)}\right\|}{\left\|\tilde{\mathbf{s}}_{k+1}^{(j+1)}\right\|} < e^{\text{rel}} \tag{4.213}$$

and an absolute criterion

$$\|\tilde{\mathbf{s}}_{i+1} - \mathbf{s}_i\| < e^{\text{abs}} \tag{4.214}$$

are used in the present work. The coupling iterations are considered converged when for all coupling quantities at least one of the criteria is fulfilled.

4.3.5 Interpolation

Due to the possibly non-matching discretizations of the individual subproblems, the coupling quantities need to be interpolated. A typical example are the fluid loads. For the numerical methods used in this work (the FEM for the structural mechanics subproblem and the FVM for the fluid mechanics subproblem), they can easily be evaluated at the cell centers of the fluid mesh. However, they need to be prescribed at the quadrature points of the structural mesh. Here, such a mapping of a quantity (the fluid loads) from a set of source points (the cell centers) to a set of target points (the quadrature points) is generally referred to as an interpolation. This terminology is justified due to the fact that if a target point corresponds exactly to a source point, the value of the quantity is not altered by the mapping process, which is in line with the definition of an interpolation. The interpolation methods explained in the following are divided into two classes. *Point cloud interpolation* refers to methods that are established solely from the positions of source and target points, while *mesh based interpolations* take the underlying computational meshes into account.

Considering a general set of N^{s} sources values $\boldsymbol{v}_i^{\mathrm{s}} \in \mathbb{R}^d$, they may be collected in a matrix

$$\mathbf{V}^{\mathrm{s}} = \begin{bmatrix} \boldsymbol{v}_1^{\mathrm{s}} & \dots & \boldsymbol{v}_{N^{\mathrm{s}}}^{\mathrm{s}} \end{bmatrix}^{\mathrm{T}}. \tag{4.215}$$

Constructing a corresponding matrix $\mathbf{V}^{\mathrm{t}}$ that collects N^{t} target values $\boldsymbol{v}_i^{\mathrm{t}} \in \mathbb{R}^d$, any interpolation method considered here may be described using an interpolation matrix $\mathbf{I} \in \mathbb{R}^{N^{\mathrm{t}} \times N^{\mathrm{s}}}$ as

$$\mathbf{V}^{\mathrm{t}} = \mathbf{I}\,\mathbf{V}^{\mathrm{s}}. \tag{4.216}$$

The construction of $\mathbf{I}$ is inherent to the interpolation method and not necessarily performed explicitly. Yet, all interpolation methods obey the properties implied by this matrix formulation, i.e. linearity and independence of the d components.

Apart from the classification as meshless and mesh-based, interpolation methods may be *consistent*, which implies that a set of constant source values results in a set of constant target values. In terms of properties of the interpolation matrix, this implies that the sum of all coefficients in one row is equal to one, i.e

$$\sum_{j=1}^{n^{\mathrm{t}}} I_{ij} = 1 \quad \text{for } i = 1, \dots n^{\mathrm{s}}. \tag{4.217}$$

Consistency is a desired feature when interpolating intrinsic quantities such as tractions or displacements. It should be noted that if extrinsic quantities such as forces need to be mapped from one discretization to another, *conservative* methods have to be applied. They ensure that the sum of the forces at all points is the same for the source and the target discretization, which implies that all coefficients in one column of $\mathbf{I}$ is equal to one. Consequently, conservative mappings cannot have the interpolation property if $N^{\mathrm{t}} \neq N^{\mathrm{s}}$, see [210] for details.

Point cloud interpolation

Considering a general set of n^{s} source points $\boldsymbol{p}_i^{\mathrm{s}}$ and a set of n^{t} target points $\boldsymbol{p}_j^{\mathrm{t}}$, the point cloud interpolation methods are based on this geometry information alone. A connectivity between the source points (possibly nodes of a mesh) is not considered.

Nearest neighbor interpolation For this straightforward interpolation method, each target value is simply set to the source value that corresponds to the source point closest to the respective target point. However, for every target point, this source point or *nearest neighbor (NN)* has to be found in an efficient way. While looping through all source points for all target points to find the smallest distance constitutes a robust algorithm, a total of $(n^{\mathrm{s}}\, n^{\mathrm{t}})$ computations of distances is required. Organizing the points in a tree data structure reduces the number of distance computations remarkably. A so-called $k-$d tree, as proposed in [23], has become a standard method for this purpose, i.e. organizing points in k-dimensional space.

Barycentric interpolation A barycentric interpolation is closely related to the nearest neighbor interpolation. Instead of taking the value at the nearest neighbor to compute the value at a source point, its three nearest neighbors are used to compute the value. This is done by parameterizing the triangle spanned by the three nearest neighbors using barycentric coordinates. Denoting the indices of the three corresponding source points by a_i, b_i and c_i, the target value is computed by

$$\boldsymbol{v}_i^{\mathrm{t}} = \boldsymbol{v}_{a_i}^{s} + \alpha_i \, \frac{\boldsymbol{v}_{b_i}^{s} - \boldsymbol{v}_{a_i}^{\mathrm{s}}}{\left|\left|\boldsymbol{p}_{b_i}^{\mathrm{s}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right|\right|} + \beta_i \, \frac{\boldsymbol{v}_{c_i}^{\mathrm{s}} - \boldsymbol{v}_{a_i}^{\mathrm{s}}}{\left|\left|\boldsymbol{p}_{c_i}^{\mathrm{s}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right|\right|}, \tag{4.218}$$

where

$$\alpha_i = \left(\boldsymbol{p}_i^{\mathrm{t}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right) \cdot \left(\boldsymbol{p}_{b_i}^{\mathrm{s}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right) \tag{4.219}$$

and

$$\beta_i = \left(\boldsymbol{p}_i^{\mathrm{t}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right) \cdot \left(\boldsymbol{p}_{c_i}^{\mathrm{s}} - \boldsymbol{p}_{a_i}^{\mathrm{s}}\right) . \tag{4.220}$$

Special attention has to be paid when using the barycentric interpolation because the three source points will not necessarily span a triangle that contains the target point. Further, the triangles spanned by them may be degenerated to a line. To avoid extrapolation in such cases, the fourth nearest neighbor is added to the set of source points. For all combinations of three of these four points, it is then checked whether they form a valid triangle that contains the target point. If no such combination is found, another nearest neighbor is added, new possible combinations are checked, and so forth. While this can be computationally expensive considering a single target point, it is rarely the case that more than three nearest neighbors are needed. In most cases, the computational meshes provide a point distribution that excludes this case.

Inverse distance weighting Interpolation methods using an inverse distance weighting can generally be formulated as

$$\boldsymbol{v}\left(\boldsymbol{p}_i^{\mathrm{t}}\right) = \begin{cases} \frac{\sum_{j=1}^{n_i^{\mathrm{n}}} w_i^{(j)}\, \boldsymbol{v}_k^{\mathrm{s}}}{\sum_{j=1}^{n_i^{\mathrm{n}}} w_i^{(j)}} & \text{if } \left\|\boldsymbol{p}^{\mathrm{t}} - \boldsymbol{p}_j^{\mathrm{s}}\right\| \neq 0 \text{ for all } j \in N_i, \\ \boldsymbol{v}_j^{\mathrm{s}} & \text{if } \left\|\boldsymbol{p}^{\mathrm{t}} - \boldsymbol{p}_j^{\mathrm{s}}\right\| = 0 \text{ for some } j \in N_i, \end{cases} \tag{4.221}$$

where $w_i^{(j)}$ denotes a weighting function and N_i denotes the set of source point indices contributing to the interpolated value at the target point $\boldsymbol{p}_i^{\mathrm{t}}$. If used as a point cloud interpolation method, a certain number of nearest neighbors or all points within a certain distance to $\boldsymbol{p}_i^{\mathrm{t}}$ may be used. In [178], the inverse distance weighting function

$$w_i^{(j)} = \frac{1}{\left\|\boldsymbol{p}_i^{\mathrm{t}} - \boldsymbol{p}_j^{\mathrm{s}}\right\|^p} \tag{4.222}$$

was proposed. Based on the author's name, the method is also known as *Shepard interpolation.*

Radial basis functions Radial basis functions (RBF) are a very flexible method for meshless interpolation. The target values are computed as a weighted sum of radial basis functions θ as

$$\boldsymbol{v}_i^{\mathrm{t}} = \sum_{j=1}^{n^{\mathrm{s}}} \boldsymbol{\lambda}_j\, \theta_j \left(\left\|\boldsymbol{p}_j^{\mathrm{s}} - \boldsymbol{p}_i^{\mathrm{t}}\right\|\right) . \tag{4.223}$$

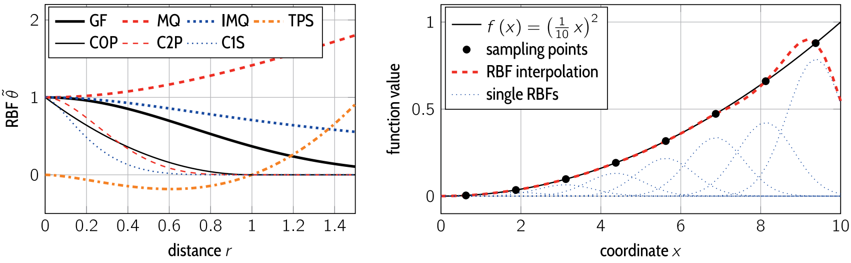


Figure 4.16: Left: Commonly applied radial basis functions (C0P, C2P, and C1S with $R_\mathrm{s} = 1$). Right: Interpolation of a parabola using 8 GF with $R_\mathrm{s} = 1$.

The weights $\boldsymbol{\lambda}_i \in \mathbb{R}^d$ are computed using the interpolation condition, i.e. by demanding that

$$\boldsymbol{v}_i^\mathrm{s} = \sum_{j=1}^{N^\mathrm{s}} \boldsymbol{\lambda}_j \, \theta_j \left(\left\|\boldsymbol{p}_j^\mathrm{s} - \boldsymbol{p}_i^\mathrm{s}\right\|\right). \tag{4.224}$$

Introducing the vector of weights

$$\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{\lambda}_1 & \dots & \boldsymbol{\lambda}_{n^\mathrm{s}} \end{bmatrix}^\mathrm{T}, \tag{4.225}$$

Eq. (4.224) can be stated in matrix notation as

$$\mathbf{V}^\mathrm{s} = \boldsymbol{\Theta}^\mathrm{s} \, \boldsymbol{\lambda}, \tag{4.226}$$

where

$$\Theta_{ij}^\mathrm{s} = \theta_j \left(\left\|\boldsymbol{p}_i^\mathrm{s} - \boldsymbol{p}_j^\mathrm{s}\right\|\right) \quad \text{with } i, j = 1, \dots n^\mathrm{s}. \tag{4.227}$$

Now, the coefficients can be computed as $\boldsymbol{\lambda} = \left(\boldsymbol{\Theta}^\mathrm{s}\right)^{-1} \mathbf{V}^\mathrm{s}$ and the target values can be obtained from

$$\mathbf{V}^\mathrm{t} = \boldsymbol{\Theta}^\mathrm{t} \, \boldsymbol{\lambda}, = \boldsymbol{\Theta}^\mathrm{t} \, \left(\boldsymbol{\Theta}^\mathrm{s}\right)^{-1} \mathbf{V}^\mathrm{s}, \tag{4.228}$$

where

$$\Theta_{ij}^\mathrm{t} = \theta_j \left(\left\|\boldsymbol{p}_i^\mathrm{t} - \boldsymbol{p}_j^\mathrm{s}\right\|\right) \quad \text{with } i = 1, \dots, n^\mathrm{t} \text{ and } j = 1, \dots n^\mathrm{s}. \tag{4.229}$$

Possible choices for the RBFs and an exemplary interpolation of a parabola are presented in Fig. 4.16. RBFs are commonly divided into functions with a compact support and functions with a global support.

A compact support implies a finite support radius $R_{\mathrm{s}}^{(j)}$ and $\theta_j\left(r\right) = 0$ for $r \geq R_{\mathrm{s}}^{(j)}$. The parameter $R_{\mathrm{s}}^{(j)}$ is introduced here for all basis functions by using

$$\theta_j\left(r\right) = \tilde{\theta}\left(\frac{r}{R_{\mathrm{s}}^{(j)}}\right) \tag{4.230}$$

as a general form. The meaning of the support radius is preserved by ensuring that, in the case of a compact support, $\tilde{\theta}(\tilde{r}) = 0$ for $\tilde{r} \leq 1$. Commonly used functions with this property include

- the C^0-*continuous polynomial (C0P)* with $\tilde{\theta} \overset{\mathrm{C0P}}{=} \left(1 - \tilde{r}\right)^2$,
- the C^2-*continuous polynomial (C2P)* with $\tilde{\theta} \overset{\mathrm{C0P}}{=} \left(1 - \tilde{r}\right)^4\left(4\,\tilde{r} + 1\right)$,
- the C^1-*continuous thin plate spline (C1S)* with $\tilde{\theta} \overset{\mathrm{C1S}}{=} 1 + \frac{80}{3}\,\tilde{r}^2 - 40\,\tilde{r}^3 + 15\,\tilde{r}^4 - \frac{8}{3}\,\tilde{r}^5 + 20\,\tilde{r}^2\ln\left(\tilde{r}\right)$.

More details on the origin of these RBFs can be found in [46, 47]. Commonly used functions without a compact support include

- the *Gaussian function (GF)* with $\tilde{\theta} \overset{\mathrm{GF}}{=} e^{-\tilde{r}^2}$,
- the *multiquadric (MQ)* with $\tilde{\theta} \overset{\mathrm{MQ}}{=} \sqrt{1 + \tilde{r}^2}$,
- the *inverse multiquadric (IMQ)* with $\tilde{\theta} \overset{\mathrm{IMQ}}{=} \frac{1}{\sqrt{1+\tilde{r}^2}}$,
- the *thin plate spline (TPS)* with $\tilde{\theta} \overset{\mathrm{TPS}}{=} \tilde{r}^2\ln\left(\tilde{r}\right)$.

A preliminary investigation of the RBFs is performed here in preparation of a more detailed analysis of their application for the mapping of coupling quantities (see Section 5.1.4) and as a mesh deformation method (see Section 5.2). The domain $[0, L]$ with $L = 10$ from Fig. 4.16 is kept as an example. The following functions are then evaluated at $n^{\mathrm{s}} = 9$ source or sampling points to be used as a basis for an interpolation with RBFs.

- Constant function $f(x) = 1$,
- parabola $f(x) = \left(\frac{2}{L}\,x - 1\right)^2$,
- step function $f(x) = \begin{cases} -1 & \text{if } x < \frac{L}{2} \\ 1 & \text{else} \end{cases}$.

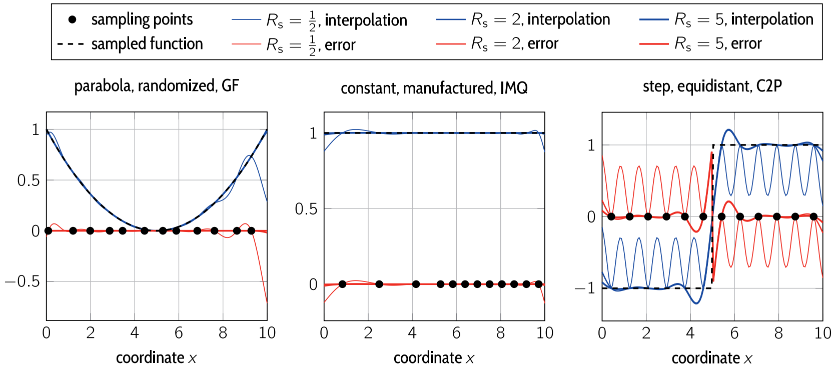


Figure 4.17: Exemplary interpolations from the study of radial basis functions and corresponding errors.

The sampling points are distributed according to one of the following schemes.

- Equidistant, with $x_i^{(\mathrm{e},L,n)} = \frac{L}{2\,n^{\mathrm{s}}} + L\left(1 - \frac{1}{n^{\mathrm{s}}}\right)\frac{i}{n^{\mathrm{s}}}$,
- randomized, with $x_i^{(\mathrm{r},L,n)} = x_i^{(\mathrm{e},L)} + e^{\mathrm{rand}}\frac{L}{2\,n^{\mathrm{s}}}$,
- manufactured, with $x_i^{(\mathrm{m},L,n)} = \begin{cases} x_i^{(\mathrm{e},\frac{L}{2},\frac{n}{4})} & \text{if } i \leq \frac{n}{4}, \\ \frac{L}{2} + x_i^{(\mathrm{e},\frac{L}{2},\frac{3\,n}{4})} & \text{else.} \end{cases}$

Using the RBFs introduced above and increasing the support radius from $R_{\mathrm{s}} = {}^{1}\!/\!_{4}$ (${}^{1}\!/\!_{4}$ of the mean source point distance) to $R_{\mathrm{s}} = 5$ in ten steps, interpolations as the ones shown in Fig. 4.17 are obtained. A global quality measure is obtained for each interpolation by evaluating it at $n^{\mathrm{t}} = 150$ equally distributed target points with $x_i^{\mathrm{t}} = {}^{L\,i}\!/\!_{n^{\mathrm{t}}}$ (the one-dimensional counterpart to $\boldsymbol{p}^{\mathrm{t}}$), which yields n^{t} target values y_i^{t} (the one-dimensional counterpart to $\boldsymbol{v}_i^{\mathrm{t}}$). Then, the mean interpolation error is computed as

$$e^{\mathrm{i}} = \sum_{i=1}^{n^{\mathrm{t}}} \sqrt{\frac{\left\|\boldsymbol{f}(\boldsymbol{p}_i^{\mathrm{t}}) - \boldsymbol{v}_i^{\mathrm{t}}\right\|^2}{n^{\mathrm{t}}}} \overset{\mathrm{1d}}{=} \sum_{i=1}^{n^{\mathrm{t}}} \sqrt{\frac{\left(f(x_i^{\mathrm{t}}) - y_i^{\mathrm{t}}\right)^2}{n^{\mathrm{t}}}}. \tag{4.231}$$

Figure 4.18 shows the errors for all considered interpolations. Included are the results for interpolations that were performed with an adaptive support

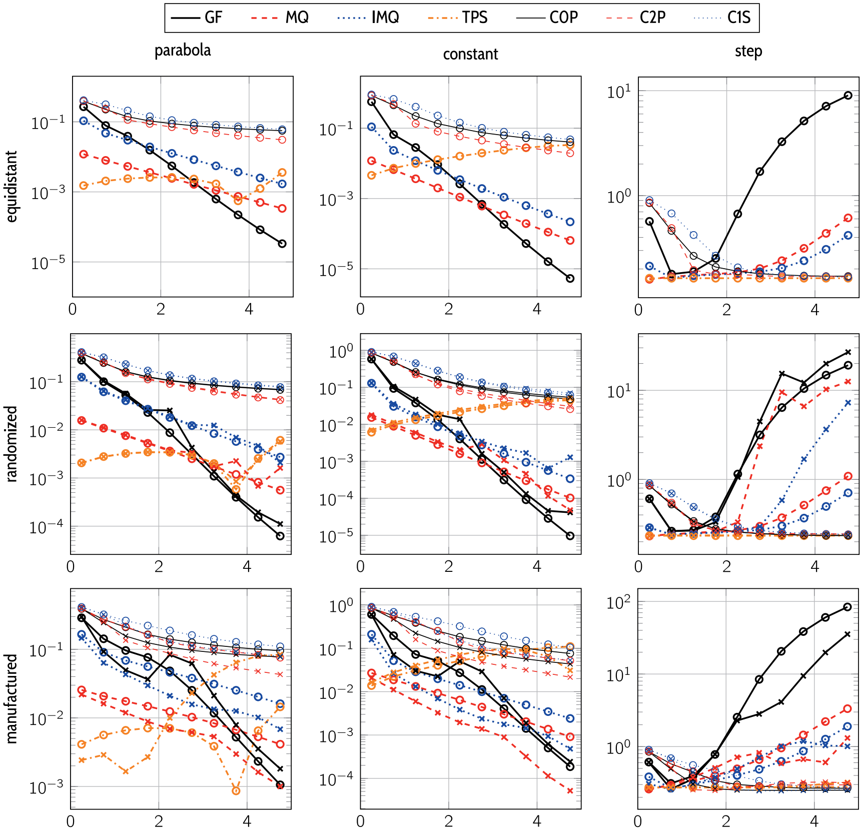


Figure 4.18: Interpolation error e^{i} (ordinates) with increasing mean support radius R_{s} (abscissas) for the interpolation of differently distributed source points sampling different functions. Marks × denote studies with adaptive $R_{\mathrm{s}}^{(j)}$, marks ○ denote studies with uniform $R_{\mathrm{s}}^{(j)} = R_{\mathrm{s}}$.

radius $R_s^{(j)}$, which was computed for each RBF by

$$R_s^{(j)} = (x_{j+1} - x_{j-1})\,\frac{1}{2\,L}\,R_{\mathrm{s}}^{(j)}. \tag{4.232}$$

This leads to a mean support radius

$$\sum_{i=1}^{n^{\mathrm{s}}} \frac{R_s^{(i)}}{n^{\mathrm{s}}} = R_s, \tag{4.233}$$

which was kept as the abscissa in Fig. 4.18. Since the mean support radius reflects a measure for the mean bandwidth of the system to be solved, this is regarded a fair comparison with the case of uniform support radii. From the results, the following conclusions can be deduced.

- For any type of RBF and source point distribution, the step function cannot be approximated satisfactory by the interpolation.
- The RBFs GF, MQ, and IMQ show a divergent behavior for the step function.
- The TPS shows an undesired convergence (divergence) behavior also for the parabola and the constant function – with increasing R_s, the error may increase.
- The RBFs with a compact support (C0P, C2P, and C1S) show a slower convergence than the RBFs with a global support.
- The GF shows the best convergence rate, while the TPS, the MQ function, and the IMQ function (in that order) show the lowest error for small support radii.
- For any convergent case, the adaptive computation of support radii yields an improvement. The largest improvement is observed for the manufactured source point distribution.

In addition to the mean interpolation error e^i, the maximum error, i.e. the maximum $\|f(x_i^\mathrm{t}) - y_i^\mathrm{t}\|$ was investigated, showing a similar convergence behavior (see Fig. A.5.2 in Section A.5). Only the MQ function is used for further investigations and applications of the RBF interpolation, as it yields the best compromise between convergence rate (better only for the GF) and accuracy for small support radii (better only for the TPS). It is used in combination with an adaptive support radius, which for every source point is set to be twice the mean distance to the two closest source points.

Mesh-based interpolation

Mesh-based interpolation makes use of the underlying computational mesh. In the best case, not only the geometry defined by the mesh is taken into account, but the interpolation method of the respective numerical method is adopted as well. A general algorithm for mesh-based interpolation then consists of two steps.

Algorithm 3 Robust projection of a point onto a mesh with elements of arbitrary geometry.

```
procedure PROJECTTOMESH(mesh, $\boldsymbol{p}$)    ▷ returns projected point $\boldsymbol{p}^{\mathrm{p}}$
    $\tilde{\boldsymbol{p}} \leftarrow$ FARAWAYFROM($\boldsymbol{p}$)
    for each face : mesh.fsiRegion.faces do    ▷ loop in a reasonable order
        $r, s \leftarrow$ PROJECTTOFACE(face, $\boldsymbol{p}$)
        if $-1 < r < 1$ and $-1 < s < 1$ then
            return $\boldsymbol{f}^{(e)}(r, s)$    ▷ see Eq. (4.56) for the $p$-FEM case
        end if
        $r \leftarrow$ MAX($-1$, MIN($1, r$))
        $s \leftarrow$ MAX($-1$, MIN($1, s$))
        if $\|\tilde{\boldsymbol{p}}^{\mathrm{p}} - \boldsymbol{p}\| > \|\boldsymbol{f}^{(e)}(r, s) - \boldsymbol{p}\|$ then
            $\tilde{\boldsymbol{p}}^{\mathrm{p}} \leftarrow \boldsymbol{f}^{(e)}(r, s)$
        end if
    end for
    for each edge : mesh.fsiRegion.edges do    ▷ loop in a reasonable order
        $r \leftarrow$ PROJECTTOEDGE(edge, $\boldsymbol{p}$)
        if $-1 < r < 1$ then
            return $\boldsymbol{e}^{(e)}(r)$    ▷ see Eq. (4.55) for the $p$-FEM case
        end if
        $r \leftarrow$ MAX($-1$, MIN($1, r$))
        if $\|\tilde{\boldsymbol{p}}^{\mathrm{p}} - \boldsymbol{p}\| > \|\boldsymbol{e}^{(e)}(r) - \boldsymbol{p}\|$ then
            $\tilde{\boldsymbol{p}}^{\mathrm{p}} \leftarrow \boldsymbol{e}^{(e)}(r)$
        end if
    end for
    return $\tilde{\boldsymbol{p}}^{\mathrm{p}}$
end procedure
```

1. Projection of the target points onto the source mesh.
2. Evaluation of a given interpolation of the source values.

When mapping displacements from the structural mesh to the fluid mesh's nodes, the finite element shape functions are used in the second step. Considering the mapping of tractions from the fluid mesh to the structural mesh's quadrature points, an interpolation is constructed using inverse distance weighting. The projection step is the same in both cases.

Projection of points onto a mesh One can implement a simple and robust algorithm to solve this task, see Alg. 3. While it is not applicable in practice due to efficiency reasons, the following considerations constitute the basis for the improved version discussed afterwards.

- A target point $\boldsymbol{p}^{\mathrm{t}}$ has to be projected onto a single element as shown

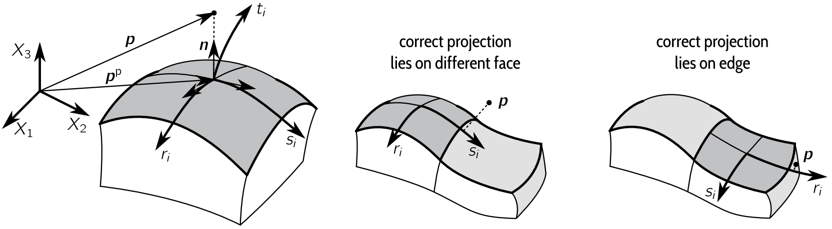

X_3
X_1
X_2
p
p^p
n
t_i
r_i
s_i
correct projection lies on different face
p
r_i
s_i
correct projection lies on edge
p
s_i
r_i

Figure 4.19: Projection of a point onto the surface of an arbitrarily curved hexahedral.

in Fig. 4.19. The corresponding minimization problem and its solution with the Newton-Raphson method are explained below. This is a computationally expensive task with respect to the high-order geometry description used for the structural mechanics subproblem.

- According to the domain $[-1, 1]$, which is the functional domain for all element parameterizations introduced in this chapter, the projection according to Fig. 4.19 may yield local coordinates outside of this domain, indicating that the point should actually be projected onto another element.

- If it is not possible to find an element for which the projection yields local coordinates inside the domain $[-1, 1]$, the point has to be projected onto the edges, see Fig. 4.19. Again, the projection result has to be checked.

- f the projection fails for all edges too, the local coordinate is set to the respective boundary value -1 or 1, which can be seen as another, final projection to the edge's node.

Considering the mapping of coupling quantities from one discretization to another, it is safe to assume only a small number of points for which this hierarchy of projections has to be performed all the way to the end. However, already the first step consumes an unacceptable amount of computational resources if performed for every element.

The first improvement of Alg. 3 therefore lies in excluding elements for which a valid projection is impossible. However, allowing for truly arbitrary element geometries, it is always possible to construct extreme cases that cause a given filtering technique to fail. Focusing on reasonable element geometries, it was found that axis-aligned bounding boxes, which are enlarged by the maximum projection distance to be expected, yield a valid criterion for the exclusion. To efficiently find all elements whose bounding box contains a given target point, they are organized in a so-called *AABB*-tree data structure, see e.g. [125, 43] for details. The first of the above subtasks is then performed for each of these elements. To further improve the efficiency, the local coordinates are set to the respective extreme value for each element instead of continuing with the second step if all projections fail. The best result, i.e the point on an element closest to the target point, is taken to be the projected point if the distance is below a given tolerance. The introduction of this tolerance is justified in view of the finite convergence criteria prescribed for the coupling algorithm and the individual field solvers.

Concerning the projection of a point to a face and an edge, a minimization problem needs to be solved to obtain the local face coordinates $\boldsymbol{r}^{(f)} = \begin{bmatrix} r & s \end{bmatrix}$ and the edge coordinate r, respectively, for a given point $\boldsymbol{p}^{\mathrm{t}}$. Following [123], the projection onto a face can be formulated as the minimization problem

$$\frac{|\boldsymbol{e}^{\mathrm{p}}|^2}{2} = \frac{1}{2}\left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) \cdot \left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) \rightarrow \min, \tag{4.234}$$

where $\boldsymbol{p}^{\mathrm{p}}(\boldsymbol{r}^{\mathrm{p}})$ is the projected point. Equation (4.234) leads to

$$\frac{\partial}{\partial \boldsymbol{r}}\frac{|\boldsymbol{e}^{\mathrm{p}}|^2}{2} = \begin{bmatrix} \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial r} \cdot (\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}) \\ \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial s} \cdot (\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}) \end{bmatrix} = 0. \tag{4.235}$$

This nonlinear system of equations is solved using the Newton-Raphson method, i.e. starting with an initial guess $\boldsymbol{r}_0^{\mathrm{p}}$ and solving

$$\left.\frac{\partial^2}{\partial \boldsymbol{r}^2}\frac{|\boldsymbol{e}^{\mathrm{p}}|^2}{2}\right|_{\boldsymbol{r}=\boldsymbol{r}_i^{\mathrm{p}}} \boldsymbol{\Delta r}^{\mathrm{p}} = -\left.\frac{\partial}{\partial \boldsymbol{r}}\frac{|\boldsymbol{e}^{\mathrm{p}}|^2}{2}\right|_{\boldsymbol{r}=\boldsymbol{r}_i^{\mathrm{p}}} \tag{4.236}$$

and updating

$$\boldsymbol{r}_{i+1}^{\mathrm{p}} = \boldsymbol{r}_i^{\mathrm{p}} + \boldsymbol{\Delta r}^{\mathrm{p}} \tag{4.237}$$

iteratively, until $\boldsymbol{\Delta r}$ falls below a given tolerance. The partial derivatives

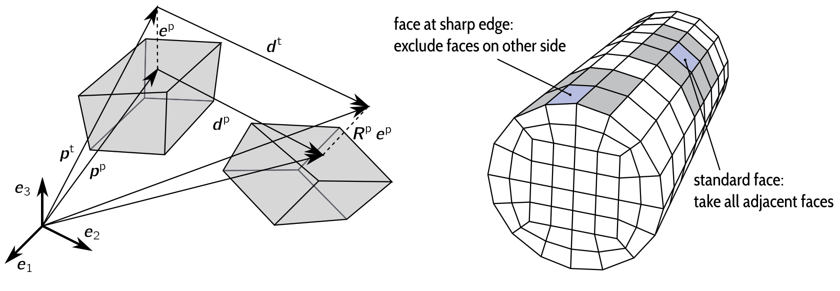


Figure 4.20: Mesh-based interpolation. Left: Correction of evaluations of displacements at projected points. Right: Determination of neighboring faces for mesh-based inverse distance weighting.

on the left hand side of (4.236) are found to be

$$\frac{\partial^2}{\partial \boldsymbol{r}^2}\frac{|\boldsymbol{e}^{\mathrm{p}}|^2}{2} = \begin{bmatrix} \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial r}\cdot\frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial r} + \frac{\partial^2 \boldsymbol{p}^{\mathrm{p}}}{\partial r^2}\cdot\left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) & \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial r}\cdot\frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial s} + \frac{\partial^2 \boldsymbol{p}^{\mathrm{p}}}{\partial r\,\partial s}\cdot\left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) \\ \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial r}\cdot\frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial s} + \frac{\partial^2 \boldsymbol{p}^{\mathrm{p}}}{\partial r\,\partial s}\cdot\left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) & \frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial s}\cdot\frac{\partial \boldsymbol{p}^{\mathrm{p}}}{\partial s} + \frac{\partial^2 \boldsymbol{p}^{\mathrm{p}}}{\partial s^2}\cdot\left(\boldsymbol{p}^{\mathrm{p}} - \boldsymbol{p}^{\mathrm{t}}\right) \end{bmatrix}. \tag{4.238}$$

Interpolation and evaluation Having found the projected point $\boldsymbol{p}^{\mathrm{p}}$, i.e. the corresponding local coordinates $\boldsymbol{r}^{\mathrm{p}}$, the quantity in consideration has to be computed at this point. Two scenarios are possible regarding the coupled simulation considered here.

1. If displacements are to be evaluated at $\boldsymbol{p}^{\mathrm{p}}$, the underlying numerical method is the FEM. A description for a continuous displacement field is therefore already given in terms of the shape functions. With the local coordinates from the projection step, Eq. (4.7) can be used for the evaluation step yielding

$$\boldsymbol{d}^{\mathrm{p}} = \mathbf{N}^{(e)}\left(\boldsymbol{r}^{\mathrm{p}}\right)\mathbf{d}^{(e)}. \tag{4.239}$$

   The degrees of freedom collected in $\mathbf{d}^{(e)}$ may be interpreted as the interpolated source values. However, this interpretation is only justified for nodal finite element methods, e.g. those applying Lagrange shape functions. In general, the degrees of freedom are weights which do not correspond to the displacement at certain points. The physical interpretation of the quantity as a displacement gives rise to a correction. Introducing a local rotation matrix $\boldsymbol{R}^{\mathrm{p}}$, possibly obtained from a polar decomposition of the deformation gradient, the evaluated displacement $\boldsymbol{d}^{\mathrm{p}}$ at $\boldsymbol{p}^{\mathrm{p}}$ should not be used directly as the displacement $\boldsymbol{d}^{\mathrm{t}}$ at $\boldsymbol{p}^{\mathrm{t}}$. Instead,

$$\boldsymbol{d}^{\mathrm{t}} = \boldsymbol{d}^{\mathrm{p}} + \left(\boldsymbol{R}^{\mathrm{p}} - \mathbf{1}\right) \boldsymbol{e}^{\mathrm{p}} \tag{4.240}$$

is used to avoid errors in cases of large rotations, as illustrated in Fig. 4.20.

2. If tractions are to be evaluated at $\boldsymbol{p}^{\mathrm{p}}$, the underlying numerical method is the FVM. As explained in Section 4.1.2, the degrees of freedom then correspond to the values at the face centers of the mesh. In this case, the inverse distance weighting method is used to compute the tractions at $\boldsymbol{p}^{\mathrm{p}}$ from the value at the face center of its parent finite volume and neighboring volumes. Then, the interpolation method is mesh-based, due to the fact that not all source points but only those corresponding to centers neighboring faces are taken into account when constructing the interpolation according to 4.221. Further, instead of taking all neighboring faces, only those with a normal close to the normal of the parent finite volume are considered to set up the inverse distance weighting interpolation. As illustrated in Fig. 4.20, this prevents issues associated with sharp edges, where the traction on either side should be described without taking the other side into account in order to avoid an influence of the discontinuity of the normal vector. In Section 5.1.4, this is addressed in more detail.

4.4 Coupling software

As mentioned previously, the coupling software *comana* is used to steer the participating solvers in a multifield simulation. It controls the data transfer, i.e. the exchange of coupling quantities, between the solvers and provides data structures for a convenient implementation of coupling algorithms. Initially, *comana* was introduced as a coupling manager, which was designed specifically for FSI problems (see [28]) and since then has matured to a C++ framework, which is generally applicable in the context of partitioned simulations, see [122].

The framework is split into a *core library* and *adapter libraries*, one of which is provided for each supported field solver. The core library provides algorithms and data structures for the implementation of a *coupling* *manager process* (CMP), which includes the coupling algorithm. Further, representations or handles the participating field solvers are created in such a CMP. They allow to remote control of the field solvers, or *field solver processes* (FSPs), i.e. through the handles, a solver may be told to

1. receive and set boundary condition values at $t_{k+1} = t_k + \Delta t_k$,
2. solve for the solution at t_{k+1},
3. evaluate and send the solution at t_{k+1},
4. proceed with the next time step, i.e. increase k.

With the above tasks, it is possible to realize all solver-specific tasks apparent in the general coupling algorithm (see Algorithm 1), e.g. the solve operator $\tilde{\boldsymbol{s}}_{k+1}^{(j+1)} = \mathcal{X}_k \circ \boldsymbol{s}_{k+1}^{(j)}$. More precisely, each evaluation of the solve operator corresponds to an execution of the first three tasks. Once a coupling iteration is converged, the fourth task is executed and the simulation proceeds with the next time step.

The main challenges in implementing a coupling software that allow to remote control field solvers in this way can be seen in

- creating a software design that allows for a convenient implementation of new adapter libraries,
- supporting field solvers that are programmed in different programming languages,
- providing an inter process communication mechanism that allows for parallelized solvers and does not rely on files,
- developing data structures that allow for a convenient and safe implementation of coupling algorithms.

In the following, an overview about how these aspects are addressed in *comana* is given. After a summary of the overall software design, the most important concepts are explained in the next section. The most important design choices regarding the implementation of coupling algorithms and the manipulation of field solvers are explained in Sections 4.4.3 and 4.4.4, respectively. They were found to yield a flexible framework and may be reused in other software projects that realize similar tasks.

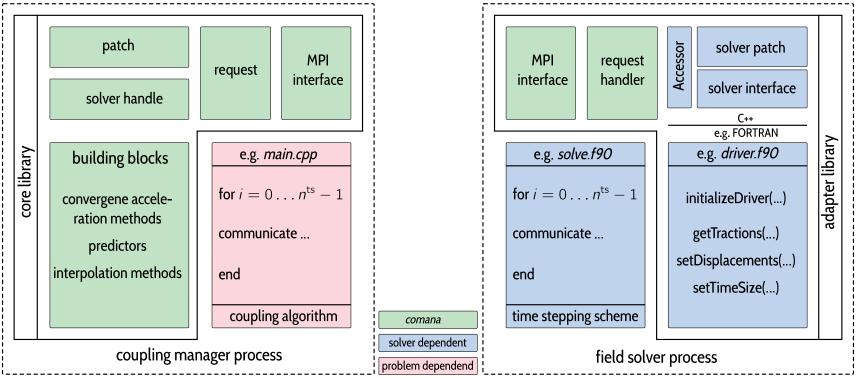


Figure 4.21: Software design of the coupling manager *comana* introduced in [122].

4.4.1 Software design

Fig. 4.21 illustrates the split into core and adapter libraries and their usage by the CMP and the FSPs. The CMP relies on the functions and data structures from the core library in order to implement the coupling algorithm. An FSP constitutes a modified field solver, i.e. a field solver, to which the respective adapter library was linked. Through the adapter library, the field solver is enabled to communicate with the CMP. The concept of fields and patches is used here as well.

From a numerical point of view, the most important data structures are the representations of the different convergence acceleration methods, predictors, interpolation methods, and convergence criteria that were introduced in Section 4.3 and denoted as *building blocks* in Fig. 4.21. In *comana*, each of these building blocks is implemented as a class, providing member functions that execute the respective operation. They can all be viewed as data processing blocks that work on arrays or vectors. Outside an implementation context, they were correspondingly introduced using operators (e.g. $\mathcal{A}$ for an arbitrary convergence acceleration method).

From an implementation point of view, data structures that do not directly represent one of these building blocks constitute the core of the software. They realize a seamless integration of the building blocks into the communication framework. On a low level, this includes container classes for the processed data (the Field class). The most important higher level classes are those implementing solver handles and representations of discretized coupling interfaces or mesh regions (the classes SolverHandle and Patch). In the following, the concepts behind these classes is explained. Details on the implementation are provided in [122].

Fields

The main container class in *comana* is the Field class. As the name suggests, a Field stores the field data, which is sent to and received from the solvers. In general, the exchanged data are evaluations of scalar, vector or higher order tensor fields at certain points in space. The total number of scalars contained in a field accordingly depends on the tensor order, on the global space dimension, and on the number of points. During the development of the coupling software, certain design choices for the Field class were found to ease the seamless integration of the building blocks.

- The data should be stored linear in memory, i.e. as a one-dimensional array. This is the only storage order that is directly supported by most inter-process communication protocols. Further, this is the way that building blocks such as predictors and convergence acceleration methods access the data (the corresponding operators process column vectors, see e.g. Eq. (4.160)).

- The data should still be accessible on a per-point basis, e.g. if a vector field is considered, the quantity at point i should be accessible as a data structure that represents a vector. This eases the implementation of building blocks such as as interpolation methods, which rely on the association of the evaluated quantity with a location in space.

The above points are realized in *comana* using a *view pattern* (see e.g. [4]). For its implementation, the Field class draws on the container classes of the C++ standard library and the linear algebra package *Eigen* (see [86]). This concept of the Field class minimizes the need for copying large data sets on their way from within the coupling algorithm to the field solvers database. In fact, only a single copy is made for some of field solvers, namely when exchanging the data between the CMP and the FSP. For most field solvers, however, additional copies within the adapter library cannot be circumvented, e.g. due to different programming languages.

Solver handles

The SolverHandle class is an abstract representation of a field solver within the CMP. To enhance the readability of this section, its name was changed to place emphasis on this view from the outside. For the explanation of the internally underlying concepts, the actual name SolverProcessGroup is more suitable. It pronounces the fact that, generally, a field solver may be parallelized such that more than one FSP is associated with it. Sending data to or receiving data from a field solver may accordingly involve communication with a group of processes, as illustrated in Fig. 4.22. Further, the exchanged data, e.g. fields, have to be scattered or gathered due to the fact that the different processes of a field solver usually represent only a part of the computational domain of the respective subproblem. As explained in more detail in Section 4.4.2, a suitable initialization of these parts of a computational domain constitutes the main challenge.

Patches

Irrespectively of a parallelization of a field solver, in the scope of surface coupled problems, only a part of the boundary of the underlying computational domain represents the coupling interface (the wet surface in FSI problems). A coupling interface is represented by the Patch class in *comana*. Fields are exchanged between the CMP and the FSP on the basis of patches. A solver patch is part of the adapter library for any (mesh-based) solver. For its implementation, a generic solver patch is provided, which allows for a flexible arrangement of tasks such as coupling quantity evaluation and mesh based interpolation. The *flexible arrangement* is explained in more detail in Section 4.4.4. As depicted in Fig. 4.22, a possible parallelization of the field solvers is hidden from the user by the Patch class. Under the hood, it is associated with the individual parts of the coupling interface, each of which is associated with one FSP. While usually, there is a one-to-one relation between patches of different field solvers, there is usually no such relation between the individual parts of the patch within the FSPs.

4.4.2 Inter process communication

Communication between the coupling manager process (CMP) and the field solver process (FSP) is realized using the open source implementation *Open-*

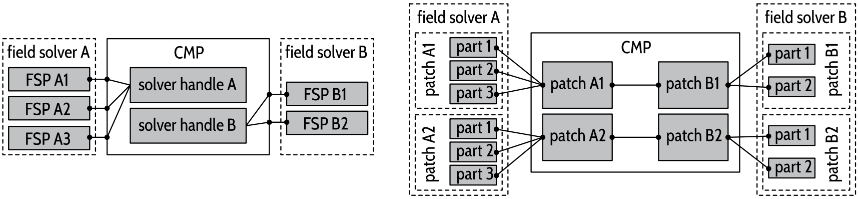


Figure 4.22: Concepts of field solver handles (left) and patches (right) in the scope of parallelized field solvers.

MPI (see [198]) of the *message passing interface (MPI) standard* [8]. It allows for different communication concepts and constitutes a widely accepted standard for handling data exchange between processes. In comana, communication is established based on a server-client model, allowing to run the CMP (server) and the FSPs (clients) independently and on different machines if desired. The MPI standard defines interfaces for sending and receiving arrays of standard data types such as characters, integers, and floating point numbers that are supported by most programming languages. In *comana*, these interfaces are wrapped, in order to send and receive fields and several other data structures.

On a higher level, the communication concept is based around *requests*, which are issued by the CMP (the server side). On the client side, they are handled by a *request handler* as illustrated in Fig. 4.21. The four tasks introduced at the beginning of this section are realized by the following requests.

Solve Tells the solver to solve the respective subproblem.

GetField Tells the solver to send a certain quantity field.

SetField Tells the solver to receive a certain quantity field.

Proceed Tells the solver to advance the state variable in time.

Further, the following requests are used to deal with adaptive time step sizes, which may be proposed by the coupling manager or the field solvers.

GetTimeStepSize Tells the solver to send the time step size.

SetTimeStepSize Tells the solver to receive a new time step size and to use it for the next Solve and Proceed requests.

Finally, requests for the initialization of mesh regions are needed (each field is associated with a certain mesh region). The request InitializePatch tells the solver to prepare a certain mesh region to participate in the coupling. Depending on the solver and the mentioned arrangement of tasks, representations of the contained elements and accessors for coupling quantities are created on the client side.

4.4.3 Implementation of coupling algorithms

With the introduced concepts, a coupling algorithm can be implemented in terms of a speaking source code, which likewise serves as an input file to a coupled simulation case. Aiming at academic research as the main application area, this design of *comana* as a C++ framework feels appropriate. Rather than parsing less readable XML files or similar input files, which are purely based on data, this allows to draw on the full capacities of C++ and use conditions, loops, etc. as desired. A coupling algorithm can then be implemented in a direct correspondence to the pseudocode introduced in Alg. 1. Listings 1 and 2 show the initialization section and the coupling algorithm section of an exemplary source code, respectively. With a few renamings, the code can be used directly in the context of a coupled simulation with the fluid solver *pimpleDyMFoam* (see [154] and Section 4.1.2) and the structural solver *AdhoC* (see [65] and Section 4.1.1). In Listing 3, a corresponding source code for a parallel coupling algorithm is given.

In the source codes, the two enum classes Location and Quantity are used very frequently. They are attached to every setField and getField request and the respective combination defines the local dimensionality of the send or received field. Requests for a combination that is not provided by a field solver will throw an exception. Setting the quantity *position* at the location *custom* tells a solver to initialize a mesh-based interpolation for the respective points, see Listing 1 (Rows 25, 26). Coupling quantities can then be evaluated directly at these points, see Listing 2 (Rows 59, 64) or Listing 3 (Rows 63, 64).

4.4.4 Field solver manipulation

In order for a standard field solver to participate in a coupled simulation, it has to be manipulated. This is done in a minimal invasive way by implementing all routines and classes needed for the communication with the

CMP and the handling of requests in the adapter library (see Fig. 4.21). Algorithms 4 and 5 show the pseudo-code for an original field solver and a modified one, respectively. The existence of a loop over time steps is the only requirement assumed to be met by any dynamic field solver. Apart from the implicit loop, the field solver manipulation consists of five function calls. The following four functions are the same for all field solvers.

initializeCoupling() Establishes the connection to the CMP via MPI.

communicateBeforeSolve() Waits for requests from the CMP and handles them. Returns on request Solve.

communicateAfterSolve() Waits for requests from the CMP and handles them. Returns true on Proceed, false on request Solve.

finalizeCoupling() Closes the connection to the CMP via MPI.

They are provided for all supported programming languages (C++, C, Fortran, MATLAB, Python), where the non-native (non-C++) variants constitute wrapper functions to the native (C++) variants.

Algorithm 4 Original implementation of a dynamic field solver.

```

problem ← setupProblem()

for $k = 1 \dots n^{\text{ts}}$ do


    solveProblem()




    updateState()
end for

```

Algorithm 5 Modified implementation of a dynamic field solver.

```
initializeCoupling()
problem ← setupProblem()
initializeDriver(problem)
for $k = 1 \dots n^{\text{ts}}$ do
    while true do
        communicateBeforeSolve()
        solveProblem()
        if communicateAfterSolve() then
            break
        end if
    end while
    updateState()
end for
finilizeCoupling()
```

The function initializeDriver() has to be implemented for each field solver and is usually directly implemented in the field solvers' programming language. Its purpose is to store a global reference to the solver data base (the data structure problem in the example in Alg. 5) such that it can be accessed later on from within functions communicateBeforeSolve() and communicateAfterSolve(). As depicted in Fig. 4.21, the solver data base itself as well as the MPI interface must be reached through these functions. Whether this is achieved purely in the native programming language of the field solver or via a C++ interface, which is common to all adapter libraries, constitutes an important design choice.

Both alternatives were tested in *comana*, and the one involving a common C++ interface was found to be favorable. At first sight, especially if only a small number of field solvers, possibly implementing completely different numerical methods, is considered, adapter libraries that are completely based on the field solvers programming language may be more suitable. However, an increasing number of solvers makes it difficult to draw on similarities between them. In the development process of *comana*, the similarity of being mesh-based, which is shared between many of the considered field solvers, was crucial for the decision to employ a common C++ interface. At the cost of exchanging data between different programming languages this allows to reuse algorithms and data structures needed, e.g. for the following task:

- Handling requests such as Solve and Proceed, which are independent of the field solvers numerical method.
- Projecting target points onto a mesh as explained in Section 4.3.5.
- Integrating quantities over a cell (e.g. a finite element face), a typical example being the integration of tractions to obtain (nodal) forces.

A customizable Mesh class, accompanied by classes for cells of different type, constitutes the core of an adapter library for a mesh-based solver. Usually, a solver-specific mesh class is derived from this abstract mesh class. The same is done for the cell classes. Their design allows for a decision on how coupling quantities are to be extracted from the solver database and how they are evaluated at the specific locations in the solver-specific derived classes. Among other things, these decisions determine the following aspects.

- Whether coupling quantities from the solver data base are extracted as an entire field, e.g. for an entire patch part (c.f. Fig. 4.22), or individually for each node, respectively each cell.

- Whether coupling quantities are interpolated and evaluated using the solvers internal methods (typical for FEM solvers) or using methods, provided by the mesh class (typical for FVM solvers).
- Whether geometrical information such as mapping functions and the related Jacobian matrices are computed within the field solver (typical solvers with special element geometries as found in the p-FEM or isogeometric methods) or using the templates for elements provided by the mesh class (standard elements like linear and quadratic lines, triangles, quadrilaterals, tetrahedrals, hexahedrals, etc. are supported by *comana*).
- Whether an interpolation or integration is done within the solver or the mesh class.

Among the seamless integration of the building blocks into the communication framework by the core concepts introduced in Section 4.4.1, this flexibility when designing a derived mesh class for a specific field solver and the well thought out data structures one can draw on during this process are what make the software most attractive.

Listing 1 Initialization section of the implementation of a CMP in an FSI simulation using the C++ framework *comana*.

```cpp
#include "comana/kernel/comana.h"

using namespace Comana;

// abbreviations
using R = Request;
using L = Location;
using Q = QuantityType;

int main()
{
    SolverHandle fSolver("pimpledymfoam"), sSolver("adhoc");

    // fluid patch
    Patch fPatch(fSolver, "Surface A", Topology::plane);
    R::initializePatch(fPatch);
    Field fVertices = R::getField(fPatch, L::vertex, Q::position);

    // structural patch
    Patch sPatch(sSolver, "Patch 3", Topology::plane);
    R::initializePatch(sPatch);
    Field sPoints = R::getField(sPatch, L::quadraturePoint, Q::position);

    // initialize custom points
    R::setField(fPatch, L::custom, Q::position, sPoints);
    R::setField(sPatch, L::custom, Q::position, fVertices);

    // initialize fields
    Field displacement = R::getField(sPatch, L::custom, Q::displacement);
    Field traction = R::getField(fPatch, L::custom, Q::traction);

    // initialize vectors
    Vector &preliminaryDisplacementVector = displacement.getComponents();
    Vector displacementVector = preliminaryDisplacementVector;
    const auto nDisplacementDof = displacementArray.size();
    Vector residual;

    // start of coupling algorithm section
```

Listing 2 Staggered coupling algorithm section of a typical implementation of a CMP in an FSI simulation using the C++ framework *comana*.

```
    // end of initialization section

    // building blocks
    PolynomialPredictor predictor(2);
    ConvergenceCriterion criterion(1e-6);
    QuasiNewtonLeastSquaresMethod accelerator(displacementVector);

    // time loop
    const auto numberOfTimeSteps=1000;
    const auto maxNumberOfIterations=50;
    for(auto k=0; k<numberOfTimeSteps; k++)
    {
        predictor.predict(displacementArray);
        for(auto j=0; j<maxNumberOfIterations; j++)
        {
            preliminaryDisplacementVector = displacementVector;

            // solve fluid subproblem
            R::setField(fPatch, L::vertex, Q::displacement, displacement);
            R::proceed(fSolver);
            R::getField(fPatch, L::custom, Q::traction, traction);

            // solve structural subproblem
            R::setField(sPatch, L::quadraturePoint, Q::traction, traction);
            R::proceed(sSolver);
            R::getField(sPatch, L::custom, Q::displacement, displacement);

            residual = preliminaryDisplacementVector - displacementVector;

            if(criterion.fulfilled(displacementVector, residual)
            {
                displacementArray = preliminaryDisplacementVector;
                R::proceed(fSolver);
                R::proceed(sSolver);
                break;
            }
            else
            {
                R::iterate(fSolver);
                R::iterate(sSolver);
                displacementVector = accelerator.updateSolution(residual);
            }
        }
    }
}
```

5 Numerical investigations

In order to study the applicability of the proposed numerical methods, investigations regarding their accuracy and efficiency were conducted. The results are presented in this chapter. In the first section, single field problems are considered in order to ensure the proper function of the individual field solvers. The main focus lies on the structural mechanics constitutive laws. In the second section, the partitioned solution approach is considered. The introduced coupling algorithms and convergence acceleration schemes are investigated in terms of robustness and efficiency. Further, benchmark problems are solved in order to validate the simulation approach. In the last section, the numerical methods are investigated with respect to their performance in cardiovascular FSI simulations.

5.1 Preliminary analyses

In this section, the individual subproblems introduced in Chapter 3 and the corresponding numerical methods introduced in Chapter 4 are investigated. The purpose of the conducted simulations is twofold. On the one hand, they give rise to investigations of the accuracy and the efficiency of the numerical methods. On the other hand, they reveal the influence of the model parameters like the parameters of the constitutive models for the structural and the fluid mechanics subproblem.

5.1.1 Structural mechanics

In this section, three aspects of the structural mechanics subproblem and its discretization are considered. At first, the superior convergence behavior the p-FEM introduced in Section 4.1.1 over the classical h-FEM is demonstrated. Secondly, the robustness of the soft-tissue material models introduced in Section 3.3.1 is investigated. Finally, suitable parameters for the isotropic models are identified in order to use those models for FSI simulations with non-trivial geometry.

Convergence analysis

As a basis for the convergence study, a cylindrically shaped artery is considered. It is loaded with an internal pressure of $p_\mathrm{i} = 16\,\mathrm{kPa}$. The displacement in axial direction is suppressed by applying suitable Dirichlet boundary conditions at both ends. Due to the symmetry, only a quarter of the cylinder is discretized. The HGO model is employed with the parameters identified in [82]. Different sets of parameters are used for the media and adventitia layer, which are assumed to have the same thickness for this test case. The sets of parameters will be considered in more detail later on in this section. They were also used in [162], where the convergence behavior of the p-FEM was investigated as well. Therein, the trunk and the tensor product space were considered, while this study is augmented by the serendipity space. The following findings from [162] are not reinvestigated.

- A p-refinement should be done isotropically. For thin-walled structures as considered in [64], the use of anisotropic ansatz spaces (different polynomial degrees in thickness and in plane direction) yields a performance gain – but this is not the case when considering the load-displacement behavior of (rather thick-walled) arteries.
- A similar convergence behavior is observed for bypass-like structures, even though they are characterized by a much smaller wall thickness, and were modeled using an isotropic (Neo-Hooke) material model in [162].
- A similar convergence behavior is observed also for arterial segments with non-circular geometry.
- A mesh with two elements in the thickness direction and eight elements in the circumferential direction (two in the quarter model considered here) yields a suitable element size.

A reference solution for the convergence analysis was computed using the same polynomial degree $p = p^\mathrm{g} = 10$ for the geometry and the shape function. For all simulations, the number of quadrature points per element was chosen to be $(p + 5)^3$ in order to avoid any influence of inaccuracy induced by the quadrature. The coarsest low-order mesh under consideration consists of 8 elements in circumferential direction (for the quarter model) and 2 elements in thickness direction. Refinements are done only in those directions, not in the axial direction. The simulation error for three quantities of interest, namely the strain energy E, the maximum displacement $\|\boldsymbol{d}^{\max}\|$ and the von Mises stress σ_{v} are investigated. For all quantities, the relative error is considered, e.g.

$$e_E = \left\| \frac{E^{\mathrm{ref}} - E}{E^{\mathrm{ref}}} \right\| \tag{5.1}$$

for the strain energy.

The results in Figure 5.1 show that the trunk space is favorable over the other ansatz spaces. The comparably low error for quadratic elements ($p = p_{\mathrm{g}} = 2$), which was discovered in [162], is observed here as well. It is attributed to the problem characteristics, which seem to be such that the solution lacks an important contribution from the additional shape functions present for $p = 3$ as opposed to $p = 2$. This may at the same time shift the convergence graph for the h-FEM with $p = p_{\mathrm{g}} = 2$ towards lower errors, such that this discretization method should be investigated for other problems as well, before making statements about its performance, which for the present case is unexpectedly high. In any case, the convergence rates of the p-FEM are even higher – and as long as the trunk space (TS) is used instead of the tensor product space (TPS), the same error can be achieved with less degrees of freedom in nearly all ranges of desired accuracies. The convergence of simulations with the serendipity space (SS) is limited, which is attributed to a lack of modes that are crucial in order to describe the correct displacements. Therefore, it is not recommended to apply this space.

Volumetric strain energy

Having applied the isochoric volumetric split for the introduced constitutive models, their robustness and physical consistency in the case of large strains shall be investigated here. For material models that are based on a strain energy density function (SEDF) with a decoupled volumetric and distortional part, an effect called *thickness thickening* is expected for large uniaxial stretches, see e.g. [92]. While a detailed investigation is out of the scope of this work, the studies conducted here ensure that this undesired effect is not observed within the physiological range of stretches an artery is subjected to. Further, a comparison of the different volumetric parts of the SEDF W_i^{vol} introduced in Section 3.2.2 and an investigation of the influence of the bulk modulus κ are of interest.

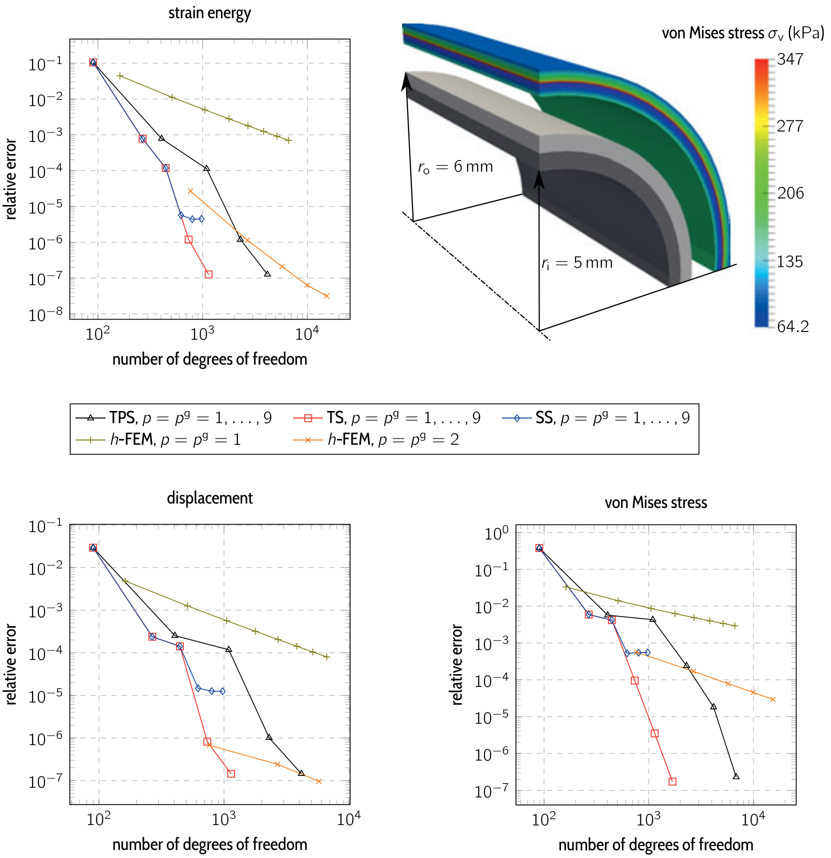


Figure 5.1: Convergence behavior of the p-FEM and the h-FEM for a two-layered model of an idealized artery.

Table 5.1: Parameters for the soft tissue material models.

HGO (media)	shear modulus	μ^{HGO}	27	kPa
	fiber stiffness	k_1	0.64	kPa
	fiber stiffness	k_2	3.54	–
	fiber angle	β	10	°
	bulk modulus	κ	10	kPa
HGO (adventitia)	shear modulus	μ^{HGO}	2.7	kPa
	fiber stiffness	k_1	0.64	kPa
	fiber stiffness	k_2	3.54	–
	fiber angle	β	10	°
	bulk modulus	κ	10	kPa
Delfino	shear modulus	a	44.2	Pa
	stress factor	b	16.7	-
	bulk modulus	κ	10	MPa
Fung	shear modulus	μ	44.2	kPa
	first Lamé coeff.	λ^{Fung}	99.7	kPa
	stress factor	c	50	kPa
	bulk modulus	κ	10	MPa

As a test case, a unit cube with symmetry boundary conditions at three of its boundaries as depicted in Figure 5.2 is considered. The symmetry boundaries can be identified as Γ^{back}, Γ^{left}, and Γ^{bottom}. At each of these boundaries, the displacement is restricted to the in-plane direction. Two different scenarios are considered for the other boundaries.

1. A uniform displacement d_x in the x-direction is prescribed at the boundary Γ^{right}. The boundaries Γ^{front} and Γ^{top} are stress-free.

2. A uniform displacement is prescribed at the boundaries Γ^{right}, Γ^{front}, and Γ^{top}, such that an isochoric deformation is obtained.

For the second scenario, the displacement to be prescribed at the boundaries Γ^{front} and Γ^{top} is obtained by considering the volume in the deformed configuration, which reads

$$V_t = (1 + d_x)\,(1 + d_y)\,(1 + d_z)\,. \tag{5.2}$$

Therein, d_x denotes the displacement at Γ^{right} as in the first scenario, while d_y and d_z denote the displacements, which are prescribed at Γ^{front} and Γ^{top},

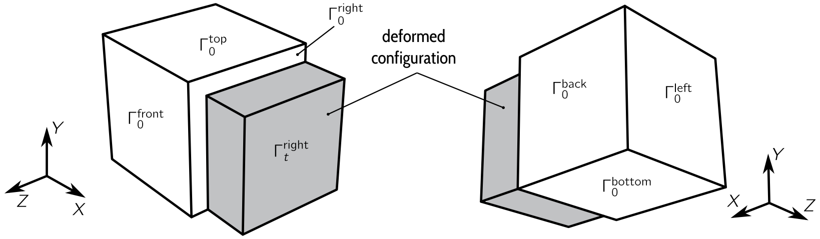

Γ_0^{right}
deformed configuration
Γ_0^{top}
Γ_0^{front}
Γ_t^{right}
Γ_0^{back}
Γ_0^{left}
Γ_0^{bottom}
Y
Z
X
Y
X
Z

Figure 5.2: Reference and deformed configuration of the unit cube considered as a test case.

respectively. From $V_t = 1$, and $d_y = d_z$, it is found that

$$d_y = d_z = \frac{1}{\sqrt{1 + d_x}} - 1. \tag{5.3}$$

Table 5.1 lists the parameters for the considered material models. The parameters for the Delfino material were taken from [57], where they were identified for the incompressible limit case. The shear modulus for the Fung model is taken to be that of the Delfino model, while the first Lamé coefficient-like parameter λ^{Fung} is determined from μ and κ according to linear elasticity. Note that λ^{Fung} does not actually represent a parameter that has an effect on the initial (linearized around zero deformation) stiffness, as shown in Section 3.3.1. The parameter c was chosen such that a similar load displacement behavior is obtained for the Fung and the Delfino model. As demonstrated in Section 3.3.1 as well, the qualitative behavior of the incompressible Delfino model is not preserved in the compressible variant. A usage of the parameters in Table 5.1 is therefore questionable, and they are only chosen as a starting point here. A more consistent set of parameters is determined in Section 5.1.1. The parameters for the HGO are taken from [82]. A bulk modulus of $\kappa = 1 \times 10^7\,\text{Pa}$ given therein is adopted for the other material models as well. The fiber directions are set to

$$\boldsymbol{a}_1 = \frac{1}{\sqrt{2}} \left(\boldsymbol{e}_1 + \boldsymbol{e}_2 \right) \tag{5.4}$$

and

$$\boldsymbol{a}_2 = \frac{1}{\sqrt{2}} \left(\boldsymbol{e}_1 - \boldsymbol{e}_2 \right). \tag{5.5}$$

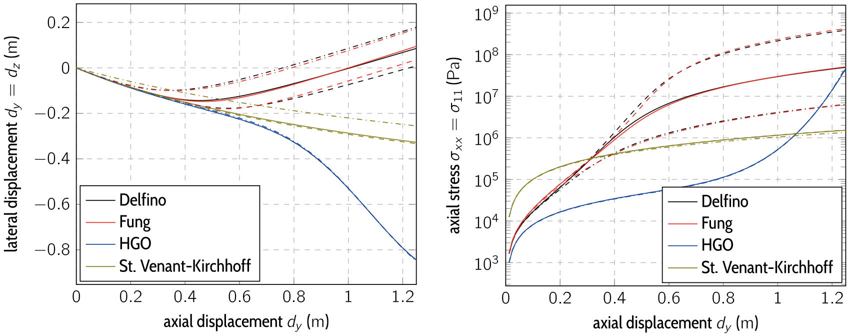


Figure 5.3: Lateral displacement (left) and axial stress (right) plotted over the axial displacement for the first scenario considered for the test case *unit cube*. Solid lines represent $\kappa = 1 \times 10^7\,\mathrm{Pa}$, dashed lines $\kappa = 1 \times 10^6\,\mathrm{Pa}$, and dash-dotted lines $\kappa = 1 \times 10^8\,\mathrm{Pa}$.
.

For the volumetric part of the soft tissue SEDFs, $W_1^{\mathrm{vol}} = \frac{\kappa}{2}(1-J)^2$ as introduced in Eq. (3.78) is considered first. As a reference, the St. Venant-Kirchhoff model is considered as well. As usual when considering arterial walls, a Young's modulus of $E = 1\,\mathrm{Mpa}$ is assumed, which, in combination with a bulk modulus of $\kappa = 10^7$, leads to

$$\mu = \frac{3\,\kappa\,E}{9\,\kappa - E} = 0.337\,\mathrm{MPa}, \tag{5.6}$$

$$\nu = \frac{3\,\kappa - E}{6\,\kappa} = 0.483. \tag{5.7}$$

In Figure 5.3, the lateral displacement $d_y = d_z$ is plotted over the axial displacement d_x for the first scenario and different bulk moduli κ. The effect of thickness thickening is not observed until the stretch reaches a value far beyond the physiological range (for $\kappa = 1 \times 10^7\,\mathrm{Pa}$, $d_y \approx 0.5\,\mathrm{m}$ marks the point where the lateral stretch no longer decreases but starts to increase). For the HGO model and the St. Venant-Kirchhoff model, it is not observed at all. Figure 5.3 further shows that the load displacement behavior for all material models except the HGO model strongly depends on the value for κ.

Regarding the second scenario, the thickness thickening effect can be observed indirectly for the HGO model. As shown in Figure 5.4, the lateral stresses $\sigma_{zz} = \sigma_{yy} = \sigma_{22} = \sigma_{33}$ are negative for moderate axial displacement. However, between $d_x = 0.8\,\mathrm{m}$ and $d_x = 0.9\,\mathrm{m}$, they become positive, which

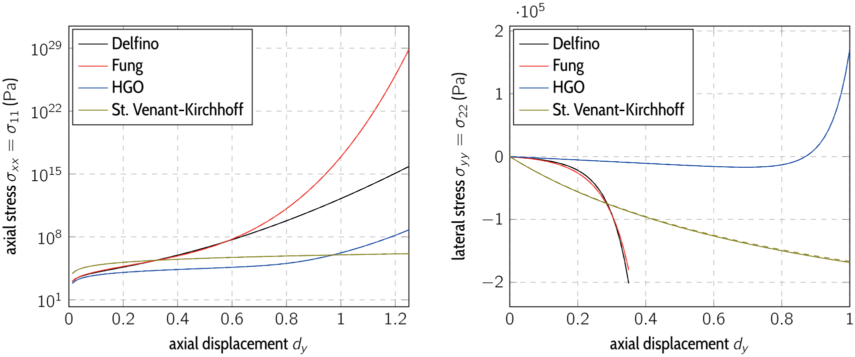


Figure 5.4: Axial stress (left) and lateral stress (right) plotted over the axial displacement for the second scenario considered for the test case *unit cube*.
.

would lead to thickness thickening if the lateral displacement was not constrained to conform to an isochoric deformation. Again, stretches in this range are far beyond the physiological limit, indicating that all material models can safely be applied in cardiovascular FSI simulations.

In [92], the unphysical effects are circumvented for a rubber-like SEDF by choosing the volumetric part $W_2^{\text{vol}} = \frac{\kappa}{50}\left(J^5 + J^{-5} - 2\right)$ introduced in Eq. (3.80). This was also done for the soft tissue material models considered here. While the results in Figure 5.5 indicate that the point at which the thickness thickening is observed first is shifted towards larger stretches, the effect can still be observed within the considered stretch ranges. Due to the fact that this constitutes only a small improvement, which is irrelevant for physiological deformation scenarios, the simpler term W_1^{vol} is used in the following for all soft tissue material models. This yields an HGO model that is very close to the compressible variant from from [82][1] and is equivalent to the one considered in [152] and [236].

While the St. Venant-Kichhoff model shows physically consistent results even for the very large stretches considered here, it is emphasized again that this model has a major drawback for strong compressions. However, this can be shown analytically – as has already been done, see e.g. [230] and Section 3.2.2.

[1] The invariant IV_C and VI_C are computed from the isochoric counterparts of $\boldsymbol{C}$ in [82] as discussed in Section 3.3.1.

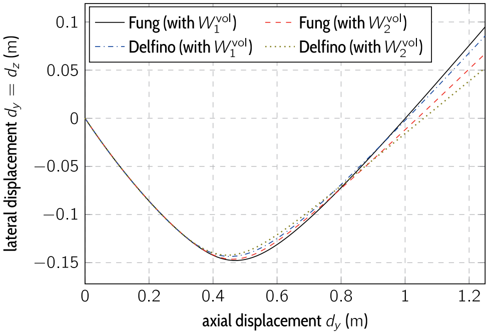


Figure 5.5: Influence of the volumetric part of the SEDF W^{vol} on the thickness thickening effect for the first scenario considered for the test case *unit cube*.

Parameter identification

According to the findings in the previous section, material parameters obtained for the incompressible variants of the material models cannot simply be adopted for the compressible variants introduced in Section 3.3.1. Instead, it is necessary to identify equivalent parameters that yield a similar load-displacement behavior. To this end, the case considered in [82] is taken as a reference. Material parameters for the HGO model are provided there, but they were identified considering prestresses, which – like the anisotropy – cannot easily be incorporated in models with a complex geometry. In the same manner as would be appropriate when taking experimental data as a reference, parameters for the isotropic models are identified, but taking simulation results as a reference.

Prestressed reference case As mentioned in Chapter 2, an artery shrinks in axial direction when it is explanted. Further, if cut along the axial direction, it springs open. Accordingly, prestresses must exist in the closed and axially stretched *load-free configuration*. One attempt to capture these prestresses in a simulation is to start the modeling process from the opened and unstretched state. This state is then assumed to correspond to a *stress-free configuration*, whose geometry has to be determined by separating the arterial layers. Here, this is done in a similar manner as in [82], as illustrated in Figure 5.6 and explained in the following.

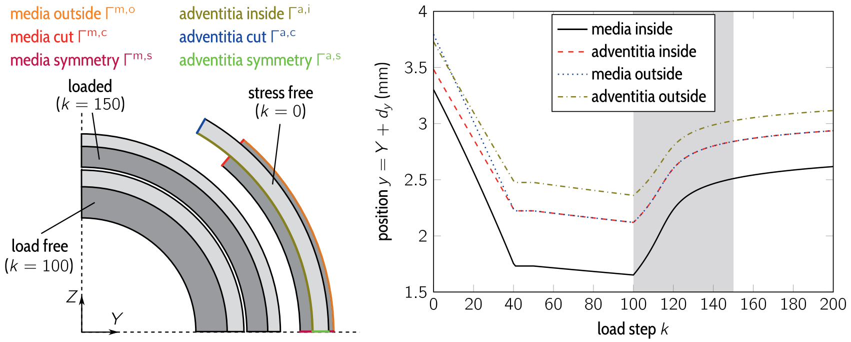


Figure 5.6: Configurations and circumferencial stress for an artery made out of HGO material considering prestresses within the vessel wall.

1. The starting point of the simulation is taken to be the opened state. This stress free configuration is meshed individually for the media and adventitia. As can be seen in Figure 5.6, the radii of both layers do not match at this state. Throughout the simulation, the displacement in z-direction at the boundaries $\Gamma^{m,s}$ and $\Gamma^{a,s}$ is set to zero, which leads to a symmetry boundary condition.

2. By applying Dirichlet boundary conditions that prescribe the displacement in y-direction at the boundaries $\Gamma^{m,c}$ and $\Gamma^{a,c}$, both layers are closed. Further, Dirichlet boundary conditions are applied on the outer boundary of the media $\Gamma^{m,o}$ and on the inner boundary of the adventitia $\Gamma^{a,o}$. Here, the prescribed displacement is determined in such a way that the two boundaries coincide at the end of the prestretching phase, which lasts for the first 40 load steps.

3. From load step 41 on, the Dirichlet boundary conditions applied on $\Gamma^{m,o}$ and $\Gamma^{a,i}$ are changed. Instead of prescribing a fixed displacement, the two boundaries are now glued together to ensure that any additional displacement is forced to be equal. This results in a small displacement from load step 45 to 46, which brings the system into an equilibrium state.

4. From load step 50 to load step 100, an axial stretch of $\lambda^a = 1.1$ is applied by prescribing the displacement in the x-direction at the boundaries representing the ends of the vessel segment.

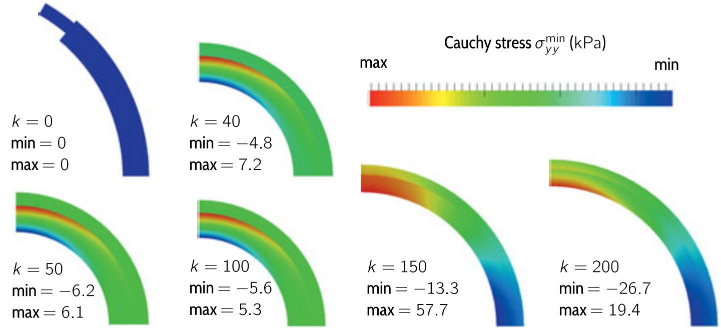


Figure 5.7: Stress states during the simulation of the prestressed reference case.

5. From load step 100 to load step 200, an internal pressure load is applied and increased to a value of $p_i = 26.66\,\mathrm{kPa}$. While the loads exceed the pressure range considered in [82] ($p_i = 0\,\mathrm{kPa}to13.33\,\mathrm{kPa}$), no unphysical effects are observed.

The stress distributions in the closed but axially unstretched state (load step 50) as well as that in the axially stretched state loaded with $p_i = 13.33\,\mathrm{kPa}$ (load step 150) shown in Figure 5.7 are in good agreement with the ones given in [82]. The study is well-suited to show the adventitia's role as a shield that protects the other layers from damage at high loads. While much higher stresses prevail in the media under low to medium loads, the stresses in adventitia reach comparable levels in the high load regime.

Optimization procedure For cases with complex geometries, it is desired to start the modeling process from the unloaded but prestretched state. In the reference case, this state was reached in load step 100, cf. Figure 5.6. In practice, it may correspond to an artery being clamped (no blood pressure) but not explanted (stretched in axial and circumferential direction). This is a situation that occurs, for example, during bypass surgeries. Besides working with closed geometries, it is desired to apply isotropic material models, due to a lack of information on the preferred fibre direction in complex geometries.

In the following, the possibility to reproduce the load-displacement behavior of the prestressed anisotropic case using a simplified isotropic model is investigated. It should be emphasized that this is an attempt to reproduce only the load-displacement behavior. Realistic stress states corresponding to the ones shown in Figure 5.6 are obviously not reproducible using such a simplified model. In view of the FSI simulations to be performed, this decision seems reasonable. While, from a structural mechanics point of view, an adequate prediction of the stresses inside the material is a major concern, only the load-displacement behavior is of interest in an FSI simulation in the scope of hemodynamics. That is, if the structural model predicts the displacement of the inner vessel wall properly, the hemodynamics are predicted well – assuming that the fluid mechanics are captured well including all boundary conditions.

For the optimization procedure, the load-displacement curve from the prestressed case during the loading phase (shaded in gray in Figure 5.6) is extracted and taken as a reference. The values are collected in a vector $\mathbf{d}^{\text{ref}}$. Trial solutions, obtained from simulations with the simplified model are denoted by $\mathbf{d}_i^{\text{trial}}$, where i denotes the iteration number within the optimization process. The MATALB function LSQNONLIN (see [197]) is then used with its standard parameters to minimize the error

$$e^{\text{opt}} = \frac{\left\|\mathbf{d}^{\text{trial}} - \mathbf{d}^{\text{ref}}\right\|}{\left\|\mathbf{d}^{\text{ref}}\right\|}. \tag{5.8}$$

The function LSQNONLIN implements the Levenberg-Marquardt algorithm (LMA) proposed in [134] and rediscovered in [140]. It was previously used for the identification of material parameters from experimental data, see e.g. [91] and the references cited therein.

The optimization process yields errors e_i^{opt}, each of which corresponds to a set of material parameters, namely

$$\mathbf{p}_i \overset{\text{Fung}}{=} \left[\begin{array}{ccc} \mu_i & \lambda_i^{\text{Fung}} & c_i \end{array}\right] \tag{5.9}$$

for the Fung model and

$$\mathbf{p}_i \overset{\text{Delfino}}{=} \left[\begin{array}{cc} a_i & b_i \end{array}\right] \tag{5.10}$$

for the Delfino model (see Section 3.3.1 for details). While a fully three-dimensional finite element simulation with the simplified model is performed in each iteration, the only input to the LMA are the corresponding trial solutions $\mathbf{d}_i^{\text{trial}}$ and an initial set of parameters $\mathbf{p}_0$. Its output are improved sets of parameters, which eventually lead to a decreased e_i^{opt}.

The geometry for the simplified model is taken to be that of the prestretched one, corresponding to the beginning of the reference loading phase

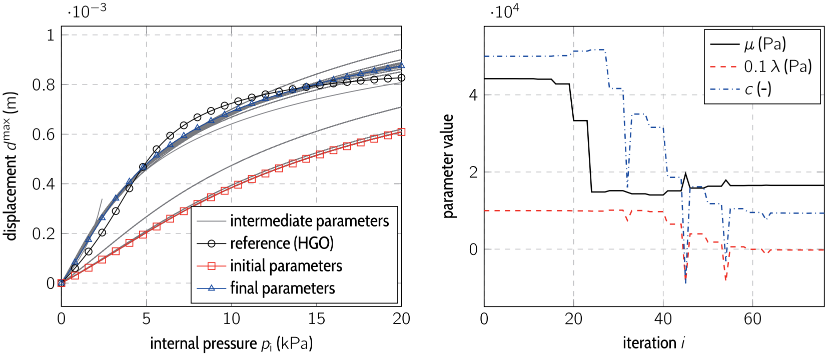


Figure 5.8: Load-displacement behavior obtained with the original Fung model for the artery case and progression of the parameter values during the optimization process.

(load step 100 in Figure 5.6). At this state, the inner and outer radii of the vessel wall were found to be $r_\mathrm{i} = 1.65\,\mathrm{mm}$ and $r_\mathrm{o} = 2.36\,\mathrm{mm}$, respectively. A distinction between media and adventitia is not made in the simplified model. Rather, the artery is modeled as one homogeneous layer.

Figure 5.8 shows the results of the optimization procedure for the Fung model. The initial values for the material parameters were taken to be those from Table 5.1. As can be seen, the LMA manages to reduce the error to some extent (from $e_0 = 0.413$ to $e_{87} = 0.0488$), however, the result is rather unsatisfactory. The load displacement curve does not match the reference very well. A similar mismatch is found for the Delfino model.

In order to find a more suitable model, the simple uniaxial tension test from the beginning of this section is considered again. This time, a third scenario is examined, where the deformation is driven by a uniform traction

$$\bar{\boldsymbol{t}} = \begin{bmatrix} t_x & 0 & 0 \end{bmatrix}^\mathrm{T}, \tag{5.11}$$

which is applied at the boundary Γ^right. As shown in Figure 5.9 (left), the mismatch prevails also for this case, which does not include any prestresses or biaxial stress states. This motivates a modification of the SEDF, since it is the only difference between the considered models. Looking at respective SEDFs, there is an obvious conceptional difference: While there is a Neo-Hookean term in the HGO model to account for the stiffness of the matrix material, such a term is missing in the Fung and the Delfino model. If this

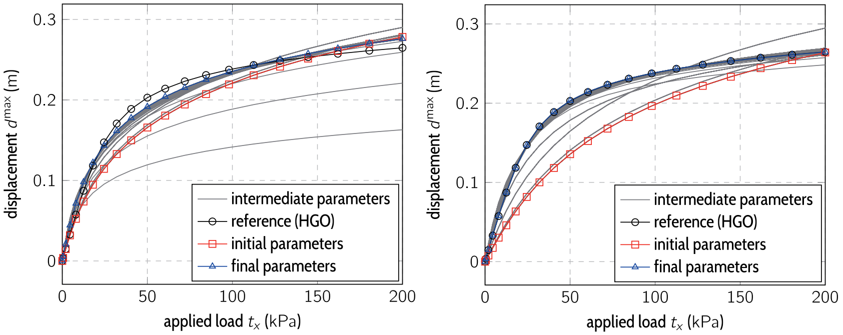


Figure 5.9: Load-displacement behavior obtained with the original Fung model (left) and the modified Fung model (right) for the uniaxial tension case.

term is included in these models as well, modified SEDFs are obtained:

$$W^{\text{Fung}^*} = W^{\text{Fung}} + W^{\text{NH}} \tag{5.12}$$

and

$$W^{\text{Delfino}^*} = W^{\text{Delfino}} + W^{\text{NH}}, \tag{5.13}$$

where W^{Fung} and W^{Delfino} denote the original SEDFs introduced in Eqs. (3.141) and (3.149), respectively. For the second Piola-Kirchhoff stresses and elasticity tensors, a corresponding addition is obtained as explained in Section 3.2.2. As can be seen from Figure 5.9 (right), the load-displacement curve for the uniaxial tension test can be represented very well by the modified Fung model. A similarly good match was found for the modified Delfino model.

Coming back to the simplified model of an artery, it is found that a very good approximation of the load displacement curve can be achieved here as well, see Figure 5.10 and 5.11. A closer look at the progression of the error given in Figure 5.12 motivates the use of the Fung model. Compared to the Delfino model, the error is slightly lower here, even though both models lead to good fits. In order to check the robustness of the identified parameters, several more optimizations were performed using different initial values $\mathbf{p}_0$. While no minimum was found in some of the cases, it showed that the same set of minimizing parameters occurred in all the successful runs, which indicates the uniqueness of the solution.

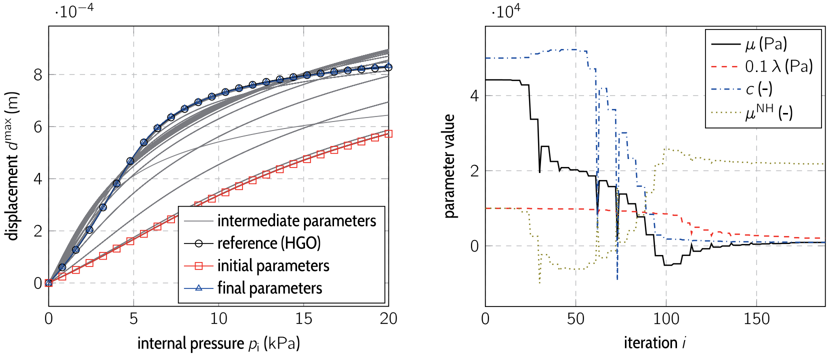


Figure 5.10: Load-displacement behavior obtained with the modified Fung model for the artery case and progression of the parameter values during the optimization process.

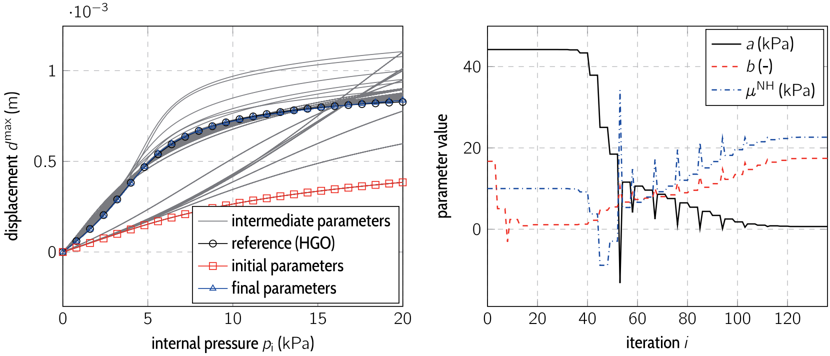


Figure 5.11: Load-displacement behavior obtained with the modified Delfino model for the artery case and progression of the parameter values during the optimization process.

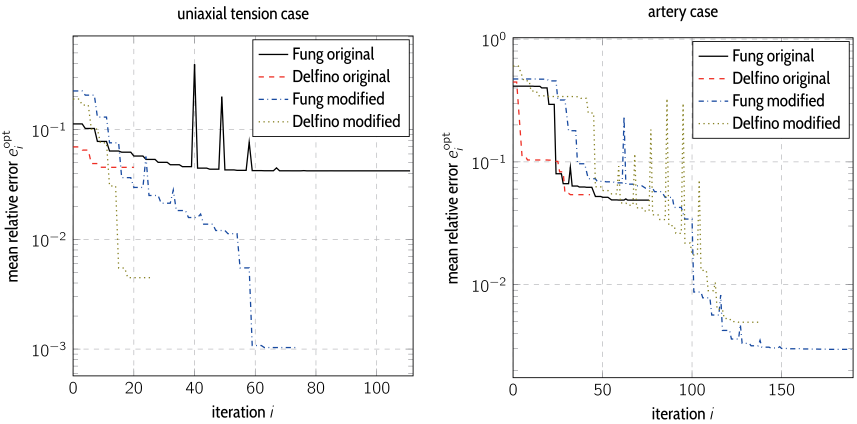


Figure 5.12: Reduction of the error during the optimization process.

5.1.2 Fluid dynamics

In this section, preliminary investigations of the fluid mechanics subproblem are presented. They serve to assess the influence of the constitutive equation, respectively the viscosity model. Therefore, the flow through a straight rigid vessel is considered. Three different diameters are examined in view of the assumption that the typically low shear rates prevailing in small arteries and arterioles lead to a strong influence of the shear thinning effect and the initial shear stress (see Section 3.3.2), while this effect is negligible for larger arteries.

The considered vessel diameters and flow rates are taken from experimental studies. The largest vessel resembles a common femoral artery of an average male. A mean diameter of $D_1 = 10\,\mathrm{mm}$ and a mean flow rate of $Q_1^{\mathrm{mean}} = 350\,\mathrm{ml/min}$ are set according to the results presented in [135]. Therein, the minimum and maximum flow rates observed during the cardiac cycle were found to be $Q_1^{\mathrm{min}} = 127\,\mathrm{ml/min}$ and $Q_1^{\mathrm{max}} = 819\,\mathrm{ml/min}$, respectively. The time-dependent flow rate $Q_1(t)$ shown in Figure 5.13 (left) was determined such that these bounds are matched approximately and a typical flow curve in the lower extremities is obtained. As a second scenario, the flow rate in an artery in the human finger as investigated in [120] is considered. According to the different specimens examined therein, the diameter is set to $D_2 = 800\,\mu\mathrm{m}$. The time-dependent mean axial veloc-

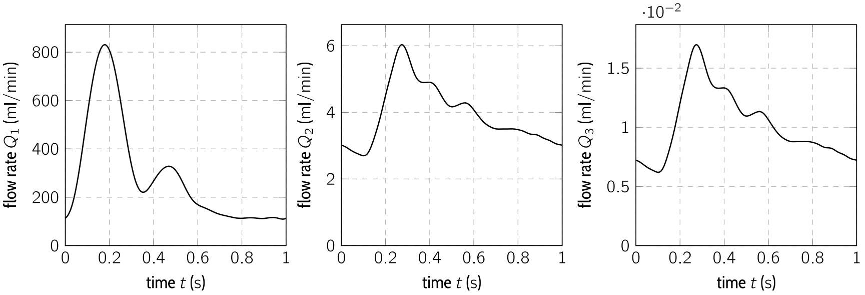


Figure 5.13: Flow rates for the femoral artery-like vessel (left), the finger artery-like vessel (middle), and the retina arteriole-like vessel (right).

ity measured in [120] was reconstructed using the *WebPlotDigitizer* (see [169]) and corresponds to the flow rate shown in Figure5.13 (middle) with a mean value of $Q^{\text{mean}} = 3.93\,\text{ml/min}$. The smallest vessel resembles an arteriole in the retina and, according to the studies in [84], has a diameter of $D_1 = 110\,\mu\text{m}$. A mean flow rate of $Q_3^{\text{mean}} = 0.0102\,\text{ml/min}$ was measured therein. Further, a ratio between minimum and maximum flow rate of 3.1 is given. The flow rate for the finger artery is taken as a basis and then scaled and shifted accordingly to arrive at the flow rate $Q_3(t)$, as shown in Figure 5.13 (right).

To start with, a stationary flow using the respective mean flow rate is considered for all vessels. Figure 5.14 shows a comparison between the different viscosity models. While minor differences can be observed, it is questionable, whether the usage of a more complex model is justified. The flattened velocity profiles prevailing for all viscosity models due to the increase viscosity for low shear rates is still very close to the one obtained for the Newtonian model. Different (constant and shear-rate-independent) viscosities would of course yield the same parabolic velocity profile in the stationary case, but the pressure drop would change.

In the pulsatile case, the shape of the velocity profiles does depend on the viscosity, as shown in Figure 5.15. Besides the effect of different viscosities in combination with the Newtonian model, the influence of the viscosity model itself is more pronounced here. It is also observed that the velocity profiles in the smaller vessels correspond very well to the parabolic Poisseuille velocity profile. Here, the differences due to different viscosities in combination with

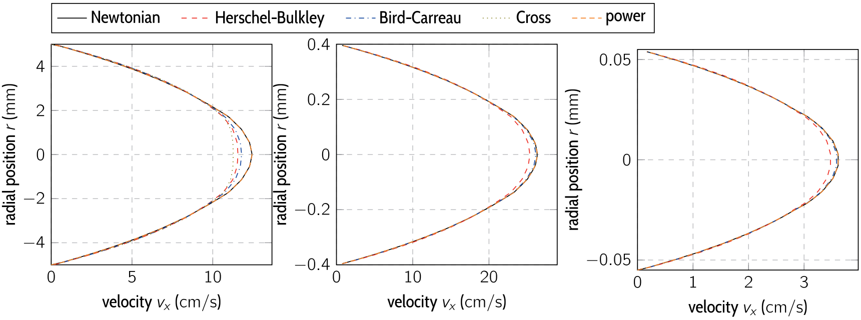


Figure 5.14: Velocity profiles for the pulsatile flow situation in the femoral artery-like vessel (left), the finger artery-like vessel (middle), and the retina arteriole-like vessel (right).

the Newtonian model are negligible, as in the stationary case. In the large vessel, the typical Womersley shape is observed. This may also be estimated a priori in terms of the Womersley number (see Eq. (3.181)), which describes the relation between viscous and inertia forces or, respectively, their effect on the velocity profile. Being proportional to the radius, it is much higher for the flow in the large vessel than for those in the smaller vessels. The differences due to the viscosity model are clearly visible, especially in the late diastolic phase (instance 0 s, 0.8 s in Figure 5.15).

Concluding, one can say that the viscosity has a major impact on the velocity profiles in pulsatile flows. The effect of a non-Newtonian behavior can be observed as well. However, the differences induced due to different mean viscosities are comparably large. It is therefore emphasized that simulations of hemodynamics, respectively the interpretations of their results, have to be done in view of the prescribed blood viscosity. In patient-specific studies, e.g. with vessel geometries obtained from CT scans, patient-specific parameters for the viscosity models should be used as well. Further, physiological velocity profiles should be prescribed at inlet boundaries, such that the distance from the inlet, from which on a fully developed flow prevails, is as short as possible. This means that Womersley profiles rather than Poisseuille profiles should be prescribed, especially for larger arteries, where the flow characteristics are such that the two choices show severe differences.

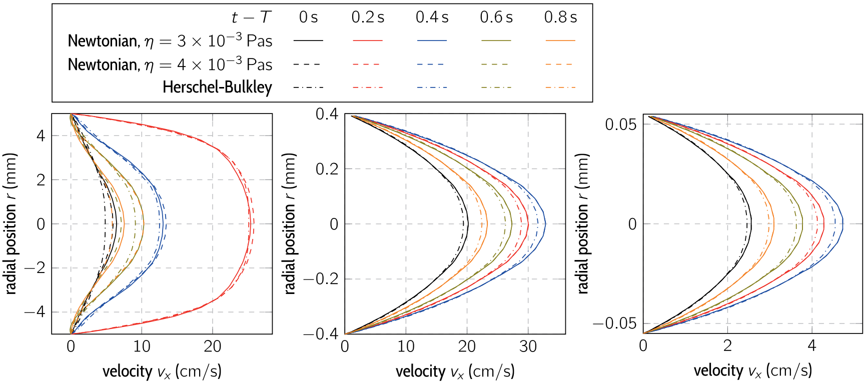


Figure 5.15: Velocity profiles for the pulsatile flow situation in the femoral artery-like vessel (left), the finger artery-like vessel (middle), and the retina arteriole-like vessel (right).

5.1.3 Reduced models

In this section, the one-dimensional network model and the windkessel models are considered. For the former, the implementation is compared to a reference solution for validation purposes, while the latter are investigated in more detail in order to be used efficiently in the scope of FSI simulations.

One-dimensional vessel model

The one-dimensional vessel model introduced in Section 3.3.3 is solved with the Taylor-Galerkin method introduced in Section 4.1.3. It was implemented in MATLAB (see [141]) and is validated considering an example from [179]. The problem setup consists of a single vessel, which is discretized here using 100 one-dimensional finite elements. It has a length of $L = 15\,\text{cm}$ and a diameter of $D = 1\,\text{cm}$. The stiffness parameter and the density are set to $\beta = 451\,352\,\text{kgs}^{-1}\text{cm}^{-2}$ and $\rho = 1\,\text{kg/cm}^3$, respectively. At the inlet, a pressure pulse

$$p_\text{p} = \begin{cases} \sin\left(\frac{\pi t}{T_\text{p}}\right) 2000\,\text{Pa} & \text{if } t < T \\ 0\,\text{Pa} & \text{else} \end{cases} \tag{5.14}$$

with a duration of $T_\text{p} = 0.167\,\text{s}$ is prescribed. As shown in Figure 5.16, the solution to the wave propagation problem obtained with the Taylor-Garlerkin method is very similar to the reference solution from [179]. The

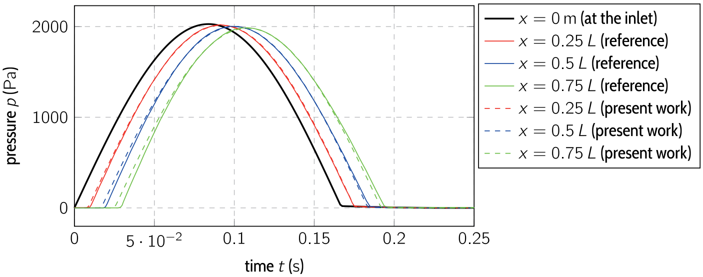


Figure 5.16: Pulse wave propagation in a single vessel: Results obtained with the Taylor-Galerkin method from [179] introduced in Section 4.1.3.

reference solution was obtained using the *WebPlotDigitizer* (see [169]), which is a tool to extract graphs from images that has been used throughout this work. Minor differences are attributed to inaccuracies induced by this process.

Windkessel models

The windkessel models introduced in Section 3.3.4 are used to determine a physiological pressure at the distal end of a fully resolved model in terms of the flow prevailing at this point. Like the test simulations for the fluid mechanics subproblem from Section 5.1.3 all cardiovascular FSI simulations considered in this work are based on a given flow through the inlet. Due to the elasticity of the vessel wall, this flow will of course not match the flow through the outlet – however, for vessel segments of reasonable length, they will show a qualitatively similar behavior. This motivates preliminary investigations of the windkessel models, which are directly based on a given flow.

The FSI simulations under consideration start from homogeneous initial conditions, meaning that, initially, there is no flow at all. However, all evaluations of hemodynamic quantities are not performed before the system has reached a periodic state. The time until a periodic state is reached, strongly depends on the way the windkessel model is coupled to the fluid mechanics subproblem. Of course, it is desirable to reach this state as quickly as possible in order to save computation time. To this end, an alternative to the technique introduced in [163] is explained here.

Given the maximum (systolic) pressure $p_{\max}$ and the minimum (diastolic) pressure $p_{\min}$ as well as a periodic flow pulse $Q(t)$, the system must be slowly driven to a corresponding state from its homogeneous initial state. Instead of $Q(t)$, a modified or *ramped* flow $Q^{\mathrm{r}} = f^{\mathrm{r}}_{T_{\mathrm{r}}}(t)\, Q(t)$ is applied, where

$$f^{\mathrm{r}}_{\mathrm{T_r}} = \begin{cases} \frac{1}{2}\left(\cos\left(\frac{\pi t}{T_{\mathrm{r}}}\right) - 1\right) & \text{if } t < T_{\mathrm{r}}, \\ 1 & \text{else.} \end{cases} \tag{5.15}$$

Due to its frequent usage in coupled simulations to slowly blend data from one state to another, a *cosine ramp* according to Eq. (5.15) is actually present as a building block in *comana*.

Influence of the initial conditions Figure 5.17 shows the pressure response for a two-component windkessel model. The parameters $C = 0.3\,\mathrm{ml/mmHg}$, $R_1 = 2.55\,\mathrm{mmHg\,s/ml}$ and $R_2 = 0.1\,\mathrm{mmHg\,s/ml}$ were determined accordingly such that a pressure range of approximately $80\,\mathrm{mmHg} < p < 120\,\mathrm{mmHg}$ is obtained in the periodic state. The flow pulse was ramped using $T_{\mathrm{r}} = 0.5$, which is a reasonable value in view of the FSI simulations, see e.g. [163]. For an unramped flow, a periodic state would be reached most quickly for an initial value of $p_0 = p_{\min} \approx 80\,\mathrm{mmHg}$, but this is not the case if a ramped pressure is fed into the windkessel model. Looking at the periodicity error introduced in Figure 5.17, it is found that $p_0 = 160\,\mathrm{mmHg}$ is more suitable in this case. The error already drops below $1\,\mathrm{mmHg}$ in the first period. Of course, only the pressure response resulting from the choice $p_0 = 0\,\mathrm{mmHg}$ can readily be applied in an FSI simulations, which demand for a smooth change from the unloaded to the loaded state. However, the pressure which is actually applied in the fluid subproblem can be ramped like the flow in order to make all choices applicable. If not stated otherwise, this is done here for all cardiovascular FSI simulations, while the initial pressure is set to twice the lowest value expected for the periodic state.

Influence of the parameters So far, only the three-component windkessel model was considered. In order to investigate the influence of the parameters, the four-component windkessel model is considered. Figure 5.18 shows the pressure response for varying values of C in combination with $L = 0\,\mathrm{mmHg\,s/ml}$ (which corresponds to the three-component model) and for varying values of L in combination with $C = 0.3\,\mathrm{ml/mmHg}$. The values for R_1 and R_2 are set to the same values as in the previous section. The

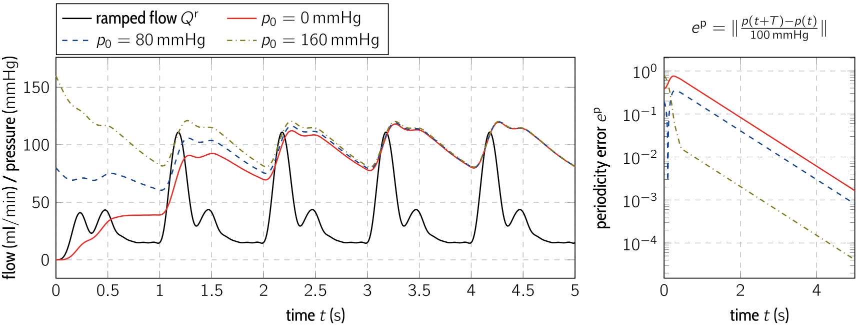


Figure 5.17: Influence of the initial pressure on the periodicity of the response of a three-component windkessel model.

results are obtained using the same flow form as above. However, only one period is considered, after which the flow remains constant. Accordingly, the applied flow is given by

$$Q^* = \begin{cases} Q(t) & \text{if } t < 1\,\mathrm{s} \\ Q(0) & \text{else.} \end{cases} \tag{5.16}$$

The parameter variations show the behavior that was to be expected according to the physical meaning of the individual components. Increasing the capacity C, which corresponds to an increased compliance of the vessel network, lower maximum pressures are obtained. Furthermore, the pressure decays more slowly. Regarding the inductivity L, which represents the inertia effects of the blood, a physically meaningful behavior is observed as well. With increasing values, the pressure is affected to a greater extent by the derivative of the flow pulse. In other words, the effects of accelerating and decelerating the blood become visible.

5.1.4 Interpolation

In order to investigate and compare the different interpolation methods introduced in Section 4.3.5, two test geometries are considered, namely a unit cube (edge length $l = 1$) and a unit sphere (radius $R = 1$). Discretizations with 54,150,384 and 864 quadrilateral elements are created for each geometry and used as a basis for the interpolation. The discretizations with 150 elements are shown as an example in Figure 5.19. The nodes of

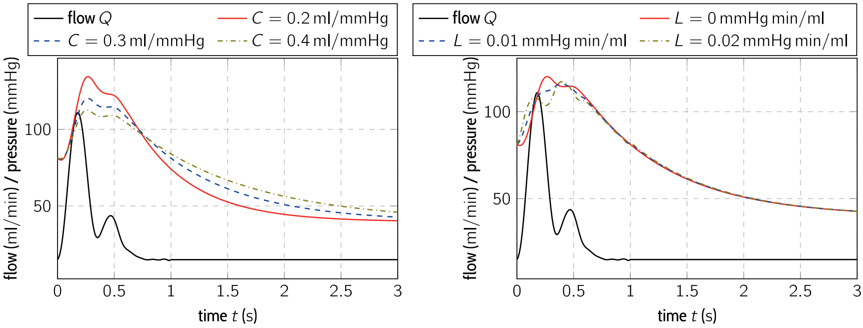


Figure 5.18: Influence of the capacity C (left) and the inductivity L (right) on the pressure response of the four-component windkessel model.

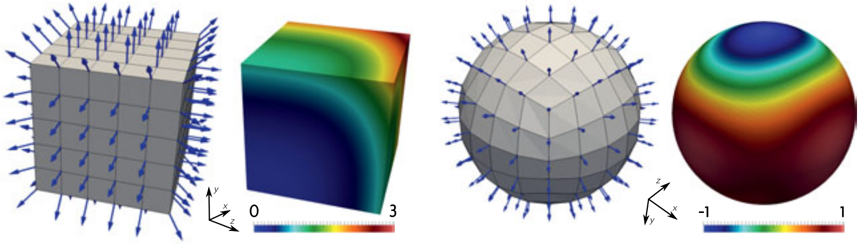


Figure 5.19: Test second coarsest quadrilateral meshes used as a basis for the interpolations and sampled functions (polynomial $f(x, y, z) = x^2 + y^3 + z^4$ and normal vector).

these source meshes serve as sampling points for the meshless interpolation methods (barycentric, nearest neighbor, radial basis functions). Further, an interpolation with linear shape functions is created, drawing on the nodal values as degrees of freedom. For the inverse distance weighting, the element centers are used as sampling points. Along with the meshes, the sampled functions are illustrated in Figure 5.19. For both geometries, the polynomial

$$f(\boldsymbol{x}) \overset{\text{polynomial}}{=} x^2 + y^3 + z^4 \tag{5.17}$$

as well as the respective normal vector are considered. While being uniquely defined for the sphere, e.g. by

$$\boldsymbol{f}(\boldsymbol{x}) \overset{\text{sphere\_normal}}{=} \boldsymbol{x}, \tag{5.18}$$

the normal vector for the cube is not well defined along its edges and at its corners. As can be seen in Figure 5.19, the mean normal of the adjacent faces is assumed here. For the inverse distance weighting, this issue is not relevant because the element centers are used as sampling points. As in Section 4.3.5, the local integration error

$$\boldsymbol{e}^{\mathrm{i}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{i}(\boldsymbol{x}) \tag{5.19}$$

is analyzed, where $\boldsymbol{i}(\boldsymbol{x})$ denotes the interpolation. As a global error measure and for visualization purposes, the interpolation is evaluated at the nodes of a mesh with 15000 quadrilaterals. They constitute the $n^{\mathrm{t}} = 14400$ target points $\boldsymbol{p}_i^{\mathrm{t}}$. Recalling that the values at the sampling or source points are collected in a matrix $\mathbf{V}^{\mathrm{s}}$ (see Eq. (4.215)) and $\mathbf{I}$ denotes a general interpolation matrix, a matrix collecting the errors at all target points can be stated as

$$\mathbf{e}^{\mathrm{i}} = \left[\ \boldsymbol{e}^{\mathrm{i}}(\boldsymbol{p}_1^{\mathrm{t}}) \ \ \ldots \ \ \boldsymbol{e}^{\mathrm{i}}(\boldsymbol{p}_{n^{\mathrm{t}}}^{\mathrm{t}})\ \right]^{\mathrm{T}} = \left[\ \boldsymbol{f}(\boldsymbol{p}_1^{\mathrm{t}}) \ \ \ldots \ \ \boldsymbol{f}(\boldsymbol{p}_{n^{\mathrm{t}}}^{\mathrm{t}})\ \right]^{\mathrm{T}} - \mathbf{I}\,\mathbf{V}^{\mathrm{s}}. \tag{5.20}$$

The global error measure can then be computed as

$$e^{\mathrm{i}} = \frac{\|\mathbf{e}^{\mathrm{i}}\|}{\sqrt{n^{\mathrm{t}}}}, \tag{5.21}$$

which corresponds to the definition in (4.231), given that $\|\mathbf{e}^{\mathrm{i}}\|$ denotes the Frobenius norm of $\mathbf{e}^{\mathrm{i}}$. Further, the maximum error $e^{\mathrm{i}}_{\mathrm{max}}$, i.e. the component with the maximum absolute value in $\mathbf{e}^{\mathrm{i}}$ is evaluated.

Results

For a qualitative overview of the interpolations, Figure 5.20 and Figure 5.21 exemplary show the interpolation error for the source meshes with 150 quadrilaterals. From the overview, it can be observed that only three of the considered scenarios yield satisfactory results for this comparably coarse source point mesh. The radial basis functions seem to be a reliable choice for the sphere geometry regardless of the sampled function. Further, the inverse distance weighting shows a very small error for the cube geometry when interpolating the normal vector. Only along the edges, where the mean normal is chosen as a reference as mentioned above, the error rises above the minimum tolerance, which is achievable despite the finite precision arithmetic underlying all implementations. The convergence characteristics shown in Figure 5.22 confirm the above statements.

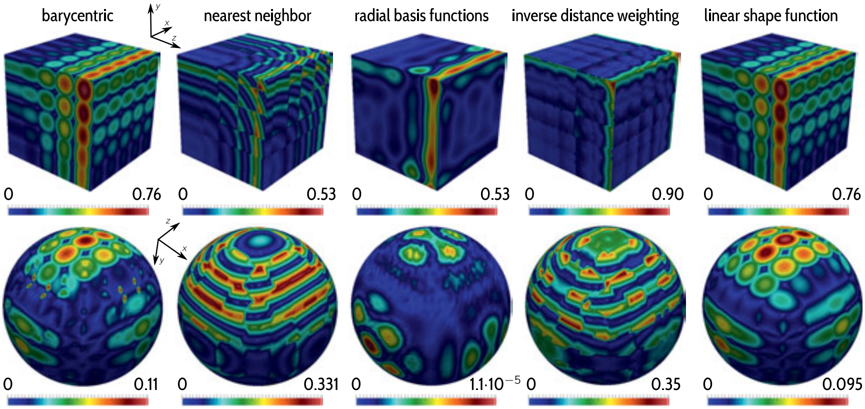


Figure 5.20: Error distribution $\|e^{\mathrm{i}}\|$ for the interpolation of the polynomial illustrated in Figure 5.19 and defined in Eq. (5.17) using the nodes (respectively the face centers) of meshes with 150 quadrilaterals as sampling points.

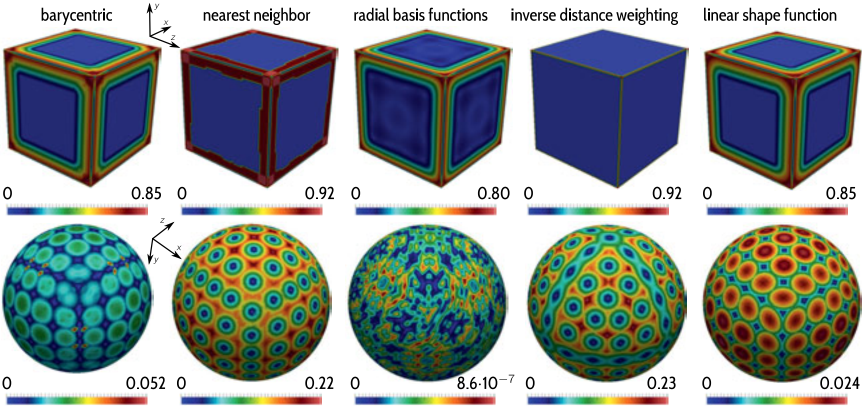


Figure 5.21: Error distribution $\|e^{\mathrm{i}}\|$ for the interpolation of the normal vector as illustrated in Figure 5.19 using the nodes (respectively the face centers) of meshes with 150 quadrilaterals as sampling points.

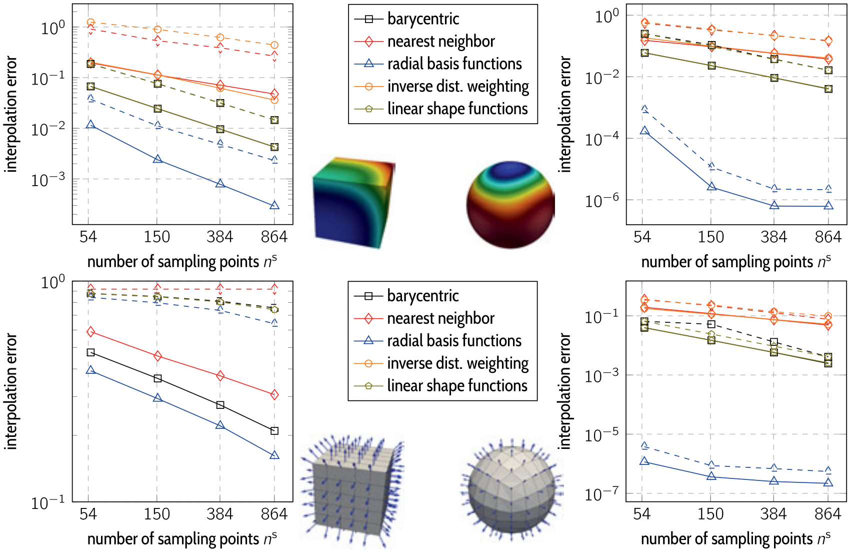


Figure 5.22: Convergence of the global interpolation error e^i (solid lines) and the maximum error $e^\mathrm{i}_\mathrm{max}$ (dashed lines) with increasing number of sampling points.

5.1.5 Load integration

In the scope of FSI simulations, spatially and temporarily varying surface tractions are prescribed at the boundary of the structural mechanics subproblem. The according integral in the weak form (see Eq. 4.2) is approximated using Gaussian quadrature. While this yields very good approximations in the case of moderate and smooth spatial variations, discontinuous or severely varying tractions demand for a modification of the quadrature rule in order to capture the applied load accurately. The FSI problems considered in the present work are solved using high-order finite elements on the structural side and classical finite volumes on the fluid side. In view of this, the structural meshes are likely to be much coarse than the fluid meshes, which yields a scenario where the prescribed fluid loads may show severe variations across one structural element face. According to the above arguments, these tractions are therefore not captured accurately if standard Gaussian quadrature is applied. In this section, a simple remedy for this issue is investigated. It is based on a partitioning of the structural element face into subcells, on which a standard Gaussian quadrature is applied.

One-dimensional integrals

Figure 5.23 (left) shows two functions.

$$f_1(x) = \cos(3\,x) \tag{5.22}$$

can be expected to be accurately integrable using Gaussian quadrature due to its smoothness while

$$f_2(x) = \begin{cases} f_1(x) & \text{if } -0.5 < x < 0.5 \\ f_1(0.5) & \text{else} \end{cases} \tag{5.23}$$

is non-smooth (only C^0-continuous). Here, the aforementioned issues are expected.

The functions f_i are integrated analytically to provide reference solutions

$$F_i^{\text{ref}} = \int_{-1}^{1} f_i \,\mathrm{d}x. \tag{5.24}$$

For the standard Gaussian quadrature, approximations are computed as

$$F_i^{(n^{\text{qp}})} = \sum_{j=1}^{n^{\text{qp}}} f_i(\hat{x}_j^{(n^{\text{qp}})})\, w_j^{(n^{\text{qp}})}, \tag{5.25}$$

where $x_j^{(n^{\mathrm{qp}})}$ denotes the jth point for a Gaussian quadrature rule with n^{qp} points in total. The corresponding weight is denoted by $w_j^{(n^{\mathrm{qp}})}$. If the considered interval $\Omega = [-1, 1]$ is split into n^{sc} subcells

$$\Omega_k = \left[-1 + \frac{2\,(k-1)}{n^{\mathrm{sc}}}, -1 + \frac{2\,k}{n^{\mathrm{sc}}}\right] \tag{5.26}$$

and Gaussian quadrature is applied on each subcell, possibly improved approximations are computed as

$$\hat{F}_i^{(n^{\mathrm{qp}})} = \sum_{k=1}^{n^{\mathrm{sc}}} \sum_{j=1}^{n^{\mathrm{qp}}} f_i(\hat{x}_{jk}^{(n^{\mathrm{qp}})})\, \hat{w}_j^{(n^{\mathrm{qp}})}, \tag{5.27}$$

where

$$\hat{x}_{jk}^{(n^{\mathrm{qp}})} = -1 + \left(\frac{2\,k - 1 + x_j^{(n^{\mathrm{qp}})}}{n^{\mathrm{sc}}}\right) \quad \text{and} \quad \hat{w}_i^{(n^{\mathrm{qp}})} = \frac{w_i^{(n^{\mathrm{qp}})}}{n^{\mathrm{sc}}} \tag{5.28}$$

hold for the modified quadrature points and weights within subcell k.

An investigation of the relative error

$$e_i^{\mathrm{q}} = \frac{F_i^{(n^{\mathrm{qp}})} - F_i^{\mathrm{ref}}}{F_i^{\mathrm{ref}}} \tag{5.29}$$

in relation to n^{qp} as shown in Figure 5.23 confirms the above expectations. Regarding $f_1(x)$, the standard Gaussian quadrature is the most performant method. The non-smooth function f_2 can only be integrated accurately, if the subcell boundaries coincide with the discontinuities. The lower bound of the error observable therein is attributed to the finite precision arithmetic underlying the computations.

Surface tractions

The technique described above for the one-dimensional case can just as well be applied for the two-dimensional integrals arising in the finite element load vector. Recalling the contribution to the weak form due to surface traction from Eq. (4.2), it is usually approximated on element level as

$$\int_{\Gamma_0^{(e)}} \mathbf{f}\, \mathrm{d}\Gamma_0^{(e)} \approx \sum_{i=1}^{n^{\mathrm{qp}}} \sum_{j=1}^{n^{\mathrm{qp}}} \mathbf{f}\left(\boldsymbol{X}\left(x_i^{(n1\mathrm{qp})}, x_j^{(n1\mathrm{qp})}\right)\right) \det\left(\hat{\boldsymbol{J}}\right)\, w_i^{(n^{\mathrm{qp}})}\, w_j^{(n^{\mathrm{qp}})} \tag{5.30}$$

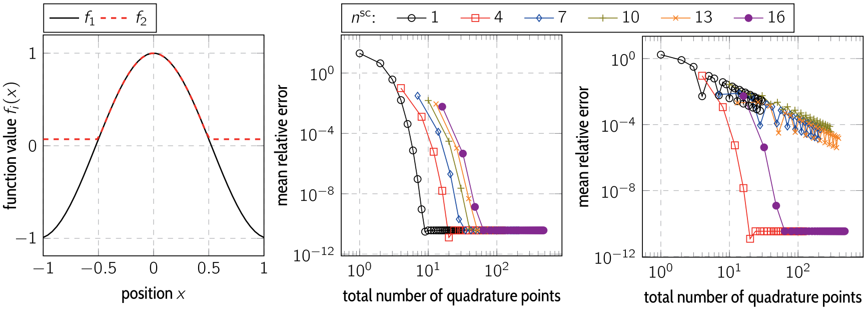


Figure 5.23: Considered function for the quadrature test (left) and approximation error for f_1 (middle) and f_2 (right).

where $\mathbf{f} = \mathbf{N}^{(e)^{\mathrm{T}}} \boldsymbol{t}$ results from the discretization of $\delta \boldsymbol{d} \cdot \boldsymbol{t}$ ($\mathbf{N}^{(e)}$ is the shape function matrix, see Section A.3.2). $\hat{\boldsymbol{J}}$ denotes the Jacobian matrix associated with the respective element face and is defined by

$$\hat{J}_{ij} = \frac{\partial X_i}{\partial r_j}. \tag{5.31}$$

Partitioning the element face as in the one-dimensional case,

$$\int\limits_{\Gamma_0^{(e)}} \mathbf{f} \, \mathrm{d}\Gamma_0^{(e)} \approx \sum_{k=1}^{n^{\mathrm{sc}}} \sum_{l=1}^{n^{\mathrm{sc}}} \sum_{i=1}^{n^{\mathrm{qp}}} \sum_{j=1}^{n^{\mathrm{qp}}} \mathbf{f}_{ijkl} \det\left(\hat{\boldsymbol{J}}\left(\hat{x}_{ik}^{(n1\mathrm{qp})}, \hat{x}_{jl}^{(n1\mathrm{qp})}\right)\right) \hat{w}_{ik}^{(n^{\mathrm{qp}})} \, \hat{w}_{jl}^{(n^{\mathrm{qp}})} \tag{5.32}$$

is obtained, where $\mathbf{f}_{ijkl} = \mathbf{f}\left(\boldsymbol{X}\left(\hat{x}_{ik}^{(n1\mathrm{qp})}, \hat{x}_{jl}^{(n1\mathrm{qp})}\right)\right)$. Of course, the finite element solution, which is based on the smooth shape functions in $\mathbf{N}^{(e)}$ may not be able to reproduce an accurate solution in the vicinity of discontinuities in the load. However, capturing the load accurately in an integral sense may already improve the overall solution significantly. A simple test case is considered to illustrate this. It consists of a beam, which is clamped at the left and subjected to pressure on the right as depicted in Figure 5.24. The beam's dimensions are 100 × 1 × 10 m. Its material behavior is described using the St. Venant-Kirchhoff model with a Young's modulus of $E = 12\,\mathrm{Pa}$ and a Poisson ratio of $\nu = 0$. The problem was also considered in [151],

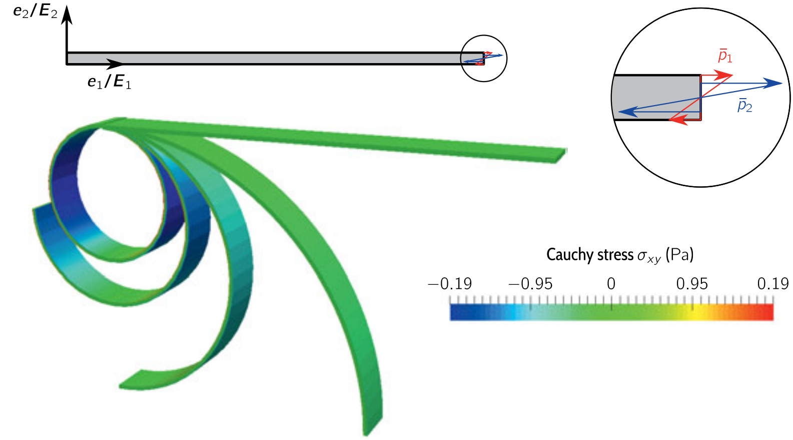


Figure 5.24: Test case *beam under follower load* considered for a comparison of the different quadrature rules used to approximate the traction integral in the FEM load vector.

and it constitutes a benchmark problem for follower loads. If

$$\bar{p}_1 = \frac{24\,\pi}{100}\left(Y - \frac{1}{2}\right) \tag{5.33}$$

holds for the prescribed pressure, the beam will approximately deform into a circle (the zero-thickness limit case will yield a perfect circle). It is assumed that a modified load

$$\bar{p}_2 = \begin{cases} \hat{p} = \alpha\,\frac{24\,\pi}{100}\left(Y - \frac{1}{2}\right) & \text{for } \frac{1}{5} \leq y < \frac{4}{5} \\ 0 & \text{else} \end{cases} \tag{5.34}$$

leads to a very similar displacement if the factor α is chosen such that an equivalent moment

$$m_z = \int\limits_{\Gamma^{\text{right}}} \left(y - \frac{1}{2}\right)\bar{p}_1\,\mathrm{d}y = \int\limits_{\Gamma^{\text{right}}} \left(y - \frac{1}{2}\right)\bar{p}_2\,\mathrm{d}y \tag{5.35}$$

is obtained. From Eq. (5.35), it is found that $\alpha = {}^{125}/_{27}$. This assumption is confirmed by a check with a discretization, where five elements in thickness directions are used such that the load $\bar{p}_2$ can be integrated exactly using standard Gaussian quadrature, see Figure 5.24. As shown therein as well, the subcell integration in combination with the original discretization (one

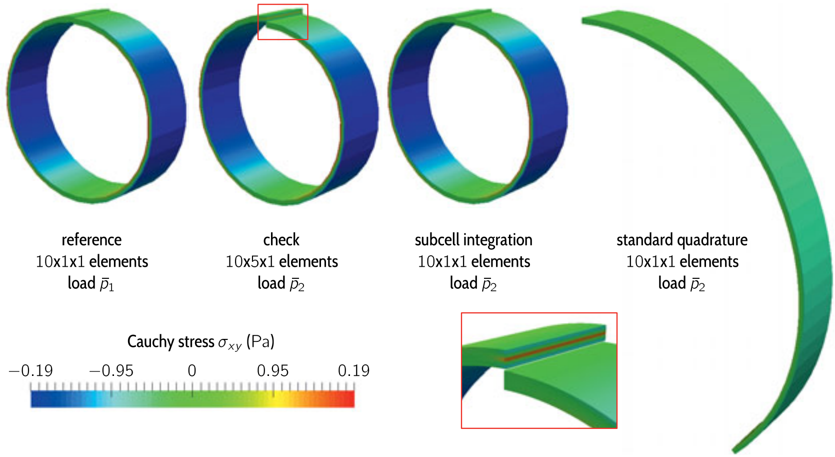


Figure 5.25: Comparison of the results for the test case *beam under follower load* obtained with different quadrature rules and a modified (discontinuous) load.

element in thickness direction) yields a similar result, while the standard quadrature does not. As in the one-dimensional case, this result is only obtained if the subcell boundaries correspond to the discontinuities in the integrand, i.e. for $n^{\mathrm{sc}} = 5$ in the present case. This example shows how an accurate approximation of the traction, at least in an integral sense, can improve the solution significantly. Of course, errors in the stresses in the vicinity of the discontinuity cannot be reduced by this technique. It is only for the check case where the element boundaries coincide with the discontinuity that the stresses can be assumed to be meaningful, see Figure 5.25.

5.2 Coupled benchmark problems

In the following, several benchmark problems are introduced. They are used to validate the coupling software and to investigate the performance of the introduced coupling algorithms and convergence acceleration methods.

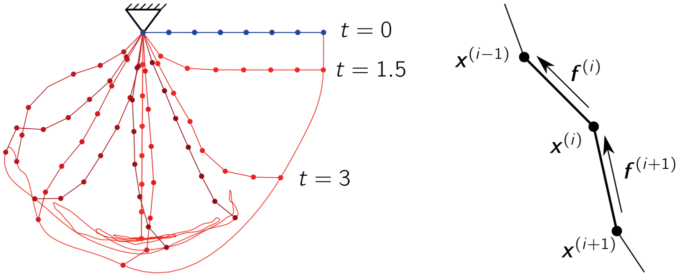


Figure 5.26: Multibody system used for the investigation of the coupling algorithms and convergence acceleration methods. Left: System behavior over time. Right: Indexing of the point masses and elastic rods.

Table 5.2: Parameters used in the simulations of the multibody system.

parameter	variable	value	unit
number of segments	n	8	-
total length	L	5	m
total mass	m	2	kg
damping coefficient	d	0.1	Ns/m
gravitational acceleration	g	10	m/s^2
elastic rod stiffness	c	100	N/m

5.2.1 Multi-body system

The numerous coupling algorithms and convergence acceleration schemes introduced in Chapter 4, Sections 4.3.1 and 4.3.2 shall be investigated in terms of efficiency and stability. Such a study has to be performed for a simplified problem that can be solved for the various possible parameter combinations in an acceptable time. To this end, a chain of point masses connected with elastic rods is considered as shown in Fig 5.26. Table 5.2 lists the parameters used in the simulations.

Governing equations

The mechanical behavior of the system is governed by the conservation of linear momentum

$$m\,\ddot{\boldsymbol{x}}^{(i)} + d\,\dot{\boldsymbol{x}}^{(i)} = \begin{bmatrix} 0 \\ -m\,g \end{bmatrix} + \boldsymbol{f}^{(i)} - \boldsymbol{f}^{(i+1)}, \qquad (5.36)$$

which must hold for every point mass i with position $\boldsymbol{x}^{(i)}(t)$. All points share the same mass m and damping coefficient d. The gravitational acceleration is denoted by g. The axial force in rod i is given by

$$\boldsymbol{f}^{(i)} = c \underbrace{\left(L^{(i)} - L_0\right)}_{\text{magnitude}} \underbrace{\left(\frac{\mathbf{x}^{(i-1)} - \mathbf{x}^{(i)}}{L^{(i)}}\right)}_{\text{direction}} = c\left(1 - \frac{L_0}{L^{(i)}}\right)\left(\mathbf{x}^{(i-1)} - \mathbf{x}^{(i)}\right), \tag{5.37}$$

where c is the spring stiffness common to all elastic rods. The current length of rod i is given by $L_i = \|\mathbf{x}_{i-1} - \mathbf{x}_i\|$, the common initial length is denoted by L^0. With the global vectors

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{x}^{(1)} \\ \vdots \\ \boldsymbol{x}^{(n)} \end{bmatrix} \quad \text{and} \quad \mathbf{f} = \begin{bmatrix} \begin{bmatrix} 0 & m\,g \end{bmatrix}^{\mathrm{T}} + \boldsymbol{f}^{(1)} \\ \begin{bmatrix} 0 & m\,g \end{bmatrix}^{\mathrm{T}} - \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)} \\ \vdots \\ \begin{bmatrix} 0 & m\,g \end{bmatrix}^{\mathrm{T}} - \boldsymbol{f}^{(n)} \end{bmatrix} \tag{5.38}$$

the equilibrium equations (5.36) can be combined to the global system

$$m\,\ddot{\mathbf{x}} + d\,\dot{\mathbf{x}} - \mathbf{f} = \mathbf{0}. \tag{5.39}$$

It is discretized in time using the Newmark-β method that introduced as a special variant of the generalized-α method in Table 4.1. Choosing $\beta = 0.25$ and $\gamma = 0.5$ the resulting nonlinear system of equations to be solved in each time step reads

$$\mathbf{g} = \left(\frac{4\,m}{\Delta t^2} - \frac{2\,d}{\Delta t}\right)(\mathbf{x}_{k+1} - \mathbf{x}_k) - \frac{4\,m}{\Delta t}\dot{\mathbf{x}}_k - m\,\ddot{\mathbf{x}}_k - d\,\dot{\mathbf{x}}_k - \mathbf{f}_{i+1} = 0. \tag{5.40}$$

The Newton-Raphson method introduced in the context of continuum mechanics in Section 4.1.1 is used to solve (5.40). A derivation of the linearized system is provided in Section A.6.

The equilibrium equations (5.39) are further solved explicitly using the central difference method. After inserting the corresponding discretizations for velocity

$$\dot{\mathbf{x}}_k = \frac{1}{2\,\Delta t}(\mathbf{x}_{k+1} - \mathbf{x}_{k-1}) \tag{5.41}$$

and acceleration

$$\ddot{\mathbf{x}}_k = \frac{1}{\Delta t^2}(\mathbf{x}_{k+1} - 2\mathbf{x}_k + \mathbf{x}_{k-1}) \tag{5.42}$$

in (5.39) one obtains

$$\mathbf{x}_{k+1} = \left(\frac{m}{\Delta t^2} + \frac{d}{2\,\Delta t}\right)^{-1} \left(\frac{m}{\Delta t^2}\,(2\,\mathbf{x}_k - \mathbf{x}_{k-1}) + \frac{d}{\Delta t}\,\mathbf{x}_k - \mathbf{f}_k\right) \tag{5.43}$$

as an update rule.

The two monolithic solution approaches serve as references for the investigation of the partitioned solution approaches. The latter are based on a coupled problem, which is obtained by condensing parts of the system into individual subproblems. One subproblem describes the movement of the point masses and another one describes the forces within the rods. The mass problem is then linear and using the Newmark discretization (again with $\beta = 0.25$ and $\gamma = 0.5$) the update rule

$$\mathbf{x}_{k+1} = \left(\frac{4\,m}{\Delta t^2} + \frac{2\,d}{\Delta t}\right)^{-1} \left[\left(\frac{4\,m}{\Delta t^2} + \frac{2\,d}{\Delta t}\right)\mathbf{x}_k + \left(\frac{4\,m}{\Delta t} + d\right)\dot{\mathbf{x}}_k + m\,\ddot{\boldsymbol{x}}_k - \bar{\mathbf{f}}_{k+1}\right] \tag{5.44}$$

is obtained. The bar over $\bar{\mathbf{f}}_{k+1}$ indicates, that within the mass problem, the forces are regarded as a prescribed coupling quantity. The force problem is obtained from Eq. (5.37) by using the position of the masses as a prescribed coupling quantity, i.e.

$$\boldsymbol{f}^{(i)} = c\left(1 - \frac{L_0}{L^{(i)}}\right)\left(\bar{\mathbf{x}}^{(i-1)} - \bar{\mathbf{x}}^{(i)}\right). \tag{5.45}$$

In order to find an appropriate time step size for the investigation of the partitioned solution approaches, a convergence study was performed. As an error measure, the position of point mass n^{pm} is considered at the time instances $t = 0\,\mathrm{s}, 2\,\mathrm{s}, \dots, 100\,\mathrm{s}$. The displacements are then collected in the vector

$$\mathbf{x}^{(n^{\mathrm{pm}})} = \begin{bmatrix} \boldsymbol{x}_0^{(n^{\mathrm{pm}})} & \boldsymbol{x}_{\frac{2}{\Delta t}}^{(n^{\mathrm{pm}})} & \dots & \boldsymbol{x}_{\frac{100}{\Delta t}}^{(n^{\mathrm{pm}})} \end{bmatrix}. \tag{5.46}$$

A monolithic, implicit simulation is performed with $\Delta t = 1 \times 10^{-3}\,\mathrm{s}$ to provide a reference solution $\mathbf{x}_{\mathrm{ref}}^{(n^{\mathrm{pm}})}$. The mean relative error is then computed as

$$e^{\mathrm{mbs}} = \frac{\left\|\mathbf{x}^{(n^{\mathrm{pm}})} - \mathbf{x}_{\mathrm{ref}}^{(n^{\mathrm{pm}})}\right\|}{\left\|\mathbf{x}_{\mathrm{ref}}^{(n^{\mathrm{pm}})}\right\|}. \tag{5.47}$$

The results shown in Figure 5.27 give rise to the following statements:

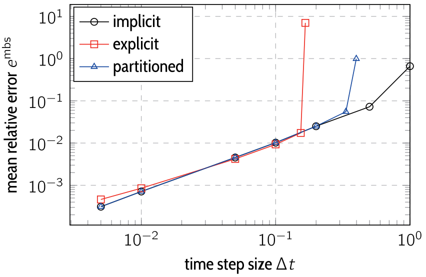


Figure 5.27: Convergence study for the multibody system using the different solution approaches. The partitioned simulations were exemplary carried out with a staggered coupling algorithm and a constant relaxation with $\omega = 0.5$.

- All solution approaches show a similar accuracy for low time step sizes ($\Delta t < 0.1\,\mathrm{s}$).
- The explicit approach using the central difference method becomes unstable for $\Delta t > 0.1\,\mathrm{s}$.
- The partitioned approach becomes unstable for $\Delta t > 0.2\,\mathrm{s}$.

Accordingly, the time step sizes $\Delta t = 0.1\,\mathrm{s}$ and $\Delta t = 0.2\,\mathrm{s}$ are used for the investigation of the coupling algorithms and convergence acceleration schemes. While for $\Delta t = 0.1$ it is expected that stability problems are not an issue, the case $\Delta t = 0.2\,\mathrm{s}$ may need some effort in terms of tuning the parameters involved in the solution approaches.

Partitioned solution approaches

The system is solved using the introduced coupling algorithms, i.e. the staggered one (with the force problem corresponding to the first subproblem), the parallel one as well as the mixed one. All convergence acceleration schemes introduced in Section 4.3.2 are employed while varying all parameters of each method. Table 5.3 lists the ranges for the considered parameters. Due to the high risk of overlooking certain parameter combinations, 1×10^{5} simulations are performed for each coupling algorithm and time step size. In each simulation, the convergence acceleration scheme as well as its parameters are determined in a random fashion. Further, the prescribed convergence tolerance e^{rel} is chosen randomly from $\left\{10^{-10}, 10^{-8}, 10^{-6}, 10^{-4}\right\}$.

This ensures that the conclusions about the efficiency of the solution approaches can be made with respect to whether a very accurate solution is demanded or not. For all simulations, the quantity of interest is the number of consecutive solver calls j^* (equal to the number of coupling iterations j only for the staggered and the mixed algorithms, c.f. Section 4.3.1).

Results

Figure 5.28 gives an overview of the staggered simulations with time step size $\Delta t = 0.1\,\mathrm{s}$. For the histogram, a window size of 1 was chosen, i.e. the bars show for how many simulations $0 < j^* \leq 1$, $1 < j^* \leq 2$, etc. Simulations, in which the maximum number of iterations did not suffice to fulfill the prescribed convergence criteria, are considered with $j^* = 100$ in Figure 5.28. The same holds for simulations that diverged or yielded a mean relative error $e^{\mathrm{psa}} > e^{\mathrm{rel}}$, where e^{psa} is computed like e^{mbs} (see Eq. 5.47) but using the monolithic solution with the respective time step size ($\Delta t = 0.1\,\mathrm{s}$ or $\Delta t = 0.2\,\mathrm{s}$) as a reference solution.

In addition to the histogram, a probability density function (PDF) is determined using a normal distribution as a kernel function. More precisely, the PDF is obtained by a superposition of normal distributions computed for each data point and reads

$$p_w\left(j^*\right) = \frac{1}{n^{\mathrm{s}}\, w} \sum_{i=1}^{n^{\mathrm{s}}} \frac{1}{\sqrt{2\,\pi}}\, e^{\frac{1}{2}\left(\frac{j^*-j_i^*}{w}\right)^2}. \tag{5.48}$$

Therein, n^{s} is the number of data points (in this case 1×10^5, the number of simulations performed for a given setting) and w is the bandwidth of the kernel distributions. According to the shape of the PDFs obtained for different bandwidths, $w = 1$ is chosen for subsequent investigations, since for $w^{\mathrm{k}} < 1$ or $w^{\mathrm{k}} > 1$ the resulting distributions are not smooth enough or too smooth, respectively.

In Figure 5.29, the PDFs for all coupling algorithms and time step sizes are shown. As expected, a first observation shows that the increase in the time step size results in an increase of j^*. Further, the standard deviations as well as the number of failed simulations (for which $j^* = 100$) is larger for the cases with $\Delta t = 0.2\,\mathrm{s}$. The characteristic quantities of each distribution given in Table 5.4 confirm this observation. The mean and minimum values as well as the standard deviations were obtained from simulations with $j^* < 100$ only, i.e. failed simulations were excluded from this statistic.

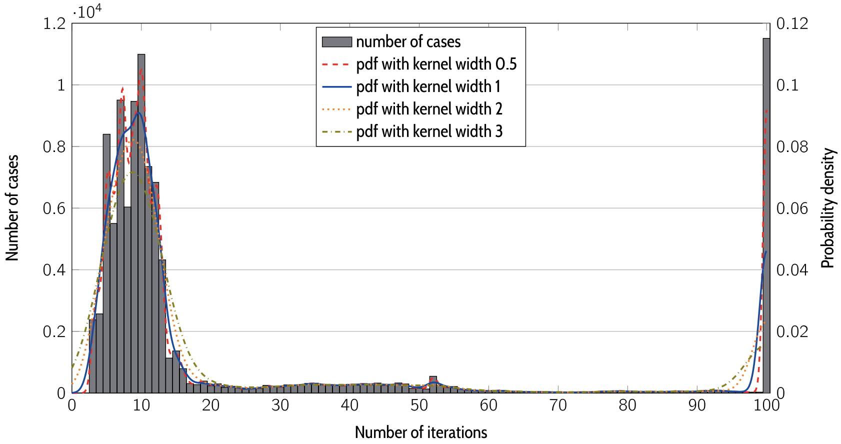


Figure 5.28: Number of simulations over number of iterations per time step for the multi-body system.

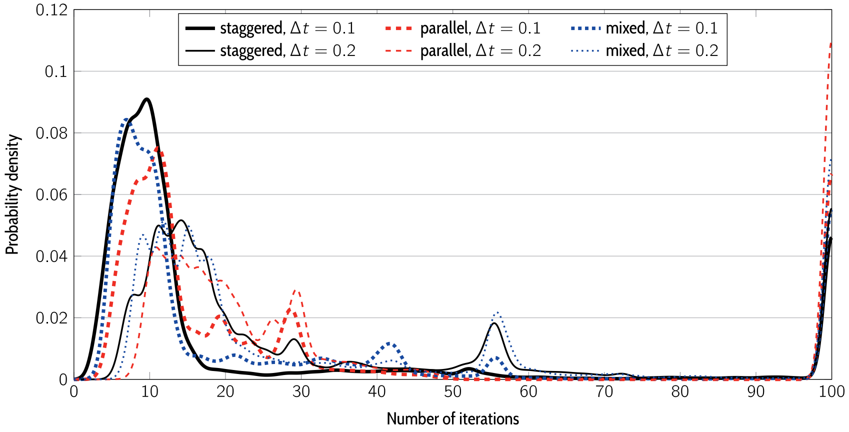


Figure 5.29: Comparison of the probability density functions determined for the different algorithms and time step sizes.

The characteristic quantities further show that the number of failed simulations is much higher for the parallel and the mixed coupling algorithms. Also the mean and minimum values are lower for the staggered simulations, the only exception being the mean number of consecutive solver calls for the parallel coupling algorithm with $\Delta t = 0.2$. However, the statistic includes all considered convergence acceleration schemes and parameter settings, some of which may be inappropriate for certain coupling algorithms and time step sizes. For a clearer picture of the performance differences, the results are investigated for the convergence acceleration schemes individually in the following.

Figure 5.30 shows the mean number of consecutive solver calls j^* over the mean relative error e^{psa} for the staggered simulations. In correspondence with the prescribed convergence criteria $(1 \times 10^{-10}$,1×10^{-8},1×10^{-6},$1 \times 10^{-4})$, the data point's abscissas cluster below these values. As expected, a strong correlation between the prescribed tolerance e^{rel} (respectively the error e^{psa}) and j^* can be observed.

Due to their qualitatively different behavior, the convergence acceleration methods are divided into two groups in Figure 5.30. One group includes all variants of the QNLS method, another group includes all other methods as well as the standard QNLS method for comparison. For better readability, only the first 5000 simulations are shown in each plot after making sure that the features observed for the full set of simulations are preserved. Again, failed simulations with $j^* = 100$ were excluded in advance. As a general observation, it is noted that the mean number of solver calls j^* shows a much smaller deviation for the QNLS methods than for the other methods. This indicates that the QNLS methods are less dependent on their parameters, which is of course a desired feature. It is also noted that the backtrack relaxation is among the most performant methods (for the staggered algorithm considered in Figure 5.30), especially for the cases with high tolerances e^{rel}.

The minimum and mean j^* (excluding failed simulations) is depicted in Figure 5.31 for all considered convergence acceleration methods, coupling algorithms and time step sizes. It is found that the differences between the coupling strategies are much more pronounced for the larger time step size. The results obtained for the smaller time step size are therefore excluded from further investigations. From the mean and minimum values of j^*, two important observations for subsequent simulations can be made.

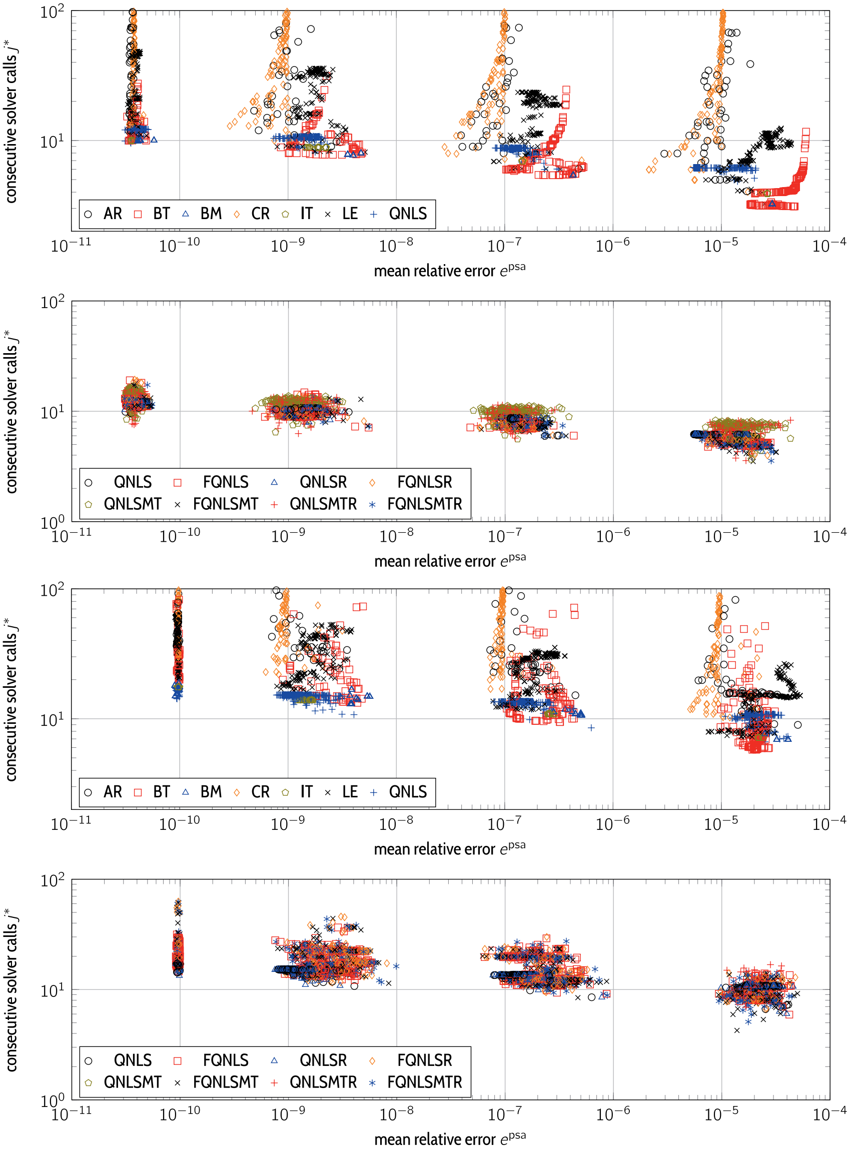


Figure 5.30: Number of coupling iterations over the mean relative error for the first 5000 successful (not diverged) simulations with $\Delta t = 0.1\,\mathrm{s}$ (top two plots) and $\Delta t = 0.2\,\mathrm{s}$ (bottom two plots).

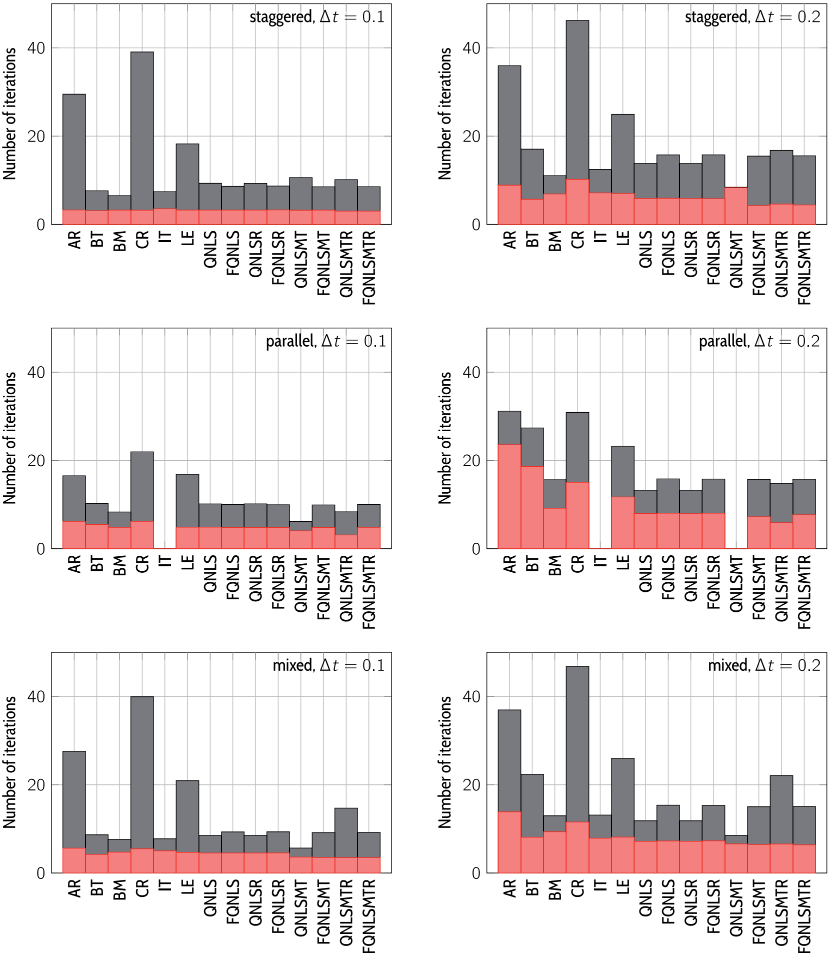


Figure 5.31: Mean and minimum number of iterations for the considered coupling algorithms and time step sizes.

- The multi time step QNLS methods are prone to instabilities if no resetting or filtering is used. In combination with the staggered coupling algorithm, only four of the (approximately 7000) simulations did not fail, which explains the small difference between the mean an minimum j^*. For the parallel algorithm, none of the simulation employing this convergence acceleration method ran successfully, for the mixed algorithm,

- The Irons-Tuck relaxation cannot be used as introduced in combination with the parallel coupling algorithm. All simulation employing this combination failed. This constitutes a pitfall that is easily overlooked because in previous investigations of the staggered algorithm, the Irons-Tuck relaxation seemed to be more stable (yet less efficient) than the QNLS methods (see [163]).

Finally, the influence of the method parameters shall be investigated. While the data obtained from the 10^5 simulations that were performed for each time step size and coupling algorithm contains much more information than can be presented in this work, selected aspects that are most interesting are considered in the following. At first, a comparison of the constant relaxation, the Aitken relaxation and the Irons-Tuck relaxation is made. Figure 5.32 shows the mean number of consecutive solver calls in relation to the initial relaxation factor ω_0. Considering the smaller time step size, it is clearly observed that the methods do not improve the efficiency of the simulations. A relaxation parameter of $\omega_0 = 1$ (or even larger) yields the smallest j^*. Regarding the larger time step size, there exist a minimum for the constant relaxation and the Aitken relaxation at $\omega_0 \approx 0.5$. Remarkably, such an optimal value does not exist for the Irons-Tuck relaxation. It seems to be completely independent of the initial relaxation factor for the considered problem and leads to much smaller j^* than the other methods. The only observable differences in j^* stems from the prescribed convergence tolerances, which as in the scatter plots on Figure 5.30 yield clearly distinguishable groups (identifiable as four individual lines for each method in Fig. 5.32). Investigating the relation between ω_0 and j^* for the parallel coupling algorithm (see Figure 5.33), it is found that there exist an optimum value smaller than one for both time step sizes. Results for the Irons-Tuck relaxation are not available, since all according simulations failed as mentioned above. The relation between ω_0 and j^* for the mixed coupling algorithm does not provide further insights and is qualitatively

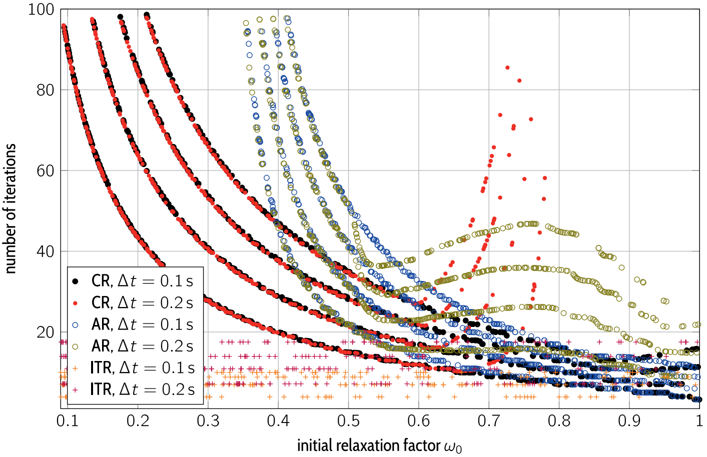


Figure 5.32: Influence of the (initial) relaxation factor ω_0 in the constant (CR), the Aitken (AR) and the Irons-Tuck relaxation (ITR) in combination with the staggered coupling algorithm.

equivalent to the one for the staggered algorithm, see Figure A.8.3).

As could be observed from the scatter plots (see e.g. Fig. 5.30), the QNLS methods were almost independent of the respective parameters listed in Table 5.3. Plots as the ones considered above for the relaxation methods show almost horizontal lines. Methods that were found to noticeably depend on their parameters were found to be the backtrack relaxation and the line extrapolation method. Figures 5.35 and 5.34 show the relation between the parameters and the number of consecutive solver calls j^*. Again, minima can be observed, which depend on the time step size and the coupling algorithm. For both methods, the staggered and the mixed coupling algorithm yield lower values for j^* and should be preferred accordingly.

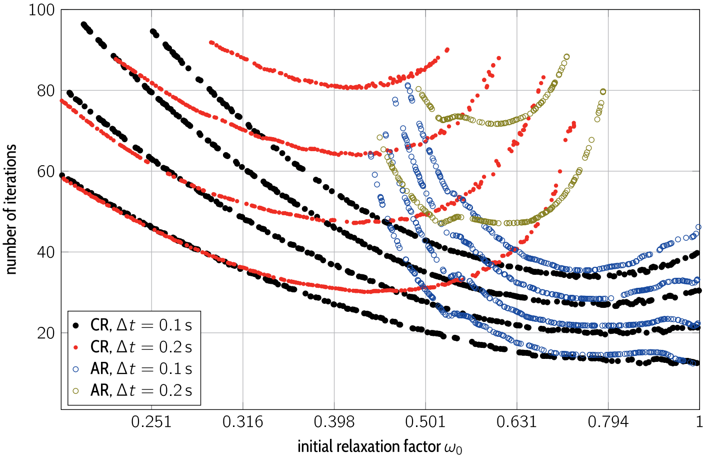


Figure 5.33: Influence of the (initial) relaxation factor ω_0 in the constant (CR), the Aitken (AR) and the Irons-Tuck relaxation (ITR) in combination with the parallel coupling algorithm.

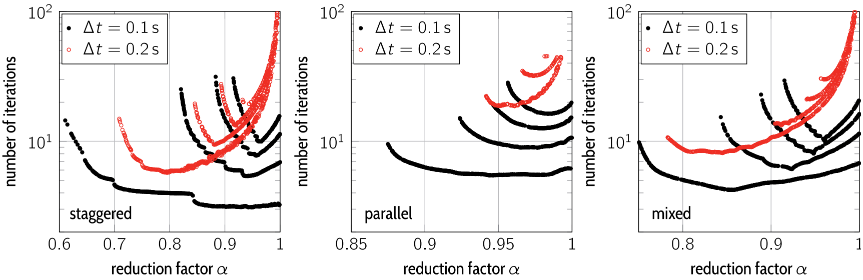


Figure 5.34: Influence of the reduction factor α in the backtrack relaxation combination with the staggered (left), the parallel (middle) and the mixed (right) coupling algorithm.

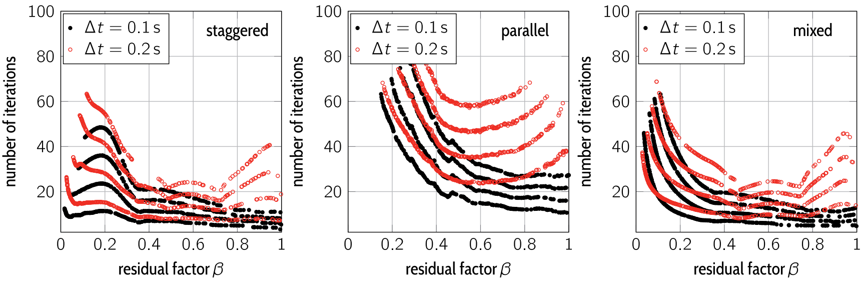


Figure 5.35: Influence of the residual factor β in the linear line extrapolation method combination with the staggered (left), the parallel (middle) and the mixed (right) coupling algorithm.

Table 5.3: Considered ranges for the parameters of the convergence acceleration schemes.

convergence acceleration method	parameter	variable	range
Aitken relaxation (AR)	initial relaxation factor	ω_0	$[10^{-3}, 1]$
backtrack relaxation (BTR)	reduction factor	α	$[0.5, 1]$
Broyden method (BM)	initial relaxation factor	ω_0	$[10^{-3}, 1]$
constant relaxation (CR)	relaxation factor	ω	$[10^{-3}, 1]$
Irons-Tuck relaxation (ITR)	initial relaxation factor	ω_0	$[10^{-3}, 1]$
Linear line extrapolation (LLE)	initial relaxation factor residual factor	α_0 β	$[10^{-3}, 1]$ $[10^{-1}, 1]$
standard QNLS (QNLS)	initial relaxation factor	ω_0	$[10^{-3}, 1]$
filtered QNLS (FQNLS)	initial relaxation factor singularity tolerance	ω_0 β	$[10^{-3}, 1]$ $[10^{-6}, 10^{-1}]$
QNLS with reset (QNLSR)	initial relaxation factor iterations before reset	ω_0 n^{r}	$[10^{-3}, 1]$ $\{6, 7, \ldots, 25\}$
filtered QNLS with reset (FQNLSR)	initial relaxation factor singularity tolerance iterations before reset	ω_0 β n^{r}	$[10^{-3}, 1]$ $[10^{-6}, 10^{-1}]$ $\{6, 7, \ldots, 25\}$
multi time step QNLS (QNLSMT)	initial relaxation factor saved time steps	ω_0 n^{sts}	$[10^{-3}, 1]$ $\{2, 3, \ldots, 20\}$
filtered multi time step QNLS (FQNLSMT)	initial relaxation factor singularity tolerance saved time steps	ω_0 β n^{sts}	$[10^{-3}, 1]$ $[10^{-6}, 10^{-1}]$ $\{2, 3, \ldots, 20\}$
multi time step QNLS with reset (QNLSMTR)	initial relaxation factor iterations before reset saved time steps	ω_0 n^{r} n^{sts}	$[10^{-3}, 1]$ $\{6, 7, \ldots, 25\}$ $\{2, 3, \ldots, 20\}$
filtered multi time step QNLS with reset (FQNLSMTR)	initial relaxation factor singularity tolerance iterations before reset saved time steps	ω_0 β n^{t} n^{sts}	$[10^{-3}, 1]$ $[10^{-6}, 10^{-1}]$ $\{2, 3, \ldots, 20\}$ $\{2, 3, \ldots, 20\}$

Table 5.4: Comparison of the considered coupling algorithms and time step sizes.

algorithm	Δt	mean	minimum	standard deviation	failed
staggered	0.1	10.2	3.0	8.2	11509
	0.2	16.2	4.3	10.1	13881
parallel	0.1	10.5	3.1	4.6	16721
	0.2	15.8	5.9	5.7	27597
mixed	0.1	10.8	3.5	9.2	13798
	0.2	16.5	6.3	10.4	17970

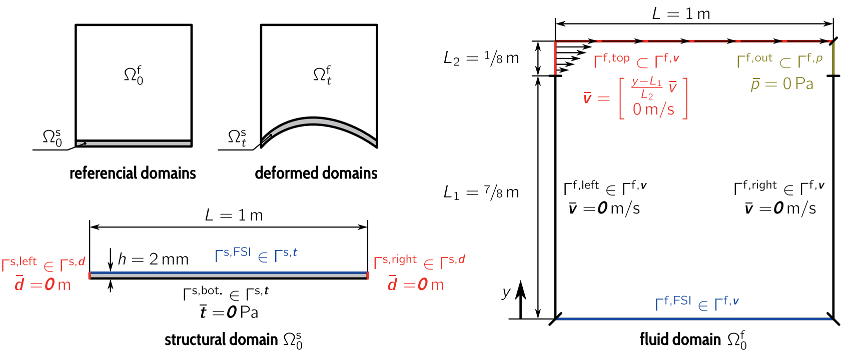

$\Omega^{\rm f}_0$
$\Omega^{\rm s}_0$
referencial domains
$\Omega^{\rm f}_t$
$\Omega^{\rm s}_t$
deformed domains
$L = 1\,\rm m$
$\Gamma^{\rm s,left} \in \Gamma^{\rm s,\boldsymbol{d}}$
$\bar{\boldsymbol{d}} = \boldsymbol{0}\,\rm m$
$h = 2\,\rm mm$
$\Gamma^{\rm s,FSI} \in \Gamma^{\rm s,\boldsymbol{t}}$
$\Gamma^{\rm s,right} \in \Gamma^{\rm s,\boldsymbol{d}}$
$\bar{\boldsymbol{d}} = \boldsymbol{0}\,\rm m$
$\Gamma^{\rm s,bot.} \in \Gamma^{\rm s,\boldsymbol{t}}$
$\bar{\boldsymbol{t}} = \boldsymbol{0}\,\rm Pa$
structural domain $\Omega^{\rm s}_0$
$L = 1\,\rm m$
$L_2 = 1/8\,\rm m$
$\Gamma^{\rm f,top} \subset \Gamma^{\rm f,\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \begin{bmatrix} \frac{y-L_1}{L_2}\,\bar{v} \\ 0\,\rm m/s \end{bmatrix}$
$\Gamma^{\rm f,out} \subset \Gamma^{\rm f,p}$
$\bar{p} = 0\,\rm Pa$
$L_1 = 7/8\,\rm m$
$\Gamma^{\rm f,left} \in \Gamma^{\rm f,\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \boldsymbol{0}\,\rm m/s$
$\Gamma^{\rm f,right} \in \Gamma^{\rm f,\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \boldsymbol{0}\,\rm m/s$
y
$\Gamma^{\rm f,FSI} \in \Gamma^{\rm f,\boldsymbol{v}}$
fluid domain $\Omega^{\rm f}_0$

Figure 5.36: Geometry and boundary conditions for the benchmark case lid-driven cavity flow.

5.2.2 Lid-driven cavity flow

The lid-driven cavity flow is a standard benchmark problem for computational fluid dynamics. As illustrated in Fig. 5.36, it is augmented by a structural problem describing a flexible bottom of the cavity to provide a benchmark problem for fluid-structure interaction problems. The two-dimensional structural model is clamped at its left and right ends. Zero traction is assumed at the bottom, while the fluid loads are prescribed at the top boundary (the wet surface). On the fluid side, the bottom boundary corresponds to the wet surface. The domain displacement as well as the velocity are prescribed here according to the displacements of the structural problem. On the bottom part of the left and right boundary of the fluid domain, a rigid wall boundary condition (zero velocity and zero domain displacement) is prescribed. On the upper part of the right boundary, where $y > L_1$, a pressure boundary condition is prescribed. Accordingly, only the velocity gradient in the normal direction is constraint (cf. Eq. (3.112)) such that fluid may flow into or out of the domain at this boundary. On the upper part of the left boundary as well as the top boundary, a velocity boundary condition is prescribed in combination with zero domain displacement. The maximum value of the x-component of the velocity is given by

$$\bar{v} = \left(1 - \cos\left(\frac{2\,\pi\,t}{5\,\rm s}\right)\right)\ \rm m/s. \tag{5.49}$$

The structural problem is discretized with high-order finite elements as introduced in Section 4.1.1. In particular, 16 quadrilateral continuum ele-

Table 5.5: Parameters for the benchmark case *lid-driven cavity flow*.

fluid	density	ρ^{f}	$1\,\mathrm{kg/m^3}$
	kinematic viscosity	ν^{f}	$0.01\,\mathrm{m/s^2}$
structure	density	ρ^{s}	$500\,\mathrm{kg/m^3}$
	Young's modulus	E	$500\,\mathrm{Pa}$
	Poisson ratio	ν	0
FEM	time step size	Δt	$0.01\,\mathrm{s}$
	spectral radius	ρ_∞	0.8
FVM	time step size	Δt	$0.01\,\mathrm{s}$

ments of equal shape and order $p = 2$ with a plain stress assumption are used. The fluid problem is discretized using the finite volume method, see Section 4.1.2. A mesh with 40 by 40 equally shaped finite volumes is used. These discretizations are comparably coarse such that a mesh dependent solution can be expected. However, the case was previously considered in [214, 145] with a similarly coarse discretization on the fluid side and shall be resembled here as closely as possible. The parameters for fluid and structure as well as the respective numerical methods are given in Tab. 5.5.

Results

In Fig. 5.37 the velocity and pressure field at a representative time instance are shown to give a general overview of the solution. Further, the deformed fluid meshes resulting from the different mesh deformation techniques introduced in Section 3.2.4 are shown. It can be observed that the two variants of Laplacian smoothing as well as the radial basis function interpolation provide good quality meshes, while the consideration of the fluid domain as pseudo solid yields an acceptable yet less well deformed mesh in terms of element size and aspect ratio.

To ensure the validity of the solution, the displacement in y-direction of the wet surface's midpoint is compared to the solution obtained in [214] and [145]. Due to the lack of information about the precise boundary conditions used in [145], the present solution as well as that from [214] differ significantly. The small difference observed between the present solution and that from [214] may result from different mesh deformation strategies. As can be seen, using the pseudo solid approach changes the results to an extend which is comparable to this difference. While the applied mesh deformation

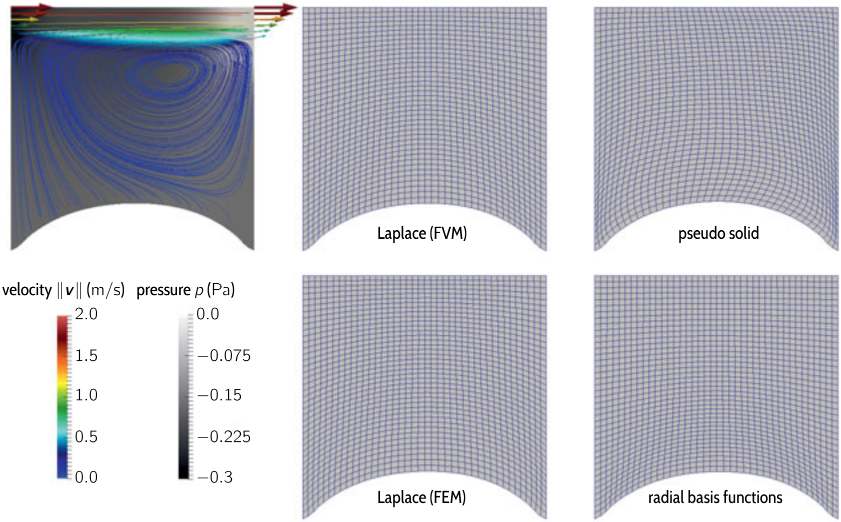


Figure 5.37: Pressure and velocity field for the lid-driven cavity flow and deformed meshes at time $t = 22.5\,\mathrm{s}$.

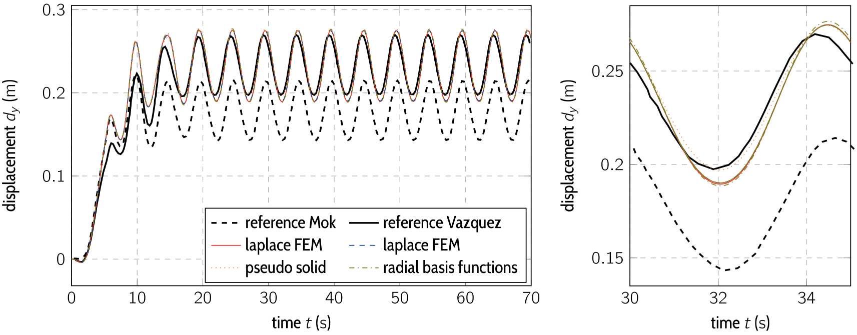


Figure 5.38: Displacement of the flexible bottoms center point in the benchmark case *lid-driven cavity flow*.

technique (not stated in [214]) could accordingly explain the differences, this is unlikely, since all other mesh deformation techniques yield the same result and the pseudo solid approach yields by far the worst mesh distortions. It is therefore believed that the difference stem from different time integration methods. While in [214] the generalized-α method is applied on the fluid side, the implicit Euler method is used here.

As done for the multi-body system, the lid-driven cavity flow is simulated using different coupling algorithms and convergence acceleration methods. The mean number of consecutive solver calls j^* (see Fig. 4.14) is compared for the introduced coupling algorithms and convergence acceleration methods that performed best in the simulation of the multi-body system in Section 5.2.1. Again, three different convergence criteria are considered, all of which result in the solution shown in Fig. 5.38 (up toa small difference in the range of the prescribed relative tolerance r^{rel}). Tables 5.6 and 5.8 list the mean number of consecutive solver calls j^* for each criterion. The comparison confirms most of the statements derived from the investigation in Section 5.2.1. This includes the issues related to the unfiltered multi time step QNLS method without resetting (QNLSMT) in combination with any coupling algorithms as well as those related to the Irons-Tuck relaxation (IT) in combination with the parallel coupling algorithms. However, some of the statements found for the multi-body system do not hold for the FSI problem considered here.

- The backtrack relaxation is not among the most performant convergence acceleration methods. In fact, in none of the simulations it managed to bring down the residual below the prescribed tolerance as indicated by the dash.

- A high tolerance does not necessarily lead to a lower j^*. Instead, for some solution methods, the (in accordance with the higher tolerance) more inaccurate solution in a certain time step leads to the need for more coupling iterations in later time steps, which may actually yield an increased j^*.

- The filtered multi-time-step QNLS method with resetting (FQNLSMTR) is not in general the most performant method. For the low tolerance case with $e^{\text{rel}} = 10^{-7}$ and $e^{\text{abs}} = 10^{-12}$, the multi time step variants of the QNLS method are actually outperformed by the single-time-step variants. However, such a low tolerance scenario may not be an appropriate reference, since a much higher tolerance is usually sufficient.

Assuming that the obtained results are generalizable, it can be stated that the FQNSLMTR should be the first choice when deciding on a convergence acceleration method for a coupled problem. In case this multi time step method fails to bring down the residual to the prescribed tolerance or leads to a diverging solution, the FQNLSR method should be tried secondly. For a trade off between efficiency and implementation effort, the Irons-Tuck relaxation may be considered, however, only in combination with the staggered or the mixed coupling algorithm. While the numbers in Tab. 5.5 indicate that the performance of the parallel and mixed coupling algorithms are considerably higher than that of the staggered one, the decision on the coupling algorithm must still be taken with care. As described in Section 4.3.1, the mean number of consecutive solver calls j^* is only an objective measure for the performance if the considered subproblems require the same amount of computation time or can be parallelized such that they require the same amount of real time. Even if this assumption holds, such that the amount of real time needed is indeed proportional to j^*, the staggered algorithm will still come with the advantage of a lower number of processes being active at the same time. This advantage is actually the dominating one, at least if more than one simulation is to be performed as explained in the following.

One may consider the example of a coupled problem, which has to be solved for two sets of parameters and which consists of two subproblems that require the same time for the solution of one time step. Assuming that the simulation shall be run on a computer with two processors, the question arises, whether to run the two simulations at the same time or one after the other. For the parallel and for the mixed algorithm, running the simulation at the same time is not a reasonable choice, since all processors are busy at all times and two simultaneously running simulations would thwart each other. Regarding the staggered algorithm, where within one simulation one of the field solver processes is always idle or paused, the operating system usually manages to distribute the processes among the processors in a way that allows for running two simulations at the same time efficiently. Assuming that the operating system does this perfectly, the usage of the parallel and mixed algorithm is only favorable, if the mean number of consecutive solver calls j^* is only half of that needed in the staggered case. In reality, such a perfect process management is of course not possible, such that the parallel and mixed algorithms are also preferable when the difference in j^* is not as large. On the other hand, also a coupled problem with subproblems that need exactly the same amount of time does not exist. That is, even if the amount of processes is unlimited or only a single simulation needs to be performed, a moderately higher j^* for the staggered algorithm may still result in lower computation times. Following these arguments and looking at the numbers in Tab. 5.6–5.8, the staggered coupling algorithm is recommended. However, the arguments indicate that this is a difficult decision, which depends on the available hardware and software. For a detailed investigation of the performance, a true measurement of computation time and real time is therefore inevitable. The decision for the staggered coupling algorithm is underpinned here by the fact, that it is the most stable one. While only the backtrack relaxation and the multi time step QNLS method without filtering fail to solve the problem in the staggered case, several other convergence acceleration methods fail if the parallel or the mixed algorithm is used.

5.2.3 Two-dimensional flag in channel flow

In [208] a benchmark case was proposed which is considered here to verify the applied partitioned solution approach. It consists of a two-dimensional channel flow as illustrated in Figure 5.39. A rigid cylinder is placed eccentrically in the channel, leading to a small difference between the flow velocity and the pressure above and below. At the downstream side of the cylinder,

Table 5.6: Performance of various convergence acceleration schemes for the benchmark case lid-driven cavity flow and high convergence tolerance $e^{\text{rel}} = 1 \times 10^{-3}$, $e^{\text{abs}} = 1 \times 10^{-8}$.

convergence acceleration method	staggered	parallel	mixed
Backtrack relaxation, $\alpha = 0.98$	-	-	-
Broyden method, $\omega_0 = 1$	7.85	-	10.1
Irons-Tuck relaxation, $\omega_0 = 0.5$	6.97	-	5.50
QNLS method, $\omega_0 = 0.5$	6.83	3.98	5.64
FQNLSR method, $\omega_0 = 0.5$	7.41	4.62	5.58
QNLSMT method, $\omega_0 = 0.5$, $n^{\text{sts}} = 10$	-	-	-
FQNLSMTR method, $\omega_0 = 0.25$, $n^{\text{sts}} = 10$	6.81	3.26	4.23

Table 5.7: Performance of various convergence acceleration schemes for the benchmark case *lid-driven cavity flow* and a moderate convergence tolerance $e^{\text{rel}} = 1 \times 10^{-5}$, $e^{\text{abs}} = 1 \times 10^{-10}$.

convergence acceleration method	staggered	parallel	mixed
Backtrack relaxation, $\alpha = 0.98$	-	-	-
Broyden method, $\omega_0 = 1$	6.83	-	5.95
Irons-Tuck relaxation, $\omega_0 = 0.5$	7.47	-	6.21
QNLS method, $\omega_0 = 0.5$	5.99	4.12	5.93
FQNLSR method, $\omega_0 = 0.5$	6.00	4.13	-
QNLSMT method, $\omega_0 = 0.5$, $n^{\text{sts}} = 10$	-	3.00	-
FQNLSMTR method, $\omega_0 = 0.25$, $n^{\text{sts}} = 10$	4.77	2.68	4.50

Table 5.8: Performance of various convergence acceleration schemes for the benchmark case *lid-driven cavity flow* and a low convergence tolerance $e^{\text{rel}} = 1 \times 10^{-7}$, $e^{\text{abs}} = 1 \times 10^{-12}$.

convergence acceleration method	staggered	parallel	mixed
Backtrack relaxation, $\alpha = 0.98$	-	-	-
Broyden method, $\omega_0 = 1$	8.71	-	10.4
Irons-Tuck relaxation, $\omega_0 = 0.5$	9.67	-	8.6
QNLS method, $\omega_0 = 0.5$	7.64	5.52	8.2
FQNLSR method, $\omega_0 = 0.5$	7.98	5.96	-
QNLSMT method, $\omega_0 = 0.5$, $n^{\text{sts}} = 10$	-	-	-
FQNLSMTR method, $\omega_0 = 0.25$, $n^{\text{sts}} = 10$	-	-	7.76

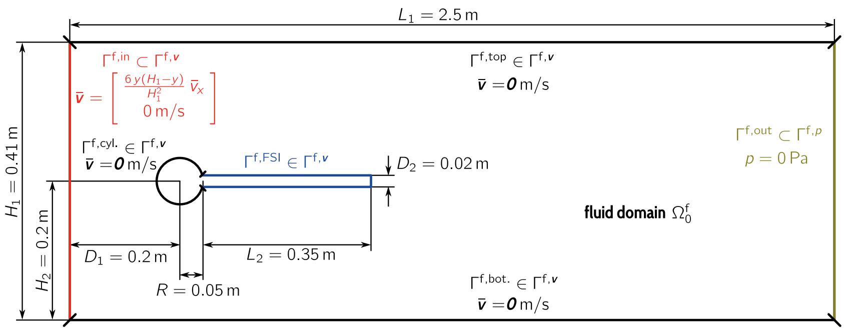

$L_1 = 2.5\,\mathrm{m}$
$\Gamma^{\mathrm{f,in}} \subset \Gamma^{\mathrm{f},\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \begin{bmatrix} \frac{6\,y(H_1-y)}{H_1^2}\,\bar{v}_x \\ 0\,\mathrm{m/s} \end{bmatrix}$
$\Gamma^{\mathrm{f,top}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \boldsymbol{0}\,\mathrm{m/s}$
$\Gamma^{\mathrm{f,out}} \subset \Gamma^{\mathrm{f},p}$
$p = 0\,\mathrm{Pa}$
$\Gamma^{\mathrm{f,cyl.}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \boldsymbol{0}\,\mathrm{m/s}$
$\Gamma^{\mathrm{f,FSI}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$
$D_2 = 0.02\,\mathrm{m}$
fluid domain Ω_0^{f}
$H_1 = 0.41\,\mathrm{m}$
$H_2 = 0.2\,\mathrm{m}$
$D_1 = 0.2\,\mathrm{m}$
$L_2 = 0.35\,\mathrm{m}$
$R = 0.05\,\mathrm{m}$
$\Gamma^{\mathrm{f,bot.}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$
$\bar{\boldsymbol{v}} = \boldsymbol{0}\,\mathrm{m/s}$

Figure 5.39: Geometry and boundary conditions for the benchmark case *two-dimensional flag in channel flow.*

a flexible flag is attached, whose movement is the subject of investigation.

The flow is driven by a parabolic velocity profile, which is prescribed at the inlet boundary $\Gamma^{\mathrm{f,in}}$. The mean flow velocity $\bar{v}$ is increased smoothly at the beginning of the simulation, such that

$$\bar{v}_x = \begin{cases} \frac{1}{2}\left(1 - \cos\left(\frac{\pi\,t}{T}\right)\right)\bar{v}_{\mathrm{max}} & \text{if } t < T \\ \bar{v}_{\mathrm{max}} & \text{else.} \end{cases} \tag{5.50}$$

Three different scenarios are considered and named FSI1, FSI2 and FSI3 hereafter as in [208]. The parameter settings for each of then is given in Table 5.9. For the scenario FSI1, the mean inflow velocity is comparably low, leading to a steady state solution, where the flag is bended upwards. For the scenarios FSI2 and FSI3, the flag eventually shows a periodic motion, induced by a von Kármán vortex street, which develops due to the fluid-structure interaction. The amplitude and period of the motion are very sensitive to the physical as well as the numerical parameters, which renders these scenarios ideal benchmark problems. The scenario FSI3 further poses a challenge for the partitioned coupling procedure. Due to the increased time step size Δt and the decreased structural density ρ^{s} as compared to the other cases, a large added mass effect is present. Without suitable convergence acceleration methods, this leads to instabilities or a large number of coupling iterations per time step. In addition, the smaller Young's modulus in the FSI3 scenario as compared to the FSI2 scenario leads to larger deformations, which poses a challenge for the mesh motion solver.

Table 5.9: Parameters for the benchmark case *two-dimensional flag in channel flow*.

			FSI1	FSI2	FSI3	
fluid	density	ρ^{f}	10^3	10^3	10^3	kg/m^3
	kinematic viscosity	ν^{f}	10^{-3}	10^{-3}	10^{-3}	m/s^2
	mean inflow velocity	$\bar{v}_{\mathrm{max}}$	0.2	1	2	m/s
structure	density	ρ^{s}	10^3	10^4	10^3	kg/m^3
	Young's modulus	E	1.4×10^6	5.6×10^6	1.4×10^6	Pa
	Poisson ratio	ν	0.4	0.4	0.4	-
FEM	time step size	Δt	10^{-3}	10^{-3}	10^{-2}	s
	Newmark parameters	β	0.25	0.25	0.25	-
		γ	0.5	0.5	0.5	-
FVM	time step size	Δt	10^{-3}	10^{-3}	10^{-2}	s

All scenarios are simulated using high-order finite elements to discretize the structural problem. The Newmark method is used here for time integration with the parameters given in Table 5.9. For the fluid problem, the finite volume method is used as the spatial discretization method while the implicit Euler method is used for time integration. A staggered coupling algorithm is used in combination with the FQNLSMTR method for convergence acceleration. As a predictor, the tangent extrapolation is applied. For a comparison with the reference solution provided in [208], the displacement of the flag tip in x-direction d_x and its displacement in y-direction d_y are considered. Further, the total drag force

$$f_x = \int\limits_{\Gamma_t^{\mathrm{f,FSI}}} \boldsymbol{\sigma}\,\boldsymbol{n} \cdot \boldsymbol{e}_x \,\mathrm{d}\Gamma_t^{\mathrm{f,FSI}} \tag{5.51}$$

and the total lift force

$$f_y = \int\limits_{\Gamma_t^{\mathrm{f,FSI}}} \boldsymbol{\sigma}\,\boldsymbol{n} \cdot \boldsymbol{e}_y \,\mathrm{d}\Gamma_t^{\mathrm{f,FSI}} \tag{5.52}$$

are considered. As introduced in Chapter 3, $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ denote unit vectors in the direction of the respective coordinate axis and $\boldsymbol{n}$ denotes the unit outward normal at the coupling interface.

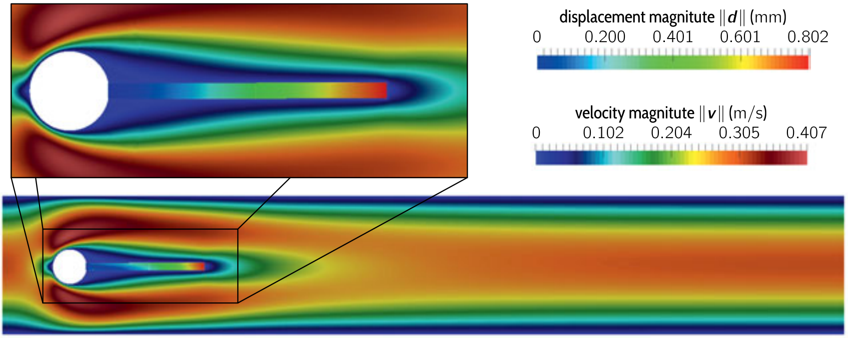


Figure 5.40: Velocity field and displacement field at $t = 10\,\mathrm{s}$ for scenario FSI1.

Results

Figure 5.40 and 5.41 show snap shots of the displacement and velocity fields for the FSI1 and FSI2 case, respectively, to give an impression of the dynamic behavior of the system. The situation in the FSI3 case phenomenologically matches that of the FSI2 case, however, the frequency of the oscillatory motion of the flag is higher, while its amplitude is lower. A quantitative evaluation of the three scenarios is given in Figure 5.42. For the FSI1 case, the displacement of the flag tip as well as the lift and the drag force are plottet for the entire simulation time of $t^{\mathrm{e}} = 10\,\mathrm{s}$. As can be seen, the system slowly reaches a steady state, where the inspected quantities are in good agreement with the reference solution from [208]. For the cases FSI2 and FSI3, appropriate time intervals at the end of the simulation time are selected. The motion shows a periodic behavior, which is in good agreement with the reference solution from [208] regarding both, amplitude and frequency.

5.2.4 Pulse wave in an elastic tube

In this benchmark case, the pulse wave propagation in an elastic tube is considered, similar to the one-dimensional test case from Section 5.1.3 but using a fully resolved simulation approach. Accordingly, a fluid and a structural subproblem are considered. Details regarding the geometry and the spatial discretizations are provided in Fig. 5.43.

At the inlet boundary of the fluid domain, a pressure boundary condition

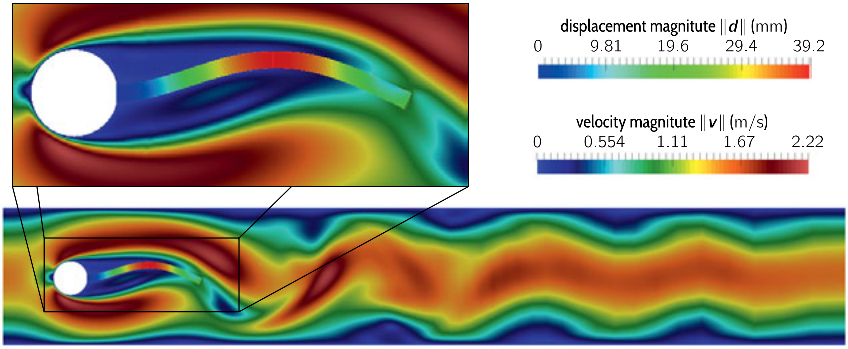


Figure 5.41: Velocity field and displacement field at $t = 10\,\mathrm{s}$ for scenario FSI2.

with

$$\bar{p} = \begin{cases} \frac{1}{2}\cos\left(\frac{2\pi}{T_\mathrm{p}}\right) 1.3332\,\mathrm{kPa} & \text{if } t < T_\mathrm{p} \\ 0\,\mathrm{kPa} & \text{else} \end{cases} \tag{5.53}$$

is prescribed. As in [75], where a similar problem is considered, the pulse duration is set to $T_\mathrm{p} = 2\,\mathrm{ms}$. At the outlet boundary, a pressure of $\bar{p}^{\mathrm{out}} = 0\,\mathrm{Pa}$ is prescribed. The fluid density and dynamic viscosity are set to $\rho = 1 \times 10^3\,\mathrm{kg/m^3}$ and $\eta = 3 \times 10^{-4}\,\mathrm{Pas}$, respectively. The structural model is clamped at both ends and assumed stress free on the outside. The tube material has a density of $1.2 \times 10^3\,\mathrm{kg/m^3}$ and its behavior is described using the St. Venant-Kirchhoff model with a Young's modulus of $E = 0.3\,\mathrm{MPa}$ and a Poisson ratio of $\nu = 0.3$. The simulation with a time step size of $\Delta t = 0.1\,\mathrm{ms}$ lasts for $n^{\mathrm{ts}} = 180$ time steps. The Newmark-β method with $\beta = 0.49$ and $\gamma = 0.9$ is used to discretize the structural subproblem in time. Since it was observed in [51] that the length of the tube has a major impact on the number of coupling iterations, three settings are considered ($L = 5\,\mathrm{cm}$, $L = 10\,\mathrm{cm}$, $L = 20\,\mathrm{cm}$).

In Fig. 5.44, an impression of the solution is given in terms of the pressure distribution at time $t = 12\,\mathrm{ms}$. It can be observed, that the solution at this time is very similar for the tubes with lengths $L = 10\,\mathrm{cm}$ and $L = 20\,\mathrm{cm}$. For the tube with $L = 5\,\mathrm{cm}$, the pulse wave has already been reflected at the end such that a different pressure distribution prevails. As shown in Fig. 5.44 as well, the number of coupling iterations per time step increases significantly with increasing L. This is the case for the entire simulation

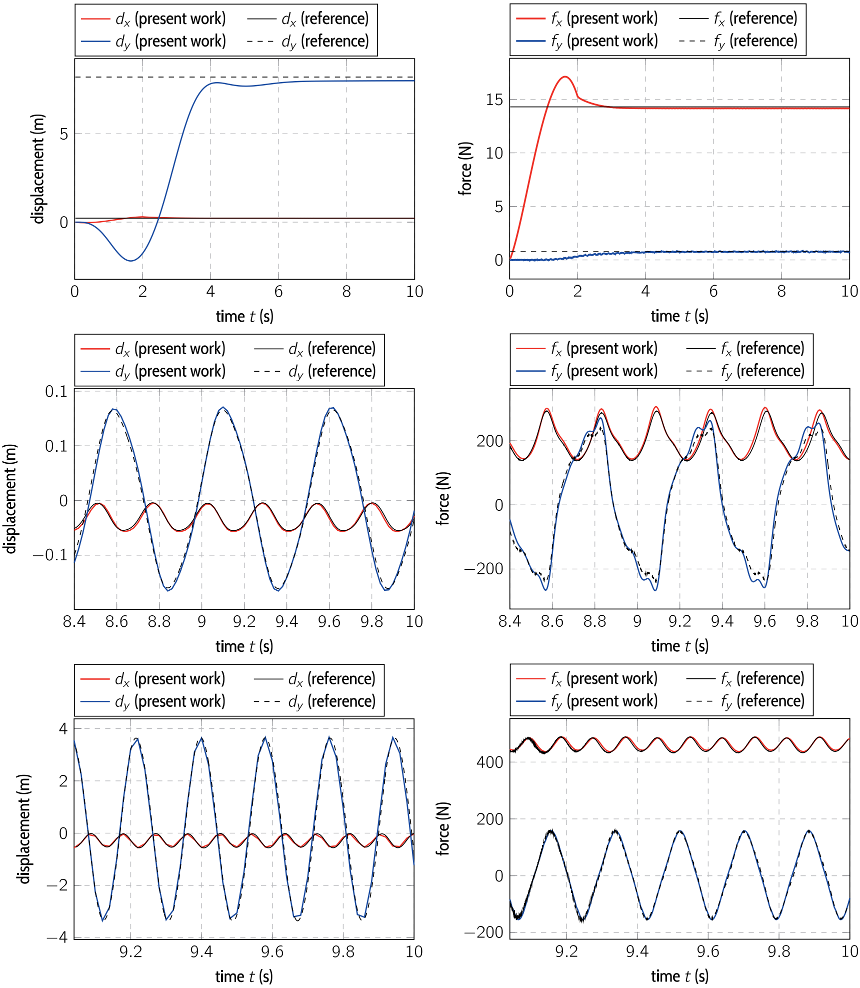


Figure 5.42: Displacement and force results for the considered scenarios FSI1, FSI2, FSI3 (from top to bottom).

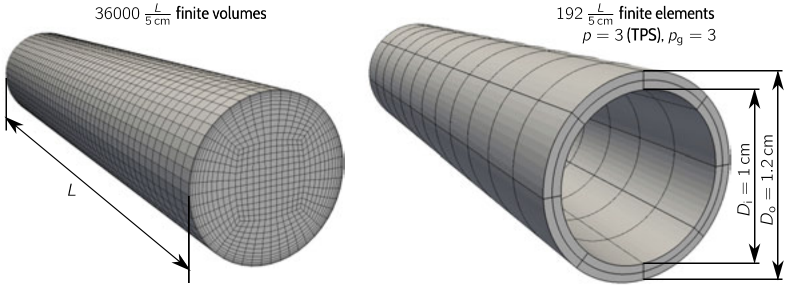

$36000 \frac{L}{5\,\text{cm}}$ finite volumes
L
$192 \frac{L}{5\,\text{cm}}$ finite elements
$p = 3$ (TPS), $p_g = 3$
$D_i = 1\,\text{cm}$
$D_o = 1.2\,\text{cm}$

Figure 5.43: Geometry and spatial discretization of the benchmark case *pulse wave in an elastic tube*.

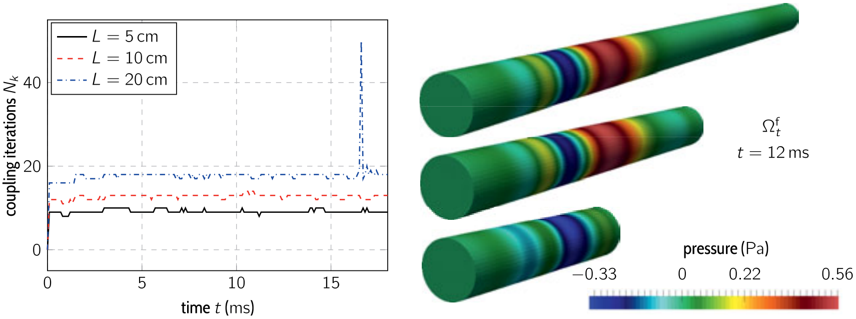


Figure 5.44: Number of coupling iterations (left) and pressure distribution at selected time instance (right) for the test case *pulse wave in flexible tube*.

course, not only once the solutions show reflection phenomena. A tube with length $L = 40\,\text{cm}$ was considered as well, however, the partitioned solution approach fails to solve the problem in this case as in [51]. In Fig. 5.45, a quantitative comparison between the different lengths is given in terms of the pressure distributions at selected time instances. It can be observed that the solution between all considered lengths is very similar as long as the pulse wave is not reflected.

While this benchmark case is based on a different boundary value problem than classical cardiovascular FSI studies, the observed limit of the partitioned solution approach in terms of L is expected to play a major role in such simulations as well. However, in view of the increased computational

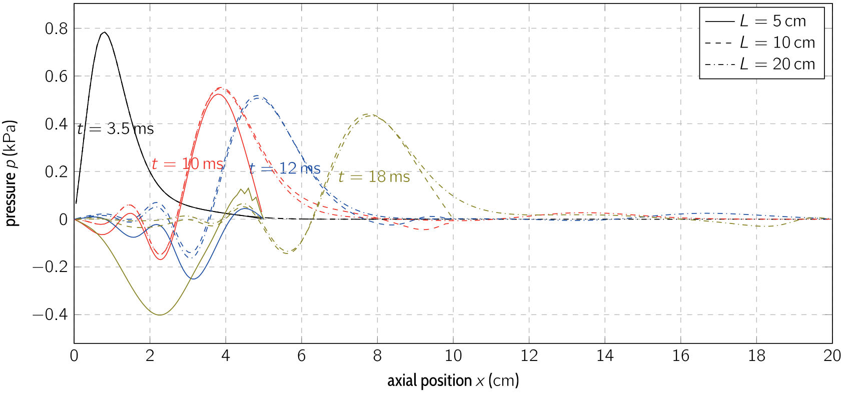


Figure 5.45: Pressure distribution at the center of the tube at selected time instances.

cost due to finer spatial discretizations and longer time intervals considered in the simulations, cardiovascular studies are typically concerned with moderate length to diameter ratios.

5.3 Arterial fluid-structure interaction

After the more general presentation of the potential of the partitioned solution approach and the individual field solvers, their ability to solve cardiovascular FSI problems is demonstrated in this section. The typical initial boundary value problem (IBVP) is introduced and solved for an idealized vessel with a simple geometry. Here, the only concern is the efficiency of the coupling strategy, while investigations of the hemodynamics are considered in the next chapter. The studies from [163] showed the severe increase in the number of coupling iterations for simulations with soft tissue material as opposed to simulations relying on rubber-like material laws. Here, only a soft-tissue material model is used and the focus is put on the partitioned coupling of the additional subproblems representing the reduced models, namely the elastic foundation and the windkessel model and its influence on the performance. In so far, the investigation follow the proposal in [164], where the same multifield problem was used to simulate the flow in an end-to-side anastomosis and the need for and investigation of the performance was emphasized. Additionally, the possibility to use adaptive time step

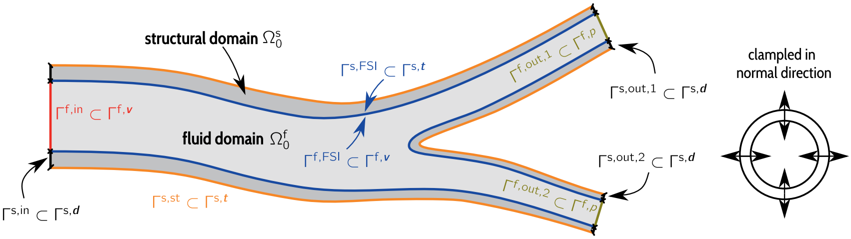

structural domain Ω_0^{s}
$\Gamma^{\mathrm{s,FSI}} \subset \Gamma^{\mathrm{s},\boldsymbol{t}}$
$\Gamma^{\mathrm{f,out,1}} \subset \Gamma^{\mathrm{f},p}$
clampled in
normal direction
$\Gamma^{\mathrm{s,out,1}} \subset \Gamma^{\mathrm{s},\boldsymbol{d}}$
$\Gamma^{\mathrm{f,in}} \subset \Gamma^{\mathrm{f},\boldsymbol{v}}$
fluid domain Ω_0^{f}
$\Gamma^{\mathrm{f,FSI}} \subset \Gamma^{\mathrm{f},\boldsymbol{v}}$
$\Gamma^{\mathrm{s,out,2}} \subset \Gamma^{\mathrm{s},\boldsymbol{d}}$
$\Gamma^{\mathrm{f,out,2}} \subset \Gamma^{\mathrm{f},p}$
$\Gamma^{\mathrm{s,st}} \subset \Gamma^{\mathrm{s},\boldsymbol{t}}$
$\Gamma^{\mathrm{s,in}} \subset \Gamma^{\mathrm{s},\boldsymbol{d}}$

Figure 5.46: Boundary value problem underlying a classical fully resolved cardiovascular FSI simulation.

sizes is considered here.

5.3.1 Initial boundary value problem

Typically, cardiovascular FSI studies are based on the classical IBVP for FSI introduced in Section 3.2. Considering only the systemic arterial circulation, the boundary conditions are likewise the same in most studies. As depicted in Fig. 5.46, the resulting problem represents a ducted flow with one inlet and possibly multiple outlets. The flow is driven by a prescribed velocity $\bar{\boldsymbol{v}}(\boldsymbol{x}, t)$ at the inlet, i.e. $\Gamma^{\mathrm{F,in}} \in \Gamma^{\mathrm{f},\boldsymbol{v}}$. At each outlet, the pressure $\bar{p}(t)$ is prescribed, i.e $\Gamma^{\mathrm{s,out},i} \in \Gamma^{\mathrm{f},p}$. While in some studies a constant pressure is assumed, which corresponds to a standard boundary condition, windkessel models are typically integrated into the fluid solver in order to determine a physiological $\bar{p}(t)$. Regarding the structural problem, the outer boundary $\Gamma^{\mathrm{s,ts}}$, which is in contact with the surrounding tissue, is typically assumed to be stress free. Alternatively, a nonzero traction $\bar{\boldsymbol{t}}$ is determined using, e.g. an elastic foundation as introduced in Section 3.3.5. In any case, $\Gamma^{\mathrm{s,ts}} \in \Gamma^{\mathrm{s},\boldsymbol{t}}$ holds, eventually with $\bar{\boldsymbol{t}} = \boldsymbol{0}$. On the structural boundaries at the inlet and the outlets, the displacement is usually directly prescribed or constraint in some way. As illustrated in Fig. 5.46, either all components are fixed or only displacements in the normal direction are prevented. In the latter case, the in-plane displacement must be constraint at least at some points in order to prevent rigid body motions.

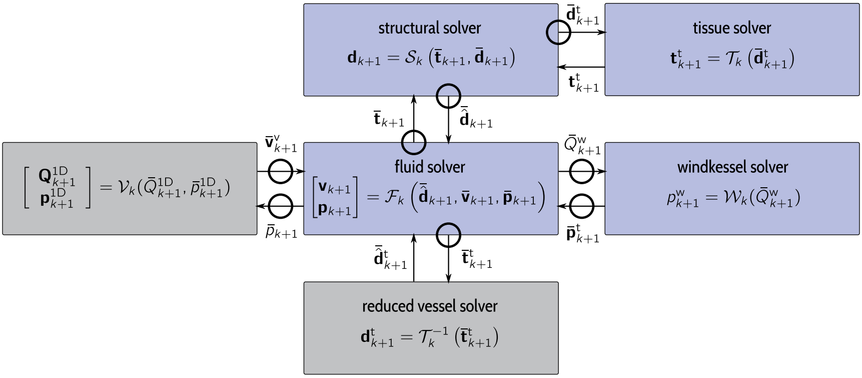

structural solver
$\mathbf{d}_{k+1} = \mathcal{S}_k\left(\bar{\mathbf{t}}_{k+1}, \bar{\mathbf{d}}_{k+1}\right)$
$\bar{\mathbf{d}}^{\mathrm{t}}_{k+1}$
tissue solver
$\mathbf{t}^{\mathrm{t}}_{k+1} = \mathcal{T}_k\left(\bar{\mathbf{d}}^{\mathrm{t}}_{k+1}\right)$
$\mathbf{t}^{\mathrm{t}}_{k+1}$
$\bar{\mathbf{t}}_{k+1}$
$\bar{\hat{\mathbf{d}}}_{k+1}$
$\bar{\mathbf{v}}^{\mathrm{v}}_{k+1}$
$\bar{Q}^{\mathrm{w}}_{k+1}$
$\begin{bmatrix}\mathbf{Q}^{\mathrm{1D}}_{k+1}\\ \mathbf{p}^{\mathrm{1D}}_{k+1}\end{bmatrix} = \mathcal{V}_k(\bar{Q}^{\mathrm{1D}}_{k+1}, \bar{p}^{\mathrm{1D}}_{k+1})$
fluid solver
$\begin{bmatrix}\mathbf{v}_{k+1}\\ \mathbf{p}_{k+1}\end{bmatrix} = \mathcal{F}_k\left(\bar{\hat{\mathbf{d}}}_{k+1}, \bar{\mathbf{v}}_{k+1}, \bar{\mathbf{p}}_{k+1}\right)$
windkessel solver
$p^{\mathrm{w}}_{k+1} = \mathcal{W}_k(\bar{Q}^{\mathrm{w}}_{k+1})$
$\bar{p}_{k+1}$
$\bar{\mathbf{p}}^{\mathrm{t}}_{k+1}$
$\bar{\hat{\mathbf{d}}}^{\mathrm{t}}_{k+1}$
$\bar{\mathbf{t}}^{\mathrm{t}}_{k+1}$
reduced vessel solver
$\mathbf{d}^{\mathrm{t}}_{k+1} = \mathcal{T}_k^{-1}\left(\bar{\mathbf{t}}^{\mathrm{t}}_{k+1}\right)$

Figure 5.47: Interdependency of the introduced subproblems in a cardiovascular FSI simulation. Circles indicate, whether a coupling quantity is computed by the respective field solver or the coupling manager.

5.3.2 Coupling algorithm

In the present work, the windkessel model and the elastic foundation are considered as individual subproblems within the partitioned solution approach. Further, the one-dimensional vessel model may be coupled to the fully resolved fluid subproblem and some segments of the vessel wall may be described using a reduced structural model instead of the fully resolved one. Figure 5.47 schematically shows the resulting interaction between all subproblems in this fully coupled case. Here, the investigations are restricted to the subproblem highlighted in blue, which yields the same simulation and modeling approach as the one followed in [164].

Recalling the coupling algorithms introduced in Section 4.3.1, any of them can readily be used to solve the typical IBVP described in the previous section. Facing the fact that the cardiovascular FSI problem may now consist of more than two subproblems, the classical coupling algorithms are no longer applicable. However, since each of the additional subproblems is only coupled to one of the fully resolved subproblems, only small modifications to the algorithms are necessary. As in [164], the following procedure underlies the modified staggered coupling algorithm.

1. Predict the displacement $\bar{\hat{\mathbf{d}}}^{(0)}_{k+1}$ and the flow $\bar{Q}^{(0)}_{k+1}$, set $i = 0$.

2. Solve the windkessel problem for the pressure ${p^{\mathrm{w}}}^{(i+1)}_{k+1}$.
3. Compute $\bar{\mathbf{p}}^{(i+1)}_{k+1}$ from ${p^{\mathrm{w}}}^{(i+1)}_{k+1}$.
4. Solve the fluid problem for the velocity $\mathbf{v}^{(i+1)}_{k+1}$ and the pressure $\mathbf{p}^{(i+1)}_{k+1}$.
5. Evaluate the traction $\bar{\mathbf{t}}^{(i+1)}_{k+1}$ and the flow ${\bar{Q}^{\mathrm{w}}}^{(i+1)}_{k+1}$.
6. Solve the tissue problem for ${\mathbf{t}^{\mathrm{t}}}^{(i+1)}_{k+1} = \bar{\mathbf{t}}^{(i+1)}_{k+1}$.
7. Solve the structural problem for $\mathbf{d}^{(i+1)}_{k+1}$.
8. Evaluate $\hat{\bar{\mathbf{d}}}^{(i+1)}_{k+1}$ and ${\bar{\mathbf{d}}^{\mathrm{t}}}^{(i+1)}_{k+1}$.
9. If all convergence criteria are fulfilled, increase k and go to Step 1. Otherwise, increase i and go to Step 2.

5.3.3 Test case

Starting with the classical IBVP introduced above, the simulation of a straight segment of an artery with a circular cross section is considered while integrating the windkessel model in the fluid solver and neglecting any influence of the surrounding tissue. Subsequently, the windkessel model is considered as a separate subproblem and finally, the tissue solver, respectively the elastic foundation, is activated. It is noted that such an on and off switching of certain parts of the overall model is one of the major advantages of the partitioned solution approach, which likewise allows to exchange certain parts by more sophisticated models.

A diameter of $D = 1\,\mathrm{cm}$ and a length of $L = 10\,\mathrm{cm}$ are chosen and the displacement at the ends of the structural model is only constraint in the normal (axial) direction. The HGO material models with the parameters from Table 5.1 is applied. The fiber directions are set as in the simulations in [82] and Section 5.1.1, such that two helically wound fiber families are obtained. The density of blood as well as that of the vessel wall are set to $\rho = 1 \times 10^3\,\mathrm{kg/m^3}$ and the Newtonian viscosity model with $\eta = 1 \times 10^{-4}\,\mathrm{Pas}$ is used. For the simulations involving the tissue solver, the inertia and damping parameters are set to zero ($m^{\mathrm{t}} = d^{\mathrm{t}} = 0$), such that a classical elastic foundation is obtained. The stiffness parameter is set to $c^{\mathrm{t}} = 10 \times 10^5\,\mathrm{Pa/m}$. While this is a comparably low value (see e.g. [164]) it ensures that the simulations with and without the tissue solver yield similar results, such that any influence on the number of coupling iterations can be attributed to the coupling between the structural subproblem and the tissue subproblem, while the influence of different flow situations induced by the consideration of the surrounding tissue can be expected to be negligible. The mesh topologies are taken to be the same as in Section 5.2.4 (see Fig. 5.43), however, coarser meshes are used (4500 finite volume on the fluid side and 24 finite elements with shape functions from the trunk space and a polynomial order of $p = p^{\mathrm{g}} = 4$ on the structural side).

Within the field of computational fluid dynamics, it is common practice to used adaptive time step sizes based on the so called *Courant number*. It describes the relation between the velocity and the size of finite volumes. In the one-dimensional case, it can be uniquely defined as

$$\mathrm{Co} = \frac{\Delta x}{h_x} = \frac{v_x \, \Delta t_k}{h_x}, \tag{5.54}$$

where h_x is the size of the finite volume and Δx is the distance a fluid particle travels in one time step. For classical pressure-corrector methods like the SIMPLE or the PISO algorithm (see Section 4.1.2) it should hold that $\mathrm{Co} < 1$ to ensure stability of the iterative solution process underlying these methods. As explained in [103], the PIMPLE algorithm allows for slightly higher Courant numbers due to the additional implicit loop. While for the larger simulations presented in the next chapter, this limit $\mathrm{Co}^{\max} > 1$ is tried to be reached as closely as possible in order to save computation time, here the classical constraint $\mathrm{Co}^{\max} = 1$ is used and the time step size is computed on this basis. Accordingly, the time step size of the coupled simulation is dictated by the fluid mechanics subproblem. As described in Section 4.4, it is received from the fluid solver and send to the structural solver using the coupling manager. Within the implicit coupling iterations, the time step size is held constant. More precisely, Δt_{k+1} is computed, based on the solution at time t_k, which was obtained using Δt_k. One could say, the time step adaptivity is realized in an explicit manner.

All simulations are performed for a time interval of 3 s, which for the prescribed flow at the inlet corresponds to three beats of the heart. For the simulations with a constant time step size, $\Delta t = 3\,\mathrm{ms}$ is chosen. In a preliminary study, this was found to be close to the maximum time step size, where a stable solution process is still possible ($\Delta t = 3.25\,\mathrm{ms}$ yields instabilities within the pressure-corrector method).

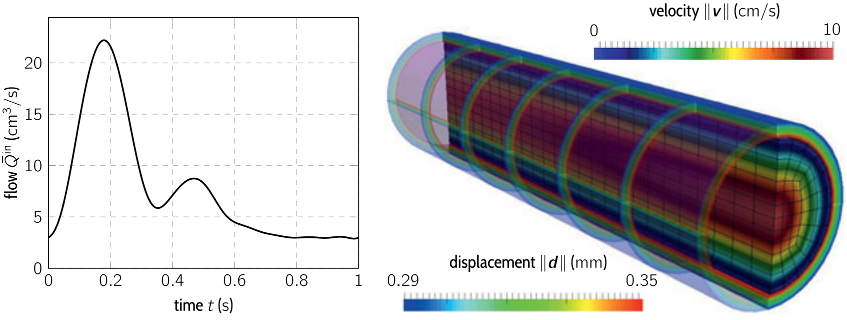


Figure 5.48: Influence of the consideration of the windkessel model as a separate subproblem on the number of coupling iterations.

5.3.4 Results

Figure 5.48 shows the distribution of the structural displacement and the fluid velocity to give an impression of the problem and the applied computational meshes. Further, the prescribed flow through the inlet is shown, which is very close to the one used in Section 5.1.2 for the large artery. In Figure 5.49 (left), the maximum displacement and the total number of coupling iterations is shown for the simulations with a constant time step size. As expected, the solution is found to be independent of, whether the windkessel model is integrated into the fluid solver (case $\mathcal{F}, \mathcal{S}$) or treated as a separate subproblem (case $\mathcal{F}, \mathcal{S}, \mathcal{W}$). Due to the low value for c^{t}, also the cases with the elastic foundation (cases $\mathcal{F}, \mathcal{S}, \mathcal{T}$ and $\mathcal{F}, \mathcal{S}, \mathcal{W}, \mathcal{T}$) show a very similar displacement behavior.

Regarding the total number of coupling iterations shown in Figure 5.49 (right), a significant increase is observed for the scenarios that treat the windkessel model and/or the foundation in a partitioned manner. While the a partitioned treatment of the windkessel model approximately doubles the number of coupling iterations, a partitioned treatment of the tissue solver seems to be less expensive. In general, the number of coupling iterations increases rapidly at the beginning of the simulations, where the system is driven from the homogeneous initial state to the range of physiological pressures $(80\,\mathrm{mmHg} to 120\,\mathrm{mmHg})$. In order to make the comparison more fair, the aspect that more residuals are checked for convergence in the fully coupled simulations (those including $\mathcal{W}$ and/or $\mathcal{T}$ as separate subproblems)

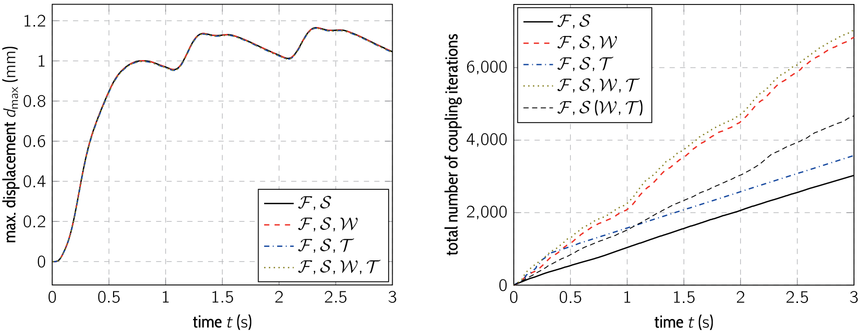


Figure 5.49: Influence of the consideration of the windkessel model as a separate subproblem on the number of coupling iterations in simulations with a constant time step size $\Delta t = 3\,\mathrm{ms}$.

has to be accounted for. As described in Section 4.3.4 all coupling quantities have to be checked, which in the fully coupled case includes the displacement and the traction at the structural models outside as well as the pressure and the flow at the outlet. Accordingly, an additional simulation without the tissue solver and with the windkessel model integrated into the fluid solver was conducted in combination with all convergence criteria prescribed in the fully coupled model. This case is denotes as $\mathcal{F}, \mathcal{S}(\mathcal{W}, \mathcal{T})$ in Figure 5.49 (right) and shows that the increase in the number of coupling iteration is not a result of the additional convergence criteria alone.

In Fig. 5.50, the results obtained for adaptive time step sizes are shown. In accordance with the dynamic behavior of the system, i.e. high flow velocities in the systolic phase and low flow velocities in the diastolic phase, the time step size varies severely over one cycle. While $\Delta t < 5\,\mathrm{ms}$ during peak flow, $\Delta t > 20\,\mathrm{ms}$ at the end of a cycle, where the flow is lowest. For the cases that involve the tissue solver $\mathcal{T}$, the adaptive time step control was constrained by an upper limit of $\Delta t^{\max} = 4.5\,\mathrm{ms}$. Only then the stability of the simulation could be guaranteed. Of course, this bound was only necessary at some phases of the simulation and an alternative approach involving a lower maximum Courant number Co would be possible as well. In any case, the adaptive time step size control leads to a remarkable increase in performance. For the classical two-field FSI problem, the total number of coupling iterations drops from 3026 to 1195, which correspond to a reduction of the computation effort to less than 40%. For the fully

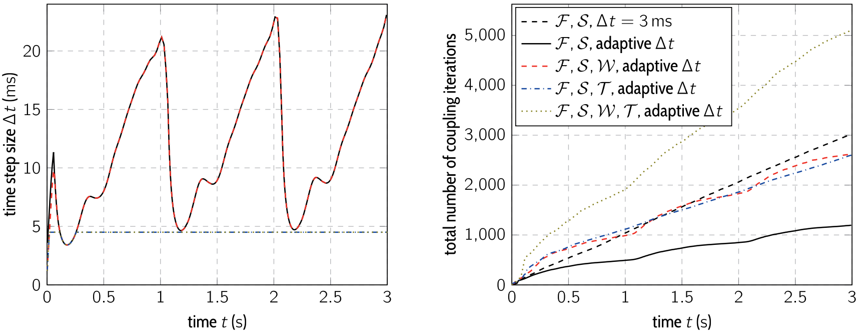


Figure 5.50: Time step size (left) and total number of coupling iterations (right) for adaptive time stepping.

coupled case $\mathcal{F}, \mathcal{S}, \mathcal{W}, \mathcal{T}$, the effort is still reduced to almost 70%, despite the comparably low $\Delta t^{\max}$.

6 Applications

In this chapter, the partitioned solution approach is applied to investigate the hemodynamics in selected arterial segments. In the first section, the influence of different parameters on the hemodynamics in segments of different shape is investigated. Secondly, the hemodynamics in different configurations of distal end-to-side anastomoses are considered. In the third section, an exemplary clinical study is performed in order to demonstrate the applicability of the approach for larger scale problems.

6.1 Hemodynamics in the scope of vessel geometry and material

In this section, the hemodynamics in three exemplary vessel segments shown in Fig. 6.1 are investigated. The main focus lies on assessing the influence of the geometry on the hemodynamics. Further, the effects of an elastic vessel wall as opposed to a rigid wall assumption are investigated. The introduced hemodynamic quantities, such as the wall shear stress, are evaluated and compared accross the segments under consideration. In addition to demonstrating the usage of the partitioned solution approach this provides references for a later investigation of the hemodynamics in bypass graft anastomoses.

As introduced in Fig. 6.1, the segments are referred to as *straight*, *curved*, and *bifurcation* in the following. The straight case is defined through its diameter $D = 1\,\text{cm}$ and its length $L = 10\,\text{cm}$. The curved case has the same cross-sectional geometry as the straight case, however, its centerline follows a 90° arc with a radius of $R = 5\,\text{mm}$. The bifurcation case is based on a patient-specific geometry. A voxel image is taken from [66] and segmented using the free software package *VTMK* (The Vascular Modeling Toolkit, see [139]). While the voxel image actually shows a cerebral aneurysm including the proximal and distal vessel network, only a small segment is extracted for the present study. Then, it is scaled in such a way that the parent vessel's diameter approximately matches that of the other cases. This is justified

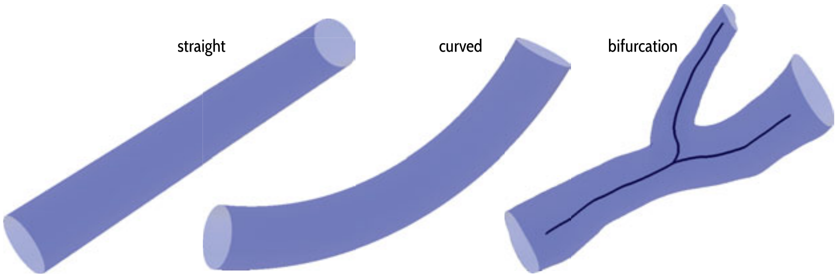


Figure 6.1: Lumen surfaces of the considered arterial segments.

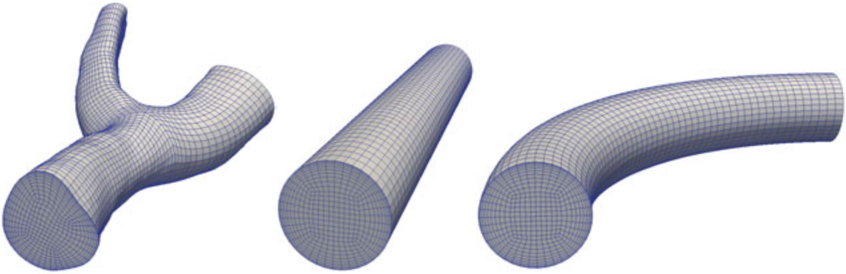


Figure 6.2: Computational fluid meshes for the considered arterial segments.

in view of the intention of the study, which is to compare the influence of different modeling aspects and to provide a reference in terms of clinical relevant hemodynamic quantities for later investigations. Summarizing, all geometries constitute fictive arterial segments. For a realistic reconstruction of geometries from medical imaging, many more aspects, for instance the blood pressure at the time the image was taken, have to be considered. Using the mesh generation technique introduced in Section 4.2, which is based on a centerline and a radius vector, structured hexahedral fluid meshes are generated for the considered geometries, see Fig. 6.2. While a radius vector with constant length ($\|\boldsymbol{r}\| = r = 5\,\mathrm{mm}$) is used for the cases *curved* and *straight*, the raytracing technique (see Fig. 4.5) is applied for case *bifurcation* case. Here, the triangulation of the lumen boundary obtained using *VMTK* from the segmented CT image constitutes the geometry definition.

At the proximal end of the segments, an inflow with the pulse shape and period reported in [226] is enforced by prescribing a temporally varying Poisseuille velocity profile. A three element windkessel model with resistances $R_1 = 2.55\,\mathrm{mmHg\,s/ml}$, $R_2 = 0.1\,\mathrm{mmHg\,s/ml}$ and a compliance of $C = 0.3\,\mathrm{ml/mmHg}$ is attached to the distal ends of the models. To arrive at a physiological pressure range, the flow is scaled by a factor of 5 before it is passed to the windkessel solvers in the straight and the curved case. For the bifurcation, it is scaled by a factor of 10, such that approximately half of the inflow passes through each outlet. The influence of the surrounding tissue is not considered in this study. All simulations are performed using an adaptive time step size based on a maximum Courant number of $\mathrm{Co}_{\mathrm{max}} = 3$. The initial step size is set to $\Delta t_0 = 1\,\mathrm{ms}$.

6.1.1 Decoupled simulations

At first, only the fluid subproblem is considered, or in other words, the vessel wall is assumed to be rigid. Figure 6.3 shows the oscillating shear index (OSI) for the decoupled fluid simulations. As expected, the maximum OSI is much larger in the bifurcation case than in the other cases. For the curved case, a region of increased OSI is observed at the inner wall (the region facing the center of the arc that defines the centerline). The maximum wall shear stress $\|\boldsymbol{\tau}^{\mathrm{w}}_{\mathrm{max}}\|$ is increased at the outer wall. For the straight case, a slight increase in the OSI is observed at the distal end. This is attributed to the Womersley velocity profile prevailing at this point as opposed to the proximal end, where a Poisseuille velocity profile is prescribed. As shown in Figures 3.9 and 5.15, the Womersley profiles are typically associated with a near-wall velocity that changes the axial direction. Accordingly, the wall shear stress (WSS) changes its direction, which is reflected by a non-zero OSI as explained in Section 3.3.7. The bifurcation case shows much higher extreme values for the OSI and for $\|\boldsymbol{\tau}^{\mathrm{w}}_{\mathrm{max}}\|$. However, it is emphasized that the considered flow situation is just one possibility within a large range of physiologically meaningful ones. In view of the same boundary conditions that were applied in all cases, the study still demonstrates the large effect that the shape of the vessel wall has on the hemodynamic quantities.

In addition to the decoupled fluid simulations, static structural simulations are performed. As for the fluid, structured hexahedral meshes are created using the centerline-radius technique. All models feature a constant wall thickness of $d = 2\,\mathrm{mm}$ was assumed. The Fung material model with

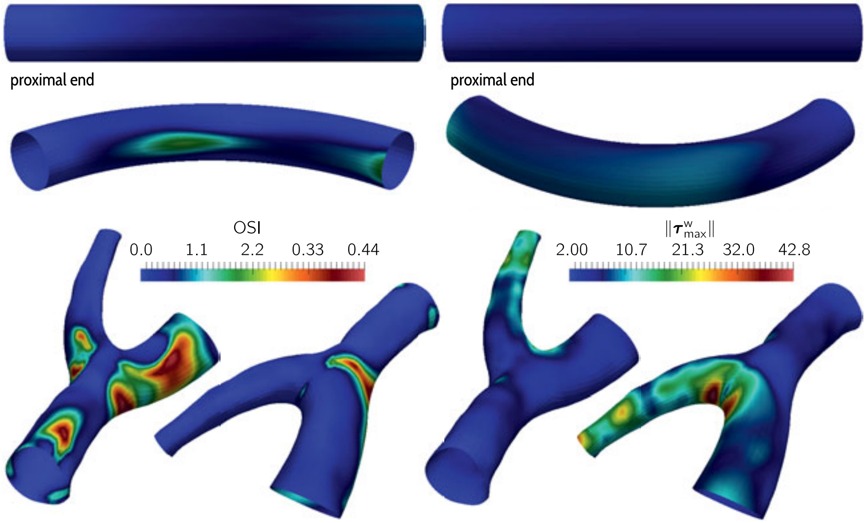


Figure 6.3: Maximum and mean wall shear stress for the decoupled simulations with a rigid wall assumption.

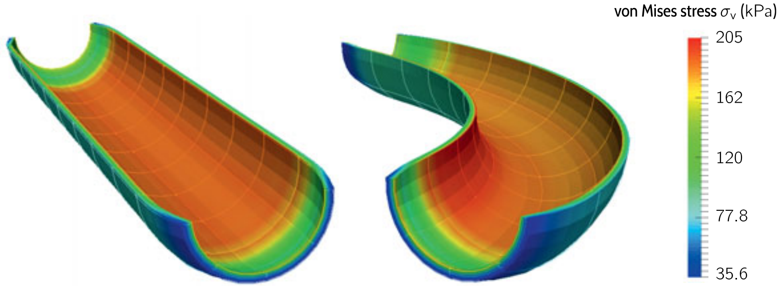


Figure 6.4: Stress distribution due to an internal pressure of $16\,\mathrm{kPa}$ for the cases *straight* and *curved*.

the parameters identified in Section 5.1.1 is used. The models are clamped at the proximal and distal ends. A spatially constant pressure, which is increased in 40 load steps to $16\,\mathrm{kPa}$ ($\approx 120\,\mathrm{mmHg}$), is prescribed at the inner wall.

Fig. 6.4 shows the stress distribution for the straight and for the curved case. It can be observed that the curvature has a small but clearly observable influence on the maximum stress, which in both cases prevails at the inside ($\sigma_\mathrm{v} = 188\,\mathrm{kPa}$ for the straight case and $\sigma_\mathrm{v} = 205\,\mathrm{kPa}$ for the curved case). The stress distribution for the bifurcation case is shown in Fig. 6.5. In accordance with the smooth geometry and the constant material properties, the stress field is found to be smooth as well, yet large variations are present. While in some regions the stress rises up to $255\,\mathrm{kPa}$, the stresses are even below $100\,\mathrm{kPa}$ in some regions of the inner surface. This can be partly attributed to a not perfectly constant wall thickness. However, in view of the findings from [162], where the effect of varying wall thicknesses was investigated in more detail, it can be assumed that the stress variations are mainly due to qualitative differences in the geometry. It is assumed that, with regard to the quality, the simulations reflect the true differences in the stress states caused by the different geometries. However, as explained in detail in Section 5.1.1, the simplified modeling approach followed here does not allow for a quantitative analysis. Since prestresses and the multi-layered structure of the artery are neglected, only the trend of increasing stresses from the straight case over the curved case to the bifurcation case can be captured. In the following, evaluations of the simulations are therefore restricted to hemodynamic quantities.

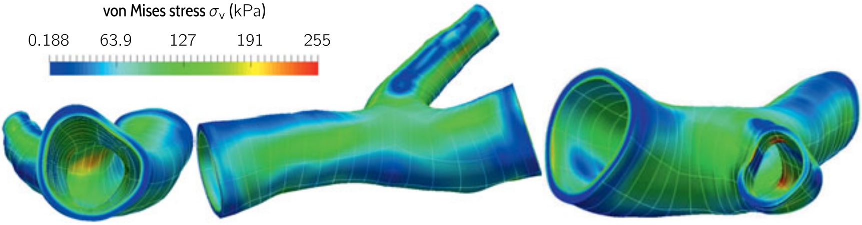


Figure 6.5: Stress distribution due to an internal pressure of $16\,\mathrm{kPa}$ for the case *bifurcation*.

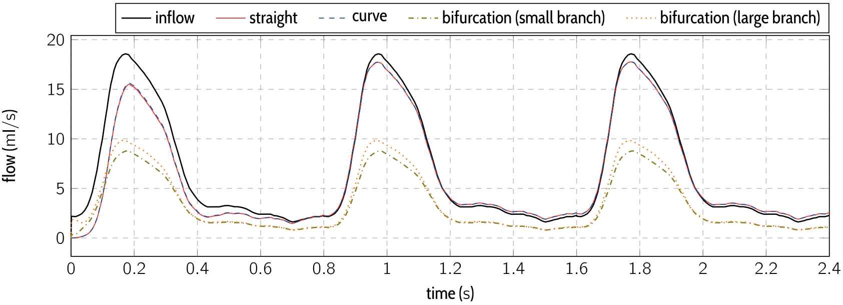


Figure 6.6: Proximal and distal flow in the differently shaped vessel segments.

6.1.2 Coupled simulations

Using the same modeling approach followed in Section 5.3, a fully resolved FSI simulation is performed for each segment. Figure 6.6 shows the flow through the inlet (equal for all models) and the flow through the individual outlets. Looking at the straight and curved case, it noted that the influence of the shape on the difference between inflow and outflow is negligible. Small deviations are only to be found during the starting phase of the simulation, where the system is driven from its homogeneous initial conditions to the pressurized state, due to the different volume changes for each model. For the bifurcation case, the flow splits almost evenly as expected from the equivalent windkessel models attached to each outlet. However, the larger pressure difference needed to drive the same flow through the smaller branch leads to clearly observable deviations especially in the systolic phase.

Fig. 6.7 shows the OSI distributions obtained from the coupled simula-

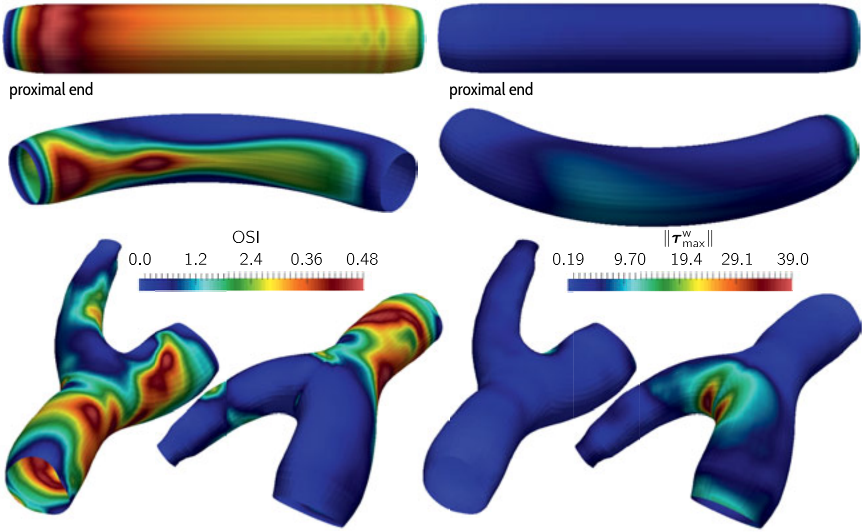


Figure 6.7: Oscillating shear index and maximum wall shear stress for the coupled simulations with elastic vessel walls.

tions. A significant difference compared to the simulations with a rigid wall assumption can be observed. However, they have to be attributed partly to the cigar-shaped flow domain, which results from the zero-displacement boundary condition at the vessel ends. For all shapes, the wall shear stresses shown in Fig. 6.7 are lower compared to those obtained for the rigid scenario. This was to be expected due to the increased diameters of the pressurized vessels. The comparison between the rigid and the elastic case is therefore restricted to illustrative purposes. For a fair comparison, the rigid model should match the geometry of the deformed vessel. Means to achieve this, which also include the possibility for a comparison of different material models, were elaborated and applied in [165, 163] and are not further considered here. This noted, Table 6.1 gives an overview on the extreme values for the wall-shear stress related quantities. Further, the maximum pressure at the proximal end is included as an indicator for the resistance of the respective segment.

Table 6.1: Comparison of the oscillating shear index and the extreme values for the wall shear stress.

	$\mathrm{OSI}_{\mathrm{max}}$	$\tau^{\mathrm{w}}_{\mathrm{max,min}}$ (Pa)	$\tau^{\mathrm{w}}_{\mathrm{max,max}}$ (Pa)
straight (rigid)	0.079	2.2	4.4
curved (rigid)	0.21	2.2	12
bifurcation (rigid)	0.44	2.0	44
straight (elastic)	0.48	0.67	2.6
curved (elastic)	0.48	0.64	8.9
bifurcation (elastic)	0.48	0.19	39

6.2 Hemodynamics in idealized end-to-side anastomoses

In this section, the focus is turned to bypass graft anastomoses. Investigating the hemodynamics in these anastomoses constitutes the target application for which the numerical methods in this work were developed. Accordingly, the main developments explained in Chapter 4 are used here in combination to arrive at an efficient and customizable modeling and simulation procedure.

The construction scheme for smooth surfaces is employed to obtain physiological meaningful geometries, which are based on a limited number of parameters. As explained in Section 4.2.1, a coarse quadrilateral control point mesh denotes the input to the scheme. Using the free modeling software *Blender* [25], such control point meshes are created for three different end-to-side anastomoses. The resulting G^1-continuous surfaces, which are taken as the lumen boundary here, are shown in Fig. 6.8. As mentioned in Chapter 2, it is assumed that cuffed bypass grafts improve the hemodynamics in the anastomosis region, which motivates the present study. In the model denoted as *standard bypass* in Fig. 6.8, the bypass graft has a constant diameter in the anastomosis region. In models denoted as *small cuff* and *large cuff*, the bypass graft increases in diameter towards the anastomosis region. Bypass grafts of these and similar types are either precuffed (see e.g. the one investigated in [207]) or prepared by the surgeon (see e.g. the one investigated in [171]). The present study is concerned with the influence of the cuff size on the hemodynamic quantities. Thanks to the developed partitioned simulation approach, the deformation of the vessel walls as well as the influence of the surrounding can be taken into account. Due to the fact that it is an exemplary study used to illustrate the potential of the simulation approach, the model parameters (fluid and vessel

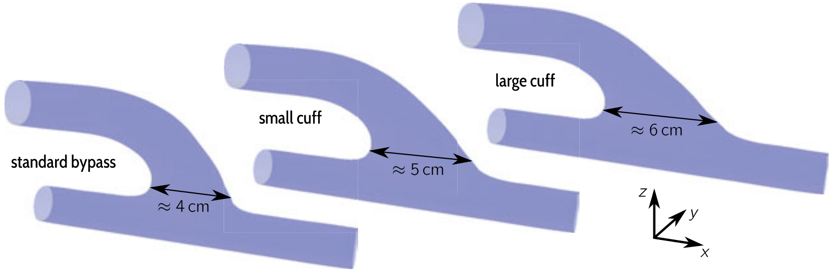


Figure 6.8: Lumen surfaces in anastomoses of standard and precuffed bypass grafts.

wall properties as well as proximal inflow and distal pressure) are kept constant. Nevertheless, it is emphasized that, in addition to the geometry, the influence of these parameters could be investigated as well.

From the G^1-continuous surfaces, tetrahedral fluid meshes are created using *VMTK* (The Vascular modeling toolkit, see [139] and Section 4.2). For the construction of the structural meshes, another G^1-continuous surface is created from the lumen surface by moving the control points in the outward normal direction according to a given wall thickness. It is assumed to be $h_\mathrm{a} = 3\,\mathrm{mm}$ for the artery and $h_\mathrm{b} = 1\,\mathrm{mm}$ for the graft. The radii of the graft and the artery are approximately $R_\mathrm{a} = 7.5\,\mathrm{mm}$ and $R_\mathrm{b} = 11\,\mathrm{mm}$. In the region of the suture line, the thicknesses are smoothly blended together. Drawing on the fact that the models are aligned with the global coordinate system as shown in Fig. 6.9, the local thickness can be described in terms of the z-coordinate as

$$h(z) = \begin{cases} h_\mathrm{a} & \text{if } z < z_\mathrm{min}, \\ \frac{z_\mathrm{max}-z}{z_\mathrm{max}-z_\mathrm{min}}\, h_\mathrm{a} + \frac{z-z_\mathrm{min}}{z_\mathrm{max}-z_\mathrm{min}}\, h_\mathrm{b} & \text{if } z_\mathrm{min} \le z \le z_\mathrm{max}, \\ h_\mathrm{b} & \text{if } z > z_\mathrm{max}. \end{cases} \tag{6.1}$$

Letting the artery's centerline be at $z = 0\,\mathrm{mm}$, the blending region extends from $z_\mathrm{min} = 10\,\mathrm{mm}$ to $z_\mathrm{max} = 15\,\mathrm{mm}$.

6.2.1 Simulation setup

The coupling procedure used for the test case in Section 5.3 is used here in the same way. The adaptive time step size is controlled by the fluid solver based on a maximum Courant number of 4. A time interval of 2.4 s

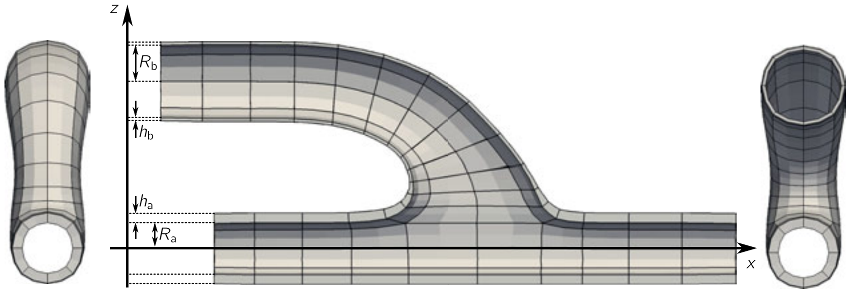


Figure 6.9: Structural mesh and geometry parameter for the anastomosis of the standard graft.

is simulated, starting with a time step size of $\Delta t_0 = 1\,\mathrm{ms}$. A flow pulse, according to the one reported in [226], is prescribed at the proximal end of the bypass graft. While the shape is left unchanged, the pulse is scaled such that physiological velocities arise. No flow is prescribed at the proximal end of the artery, assuming that the upstream part is completely occluded.

To account for the surrounding tissue, an elastic foundation solver is used. Its inertia and damping parameters are set to $m^{\mathrm{t}} = d^{\mathrm{t}} = 0$, ans the stiffness parameter is set to $c^{\mathrm{t}} = 10^6\mathrm{Pa/m}$. A three-element windkessel model is used to describe the influence of the distal arterial network. Its parameters are set to $R_1 = 2.55\,\mathrm{mmHg\,s/ml}$, $R_2 = 0.1\,\mathrm{mmHg\,s/ml}$, and $C = 0.3\,\mathrm{ml/mmHg}$, which yields a physiological pressure in the rage of approximately $80\,\mathrm{mmHg} to 120\,\mathrm{mmHg}$ if fed with the inflow pulse. In order to smoothly drive the system from the homogeneous initial state to this range of pressures, the coupling of the three-dimensional FSI simulation and the windkessel model is not enforced directly. Instead, the pressure obtained from the windkessel model is multiplied by a ramp function (see Section 3.3.4) for a duration of $T_r = 0.1\,\mathrm{s}$. Further, the coupling between the fluid and the structural solver within the three-dimensional FSI simulation is smoothly activated by multiplying the tractions by a ramp function with $T_r = 0.5\,\mathrm{s}$.

The material behavior of the artery is described using the modified Fung model with the parameters identified in Section 5.1.1. The bypass material is known to be much stiffer. As in [164], it is described using the Neo-Hooke model according to Eq. (3.98) with $E = 5 \times 10^7\,\mathrm{Pa}$ and $\nu = 0.49$.

The parameters are based on the compliance measurements conducted in the scope of [124].

In order to keep the simulation time at a reasonable level, comparably coarse spatial discretizations are chosen. For the fluid subproblem, a finite volume mesh with 345456 cells is used. On the structural side, 164 high-order finite elements with shape functions from the trunk space are used. A polynomial degree of $p = 5$ is chosen for the in-plane direction, while $p = 2$ in the thickness direction. The geometry is described using polynomials of order $p_{\mathrm{g}} = 4$. This leads to subparametric element formulations and a mesh that represents the geometry of the G^1-continuous surface exactly (up to round-off errors).

6.2.2 Results

In Fig. 6.10, the flow and pressure at the proximal end of the bypass (inflow) and the distal end of the artery (outflow) are shown for the course of the simulation. It is observed that, due to non optimal initial conditions in the windkessel model (compare Section 3.3.4), the pressure does not reach a periodic state in the considered time interval. Instead, a slow increase is observed. While in previous simulations all hemodynamic quantities were not evaluated before a periodic state was reached, this is not possible here. Instead, the last period, i.e. the interval $2.2\,\mathrm{s} < t \leq 3\,\mathrm{s}$, is taken as a basis for evaluation. Looking at the evaluation of the pressure, it is expected that the system is already very close to the desired periodic state. However, it is noted that the following results represent a state where the blood pressure increases slowly, which of course is still a situation that is found in reality. In reality, the blood pressure does in fact undergo continuous changes. In the simulation, the periodic state is only needed as a basis for comparisons and interpretations on a scenario that does not depend on the initial condition. For the present purpose of demonstrating the application of the solution approach, this is not regarded as important.

Figure 6.11 shows the distribution of the oscillating shear index (OSI) and the local maximum wall shear stress for the variant *small cuff*. The distribution is qualitatively very similar for the other variants. In all cases, the heel, toe, and bed can be identified as critical regions due to extreme maximum wall shear stresses, the high wall shear stress gradients, and the high values for the oscillating shear index. These findings was also observed experimentally (see [93]) and have been reproduced previously using the

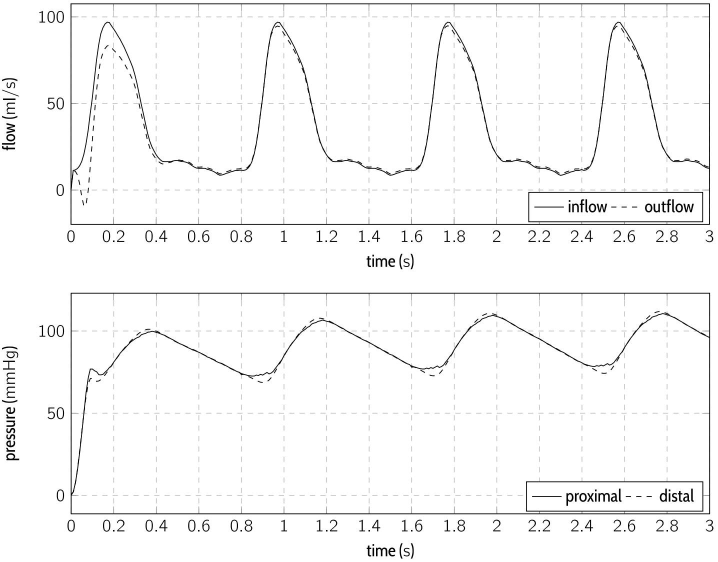


Figure 6.10: Flow and pressure for the variant *small cuff* over the course of the simulation.

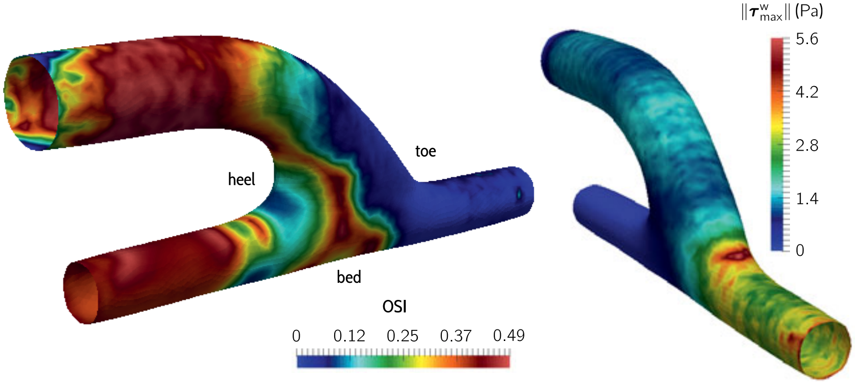


Figure 6.11: Oscillating shear index and maximum wall shear stress for variant B.

present simulation approach in [164]. However, different geometries and more realistic material models are considered here. Looking more closely at the distributions of the OSI and $\boldsymbol{\tau}^{\mathrm{w}}_{\max}$ shown in Fig. 6.12, a region with an increased OSI index is observed close to the anastomosis toe for the standard variant. In the cuffed variant, this undesired effect is almost completely eliminated. As elaborated in more detail in [162] and [164], this can be attributed to vortices in the flow, whose formation and collapsing is very sensitive to changes in the curvature of the vessel wall. The contour plots of the velocity on a slice through the middle of the anastomosis at specific times in the cardiac cycle shown in Fig. 6.13 illustrate this. The comparison between the FSI simulation and pure fluid simulations with a rigid wall assumption further shows that the elasticity of the vessel has an influence on this effect. Considering the maximum wall shear stress $\|\boldsymbol{\tau}^{\mathrm{w}}_{\max}\|$ in Fig. 6.12, the rigid-wall assumption leads to increased values. From the view on the anastomosis region selected for the presentation of the distribution, the significant changes in shape due to the elasticity of the vessel wall can be observed as well.

Table 6.2 shows a comparison of the extreme values for the mentioned hemodynamic quantities. In accordance with the qualitative similarity, the quantitative differences between the models are comparably small as well. Still, the expected improvement due to the cuff is clearly observable. While the *small cuff* does not affect the maximum OSI, the *large cuff* at least re-

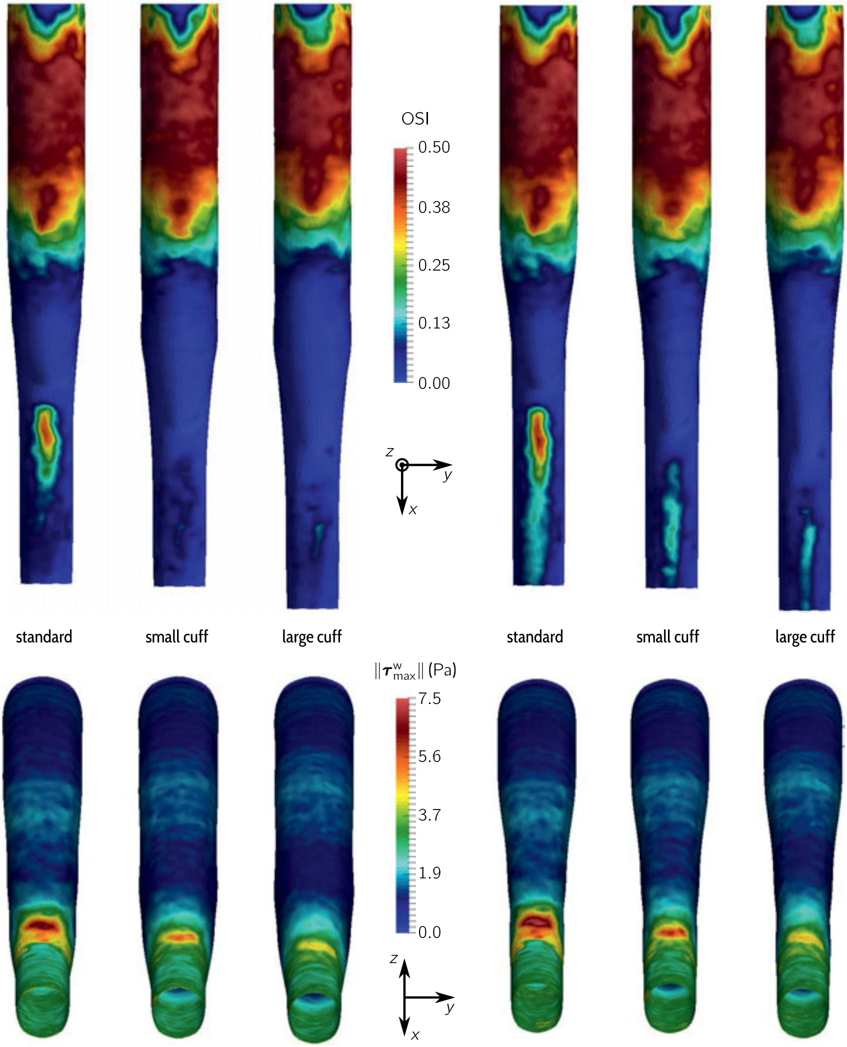


Figure 6.12: Comparison of the oscillating shear index (top) and the maximum wall shear stress (bottom) for the FSI simulations with an elastic vessel wall (left) and the fluid simulations with a rigid wall assumption (right).

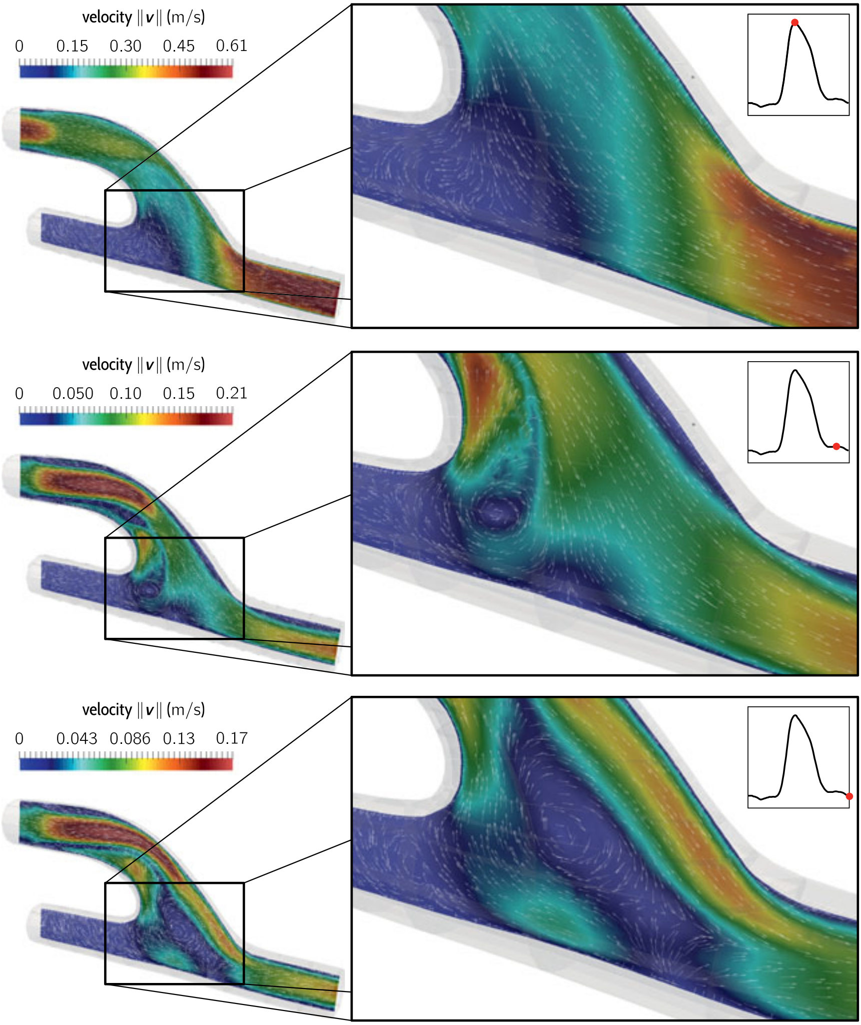


Figure 6.13: Velocity at a slice through the anastomosis with a small cuff at time $t = 2.57\,\mathrm{s}$, $t = 2.90\,\mathrm{s}$, and $t = 3.0\,\mathrm{s}$ (from top to bottom).

Table 6.2: Comparison of the maximum oscillating shear index and the extreme values for the wall shear stress.

	$\mathrm{OSI}_{\max}$ (–)	$\lVert\boldsymbol{\tau}^{\mathrm{w}}_{\max,\min}\rVert$ (Pa)	$\lVert\boldsymbol{\tau}^{\mathrm{w}}_{\max,\max}\rVert$ (Pa)
standard bypass	0.49	9.16×10^{-5}	6.84
small cuff	0.49	7.86×10^{-5}	5.59
large cuff	0.48	1.29×10^{-4}	4.83

duces the maximum OSI by 2% as compared to the *standard bypass*. Clearly, the cuffs have more influence on the extreme values for the maximum wall shear stress $\boldsymbol{\tau}^{\mathrm{w}}_{\max}$. This is observed for the largest maximum wall shear stress during one cardiac cycle $\boldsymbol{\tau}^{\mathrm{w}}_{\max,\max}$ as well as for the smallest maximum wall shear stress $\boldsymbol{\tau}^{\mathrm{w}}_{\max,\min}$. A mathematical definition for these quantities is provided in Equations (3.189) and (3.190). Compared to the *standard bypass*, the maximum value is reduced by 29% and the minimum value is increased by 41% for the *large cuff*, both of which constitute an improvement in view of the discussion in Section 3.3.7. However, it should be noted that – for a reliable evaluation of the quality of different anastomoses – it would be necessary to evaluate more flow scenarios, such as carefully designed rest and exercise conditions as well as different distributions of the total inflow to the two possible inlets.

6.3 Hemodynamics in a patient specific anastomosis

In this section, the partitioned solution approach is exemplarily applied for a patient-specific setting. The case has been studied previously in [226], but assuming the vessel wall to be rigid. Many of the parameters, e.g. those regarding the mechanical properties of the vessel wall as well as those of the patient's blood, are unknown. Thus, typical values from the literature have to be assumed. Therefore, the simulation results cannot be interpreted as a reasonable approximation of the real hemodynamics in the circulatory system of the respective patient. The study case can still give an impression of the use of numerical simulations in surgical planning and ensure the applicability of the developed methods for investigations of this type.

6.3.1 Study case

The considered patient is a 67-year-old female with occlusive peripheral arterial disease. Fig. 6.14 shows the reconstructed lumen surface, which is available through the *Cardiovascular Online Repository* [73]. Smaller arteries that branch off the common and external iliac artery as well as the femoral artery were removed (c.f. Figure 2.2). While a strong progression of the disease was observed throughout the larger arteries of the lower extremities, the region with the most severe symptoms is clearly identified as the bifurcation, where the left common iliac artery branches into the internal and external iliac arteries. Proximal to the bifurcation, an aneurysm with a dissection (separation of the arterial layers) was found. Distal to the bifurcation in the external iliac artery, a severe stenosis had developed. More details on the patient's medical history and the progression of the disease are provided in [226]. The studies presented therein are concerned with the introduction of a cardiovascular simulation software, which is designed to assist vascular surgeons in planning revascularization procedures. The software allows to modify the preoperatively acquired medical images – magnetic resonance imaging (MRI) in the present case – and study different alternatives for the planned intervention. For the study case at hand, the postoperative model shown in Figure 6.14 was modified after the patient underwent the surgery, such that it closely reflects the geometry obtained from postoperatively acquired MRI data. While the patient was symptom-free at that time, the postoperative images revealed that the bypassed external iliac arteries had occluded completely, which is attributed to the low flow prevailing therein after the surgery. The present study reproduces this flow situation as shown below.

In succession to the creation of pre- and postoperative models, a surgical planning procedure with the aid of the software introduced in [226] then includes a one-dimensional and a three-dimensional analysis. While the one-dimensional results are obtained on a standard workstation within minutes, the three-dimensional analyses take several hours. Both simulations provide insight into the prevailing flow and pressure distributions. However, the one-dimensional model does not include any effects induced by geometric features such as curvatures and non-circular cross sections, while the three-dimensional analysis does not include any effects induced by the elasticity of the vessel walls. The partitioned solution approach provides an ideal basis for more sophisticated modeling and simulation approaches. Of course, extensive software engineering would have to be performed from the development of the solution approach to obtain a software that is applicable in clinical practice, ensuring a proper user interaction and acceptable

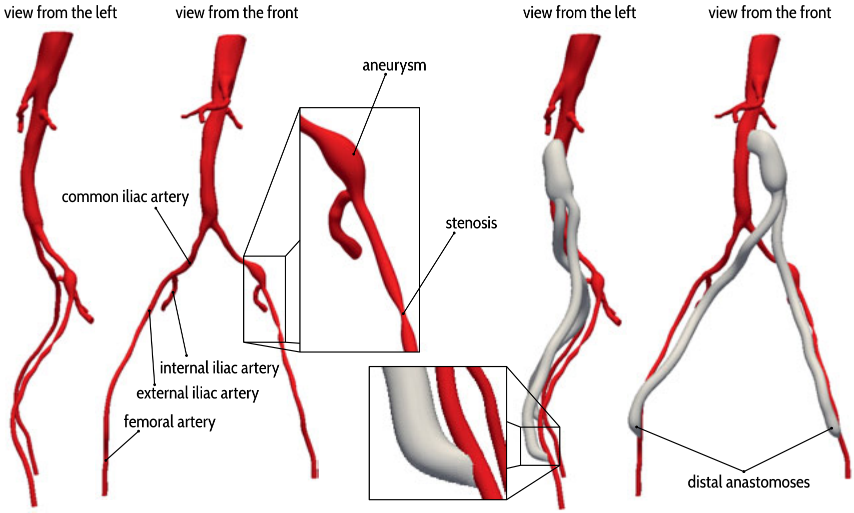


Figure 6.14: Preoperative (left) and idealized postoperative (right) vessel geometry. The lumen surfaces were reconstructed from segmented MRI data as detailed in [226].

computation times[1]. Further, additional techniques would be needed in order to automatize parts of the modeling process. Nevertheless, a fictive surgical planning procedure based on the fully coupled FSI problem can be conducted as explained in the following.

6.3.2 Modeling and simulation approach

With the modeling techniques developed around the partitioned solution approach, the fluid-structure interaction segment of the arterial tree shown in Fig. 6.14 can be investigated in detail. Since the geometry was obtained from medical images that were recorded when the artery was under pressure, the simulation is not performed for the full pressure range, i.e. from $80\,\mathrm{mmHg} to 12\,\mathrm{mmHg}$, but only for the *pulse pressure*, i.e. from $0\,\mathrm{mmHg} to 40\,\mathrm{mmHg}$. To solve the problem in acceptable time, only the anastomosis region is modeled in a fully resolved three-dimensional way. Further, only the anastomosis on the left side (see Fig. 6.14) is considered. The vessels proximal to this anastomosis as well as the entire right side are

[1] With the solvers applied in this work, the one-dimensional analyses can be done within minutes as well but the fully coupled FSI simulation takes several days.

modeled using the one-dimensional vessel model introduced in Section 3.3.3. In order to simulate the blood flow in the network, the vessel solver based on the Taylor-Galerkin method described in Section 4.1.3 is augmented by the boundary conditions introduced in [179], allowing for bifurcating and joining vessels.

The proximal ends of the three-dimensional model are coupled to the one-dimensional model by exchanging the flow and pressure. To this end, the flow obtained from the vessel solver is distributed to the inlet boundary of the fluid model in terms of a Womersely velocity profile. The pressure at this boundary is averaged and prescribed in the vessel solver. At the distal end, a two-element windkessel model is used to account for the influence of the downstream vessel network. It delivers the pressure at the outlet boundary of the fluid domain in terms of the flow, which is evaluated from the velocity by integration over the boundary. The fluid and structural solver exchange the surface traction and the displacement as usual. Additionally, the structural solver is coupled to a tissue solver that delivers tractions in terms of displacements.

A sub-stepping scheme is used, because the one-dimensional vessel solver demands for a much smaller time step size than the three-dimensional fluid solver. For every macro time step performed by the other solvers, the vessel solver performs 20 micro time steps. This allows for an implicit coupling despite the explicit nature of the Taylor-Galerkin method. Given the known average pressure at the proximal ends of the anastomosis region collected in a vector $\bar{\mathbf{p}}_k^{1\mathrm{D}}$ and a corresponding trial pressure at the current time step $\bar{\mathbf{p}}_{k+1}^{1\mathrm{D},(j)}$, where k still counts the macro time steps, the values at the intermediate steps are computed using a linear interpolation. At any time $t_k < t < t_{k+1}$, these intermediate values are then computed by

$$\bar{\mathbf{p}}^{1\mathrm{D}\,(j)}(t) = \frac{t_{k+1} - t}{\Delta t_k}\,\bar{\mathbf{p}}_k^{1\mathrm{D}} + \frac{t - t_k}{\Delta t_k}\,\bar{\mathbf{p}}_{k+1}^{1\mathrm{D}\,(j)}, \tag{6.2}$$

where Δt_k denotes the macro time step size.

The final coupling procedure is shown in Algorithm 6. For the present application, it was found to be preferable over algorithms with an additional inner loop to equilibrate pairs of fields separately as done in [68]. To avoid confusions, the evaluation and solver operators needed to achieve the above coupling are combined, such that each combined operator delivers every output that is needed from the contained solver. The substepping procedure is hidden in the same way, arriving at the operators $\tilde{\mathcal{F}}$, $\tilde{\mathcal{S}}$, $\tilde{\mathcal{V}}$, $\tilde{\mathcal{T}}$, and $\tilde{\mathcal{W}}$ for the fluid, structural, vessel, tissue, and windkessel solver, respectively. It is noted that due to an implementation detail in the actual code that was used for the present case, the pressure and velocity fields were predicted based on a constant extrapolation. The displacement was predicted using the tangent extrapolation techniques as usual (see 4.3.3).

Algorithm 6 Coupling algorithm for multifield cardiovascular FSI simulations.

```
for $k = 0 \dots n_{\mathrm{ts}} - 1$ do                                                      ▷ Start of the time loop
    $\bar{\hat{\mathbf{d}}}^{(0)}_{k+1} = \mathcal{P}_k \circ \bar{\hat{\mathbf{d}}}_k$                ▷ Predict lumen displacements
    $\bar{Q}^{\mathrm{w}\,(0)}_{k+1} = \mathcal{P}_k \circ \bar{Q}^{\mathrm{w}}_k$                       ▷ Predict distal flow
    $\bar{\mathbf{p}}^{\mathrm{1D}\,(0)}_{k+1} = \mathcal{P}_k \circ \bar{\mathbf{p}}^{\mathrm{1D}}_k$         ▷ Predict proximal pressures
    for $j = 0 \dots n_{\max} - 1$ do                                                   ▷ Start of the coupling loop
        $\bar{\mathbf{p}}^{(j+1)}_{k+1} = \tilde{\mathcal{W}}_k \left(\bar{Q}^{\mathrm{w},(j)}\right)$             ▷ Call the windkessel solver
        $\bar{\mathbf{v}}^{(j+1)}_{k+1} = \tilde{\mathcal{V}}_k \left(\mathbf{p}^{\mathrm{1D},(j)}_{k+1}\right)$     ▷ Call the vessel solver
        $\begin{bmatrix} \mathbf{t}^{(j+1)}_{k+1} \\ \mathbf{p}^{\mathrm{1D}\,(j+1)}_{k+1} \\ Q^{\mathrm{w}\,(j+1)}_{k+1} \end{bmatrix} = \tilde{\mathcal{F}}_k \left(\bar{\hat{\mathbf{d}}}^{(j+1)}_{k+1}, \bar{\mathbf{v}}^{(j+1)}_{k+1}, \bar{\mathbf{p}}^{(j+1)}_{k+1}\right)$     ▷ Call the fluid solver
        $\mathbf{t}^{\mathrm{t}\,(j+1)}_{k+1} = \tilde{\mathcal{T}}_k \left(\bar{\mathbf{d}}^{\mathrm{t}\,(j+1)}_{k+1}\right)$     ▷ Call the tissue solver
        $\begin{bmatrix} \tilde{\bar{\hat{\mathbf{d}}}}^{(j+1)}_{k+1} \\ \tilde{\bar{\mathbf{d}}}^{\mathrm{t}\,(j+1)}_{k+1} \end{bmatrix} = \tilde{\mathcal{S}}_k \left(\mathbf{t}^{\mathrm{t}\,(j+1)}_{k+1}, \mathbf{t}^{(j)}_{k+1}\right)$     ▷ Call structural solver
        if CONVERGED(...) then                                                   ▷ Check convergence of all fields
            $\bar{\hat{\mathbf{d}}}_{k+1} = \tilde{\bar{\hat{\mathbf{d}}}}^{(j+1)}_{k+1}$
            $\bar{Q}^{\mathrm{w}}_{k+1} = \bar{Q}^{w\,(j+1)}_{k+1}$
            $\bar{\mathbf{p}}^{\mathrm{1D}}_k = \bar{\mathbf{p}}^{\mathrm{1D}\,(j+1)}_k$
            break                                                              ▷ Proceed with the next time step
        else
            $\bar{\hat{\mathbf{d}}}^{(j+1)}_{k+1} = \mathcal{A}^{(j)}_k \left(\tilde{\bar{\hat{\mathbf{d}}}}^{(j+1)}_{k+1}\right)$     ▷ Accelerate convergence
        end if
    end for                                                                     ▷ End of the coupling loop
end for                                                                         ▷ End of the time loop
```

One-dimensional network model

The one-dimensional network model is created by combining the data from [226] and [179]. For each vessel in the network, its length L, cross sectional area A_0 and stiffness coefficient β have to be prescribed. The geometric data of the bypass is set according to the three-dimensional model from [226], which is shown in Fig. 6.14. The vessel parameters are taken from [179], which also served as a basis for the implementation of the vessel

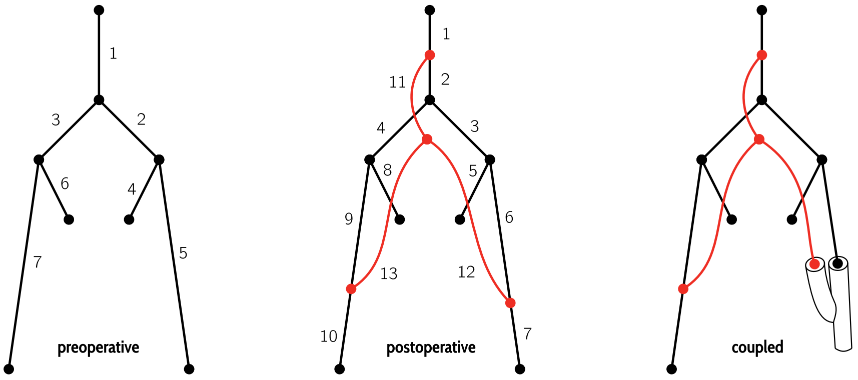


Figure 6.15: One-dimensional network models and vessel indices for the uncoupled (left and middle) and coupled (right) simulations.

Table 6.3: Properties of the vessels in the one-dimensional network model (preoperative).

	vessel	L (cm)	A_0 (cm^2)	β (kg s^{-2} m^{-2})	parents	daughters
1	abdominal aorta	5	0.578	399	-	1, 2
2	left common iliac artery	5.8	0.328	649	1	4, 5
3	right common iliac artery	5.9	0.328	649	1	6, 7
4	left internal iliac artery	5	0.181	3134	2	-
5	left external iliac artery	14.4	0.252	1493	2	-
6	right internal iliac artery	5.1	0.181	3134	3	-
7	right external iliac artery	14.5	0.252	1493	3	-

solver. Therein, the study of a full body network consisting of 55 arteries is presented. Here, only those arteries present in the model from [226] are considered, i.e. a section of the abdominal aorta and the common, internal and external iliac arteries. The femoral arteries are not considered explicitly but taken to be the continuation of the external iliac arteries. The resulting data for the vessel networks before (preoperative) and after (postoperative) the bypass surgery are given in Tables 6.3, 6.4. To account for the stenosis in the left external iliac artery, its diameter was decreased by a factor of ten. The topologies of the networks are illustrated in Figure 6.15. It further shows the model used in combination with the fully resolved FSI simulation of the anastomosis region, where the distal end of the left external iliac artery and the left bypass leg is coupled to the three-dimensional model.

Table 6.4: Properties of the vessels in the one-dimensional network model (postoperative).

	vessel	L (cm)	A_0 (cm^2)	β (kg s^{-2} cm^{-2})	parents	daughters
1	abdominal aorta I	2.5	0.578	399	-	$2, 11$
2	abdominal aorta II	2.5	0.578	399	-	$3, 4$
3	left common iliac	5.8	0.328	649	2	$5, 6$
4	right common iliac	5.9	0.328	649	2	$8, 9$
5	left internal iliac	5	0.181	3134	3	-
6	left external iliac I	4.4	0.252	1493	3	7
7	left external iliac II	10	0.252	1493	$6, 11$	-
8	right internal iliac	5.1	0.181	3134	4	-
9	right external iliac I	4.5	0.252	1493	4	10
10	right external iliac II	10	0.252	1493	$9, 12$	-
11	common bypass	4.5	0.252	5000	1	$12, 13$
12	left bypass	13	0.252	5000	11	7
13	right bypass	13	0.252	5000	11	10

At all distal ends, two-element windkessel models are used to prescribe a physiological pressure. The resistances and compliances are set to $R = 7.52\,\mathrm{s\,mmHg/ml}$ and $C = 0.013\,\mathrm{ml/mmHg}$ for all models. In view of the exemplary character of this study, this simplification is regarded as justified here as long as physiologically meaningful flow rates are obtained. A more accurate strategy based on additional flow measurements is addressed below along with the description of the results.

Three-dimensional anastomosis model

The three-dimensional meshes for the fully resolved model of the anastomosis region are created similarly to those for the idealized anastomoses considered in the previous section. This time, however, the surface representation from [226] is used to construct the coarse quadrilateral mesh, which forms the basis for the subsequent mesh generation process illustrated in Fig. 6.16. Using the modeling software *Blender* (see [25]), the manually constructed quadrilateral mesh (see Fig. 6.16, A) is refined and smoothed using a Catmull-Clark subdivision scheme (see [37]). The resulting quadrilateral mesh (see Fig. 6.16, B) is again modified manually in order to closely match the reference surface from [226]. After computing the smooth lumen surface (see Fig. 6.16, C) using the construction scheme introduced in Section 4.2, a second surface (representing the vessel's outer boundary) is obtained by extruding the lumen surface (see Fig. 6.16, D). A structural mesh (see Fig.6.16, E) is then created based on these boundaries.

Each of the hexahedral elements corresponds to one of the quadrilaterals. The polynomial degree for the geometry description is chosen as $p_\mathrm{G} = 5$. A tetrahedral fluid mesh (see Fig. 6.16, F) is constructed from a triangulation of the smoothed lumen surface using *VMTK*. It is noted that only the construction of the quadrilateral meshes A and B requires the input from a user. The subsequent mesh generation process can be done in a fully automatic way.

As before, the stiffness of the elastic foundation underlying the tissue solver is set to $c^\mathrm{t} = 1 \times 10^6\,\mathrm{Pa/m}$, while the inertia and damping parameters are set to zero. An initial time step size of $\Delta t_0 = 1\,\mathrm{ms}$ is chosen. In subsequent time steps the fluid solver dictates a Δt_k based on a maximum Courant number of 10. The convergence of each coupled field is checked based on a relative tolerance of $e^\mathrm{rel} = 10^{-5}$ and an absolute tolerance of $e^\mathrm{abs} = 10^{-12}$, see Section 4.3.4 for details. Mass-proportional structural damping with a parameter of $\zeta_0 = 0.01$ is applied. As investigated in [163], this stabilizes the simulation while altering the results only to a negligible extent.

6.3.3 Results – one-dimensional analysis

In view of the application of cardiovascular FSI simulation in the clinic, an appropriate starting point would be to get an impression of the overall flow situation. For this purpose, the one-dimensional model alone can already provide some insight. It yields the flow Q^1D and the pressure p^1D (averaged over the cross section) at every point within the considered vessels. Figure 6.17 shows the pressure distribution within the network that represents the preoperative state at time of maximum flow through the inlet (the proximal end of the abdominal aorta). The decreased flow in the left external iliac artery due to the narrowing in this vessel is clearly visible. Of course, the flow separation at bifurcations within the network strongly depends on the windkessel models. While, for the sake of simplicity, the same parameters are chosen here at each distal end, flow measurements at these points can be used to tune the parameters such that the simulation results match the measurement. In [226], a simple iterative procedure is proposed for this purpose. Here, the physiological consistency of the result, i.e. a decreased flow distal to the narrowed vessel, is regarded as sufficient. The flow distribution in Fig. 6.17 further proves the physical consistency of the treatment of bifurcations, since the flow at the distal ends of all vessels

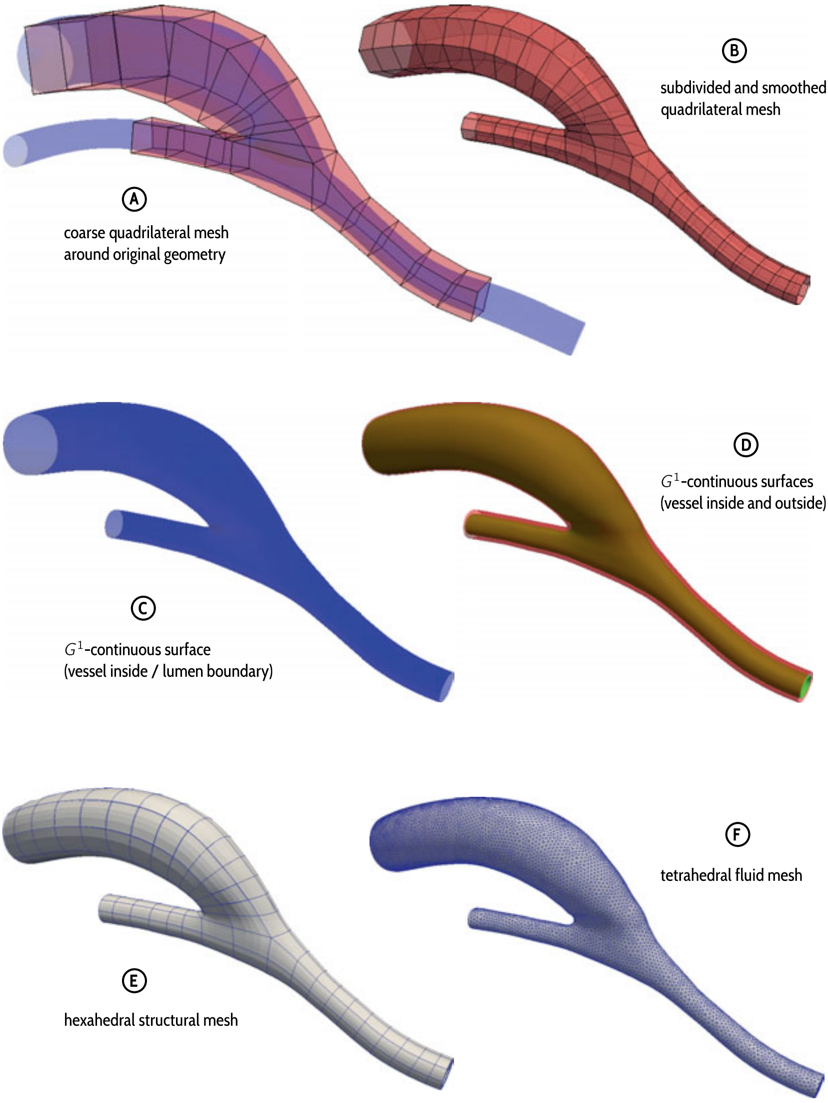

B
subdivided and smoothed quadrilateral mesh
A
coarse quadrilateral mesh around original geometry
D
G^1-continuous surfaces (vessel inside and outside)
C
G^1-continuous surface (vessel inside / lumen boundary)
F
tetrahedral fluid mesh
E
hexahedral structural mesh

Figure 6.16: Modeling steps for the patient specific exemplary case.

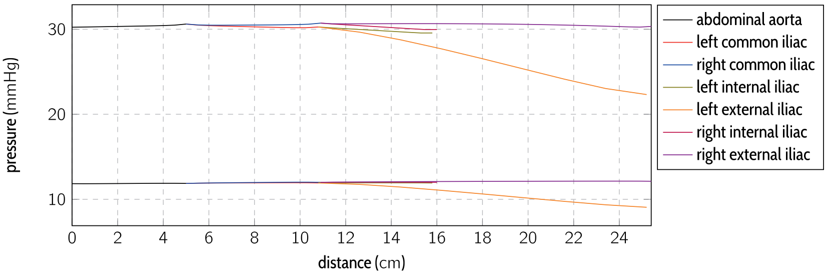


Figure 6.17: Pressure distribution in the preoperative vessel network at the time of minimum and maximum inflow.

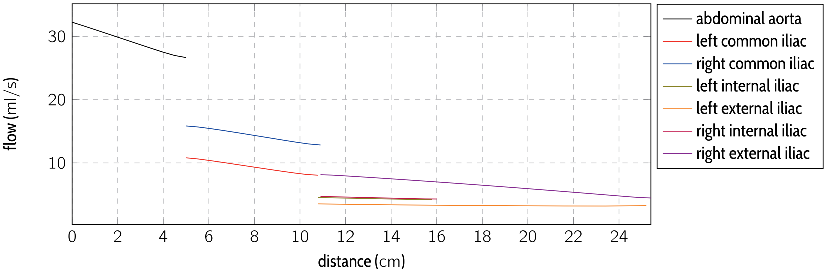


Figure 6.18: Flow distribution in the preoperative vessel network at the time of maximum inflow.

matches the sum of the flow at the proximal end of their respective daughter vessels. The pressure distributions shown in Fig. 6.18 are likewise consistent in a physiological and physical way. Here, the pressure is continuous at bifurcations, and the pressure drop in the narrowed vessel is much larger than that in the corresponding healthy artery. Of course, in addition to looking at the pressure and flow distributions in the network at a certain time, the evolution of these quantities at a certain point can be investigated as well. Fig. 6.19 shows the flow at the proximal end and the pressure at the distal end of all arteries. Again, the decreased flow in the left external iliac artery due to the stenosis is clearly visible.

The same investigations can be made for the postoperative network model. Figure 6.20 shows the flow distribution for this situation. For com-

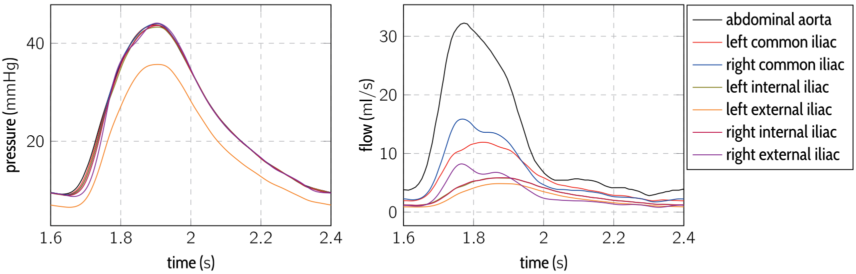


Figure 6.19: Flow at the proximal ends and pressure at the distal ends over the last period considered in the one-dimensional simulation.

pleteness, the pressure distribution is given in Fig. A.9.4. It is found to be nearly constant throughout the network – as can be expected for healthy vessels without narrowing. This one-dimensional analysis already indicates that the flow in the left external iliac artery normalizes due to the bypass. Further, it can be noted that only a very low flow prevails in the common iliac arteries in the postoperative model. This can also be seen in the evolution of the flow and pressure over time, see Fig. 6.21. In [226], it is in fact reported that a follow-up measurement of the flow rates showed that the common iliac arteries were almost completely occluded. The low flow rates after the surgery may serve as an explanation for this – and the one-dimensional analysis could serve as a basis to develop a surgical plan to counteract this issue. While it is not intended here to suggest an alternative treatment for this specific case, the findings might generally suggest that a smaller graft-diameter or a further upstream anastomosis location could be preferable. Of course, any of such decisions will always be constrained by anatomic features such as limited operative accesses or the progression of the disease. Summarizing the investigations of the one-dimensional analysis, flow and pressure distribution within the arterial network can already provide insight to the advantages and drawbacks of certain surgical plans. The achievement of the main goal of a bypass surgery, i.e. to restore the blood supply in regions distal to stenoses, can be investigated in a flexible and computationally inexpensive manner. However, the actual flow patterns in the anastomosis region, which affect the progression of intimal hyperplasia cannot be obtained from the cross-sectional averages provided by the one-dimensional analysis. This can only be achieved through the

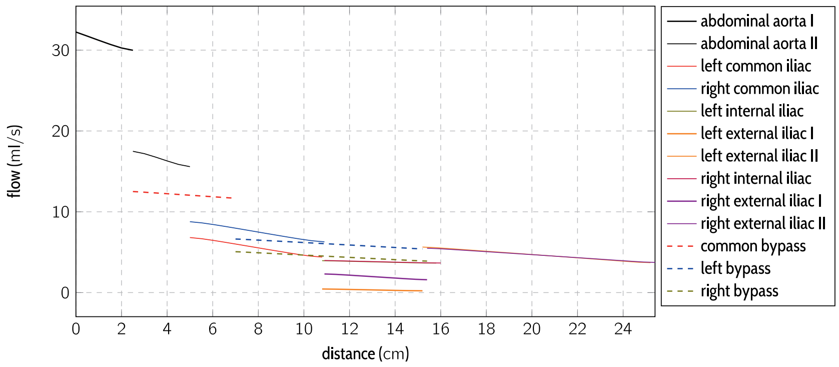


Figure 6.20: Flow distribution in the postoperative vessel network at the time of maximum inflow.

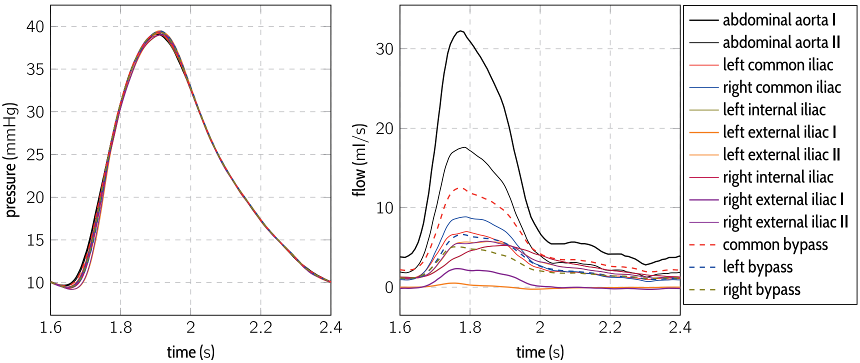


Figure 6.21: Flow at the proximal ends and pressure at the distal ends over the last period considered in the one-dimensional simulation.

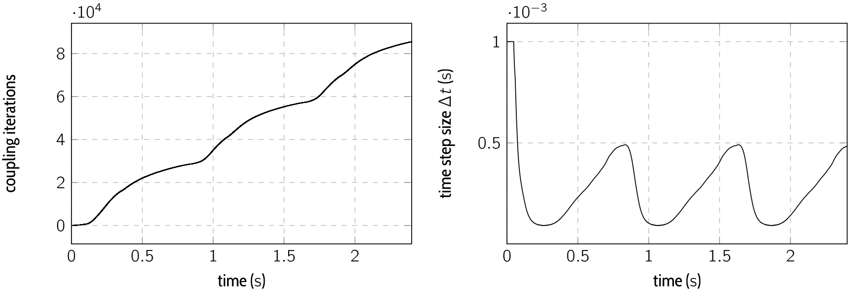


Figure 6.22: Total number of coupling iterations and time step size for the fully coupled anastomosis simulation.

fully-resolved three-dimensional FSI simulation.

6.3.4 Results – three-dimensional analysis

Before turning to the hemodynamic-related results of the coupled FSI simulation, a short note is made on the coupling iterations and the time step size. As shown in Fig. 6.22, the adaptive time stepping reduces the computational efforts drastically. A constant time step size would have to be smaller than $\Delta t_{\min} = 9.2 \times 10^{-5}\,\mathrm{s}$, such that 26087 instead of 13304 time steps had to be performed. It is further noted that the number of coupling iterations stays at a moderate level throughout the simulation. The maximum number of coupling iterations in a time step is found to be 13, while the mean is 6.8. The additional field does not seem to have a large influence on the convergence rate of the underlying modified fixed-point iteration. However, a direct comparison with the previous simulations is questionable, since they were performed for the real pressure range, while only the pulse pressure is used here. The findings from [163] indicate that this will lead to a decrease in the number of coupling iterations, such that two effects on the efficiency could be mixed here.

Hemodynamics

As before, the three-dimensional simulations deliver structural displacements as well as fluid velocities and pressures, from which the hemodynamic quantities of interest can be computed. Since the fully coupled multifield model considered here includes a one-dimensional description of the anastomosis' surrounding, it is also possible to study the flow and pressure distribution in the respective parts of the arterial system. Figure 6.23 gives an overview about the results that could be used for surgical planning purposes. In clinical practice, several alternative revascularization procedures could be compared based on such an overview.

While the overview in Fig. 6.23 includes the most important characteristics about the flow situation in the anastomosis region, like the maximum oscillating shear index and the extreme values for the wall shear stress, the corresponding distributions can also be visualized as before. Figures 6.24 and 6.25 show these distributions for the fully coupled simulation. As explained above, it reflects the flow situation directly after surgery, where a significant flow through the external iliac artery was still present. According to the measurements six month after surgery, reported in [226], the external iliac arteries occluded afterwards, which results in a very different flow situation – similar to the one considered in Section 6.2 for the idealized anastomoses. Figures 6.24 and 6.25 therefore show a comparison of the fully coupled simulation and a second FSI simulation without flow through the external iliac artery. This second simulation was not obtained using the fully coupled approach but based on the classical approach followed in the previous studies, i.e. without a one-dimensional vessel solver. The flow at the proximal end of the bypass is directly prescribed based on the follow-up measurements from [226]. The streamlines at the time of maximum inflow in Fig. 6.26 further illustrate the difference in the flow situations. The vortex prevailing in the scenario without inflow from the artery that was also observed for the idealized anastomoses (see Fig. 6.13) disappears in the scenario, where that flow is present.

Structural results

From the structural solver, displacements and stresses are obtained during the simulation. Figure 6.27 shows the anastomosis in the undeformed state Ω_0 and the deformed state Ω_t at the time of maximum pressure (at $t = 1.87\,\mathrm{s}$). Keeping in mind that the simulation was only performed for the pulse pressure range, the deformations only reflect the displacements of the artery within one cardiac cycle. Correspondingly, they are much smaller than those determined in the previous FSI simulations. For a better illustration and comparison of the two states, three cross sections with notable

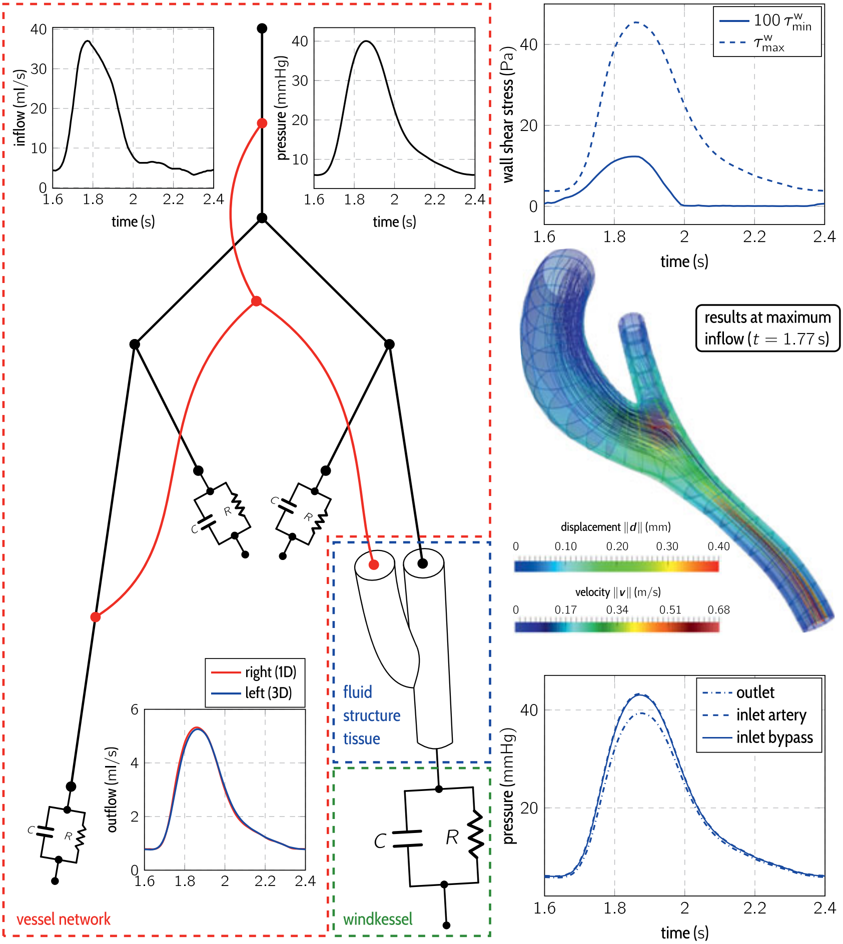


Figure 6.23: Flow and pressure over one cardiac cycle at selected points of the coupled postoperative model.

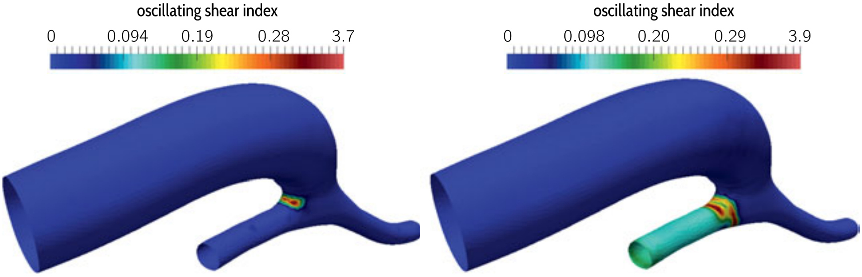


Figure 6.24: Distribution of the oscillating shear index obtained from the fully coupled simulation (left) and the classical FSI simulation without inflow through the host artery.

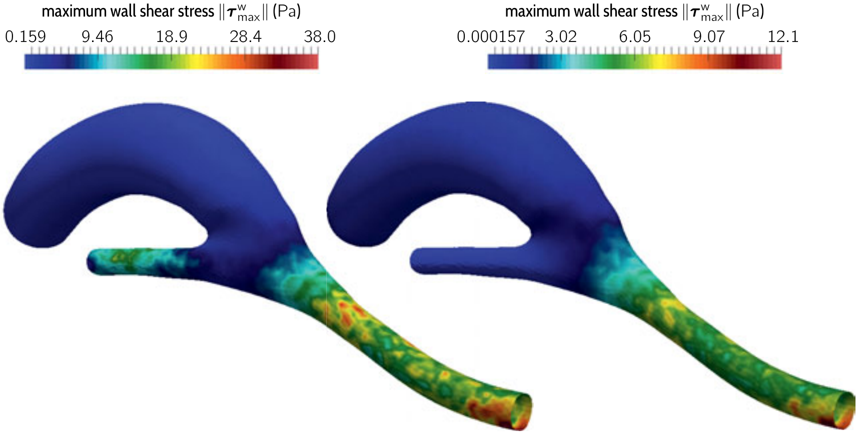

maximum wall shear stress $\|\boldsymbol{\tau}^{\mathrm{w}}_{\mathrm{max}}\|$ (Pa)
0.159
9.46
18.9
28.4
38.0
maximum wall shear stress $\|\boldsymbol{\tau}^{\mathrm{w}}_{\mathrm{max}}\|$ (Pa)
0.000157
3.02
6.05
9.07
12.1

Figure 6.25: Distribution of the maximum wall shear stress obtained from the fully coupled simulation (left) and the classical FSI simulation without inflow through the host artery.

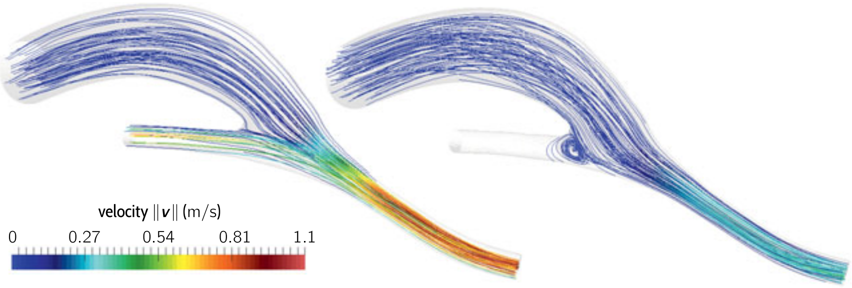


Figure 6.26: Streamlines computed from the velocity fields at the time of maximum inflow from the fully coupled simulation (left) and the classical FSI simulation without inflow through the host artery.

shape changes were highlighted in Fig. 6.27. It should be noted that, generally, there are no loads acting on the structure in the undeformed state. However, regarding the present study, the vessels were certainly under pressure at the time the MRI image was taken. Therefore, the stresses acting on the artery and the graft can only be investigated using additional simulations. One may be tempted to conduct a simulation using the actual physiological pressure level – then, however, the appropriate undeformed state would be unknown. A standard approach would be to solve an inverse problem based on an assumed load that prevailed at the time the geometry was captured. Usually, the diastolic pressure is assumed in such cases, see e.g. [191]. Still, for a truly realistic structural model, such computations would have to include the prestresses prevailing in arteries when they are unloaded.

6.3.5 Discussion

The above thoughts on the structural mechanics model highlight the great challenges in developing a realistic structural model for patient specific arteries and vessels in the scope of cardiovascular FSI simulations. If material parameters are used that were identified based on an unloaded undeformed state, simulations covering the full pressure constituted the better choice. As shown in the previous simulations, the partitioned solution approach is applicable also in cases which were found to be more prone to instabilities than simulations based on the pulse pressure only, see [163]. While the

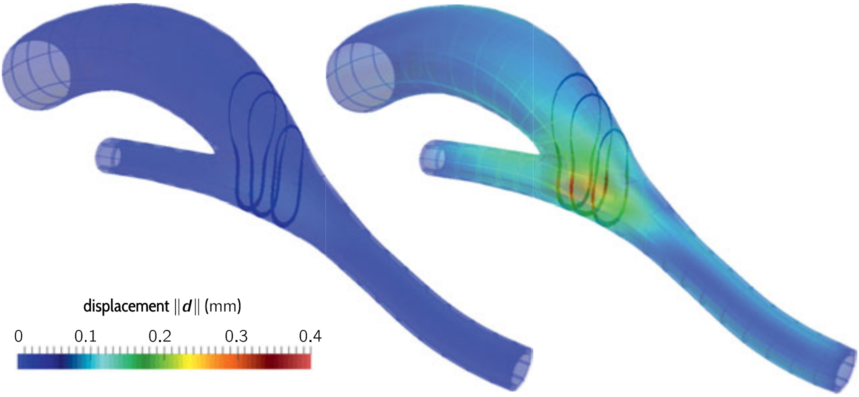


Figure 6.27: Left: Undeformed configuration of the structural domain. Right: Deformed configuration at the time of maximum pressure.

parameters for the structural and the reduced models have to be refined, the exemplary study of the patient-specific anastomosis still illustrates the potential of the partitioned solution approach as a tool for the analysis of clinical relevant problems. It is regarded as certain that improved models like this would also allow to solve the resulting FSI problems using the developed numerical methods. With increasing computational resources, corresponding simulations may be conducted on a regular basis in future clinics. It is only due to the partitioned nature of the approach and the resulting flexibility of combining different models and to use specialized solvers that corresponding high fidelity models can be used as a basis. Concluding, it was shown that the individual models from Chapter 3 and the corresponding numerical methods for their solution from Chapter 4 allow for a very detailed description of the hemodynamics while keeping the computational cost at an acceptable level.

7 Summary and Outlook

Driven by the goal to predict the local hemodynamics in segments of the arterial system, several modeling and simulation methods were developed and investigated in detail in this work. They enable three-dimensional simulations of the fluid-structure interaction (FSI) in such regions that allow for in-depth investigations of clinically relevant issues, e.g. regarding the influence of the vessel geometry and material on hemodynamic quantities such as the wall shear stress. Reduced models of the surrounding are used to provide realistic boundary conditions to the fully resolved FSI problem. The resulting multifield problem is solved using a partitioned solution approach. The main achievement is the development of numerical methods that are needed to realize such a partitioned simulation in which specialized field solvers applying different numerical methods are coupled.

Partitioned cardiovascular FSI simulations are prone to instabilities due to the large added mass effect resulting from the similar density of the fluid and structure. Accordingly, a major part of this work is concerned with the investigation of convergence acceleration methods that ensure stability and efficiency despite this nature of the problem. Together with predictors and suitable convergence criteria, they were used in combination with staggered, parallel, and mixed coupling algorithms. Using a simple test problem, the most efficient method was found to be a quasi-Newton least squares method with filtering. The findings from this study were confirmed in simulations of more complex benchmark problems that at the same time serve as a validation of the developed coupling software. Due to the non-matching discretizations of the fluid and the structural subproblem, loads and displacements have to be mapped from one computational mesh to another. Based on a detailed investigation of different interpolation methods suitable for this task, the recommendation to use mesh-based interpolation methods is given. If not available, radial basis functions constitute an acceptable alternative – with the drawback that the coupling interface must be smooth or split into parts that are smooth.

Drawing on the experiences from the two-field FSI problem, a partitioned solution approach for the augmented problem including the reduced models was introduced. In addition to the fluid and the structural solver, a one-dimensional blood flow solver and several types of windkessel models were included in the simulation approach. A coupling algorithm in which all fields are solved subsequently was found preferable over approaches where pairs of fields are coupled in an inner iteration. The simulation method was designed such that an adaptive time step size can be dictated by one of the participating solvers (here the fluid solver), which significantly decreases the computation time. Further, a sub-stepping scheme can be applied for solvers that demand for a particularly small time step size (here the one-dimensional vessel solver). Thanks to the partitioned nature of the approach, all solvers can be exchanged and improved independently from each other. For an efficient solution of the structural mechanics subproblem, high-order finite elements were used as the preferred spatial discretization method. They proved to be well-suited especially for the simulation of arteries and bypass grafts which are nearly incompressible, highly anisotropic, and possibly thin-walled – all of which may induce locking if standard finite elements are used. In order to create computational meshes with curved elements, which are needed to utilize the advantages of the high-order method, a surface construction scheme was introduced. It is based on a set of minimizers and constraints that can be mixed freely to achieve a certain smoothness of the surface. A set like this turned out to be a very useful basis for the construction of idealized and patient-specific geometries. Being based on a piecewise polynomial description, the surface geometries can be directly used to obtain iso- or subparametric element formulations.

To prove the applicability of the methods, several exemplary studies were conducted. The general possibility to investigate the influence of vessel shape and material on the hemodynamics was demonstrated for idealized healthy arteries of different shape and distal end-to-side anastomoses of differently cuffed bypass grafts. The evaluations of hemodynamic quantities like the oscillating shear index and the maximum wall shear stress over one cardiac cycle correspond well to explanations and theories about critical regions in anastomoses. The influence of the cuff size was found to be unexpectedly low for the considered scenario. However, the studies were of exemplary type and their main purpose can be seen in demonstrating the developed numerical methods. In the same spirit, a patient-specific simulation of the distal anastomosis of an aorto-femoral bypass was performed. The partitioned solution approach was found to be very suitable for the coupling of the fully-resolved model of the anastomosis region based on medical images and the reduced order models of the surrounding vessel network.

Concluding, it can be said the modeling and simulation approach performed well in addressing clinically relevant questions related to the hemodynamics in the arterial system. However, many of the developed methods are generally applicable for the solution of strongly coupled multifield problems. Convergence acceleration methods and predictors may even be applied to improve the convergence of vector sequences out of the scope of coupled problems. Further, the methods for geometry and mesh generation as well as the interpolation methods can be useful in many different applications.

Outlook on future works

The limitations elaborated at the end of Chapter 6 revealed that a refinement of the structural model is one of the biggest challenges on the way to an improved simulation approach. While state-of-the-art material models are employed in this work, the accurate modeling of prestresses in arteries remains an unsolved problem. The fact that the geometry for patient-specific models is usually obtained from medical images taken when the artery is in a loaded state further complicates this issue. However, there are ongoing efforts to refine the modeling approaches and numerical methods to solve the inverse problem arising due to this situation, see e.g. [191]. Another major challenge that arises especially in patient specific simulations is the identification of material parameters. Non-invasive methods to obtain information about the mechanical behavior of arteries are just starting to be developed. In fact, [54] is the only reference found in the scope of this thesis that addresses this question. While limited to the small deformation regime, future works could further develop the method introduced therein and combine it with the means to address the prestressing.

The above tasks constitute major developments in the field of soft-tissue modeling and are not likely to be completed in the following years. Until a truly realistic structural modeling approach for cardiovascular FSI studies is available, the idealized models used in this work may still serve as a basis for clinically relevant investigations. Non-patient specific simulations based on typical or extreme values can yield valuable insight into flow situations in critical regions. While the investigation of anastomoses was the driving force behind developing the simulation approach for this work, many other applications can be targeted using the same methods. This includes the assessment of the hemodynamics in stented vessels, the influence of non-homogeneous surrounding tissue, and the simulation of blood flow through stent-grafts. While the partitioned solution approach can be applied as-is, the boundary conditions at the coupling interfaces between the one-dimensional and the three-dimensional model should be improved. A desired feature that is not covered by the solvers developed in the scope of this work could be non-reflecting boundary conditions that not only couple flow and pressure, but also the structural deformations.

The fact that the structural model as well as the mentioned interface boundary conditions are not yet fully satisfactory gives just one direction for future works. Another direction is given by the potential of the partitioned solution approach. The decreased computational effort due to the reduced modeling approach of the surrounding allows for a usage of the approach in the scope of mathematical optimization methods. While the current work provides methods to predict local hemodynamic quantities, optimization methods could provide improved shapes or graft materials that, e.g., minimize the oscillating shear index in the anastomosis region. While the general applicability of adjoint methods for this purpose was already shown in [98], the general applicability of adjoint methods for this purpose was already shown, future works should further develop this promising approach.

Appendix

A.1 Tensor algebra

For an in-depth treatment of tensor algebra as well as tensor analysis see [26, 230, 5]. Here, only non-standard operators are listed which are not uniquely defined in literature.

A.1.1 Contractions and scalar products

Considering first-order tensors $\boldsymbol{a}$ and $\boldsymbol{b}$, second-oder tensors $\boldsymbol{A}$ and $\boldsymbol{B}$, and fourth-order tensors $\mathcal{A}$ and $\mathcal{B}$, the following operations are defined according to the notation proposed in [48].

$$\boldsymbol{a} \cdot \boldsymbol{b} = a_i \, b_i \tag{A.1}$$

$$(\boldsymbol{A} \, \boldsymbol{a})_i = A_{ij} \, a_j \tag{A.2}$$

$$\boldsymbol{A} \cdot \boldsymbol{B} = A_{ij} \, B_{ij} \tag{A.3}$$

$$(\boldsymbol{A} \, \boldsymbol{B})_{ik} = A_{ij} \, B_{jk} \tag{A.4}$$

$$(\mathcal{A} \, \boldsymbol{A})_{ij} = A_{ijkl} \, B_{kl} \tag{A.5}$$

$$(\mathcal{A} \mathcal{B})_{ijmn} = A_{ijkl} \, B_{klmn} \tag{A.6}$$

A.1.2 Dyadic products of second order tensors

Considering two arbitrary second-order tensors $\boldsymbol{A}$ and $\boldsymbol{B}$, the following operators are defined:

$$(\boldsymbol{A} \otimes \boldsymbol{B})_{ijkl} = A_{ij} \, B_{kl} \tag{A.7}$$

$$(\boldsymbol{A} \, \underline{\otimes} \, \boldsymbol{B})_{ijkl} = A_{ik} \, B_{jl} \tag{A.8}$$

$$(\boldsymbol{A} \, \overline{\otimes} \, \boldsymbol{B})_{ijkl} = A_{il} \, B_{jk} \tag{A.9}$$

$$(\boldsymbol{A} \, \underline{\overline{\otimes}} \, \boldsymbol{B})_{ijkl} = \frac{1}{2} \, (A_{ik} \, B_{jl} + A_{il} \, B_{jk}) \tag{A.10}$$

A.1.3 Special fourth order tensors

Fourth-order tensors constructed from the dyadic product of the second-order identity tensor $\mathbf{1}$ are helpful when working with tensor expressions. The following definitions are used in this work:

$$\mathcal{I} = \mathbf{1} \underline{\otimes} \mathbf{1} \qquad \rightarrow \qquad \mathcal{I}\,\boldsymbol{A} = \boldsymbol{A} \tag{A.11}$$

$$\mathcal{T} = \mathbf{1} \otimes \mathbf{1} \qquad \rightarrow \qquad \mathcal{T}\,\boldsymbol{A} = \operatorname{tr}(\boldsymbol{A})\,\mathbf{1} \tag{A.12}$$

$$\mathcal{S} = \mathbf{1} \overline{\underline{\otimes}} \mathbf{1} \qquad \rightarrow \qquad \mathcal{S}\,\boldsymbol{A} = \operatorname{sym}(\boldsymbol{A}) \tag{A.13}$$

$$\mathcal{D} = \infty - \frac{1}{3}\mathcal{T} \qquad \rightarrow \qquad \mathcal{D}\,\boldsymbol{A} = \operatorname{dev}(\boldsymbol{A}) \tag{A.14}$$

A.2 Continuum mechanics

A.2.1 Neo-Hookean elasticity tensor

Recalling the second Piola-Kirchhoff stress as given in (3.103), it can be rewritten as

$$\boldsymbol{S} \overset{\text{NH2}}{=} \mu\, J^{\frac{2}{3}} \mathbf{1} - \frac{1}{3}\,\mu\, J^{\frac{2}{3}}\, \mathrm{I}_{\boldsymbol{C}} \boldsymbol{C}^{-1} + \frac{\kappa}{2}\, J^2\, \boldsymbol{C}^{-1} - \frac{\kappa}{2}\, \boldsymbol{C}^{-1}. \tag{A.15}$$

The tangent modulus can be obtained in a systematic way by taking the derivative of each summand separately. For the first summand

$$\frac{\partial}{\partial \boldsymbol{C}} \left(\mu\, J^{\frac{2}{3}}\, \mathbf{1} \right) = -\frac{1}{3}\, \mu\, J^{-\frac{2}{3}}\, \mathbf{1} \otimes \boldsymbol{C}^{-1} \tag{A.16}$$

is obtained. The second one results in

$$\begin{aligned} \frac{1}{3}\,\mu\, \frac{\partial}{\partial \boldsymbol{C}} \left(\mathrm{I}_{\boldsymbol{C}}\, \boldsymbol{C}^{-1}\, J^{-\frac{2}{3}} \right) &= \frac{1}{3}\,\mu\, J^{\frac{2}{3}}\, \frac{\partial}{\partial \boldsymbol{C}} \left(\mathrm{I}_{\boldsymbol{C}}\, \boldsymbol{C}^{-1} \right) + \frac{1}{3}\,\mu\, \mathrm{I}_{\boldsymbol{C}}\, \boldsymbol{C}^{-1} \frac{\partial J^{\frac{2}{3}}}{\partial \boldsymbol{C}} \\ &= \frac{1}{3}\,\mu\, J^{\frac{2}{3}} \left(\mathbf{1} \otimes \boldsymbol{C}^{-1} - \mathrm{I}_{\boldsymbol{C}}\, \boldsymbol{C}^{-1} \overline{\underline{\otimes}} \boldsymbol{C}^{-1} - \frac{1}{3}\, \mathrm{I}_{\boldsymbol{C}}\, \boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} \right). \end{aligned} \tag{A.17}$$

For the third and fourth summand,

$$\begin{aligned} \frac{\kappa}{2}\, \frac{\partial}{\partial \boldsymbol{C}} \left(J^2\, \boldsymbol{C}^{-1} \right) &= \frac{\kappa}{2}\, \frac{\partial J^2}{\partial \boldsymbol{C}}\, \boldsymbol{C}^{-1} + \frac{\kappa}{2}\, J^2\, \frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{C}} \\ &= \frac{\kappa}{2}\, J^2 \left(\boldsymbol{C}^{-1} \otimes \boldsymbol{C}^{-1} - \boldsymbol{C}^{-1} \overline{\underline{\otimes}} \boldsymbol{C}^{-1} \right) \end{aligned} \tag{A.18}$$

and

$$\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{C}} = -\boldsymbol{C}^{-1} \overline{\underline{\otimes}} \boldsymbol{C}^{-1} \tag{A.19}$$

are obtained, respectively. The summation of Eq. (A.16-A.19) results in the expression given for $\boldsymbol{\mathcal{C}}$ in (3.105).

A.3 Finite elements

A.3.1 Weak form

As mentioned in Section 4.1.1, some algebra is necessary to derive the final weak form (4.2) from the weighted residual form (4.1). To start off with, the static internal load term in (4.1) is integrated by parts, which yields

$$\int\limits_{\Omega_0^{\mathrm{s}}} \mathrm{Div}(\boldsymbol{P}) \cdot \boldsymbol{\delta d} \, \mathrm{d}\Omega_0^{\mathrm{s}} = \int\limits_{\Omega_0^{\mathrm{s}}} \boldsymbol{P} \cdot \mathrm{Grad}(\boldsymbol{\delta d}) \; \mathrm{d}\Omega_0^{\mathrm{s}} - \int\limits_{\Gamma_0^{\mathrm{s}}} \boldsymbol{P} \, \boldsymbol{N} \cdot \boldsymbol{\delta d} \, \mathrm{d}\Gamma_0^{\mathrm{s}}. \tag{A.20}$$

Further,

$$\boldsymbol{\delta E} = \frac{1}{2} \Big(\left(\mathrm{Grad}(\boldsymbol{\delta d})\right)^{\mathrm{T}} \, \boldsymbol{F} + \boldsymbol{F}^{\mathrm{T}} \, \mathrm{Grad}(\boldsymbol{\delta d}) \Big) \tag{A.21}$$

holds for the variation of the Green-Lagrange strain tensor. Drawing on the symmetry of $\boldsymbol{S}$ and $\boldsymbol{\delta E}$, it holds that

$$\boldsymbol{P} \cdot \mathrm{Grad}(\boldsymbol{\delta d}) = \boldsymbol{S} \cdot \boldsymbol{F}^{\mathrm{T}} \, \mathrm{Grad}(\boldsymbol{\delta d}) = \boldsymbol{S} \cdot \boldsymbol{\delta E}. \tag{A.22}$$

as detailed in [230, p. 84]. Together with (A.21), (A.22) provides the link between (4.1) and (4.2) for the general case, i.e. without assuming an underlying minimization problem.

A.3.2 Special matrices

The element interpolation matrix

$$\mathbf{N}^{(e)} = \begin{bmatrix} N_1^{(e)} & 0 & 0 & & N_{n_\mathrm{e}}^{(e)} & 0 & 0 \\ 0 & N_1^{(e)} & 0 & \dots & 0 & N_{n_\mathrm{e}}^{(e)} & 0 \\ 0 & 0 & N_1^{(e)} & & 0 & 0 & N_{n_\mathrm{e}}^{(e)} \end{bmatrix} \tag{A.23}$$

allows for an evaluation of quantities which are defined through an element degree of freedom vector, e.g. the displacement vector $\mathbf{d}^{(e)}$, inside the element as

$$\boldsymbol{d} = \mathbf{N}^{(e)} \, \mathbf{d}^{(e)} \quad \text{in } \Omega^{(e)}. \tag{A.24}$$

The element strain-displacement matrix

$$\mathbf{B}_i^{(e)} = \begin{bmatrix} F_{11}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{21}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{31}\,\frac{\partial N_i^{(e)}}{\partial r} \\ F_{12}\,\frac{\partial N_i^{(e)}}{\partial s} & F_{22}\,\frac{\partial N_i^{(e)}}{\partial s} & F_{32}\,\frac{\partial N_i^{(e)}}{\partial s} \\ F_{13}\,\frac{\partial N_i^{(e)}}{\partial t} & F_{23}\,\frac{\partial N_i^{(e)}}{\partial t} & F_{33}\,\frac{\partial N_i^{(e)}}{\partial t} \\ F_{11}\,\frac{\partial N_i^{(e)}}{\partial s} + F_{12}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{21}\,\frac{\partial N_i^{(e)}}{\partial s} + F_{22}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{31}\,\frac{\partial N_i^{(e)}}{\partial s} + F_{32}\,\frac{\partial N_i^{(e)}}{\partial r} \\ F_{12}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{13}\,\frac{\partial N_i^{(e)}}{\partial s} & F_{22}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{23}\,\frac{\partial N_i^{(e)}}{\partial s} & F_{32}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{33}\,\frac{\partial N_i^{(e)}}{\partial s} \\ F_{11}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{13}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{21}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{23}\,\frac{\partial N_i^{(e)}}{\partial r} & F_{31}\,\frac{\partial N_i^{(e)}}{\partial t} + F_{33}\,\frac{\partial N_i^{(e)}}{\partial r} \end{bmatrix} \tag{A.25}$$

allows for a computation of

$$\delta\mathbf{E}^{\mathrm{V}} = \mathbf{B}^{(e)}\,\delta\mathbf{d}^{(e)} \tag{A.26}$$

and

$$\boldsymbol{\Delta}\mathbf{E}^{\mathrm{V}} = \mathbf{B}^{(e)}\,\boldsymbol{\Delta}\mathbf{d}^{(e)} \tag{A.27}$$

A.3.3 Assembly

In the finite element method, global matrices and vectors are assembled from the counterparts on element level, e.g. $\mathbf{M}$ from $\mathbf{M}^{(e)}$ and $\mathbf{d}$ from $\mathbf{d}^{(e)}$. To this end, all degrees of freedom are identified with a unique global index g and an element location vector that contains the global indices of the element's $(d\,n_e)$ degrees of freedom is introduced as

$$\mathbf{l}^{(e)} = \left[\begin{array}{cccc} g_1 & g_2 & \dots & g_{(d\,n_e)} \end{array}\right], \tag{A.28}$$

where d denotes the number of space dimensions. Now, the assembly process can be realized according to Alg. 7.

Algorithm 7 Assembly of global matrices and vectors in the finite element method.

```
for $e = 1 \dots n^{\mathrm{e}}$ do
    for $i = 1\ \dots\ (d\,n_e)$ do
        $d_{l_i} = d_i^{(e)}$
        for $j = 1\ \dots\ (d\,n_e)$ do
            $M_{l_i\,l_j} = M_{ij}^{(e)}$
        end for
    end for
end for
```

Table A.3.1: Relation between edge coordinates and element coordinates

node	1	2	3	4	5	6	7	8
l	1	2	1	2	1	2	1	2
m	1	1	2	2	1	1	2	2
n	1	1	1	1	2	2	2	2

A.3.4 Voigt notation

As done for second-order tensors in (4.14) and (4.15), fourth-order tensors that have the symmetries of the elasticity tensor introduced in (3.74) can be represented using the Voigt notation as

$$\mathcal{C} \to \mathbf{C}^{\mathrm{V}} = \begin{bmatrix} \mathcal{C}_{1111} & \mathcal{C}_{1122} & \mathcal{C}_{1133} & \mathcal{C}_{1112} & \mathcal{C}_{1123} & \mathcal{C}_{1113} \\ \mathcal{C}_{2211} & \mathcal{C}_{2222} & \mathcal{C}_{2233} & \mathcal{C}_{2212} & \mathcal{C}_{2223} & \mathcal{C}_{2213} \\ \mathcal{C}_{3311} & \mathcal{C}_{3322} & \mathcal{C}_{3333} & \mathcal{C}_{3312} & \mathcal{C}_{3323} & \mathcal{C}_{3313} \\ \mathcal{C}_{1211} & \mathcal{C}_{1222} & \mathcal{C}_{1233} & \mathcal{C}_{1212} & \mathcal{C}_{1223} & \mathcal{C}_{1213} \\ \mathcal{C}_{2311} & \mathcal{C}_{2322} & \mathcal{C}_{2333} & \mathcal{C}_{2312} & \mathcal{C}_{2323} & \mathcal{C}_{2313} \\ \mathcal{C}_{1311} & \mathcal{C}_{1322} & \mathcal{C}_{1333} & \mathcal{C}_{1312} & \mathcal{C}_{1323} & \mathcal{C}_{1313} \end{bmatrix} . \tag{A.29}$$

Considering the product arising in the linearized weak form (4.4), it can be reformulated as

$$\boldsymbol{\delta E} \cdot \mathcal{C}\, \boldsymbol{\Delta E} \to \delta\mathbf{E}^{\mathrm{V}} \cdot \mathbf{C}^{\mathrm{V}}\, \boldsymbol{\Delta}\mathbf{E}. \tag{A.30}$$

A.3.5 Nodal shape function indices

Recalling the nodal mapping

$$\boldsymbol{n}_i^{(e)} = N_l^{(1\mathrm{D})}(r^{(e)})\, N_m^{(1\mathrm{D})}(s^{(e)})\, N_n^{(1\mathrm{D})}(s^{(e)})\, \boldsymbol{X}_n^{(e,\mathrm{n})}, \tag{A.31}$$

from (4.52), the relations between the index i and the indices l, m, and n for the node numbering of a hexahedral element as shown in Fig. 4.1 are those given in Tab. A.3.1.

A.3.6 Face and edge coordinates

For the node numbering of a hexahedral element as shown in Fig. A.3.1, the relations between the element coordinates and the face and the edge coordinates, respectively, are those given in Tab.A.3.3 and Tab. A.3.2.

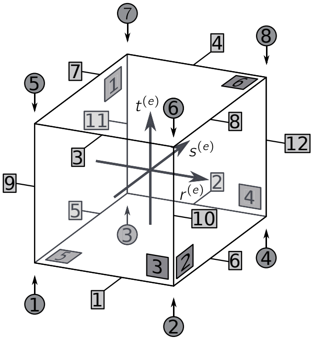


Figure A.3.1: Numbering of nodes, faces and edges of a hexahedral finite element.

Table A.3.2: Relation between edge coordinates and element coordinates

edge	1	2	3	4	5	6	7	8	9	10	11	12
r_a	$r^{(e)}$	$r^{(e)}$	$r^{(e)}$	$r^{(e)}$	$s^{(e)}$	$s^{(e)}$	$s^{(e)}$	$s^{(e)}$	$t^{(e)}$	$t^{(e)}$	$t^{(e)}$	$t^{(e)}$
r_b	$s^{(e)}$	$s^{(e)}$	$-s^{(e)}$	$-s^{(e)}$	$t^{(e)}$	$t^{(e)}$	$-t^{(e)}$	$-t^{(e)}$	$r^{(e)}$	$r^{(e)}$	$-r^{(e)}$	$-r^{(e)}$
r_c	$t^{(e)}$	$-t^{(e)}$	$t^{(e)}$	$-t^{(e)}$	$r^{(e)}$	$-r^{(e)}$	$r^{(e)}$	$-r^{(e)}$	$s^{(e)}$	$-s^{(e)}$	$s^{(e)}$	$-s^{(e)}$

A.4 Taylor-Galerkin method

The global systems (4.87) and (4.88) are assembled from their counterpart on element level, which can be computed as follows. While the same matrices are given in [89], a different notation – one commonly used in the scope of the finite-element method – is applied here. This allows for a more flexible tuning of the method, e.g. by using different shape functions or quadrature rules.

A.4.1 Left-hand side

Introducing the interpolation matrix

$$\mathbf{N} = \begin{bmatrix} N_1^{1\mathrm{D}}(r^{(e)}) & 0 \\ 0 & N_2^{1\mathrm{D}}(r^{(e)}) \end{bmatrix}, \tag{A.32}$$

which is assumed to be the same for each element e and applying the usual finite element approximations

$$A^{1\mathrm{D}}(x) \approx \mathbf{N}\,\mathbf{A}^{(e)} \quad \text{and} \quad Q^{1\mathrm{D}}(x) \approx \mathbf{N}\,\mathbf{Q}^{(e)} \quad \text{in } \Omega^{(e)} \tag{A.33}$$

Table A.3.3: Relation between face coordinates and element coordinates

face	1	2	3	4	5	6
r_a	$r^{(e)}$	$r^{(e)}$	$r^{(e)}$	$r^{(e)}$	$s^{(e)}$	$s^{(e)}$
r_b	$s^{(e)}$	$s^{(e)}$	$-s^{(e)}$	$-s^{(e)}$	$t^{(e)}$	$t^{(e)}$
r_c	$t^{(e)}$	$-t^{(e)}$	$t^{(e)}$	$-t^{(e)}$	$r^{(e)}$	$-r^{(e)}$

with

$$\mathbf{A}^e = [A_1^e\ A_2^e]^{\mathrm{T}} \quad \text{and} \quad \mathbf{Q}^e = [Q_1^e\ Q_2^e]^{\mathrm{T}}, \tag{A.34}$$

the element mass matrix reads

$$\mathbf{M}^{(e)} = \frac{h}{2} \int\limits_{-1}^{1} \mathbf{N}^T \mathbf{N} \, \mathrm{d}r = \frac{h}{2} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}. \tag{A.35}$$

Therein, h denotes the length common to all elements, such that $h/2$ can be interpreted as $\partial x/\partial r^{(e)}$. As in the scope of the FEM for structural mechanics, the global left hand side $\mathbf{M}$ may be assembled according to Alg. 7.

A.4.2 Right-hand side

The right hand sides are split into the individual terms arising in (4.85) such that

$$\mathbf{R}_k^{(A,e)} = \mathbf{R}_k^{(A1,e)} + \mathbf{R}_k^{(A2,e)} + \mathbf{R}_k^{(A3,e)} \tag{A.36}$$

and

$$\mathbf{R}_k^{(Q,e)} = \mathbf{R}_k^{(Q1,e)} + \mathbf{R}_k^{(Q2,e)} + \mathbf{R}_k^{(Q3,e)}. \tag{A.37}$$

For the summands, one obtains

$$\mathbf{R}_k^{(A1,e)} = \frac{h^{\mathrm{e}}}{2} \int\limits_{-1}^{1} \mathbf{N}^T \mathbf{N} \, \mathbf{A}_k^{(e)} \, \mathrm{d}r = \mathbf{M}^{\mathrm{e}} \, \mathbf{A}_k^{(e)}, \tag{A.38}$$

$$\mathbf{R}_k^{(Q1,e)} = \frac{h^{\mathrm{e}}}{2} \int\limits_{-1}^{1} \mathbf{N}^T \mathbf{N} \, \mathbf{Q}_k^{(e)} \, \mathrm{d}r = \mathbf{M}^{(e)} \, \mathbf{Q}_k^{(e)} \tag{A.39}$$

$$\mathbf{R}_k^{(A2,e)} = \frac{h^{\mathrm{e}}}{2} \int\limits_{-1}^{1} \mathbf{B}^T \left(\mathbf{N}\,\mathbf{Q}_k^{(e)} - \frac{\Delta t}{2}\, K_R \frac{\mathbf{N}\,\mathbf{Q}^{(e)}}{\mathbf{N}\,\mathbf{A}_k^{(e)}} \right) \mathrm{d}r \tag{A.40}$$

$$\mathbf{R}_k^{(Q2,e)} = \frac{h^{\mathrm{e}}}{2} \int\limits_{-1}^{1} \mathbf{B}^T \left(\frac{\alpha \left(\mathbf{N}\,\mathbf{Q}_k^{(e)}\right)^2}{\mathbf{N}\,\mathbf{A}_k^{(e)}} + \frac{\beta \left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^{\frac{2}{3}}}{3\,\rho\,A_0} - \frac{\Delta t\,\alpha\,K_R \left(\mathbf{N}\,\mathbf{Q}_k^{(e)}\right)^2}{\left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^2} \right) \mathrm{d}r \tag{A.41}$$

$$\mathbf{R}_k^{(A3,e)} = \mathbf{0} \tag{A.42}$$

$$\begin{aligned} \mathbf{R}_k^{(Q3,e)} = \frac{h^{\mathrm{e}}}{2} \int\limits_{-1}^{1} \mathbf{B}^T & \left(\frac{K_R\,\mathbf{N}\,\mathbf{Q}_k^{(e)}\,\mathbf{B}\,\mathbf{Q}_k^{(e)}}{\left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^2} - \frac{2\,K_R\,\mathbf{N}\,\mathbf{Q}_k^{(e)}\,\mathbf{B}\,\mathbf{Q}_k^{(e)}}{\left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^2} \right. \\ & \left. - \frac{K_R \left(\mathbf{N}\,\mathbf{Q}_k^{(e)}\right)^2 \mathbf{B}\,\mathbf{A}_k^{(e)}}{\left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^3} + \frac{K_R\,\beta \left(\mathbf{N}\,\mathbf{A}_k^{(e)}\right)^{\frac{1}{2}} \mathbf{B}\,\mathbf{A}_k^{(e)}}{2\,\rho\,A_0} \right) \mathrm{d}r \end{aligned} \tag{A.43}$$

A.5 Radial basis functions

In addition to the mean error shown in Section 4.3.5, Fig. 4.18, the maximum error for each case considered there is shown in Fig. A.5.2.

A.6 Multi body system

From the structure of the system, the tangent matrix $\mathbf{K}$ according to the directional derivative $\mathrm{D}\mathbf{g}_{\mathbf{x}}\,[\boldsymbol{\Delta}\mathbf{x}] = \mathbf{K}\,\boldsymbol{\Delta}\mathbf{x}$ can be identified to be a band matrix

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^{(1,1)} & \mathbf{K}^{(1,2)} & & & & \\ \mathbf{K}^{(2,1)} & \mathbf{K}^{(2,2)} & \mathbf{K}^{(2,3)} & & & \\ & & \ddots & & & \\ & & & \mathbf{K}^{(n-1,n-2)} & \mathbf{K}^{(n-1,n-1)} & \mathbf{K}^{(n-1,n)} \\ & & & & \mathbf{K}^{(n,n-1)} & \mathbf{K}^{(n,n)} \end{bmatrix} \tag{A.44}$$

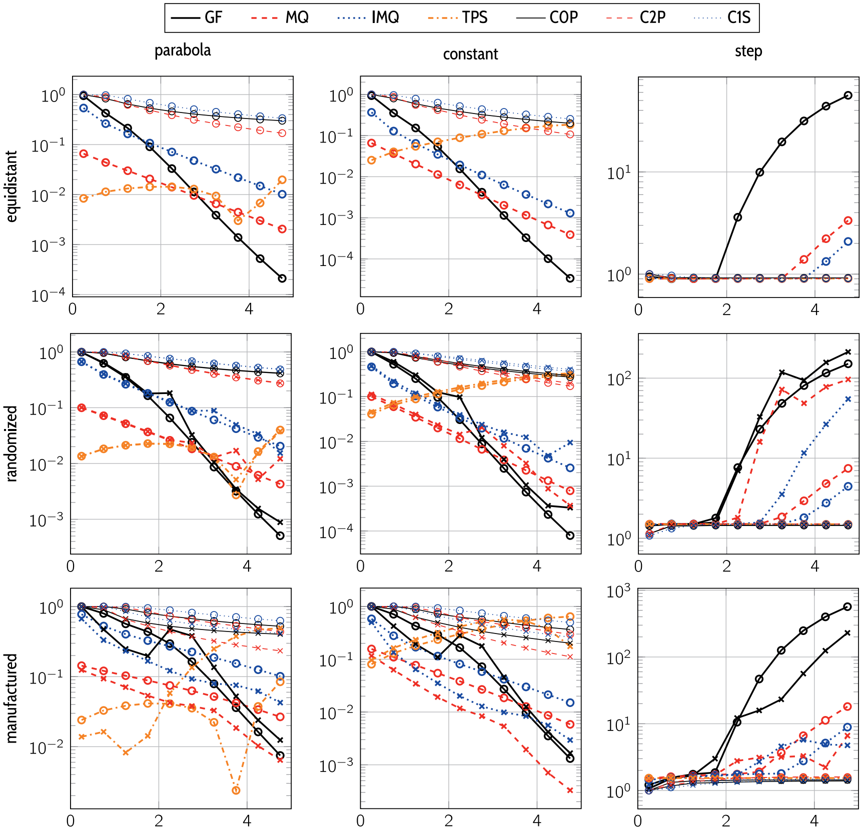


Figure A.5.2: Maximum interpolation error $\|f(x_i^{\mathrm{t}}) - y_i^{\mathrm{t}}\|$ (ordinates) with increasing mean support radius R_{s} (abscissas) for the interpolation of differently distributed source points sampling different functions. Marks $\times$ denote studies with adaptive $R_{\mathrm{s}}^{(i)}$, marks $\circ$ denote studies with uniform $R_{\mathrm{s}}^{(i)} = R_{\mathrm{s}}$.

with the blocks $\mathbf{K}^{(i,j)} = \mathrm{D}\boldsymbol{g}^{(i)}_{\boldsymbol{x}^{(j)}}\left[\boldsymbol{\Delta x}^{(j)}\right]$. The individual derivatives are determined as

$$
\begin{aligned}
\mathbf{K}^{(i,i-1)} &= \begin{bmatrix} \dfrac{\partial \mathbf{f}^{(i)}}{\partial x_1^{(i-1)}} & \dfrac{\partial \mathbf{f}^{(i)}}{\partial x_2^{(i-1)}} \end{bmatrix}, \\
\mathbf{K}^{(i,i)} &= \frac{1}{\beta \Delta t^2}\begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} + \frac{\gamma}{\beta \Delta t}\begin{bmatrix} d & 0 \\ 0 & d \end{bmatrix} \\
&\quad + \begin{bmatrix} -\dfrac{\partial \mathbf{f}^{(i)}}{\partial x_1^{(i)}} + \dfrac{\partial \mathbf{f}^{(i+1)}}{\partial x_1^{(i)}} & -\dfrac{\partial \mathbf{f}^{(i)}}{\partial x_2^{(i)}} + \dfrac{\partial \mathbf{f}^{(i+1)}}{\partial x_2^{(i)}} \end{bmatrix} \\
\mathbf{K}^{(i,i+1)} &= \begin{bmatrix} \dfrac{\partial \mathbf{f}^{(i+1)}}{\partial x_1^{(i+1)}} & \dfrac{\partial \mathbf{f}^{(i+1)}}{\partial x_2^{(i+1)}} \end{bmatrix}.
\end{aligned}
\tag{A.45}
$$

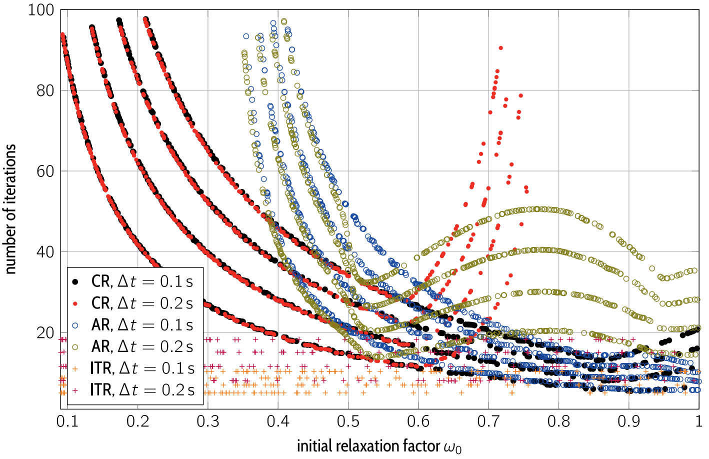


Figure A.8.3: Influence of the (initial) relaxation factor ω_0 in the constant, the Aitken and the Irons-Tuck relaxation in combination with the mixed coupling algorithm.

In each time step, a Newton-Raphson iteration

$$\mathbf{K}|_{\tilde{\mathbf{x}}^{(i)}}\,\boldsymbol{\Delta}\mathbf{x}^{(i)} = -\mathbf{g}|_{\tilde{\mathbf{x}}^{(i)}} \tag{A.46}$$

$$\tilde{\mathbf{x}}^{(i+1)} = \tilde{\mathbf{x}}^{(i)} + \boldsymbol{\Delta}\mathbf{x}^{(i)} \tag{A.47}$$

starting with an initial guess $\tilde{\mathbf{x}}^{(0)} = \mathbf{x}_i$ is performed until $\|\mathbf{g}\| < e^{\mathrm{tol}}$ after N iterations. Then, after setting $\mathbf{x}_{i+1} = \tilde{\mathbf{x}}^{(N)}$, the next time step is processed.

A.7 Coupling software

Listing 3 shows the source code of the algorithm section of a coupling manager process implementing a parallel coupling algorithm.

A.8 Preliminary investigations

Figure 5.32 shows the influence of the parameter ω_0 on the mean number of consecutive solver calls j^* for the mixed coupling algorithm in addition to Figure 5.32 and 5.33, covering the staggered and the parallel algorithm, respectively.

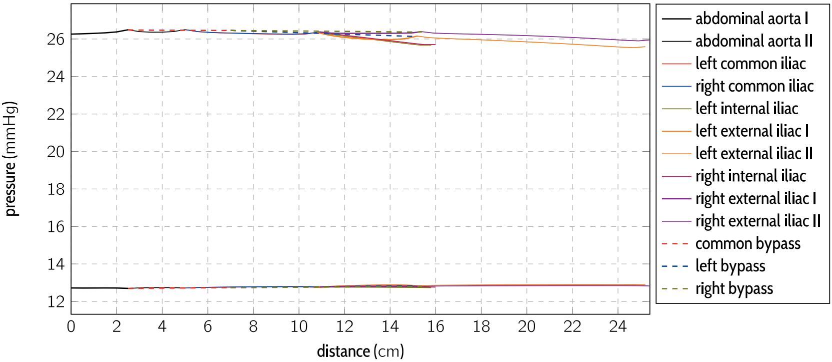


Figure A.9.4: Pressure distribution in the postoperative vessel network at the time of minimum and maximum inflow.

A.9 Applications

In this section, additional information and results obtained in the scope of Chapter 6 are given. Fig. A.9.4 shows the pressure distribution in the postoperative vessel network considered in Section 6.3.

Listing 3 Parallel coupling algorithm section of a typical implementation of a CMP in an FSI simulation using the C++ framework *comana*.

```
    // end of initialization section

    // building blocks
    PolynomialPredictor predictor(2);
    ConvergenceCriterion convergenceCriterion(1e-6);
    QuasiNewtonLeastSquaresMethod accelerator(displacementVector);

    // time loop
    const auto numberOfTimeSteps=1000;
    const auto maxNumberOfIterations=50;
    for(auto t=0; t<numberOfTimeSteps; t++)
    {
        predictor.predict(displacementArray);
        for(auto j=0; j<maxNumberOfIterations; j++)
        {
            preliminarySolutionVector = solutionVector;

            scatter(solution, displacement, traction);

            // solve both subproblems
            R::setField(fPatch, L::vertex, Q::displacement, displacement);
            R::setField(sPatch, L::quadraturePoint, Q::traction, traction);
            R::proceed(fSolver);
            R::proceed(sSolver);
            R::getField(fPatch, L::custom, Q::traction, traction);
            R::getField(sPatch, L::custom, Q::displacement, displacement);

            gather(displacement, traction, solution);
            residual = preliminarySolutionVector - solutionVector;

            if(criterion.fulfilled(solutionVector,residual)
            {
                solutionVector = preliminarySolutionVector;
                R::proceed(fSolver);
                R::proceed(sSolver);
                break;
            }
            else
            {
                R::iterate(fSolver);
                R::iterate(sSolver);
                solutionVector = accelerator.updateSolution(residual);
            }
        }
    }
}
```

Bibliography

[1] L. Achaba, M. Mahfouda, and S. Benhadida. "Numerical study of the non-Newtonian blood flow in a stenosed artery using two rheological models". In: *Thermal Science* 20.2 (2016), pp. 449–460.

[2] A. Aitken. "Studies in Practical Mathematics. On the Iterative Solution of a System of Linear Equations". In: *Proceedings of the Royal Society of Edinburgh. Section A. Mathematical and Physical Sciences* 63.1 (1950), pp. 52–60.

[3] B. Alberts, A. Johnson, L. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of the Cell.* New York: Garland Science, 2002.

[4] A. Alexandrescu. *Modern C++ Design: Generic Programming and Design Patterns Applied.* Boston et al.: Addison-Wesley, 2001.

[5] H. Altenbach. *Kontinuumsmechanik.* Berlin and Heidelberg: Springer Vieweg, 2015.

[6] M. Anliker, R. Rockwell, and E. Ogden. "Nonlinear analysis of flow pulses and shock waves in arteries". In: *Zeitschrift für angewandte Mathematik und Physik ZAMP* 22.2 (1971), pp. 217–246.

[7] L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi, and D. A. Steinman. "An image-based modeling framework for patient-specific computational hemodynamics". In: *Medical & Biological Engineering & Computing* 46.11 (2008), pp. 1097–1112.

[8] Argonne National Laboratory. *The Message Passing Interface (MPI) standard.* 2017. URL: `http://www.mcs.anl.gov/research/projects/mpi` (visited on 12/20/2017).

[9] V. I. Arnold, V. V. Kozlov, and A. I. Neishtadt. *Mathematical Aspects of Classical and Celestial Mechanics.* Berlin and Heidelberg: Springer, 2006.

[10] G. Ateshian and K. Costa. "A frame-invariant formulation of Fung elasticity". In: *Journal of Biomechanics* 42.6 (2009), pp. 781–785.

[11] J. Awrejcewicz. *Classical Mechanics.* New York: Springer, 2012.

[12] U. Ayachit. *The ParaView Guide: A Parallel Visualization Application.* USA: Kitware, Inc., 2015.

[13] L. Badimon and G. Vilahur. "Thrombosis formation on atherosclerotic lesions and plaque rupture". In: *Journal of Internal Medicine* 276.6 (2014), pp. 618–632.

[14] F. Ballarin, E. Faggiano, S. Ippolito, A. Manzoni, A. Quarteroni, G. Rozza, and R. Scrofani. "Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a POD-Galerkin method and a vascular shape parametrization". In: *Journal of Computational Physics* 315.Supplement C (2016), pp. 609–628.

[15] P. D. Ballyk, C. Walsh, J. Butany, and M. Ojha. "Compliance mismatch may promote graft-artery intimal hyperplasia by altering suture-line stresses". In: *Journal of Biomechanics* 31.3 (1997), pp. 229–237.

[16] K. J. Bathe. *Finite Element Procedures.* Englewood Cliffs: Prentice Hall, 1996.

[17] J. Baulmann, J. Nürnberger, J. Slany, R. Schmieder, A. chmidt-Trucksäss, D. Baumgart, P. Cremerius, O. Hess, K. Mortensen, and T. Weber. "Arterielle Gefäßsteifigkeit und Pulswellenanalyse (Arterial stiffness and pulse wave analysis)". In: *Deutsche Medizinische Wochenschrift* 135.S 01 (2010), S4–S14.

[18] Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and Y. Zhang. "Isogeometric fluid-structure interaction: theory, algorithms, and computations". In: *Computational Mechanics* 43.1 (2008), pp. 3–37.

[19] Y. Bazilevs, M. C. Hsu, D. J. Benson, S. Sankaran, and A. L. Marsden. "Computational fluid-structure interaction: methods and application to a total cavopulmonary connection". In: *Computational Mechanics* 45 (2009), pp. 77–89.

[20] Y. Bazilevs, M. C. Hsu, Y. Zhang, W. Wang, T. Kvamsdal, S. Hentschel, and J. G. Isaksen. "Computational vascular fluid-structure interaction: methodology and application to cerebral aneurysms". In: *Biomechanics and Modeling in Mechanobiology* 9.4 (2010), pp. 481–498.

[21] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar. *Computational Fluid-Structure Interaction: Methods and Applications.* Wiley Series in Computational Mechanics. John Wiley & Sons, 2013.

[22] A. Beckert and H. Wendland. "Multivariate interpolation for fluid-structure-interaction problems using radial basis functions". In: *Aerospace Science and Technology* 5 (2001), pp. 125–134.

[23] J. L. Bentley. "Multidimensional Binary Search Trees Used for Associative Searching". In: *Communications of the ACM* 18.9 (1975), pp. 509–517.

[24] M. Bercovier and O. Volpin. "G1 Hierarchical Bézier Surface over Arbitrary Meshes". In: *Computer Graphics Forum* 18.4 (1999), pp. 223–236.

[25] Blender Foundation. *Blender, Free And Open 3D Creation Software.* 2017. URL: `http://www.blender.org/` (visited on 10/28/2018).

[26] J. Bonet and R. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis.* New York: Cambridge University Press, 2008.

[27] G. P. Bonneau and S. Hahmann. "Flexible G1 interpolation of quad meshes". In: *Graphical Models* 76.6 (2014), pp. 669–681.

[28] S. Brändli, M. Abdel-Maksoud, and A. Düster. "A FEM-BEM approach for Fluid-Structure Interaction". In: *Proceedings in Applied Mathematics and Mechanics* 11 (2011), pp. 457–458.

[29] C. G. Broyden. "A class of methods for solving nonlinear simultaneous equations". In: *Mathematics of Computations* 19 (1965), pp. 577–593.

[30] J. R. Buchanan, C. Kleinstreuer, S. Hyun, and G. A. Truskey. "Hemodynamics simulation and identification of susceptible sites of atherosclerotic lesion formation in a model abdominal aorta". In: *Journal of Biomechanics* 36.8 (2003), pp. 1185–1196.

[31] H. J. Bungartz, F. Lindner, M. Mehl, and B. Uekermann. "A plug-and-play coupling approach for parallel multi-field simulations". In: *Computational Mechanics* 55.6 (2015), pp. 1119–1129.

[32] H. J. Bungartz, M. Mehl, and M. Schäfer, eds. *Fluid-Structure Interaction II, Modelling, Simulation, Optimisation.* Vol. 73. Lecture Notes in Computational Science and Engineering. Springer, 2010.

[33] H. J. Bungartz and M. Schäfer, eds. *Fluid-Structure Interaction, Modelling, Simulation and Optimisation.* Vol. 53. Lecture Notes in Computational Science and Engineering. Springer, 2006.

[34] L. S. Caretto, A. D. Gosman, S. V. Patankar, and D. B. Spalding. "Two calculation procedures for steady, three-dimensional flows with recirculation". In: *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics: Vol. II Problems of Fluid Mechanics.* Ed. by H. Cabannes and R. Temam. Berlin and Heidelberg: Springer, 1973, pp. 60–68.

[35] C. G. Caro, J. . Fitz-Gerald, and R. C. Schroter. "Atheroma and arterial wall shear - Observation, correlation and proposal of a shear dependent mass transfer mechanism for atherogenesis". In: *Proceedings of the Royal Society of London B: Biological Sciences* 177.1046 (1971), pp. 109–133.

[36] P. J. Carreau, I. F. MacDonald, and R. B. Bird. "A nonlinear viscoelastic model for polymer solutions and melts—II". In: *Chemical Engineering Science* 23.8 (1968), pp. 901–911.

[37] E. Catmull and J. Clark. "Recursively generated B-spline surfaces on arbitrary topological meshes". In: *Computer-Aided Design* 10.6 (1978), pp. 350–355.

[38] S. Chandra, S. S. Raut, A. Jana, R. W. Biederman, M. Doyle, S. C. Muluk, and E. A. Finol. "Fluid-Structure Interaction Modeling of Abdominal Aortic Aneurysms: The Impact of Patient-Specific Inflow Conditions and Fluid/Solid Coupling". In: *Journal of Biomechanical Engineering* 135.8 (2013), pp. 081001-1–081001-14.

[39] Q. Chen and I. Babuška. "Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle". In: *Computer Methods in Applied Mechanics and Engineering* 128 (1995), pp. 405–417.

[40] Q. Chen and I. Babuška. "The optimal symmetrical points for polynomial interpolation of real functions in the tetrahedron". In: *Computer Methods in Applied Mechanics and Engineering* 137 (1996), pp. 89–94.

[41] J. J. Chiu and S. Chien. "Effects of Disturbed Flow on Vascular Endothelium: Pathophysiological Basis and Clinical Perspectives". In: *Physiological Reviews* 91.1 (2011), pp. 327–387.

[42] J. Chung and G. M. Hulbert. "Time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method". In: *Journal of Applied Mechanics, Transactions ASME* 60.2 (1993), pp. 371–375.

[43] M. G. Coutinho. *Dynamic Simulations of Multibody Systems.* New York: Springer-Verlag, 2001.

[44] M. M. Cross. "Rheology of non-Newtonian fluids: A new flow equation for pseudoplastic systems". In: *Journal of Colloid Science* 20.5 (1965), pp. 417–437.

[45] S. M. Damián. "An Extended Mixture Model for the Simultaneous Treatment of Short and Long Scale Interfaces". Doctoral thesis. Universidad Nacional del Litoral, 2013.

[46] A. de Boer, M. S. van der Schoot, and H. Bijl. "Mesh deformation based on radial basis function interpolation". In: *Computers & Structures* 85.11 (2007), pp. 784–795.

[47] A. de Boer, A. H. van Zuijlen, and H. Bijl. "Radial Basis Functions for Interface Interpolation and Mesh Deformation". In: *Advanced Computational Methods in Science and Engineering.* Ed. by B. Koren and K. Vuik. Berlin and Heidelberg: Springer, 2010, pp. 143–178.

[48] R. de Boer. *Vektor- und Tensorrechnung für Ingenieure.* Berlin and Heidelberg: Springer, 1982.

[49] F. De Santis, C. Pattaro, G. Mani, P. P. Pramstaller, G. Loreni, and G. Martini. "Factors Affecting Long-Term Results of Above-Knee Femoropopliteal Bypass: A Single-Center Contemporary Study". In: *Vascular and Endovascular Surgery* 50.2 (2016), pp. 72–79.

[50] E. S. Debus and W. Gross-Fengels. *Operative und interventionelle Gefäßmedizin.* Berlin and Heidelberg: Springer, 2012.

[51] J. Degroote, K. J. Bathe, and J. Vierendeels. "Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction". In: *Computers & Structures* 87.11 (2009), pp. 793–801.

[52] J. Degroote, R. Bruggeman, R. Haelterman, and J. Vierendeels. "Stability of a coupling technique for partitioned solvers in {FSI} applications". In: *Computers & Structures* 86.23–24 (2008), pp. 2224–2234.

[53] J. Degroote, R. Haelterman, S. Annerel, P. Bruggeman, and J. Vierendeels. "Performance of partitioned procedures in fluid-structure interaction". In: *Computers & Structures* 88.7-8 (2010), pp. 446–457.

[54] J. Degroote, M. Hojjat, E. Stavropoulou, R. Wüchner, and K. U. Bletzinger. "Partitioned solution of an unsteady adjoint for strongly coupled fluid-structure interactions and application to parameter identification of a one-dimensional problem". In: *Structural and Multidisciplinary Optimization* 47.1 (2013), pp. 77–94.

[55] J. Degroote and J. Vierendeels. "Multi-level quasi-Newton coupling algorithms for the partitioned simulation of fluid-structure interaction". In: *Computer Methods in Applied Mechanics and Engineering* 225-228.Supplement C (2012), pp. 14–27.

[56] J. Degroote and J. Vierendeels. "Multi-solver algorithms for the partitioned simulation of fluid-structure interaction". In: *Computer Methods in Applied Mechanics and Engineering* 200.25 (2011), pp. 2195–2210.

[57] A. Delfino, N. Stergiopulos, J. E. Moore, and J. J. Meister. “Residual strain effects on the stress field in a thick wall finite element model of the human carotid bifurcation”. In: *Journal of Biomechanics* 30.8 (1997), pp. 777–786.

[58] J. M. Dolan, J. Kolega, and H. Meng. “High Wall Shear Stress and Spatial Gradients in Vascular Pathology: A Review”. In: *Annals of Biomedical Engineering* 41.7 (2013), pp. 1411–1427.

[59] S. Doll and K. Schweizerhof. “On the Development of Volumetric Strain Energy Functions”. In: *Journal of Applied Mechanics* 67.1 (2000), pp. 17–21.

[60] J. Donea, A. Huerta, J. P. Ponthot, and A. Rodríguez-Ferran. “Arbitrary Lagrangian-Eulerian Methods”. In: *Encyclopedia of Computational Mechanics*. Ed. by E. Stein, R. de Borst, and T. J. R. Hughes. John Wiley & Sons, Ltd, 2004.

[61] J. Dong, Z. Sun, K. Inthavong, and J. Tu. “Fluid-structure interaction analysis of the left coronary artery with variable angulation”. In: *Computer Methods in Biomechanics and Biomedical Engineering* 18.14 (2015), pp. 1500–1508.

[62] J. M. W. Donker, G. H. Ho, A. te Slaa, H. G. W. de Groot, J. C. H. van der Waal, E. J. Veen, and L. van der Laan. “Midterm Results of Autologous Saphenous Vein and ePTFE Pre-Cuffed Bypass Surgery in Peripheral Arterial Occlusive Disease”. In: *Vascular and Endovascular Surgery* 45.7 (2011), pp. 598–603.

[63] DuPont. *DuPont USA*. 2017. URL: `http://www.dupont.com` (visited on 11/27/2017).

[64] A. Düster, H. Bröker, and E. Rank. “The p-version of the finite element method for three-dimensional curved thin walled structures”. In: *International Journal for Numerical Methods in Engineering* 52.7 (2001), pp. 673–703.

[65] A. Düster and S. Kollmannsberger. *AdhoC^4 – User’s Guide*. Lehrstuhl für Computation in Engineering, TU München, Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik, TU Hamburg-Harburg. 2010.

[66] Emory University, OROBIX – Open eyes technology for image analysis. *AneuriskWeb – The Aneurisl dataset repository*. 2017. URL: `http://ecm2.mathcs.emory.edu/aneuriskweb/index` (visited on 01/14/2018).

[67] S. Epstein, M. Willemet, P. J. Chowienczyk, and J. Alastruey. “Reducing the number of parameters in 1D arterial blood flow modeling: less is more for patient-specific simulations”. In: *American Journal of Physiology-Heart and Circulatory Physiology* 309.1 (2015), H222–H234.

[68] P. Erbts, S. Hartmann, and A. Düster. “A partitioned solution approach for electro-thermo-mechanical problems”. In: *Archive of Applied Mechanics* 85 (2015), pp. 1075–1101.

[69] S. Erlicher, L. Bonaventura, and O. S. Bursi. “The analysis of the Generalized-α method for non-linear dynamic problems”. In: *Computational Mechanics* 28.2 (2002), pp. 83–104.

[70] Eurostat. *Cardiovascular diseases statistics*. 2016. URL: `http://ec.europa.eu/eurostat/statistics-explained/index.php/Cardiovascular_diseases_statistics` (visited on 10/21/2017).

[71] J. H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics.* Berlin and Heidelberg: Springer-Verlag, 2002.

[72] C. A. Figueroa, S. Baek, C. A. Taylor, and H. J. D. “A Computational Framework for Fluid-Solid-Growth Modeling in Cardiovascular Simulations”. In: *Computer Methods in Applied Mechanics and Engineering* 198.45–46 (2009), pp. 3583–3602.

[73] C. A. Figueroa and J. Ku. *SimTK – Cardiovascular Model Repository.* 2018. URL: `https://simtk.org/projects/cv-gmodels` (visited on 04/10/2018).

[74] P. J. Flory. “Thermodynamic relations for high elastic materials”. In: *Transactions of the Faraday Society* 57 (1961), pp. 829–838.

[75] L. Formaggia, J. F. Gerbeau, F. Nobile, and A. Quarteroni. “On the coupling of 3D and 1D Navier-Stokes equations for flow problems in compliant vessels”. In: *Computer Methods in Applied Mechanics and Engineering* 191.6–7 (2001), pp. 561–582.

[76] L. Formaggia, D. Lamponi, and A. Quarteroni. “One-dimensional models for blood flow in arteries”. In: *Journal of Engineering Mathematics* 47.3 (2003), pp. 251–276.

[77] C. Förster, W. A. Wall, and E. Ramm. “Artificial Added Mass Instabilities in Sequential Staggered Coupling of Nonlinear Structures and Incompressible Flows”. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), pp. 1278–1293.

[78] O. Frank. “Die Grundform des arteriellen Pulses”. In: *Zeitschrift für Biologie* 37 (1899), pp. 483–526.

[79] D. L. Fry. “Acute Vascular Endothelial Changes Associated with Increased Blood Velocity Gradients”. In: *Circulation Research* 22.2 (1968), pp. 165–197.

[80] Y. C. Fung, K. Fronek, and P. Patitucci. “Pseudoelasticity of arteries and the choice of its mathematical expression”. In: *American Journal of Physiology - Heart and Circulatory Physiology* 237.5 (1979), H620–H631.

[81] T. G. Gallinger. “Effiziente Algorithmen zur partitionierten Lösung stark gekoppelter Probleme der Fluid-Struktur-Wechselwirkung”. Doctoral thesis. Technische Universität München, 2010.

[82] T. C. Gasser, C. A. J. Schulze-Bauer, and G. A. Holzapfel. “A Three-dimensional Finite Element Model for Arterial Clamping”. In: *Journal of Biomechanical Engineering* 124.4 (2002), pp. 355–363.

[83] D. N. Ghista and F. Kabinejadian. “Coronary artery bypass grafting hemodynamics and anastomosis design: a biomedical engineering review”. In: *BioMedical Engineering OnLine* 12.1 (2013), pp. 1–28.

[84] E. D. Gilmore, C. Hudson, R. K. N., R. T. Harvey, M. Mandelcorn, W. C. Lam, and R. G. Devenyi. “Retinal Arteriolar Diameter, Blood Velocity, and Blood Flow Response to an Isocapnic Hyperoxic Provocation in Early Sight-Threatening Diabetic Retinopathy”. In: *Investigative Ophthalmology & Visual Science* 48.4 (2007), p. 1744.

[85] B. Gschaider, H. Nilsson, H. Rusche, H. Jasak, M. Beaudoin, and V. Skuric. *FoamExtend - Open Source CFD Toolbox*. 2017. URL: `https://sourceforge.net/projects/foam-extend/` (visited on 12/28/2017).

[86] G. Guennebaud and B. Jacob. *Eigen v3*. 2010. URL: `http://eigen.tuxfamily.org` (visited on 01/18/2018).

[87] R. Haelterman, A. E. J. Bogaers, K. Scheufele, B. Uekermann, and M. Mehl. "Improving the performance of the partitioned QN-ILS procedure for fluid-structure interaction problems: Filtering". In: *Computers & Structures* 171.Supplement C (2016), pp. 9–17.

[88] S. Hahmann, G. P. Bonneau, and B. Caramiaux. "Bicubic G^1 Interpolation of Irregular Quad Meshes Using a 4-Split". In: *Advances in Geometric Modeling and Processing: 5th International Conference, GMP 2008, Hangzhou, China, 2008. Proceedings*. Ed. by F. Chen and B. Jüttler. Berlin and Heidelberg: Springer Berlin Heidelberg, 2008, pp. 17–32.

[89] B. Hamnetner. "Implementierung eines eindimensionalen Blutflussmodells mittels Finiter Elemente Methode". Diploma thesis. Technische Universität Wien, 2008.

[90] S. Hartmann. "Finite-Elemente Berechnung inelastischer Kontinua". Postdoctoral thesis. Universität Kassel, 2003.

[91] S. Hartmann and R. R. Gilbert. "Identifiability of material parameters in solid mechanics". In: *Archive of Applied Mechanics* 88.1 (2018), pp. 3–26.

[92] S. Hartmann and P. Neff. "Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility". In: *International Journal of Solids and Structures* 40.11 (2003), pp. 2767–2791.

[93] H. Haruguchi and S. Teraoka. "Intimal hyperplasia and hemodynamic factors in arterial bypass and arteriovenous grafts: a review". In: *Journal of Artificial Organs* 6.4 (2003), pp. 227–235.

[94] X. He and D. N. Ku. "Pulsatile Flow in the Human Left Coronary Artery Bifurcation: Average Conditions". In: *Journal of Biomechanical Engineering* 118.1 (1996), pp. 74–82.

[95] Healthwise Staff. *Aortobifemoral bypass*. 2017. URL: `https://myhealth.alberta.ca/Health/pages/conditions.aspx?hwid=zm2720` (visited on 12/22/2017).

[96] Healthwise Staff. *Femoropopliteal (fem-pop) bypass*. 2017. URL: `https://myhealth.alberta.ca/Health/pages/conditions.aspx?hwid=zm2718` (visited on 12/22/2017).

[97] M. Heise. "Der Einfluss der hydraulischen Impedanz auf die Prognose und lokale Hämodynamik femoro-infrainguinaler Rekonstruktionen". Postdoctoral thesis. Humboldt-Universität zu Berlin, 2003.

[98] J. P. Heners, L. Radtke, M. Hinze, and A. Düster. "Adjoint shape optimization for fluid-structure interaction of ducted flows". In: *Computational Mechanics* 61.3 (2018), pp. 259–276.

[99] W. H. Herschel and R. Bulkley. "Konsistenzmessungen von Gummi-Benzollösungen". In: *Kolloid-Zeitschrift* 39.4 (1926), pp. 291–300.

[100] M. T. C. Hoedt, H. van Urk, W. C. J. Hop, A. van der Lugt, and C. H. A. Wittens. “A Comparison of Distal End-to-side and End-to-end Anastomoses in Femoropopliteal Bypasses”. In: *European Journal of Vascular and Endovascular Surgery* 21.3 (2001), pp. 266–270.

[101] G. A. Holzapfel, T. C. Gasser, and R. W. Ogden. “A New Constitutive Framework for Arterial Wall Mechanics and a Comparative Study of Material Models”. In: *Journal of elasticity and the physical science of solids* 61.1 (2000), pp. 1–48.

[102] G. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering.* Chichester and other: Wiley, 2000.

[103] T. Holzmann. *Mathematics, Numerics, Derivations and OpenFOAM.* Leoben: Holzmann CFD, 2017.

[104] 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”. In: *Computational Mechanics* 54.4 (2014), pp. 1055–1071.

[105] T. J. R. Hughes and J. Lubliner. “On the one-dimensional theory of blood flow in the larger vessels”. In: *Mathematical Biosciences* 18.1 (1973), pp. 161–170.

[106] J. D. Humphrey and K. R. Rajagopal. “A constrained mixture model for growth and remodeling of soft tissue”. In: *Mathematical Models and Methods in Applied Sciences* 12.3 (2002), pp. 407–430.

[107] B. Irons and R. C. Tuck. “A version of the Aitken Accelerator for Computer implementation”. In: *International Journal for Numerical Methods in Engineering* 1 (1969), pp. 275–277.

[108] J. G. Isaksen, Y. Bazilevs, T. Kvamsdal, Y. Zhang, J. H. Kaspersen, K. Waterloo, B. Romner, and T. Ingebrigtsen. “Determination of Wall Tension in Cerebral Artery Aneurysms by Numerical Simulation”. In: *Stroke* 39.12 (2008), pp. 3172–3178.

[109] M. Ishigaki, S. Abe, Y. Sibamoto, and T. Yonomoto. “Influence of mesh non-orthogonality on numerical simulation of buoyant jet flows”. In: *Nuclear Engineering and Design* 314.Supplement C (2017), pp. 326–337.

[110] R. I. Issa. “Solution of the implicitly discretised fluid flow equations by operator-splitting”. In: *Journal of Computational Physics* 62.1 (1986), pp. 40–65.

[111] H. Jasak. “Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows”. Dissertation. Imperial College of Science, Technology and Medicine, 1996.

[112] R. Jayendiran, B. M. Nour, and A. Ruimi. “Dacron graft as replacement to dissected aorta: A three-dimensional fluid-structure-interaction analysis”. In: *Journal of the Mechanical Behavior of Biomedical Materials* 78 (2018), pp. 329–341.

[113] W. Joppich and M. Kürschner. “MpCCI – a tool for the simulation of coupled applications”. In: *Concurrency and Computation: Practice and Experience* 18.2 (2006), pp. 183–192.

[114] F. Kabinejadian and D. N. Ghista. “Compliant model of a coupled sequential coronary arterial bypass graft: Effects of vessel wall elasticity and non-Newtonian rheology on blood flow regime and hemodynamic parameters distribution”. In: *Medical Engineering & Physics* 34.7 (2012), pp. 860–872.

[115] F. P. Kärrholm. “Numerical Modelling of Diesel Spray Injection, Turbulence Interaction and Combustion”. Doctoral thesis. Chalmers University of Technology, 2008.

[116] C. T. Kelley. Society for Industrial and Applied Mathematics, 2003.

[117] Khronos Group. *OpenGL - The Standard for High Performance Graphics*. 2018. URL: `https://www.opengl.org/` (visited on 01/11/2018).

[118] S. Kim. “A Study of Non-Newtonian Viscosity and Yield Stress of Blood in a Scanning Capillary-Tube Rheometer”. Doctoral thesis. Drexel University, 2002.

[119] G. Királyfalvi and B. A. Szabó. “Quasi-regional mapping for the p-version of the finite element method”. In: *Finite Elements in Analysis and Design* 27.1 (1997), pp. 85–97.

[120] M. Klarhöfer, B. Csapo, C. Balassy, J. C. Szeles, and E. Moser. “High-resolution blood flow velocity measurements in the human finger”. In: *Magnetic Resonance in Medicine* 45.4 (2001), pp. 716–719.

[121] C. Kleinstreuer, M. Nazemi, and J. P. Archie. “Hemodynamics Analysis of a Stenosed Carotid Bifurcation and its Plaque-Mitigating Design”. In: *Journal of Biomechanical Engineering* 133.3 (1991), pp. 330–335.

[122] M. König, L. Radtke, and A. Düster. “A flexible C++ framework for the partitioned solution of strongly coupled multifield problems”. In: *Computers & Mathematics with Applications* 72.7 (2016), pp. 1764–1789.

[123] A. Konyukhov and K. Schweizerhof. “Closest Point Projection Procedure and Corresponding Curvilinear Coordinate System”. In: *Computational Contact Mechanics: Geometrically Exact Theory for Arbitrary Shaped Bodies*. Berlin and Heidelberg: Springer, 2013, pp. 35–62.

[124] C. Körner. “Die Silikon-beschichtete Polyesterprothese: Konzept, physikalische und biologische Eigenschaften”. Doctoral thesis. Bayerische Julius-Maximilians-Universität Würzburg, 2007.

[125] R. Krause and E. Rank. “A fast algorithm for point-location in a finite element mesh”. In: *Computing* 57.1 (1996), pp. 49–62.

[126] D. N. Ku, D. P. Giddens, C. K. Zarins, and S. Glagov. “Pulsatile flow and atherosclerosis in the human carotid bifurcation. Positive correlation between plaque location and low oscillating shear stress.” In: *Arteriosclerosis, Thrombosis, and Vascular Biology* 5.3 (1985), pp. 293–302.

[127] D. Kuhl and M. A. Crisfield. “Energy-conserving and decaying Algorithms in nonlinear structural dynamics”. In: *International Journal for Numerical Methods in Engineering* 45.5 (1999), pp. 569–599.

[128] U. Küttler, M. Gee, C. Förster, A. Comerford, and W. A. Wall. “Coupling strategies for biomedical fluid-structure interaction problems”. In: *International Journal for Numerical Methods in Biomedical Engineering* 26.3-4 (2010), pp. 305–321.

[129] U. Küttler and W. A. Wall. “Fixed-point fluid-structure interaction solvers with dynamic relaxation”. In: *Computational Mechanics* 43.1 (2008), pp. 61–72.

[130] U. Küttler and W. A. Wall. “Vector Extrapolation for Strong Coupling Fluid-Structure Interaction Solvers”. In: *Journal of Applied Mechanics* 76.2 (2009), pp. 021205–021205-7.

[131] E. L. Kyung, S. L. Jeong, and Y. Jung. “A numerical study on steady flow in helically sinuous vascular prostheses”. In: *Medical Engineering & Physics* 33.1 (2011), pp. 38–46.

[132] J. W. Lambert. “On the nonlinearities of fluid flow in nonrigid tubes”. In: *Journal of the Franklin Institute* 266.2 (1958), pp. 83–102.

[133] M. Lei, J. P. Archie, and C. Kleinstreuer. “Computational design of a bypass graft that minimizes wall shear stress gradients in the region of the distal anastomosis”. In: *Journal of Vascular Surgery* 25.4 (1997), pp. 637–646.

[134] K. Levenberg. “A method for the solution of certain non-linear problems in least squares”. In: *Quarterly Journal of Applied Mathmatics* 2.2 (1944), pp. 164–168.

[135] P. Lewis, J. V. Psaila, W. T. Davies, K. McCarty, and J. P. Woodcock. “Measurement of volume flow in the human common femoral artery using a duplex ultrasound system”. In: *Ultrasound in Medicine & Biology* 12.10 (1986), pp. 777–784.

[136] B. B. Lieber and D. P. Giddens. “Post-stenotic core flow behavior in pulsatile flow and its effects on wall shear stress”. In: *Journal of Biomechanics* 23.6 (1990), pp. 597–605.

[137] F. Loth, P. F. Fischer, and H. S. Bassiouny. “Blood Flow in End-to-Side Anastomoses”. In: *Annual Review of Fluid Mechanics* 40.1 (2008), pp. 367–393.

[138] A. J. Macleod. “Acceleration of vector sequences by multi-dimensional Δ^2 methods”. In: *Communications in Applied Numerical Methods* 2.4 (1986), pp. 385–392.

[139] S. Manini and L. Antiga. *The Vascular Modeling Toolkit*. 2017. URL: `http://www.vmtk.org` (visited on 11/29/2017).

[140] D. W. Marquardt. “An Algorithm for Least-Squares Estimation of Nonlinear Parameters”. In: *Journal of the Society for Industrial and Applied Mathematics* 11.2 (1963), pp. 431–441.

[141] MATLAB. *Version 9.2.0 (R2017a)*. Natick, Massachusetts: The MathWorks Inc., 2017.

[142] H. Meng, V. M. Tutino, J. Xiang, and A. Siddiqui. “High WSS or Low WSS? Complex Interactions of Hemodynamics with Intracranial Aneurysm Initiation, Growth, and Rupture: Toward a Unifying Hypothesis”. In: *American Journal of Neuroradiology* 35.7 (2014), pp. 1254–1262.

[143] P. M. Michaelsen, B. Pritz, and M. Gabi. “A Fluid-structure-interaction Tool by Coupling of Existing Codes”. In: *Proceedings of the 20th European MPI Users’ Group Meeting*. EuroMPI ’13. New York: ACM, 2013, pp. 157–162.

[144] S. Minami and S. Yoshimura. “Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid-structure interactions”. In: 64 (2010), pp. 1129–1147.

[145] D. P. Mock. “Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion”. Doctoral thesis. 2001.

[146] P. Moireau, N. Xiao, M. Astorinot, C. A. Figueroa, D. Chapellet, C. A. Taylor, and J. F. Gerbeau. “External tissue support and fluid-structure simulation in blood flows”. In: *Journal of Biomechanics* 11.1 (2012), pp. 1–18.

[147] P. D. Morris, A. Narracott, H. von Tengg-Kobligk, D. A. Silva Soto, S. Hsiao, A. Lungu, P. Evans, N. W. Bressloff, P. V. Lawford, D. R. Hose, and J. P. Gunn. “Computational fluid dynamics modelling in cardiovascular medicine”. In: *Heart* 102.1 (2016), pp. 18–28.

[148] F. Moukalled, L. Mangani, and M. Darwish. *The Finite Volume Method in Computational Fluid Dynamics*. Cham and other: Springer, 2016.

[149] K. Müller, D. A. Fedosov, and G. Gompper. “Margination of micro- and nanoparticles in blood flow and its effect on drug delivery”. In: *Scientific Reports* 4 (2014), p. 4871.

[150] News Republica. *Side effects angioplasty, how many of us are aware of angioplasty risks*. 2017. URL: `http://newsrepublica.com/side-effects-angioplasty-many-us-aware-angioplasty-risks/` (visited on 11/20/2017).

[151] N. Noël and B. A. Szabó. “Formulation of geometrically non-linear problems in the spatial reference frame”. In: *International Journal for Numerical Methods in Engineering* 40.7 (), pp. 1263–1280.

[152] D. R. Nolan, A. L. Gower, M. Destrade, R. W. Ogden, and J. P. McGarry. “A robust anisotropic hyperelastic formulation for the modelling of soft tissue”. In: *Journal of the Mechanical Behavior of Biomedical Materials* 39 (2014), pp. 48–60.

[153] M. Ohyama, K. Mizushige, H. Ohyama, T. Takahashi, N. Hosomi, S. Ichihara, and M. Kohno. “Carotid turbulent flow observed by convergent color Doppler flowmetry in silent cerebral infarction”. In: *The International Journal of Cardiovascular Imaging* 18.2 (2002), pp. 119–124.

[154] OpenCFD Ltd (ESI Group). *Open FOAM - Official home of The Open Source Computational Fluid Dynamics (CFD) Toolbox*. 2017. URL: `https://www.openfoam.com/` (visited on 12/18/2017).

[155] C. Papazov. “Local, Polynomial G^1 PN Quads”. In: *Advances in Visual Computing: 10th International Symposium, ISVC 2014, Las Vegas, NV, USA, December 8-10, 2014, Proceedings, Part I*. Cham: Springer International Publishing, 2014, pp. 63–74.

[156] S. V. Patankar and D. B. Spalding. “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows”. In: *International Journal of Heat and Mass Transfer* 15.10 (1972), pp. 1787–1806.

[157] M. Peckham, A. Knibbs, and S. Paxton. *Faculty of Biological Sciences at Leeds: The Histology Guide*. 2017. URL: `http://www.histology.leeds.ac.uk/circulatory/arteries.php` (visited on 12/12/2017).

[158] V. Peiffer, A. A. Bharath, S. J. Sherwin, and P. D. Weinberg. “A Novel Method for Quantifying Spatial Correlations Between Patterns of Atherosclerosis and Hemodynamic Factors”. In: *Journal of Biomechanical Engineering* 135.2 (2013), pp. 021023–021023-11.

[159] V. Peiffer, S. J. Sherwin, and P. D. Weinberg. “Does low and oscillatory wall shear stress correlate spatially with early atherosclerosis? A systematic review”. In: *Cardiovascular Research* 99.2 (2013), pp. 242–250.

[160] A. Quarteroni, M. Tuveri, and A. Veneziani. “Computational Vascular Fluid Dynamics: Problems, Models and Methods”. In: *Computing and Visualization in Science* 2.4 (2000), pp. 163–197.

[161] L. Radtke. “Ein partitionierter Ansatz zur Simulation der Fluid-Struktur-Interaktion an der Schnittstelle zu künstlichen Blutgefäßen und endoluminalen Gefäßstützen”. Master thesis. Technische Universität Hamburg-Harburg, 2013.

[162] L. Radtke, M. König, and A. Düster. “The influence of geometric imperfections in cardiovascular FSI simulations”. In: *Computers & Mathematics with Applications* 74.7 (2017), pp. 1675–1689.

[163] L. Radtke, A. Larena-Avellaneda, E. S. Debus, and A. Düster. “Convergence acceleration for partitioned simulations of the fluid-structure interaction in arteries”. In: *Computational Mechanics* 57.6 (2016), pp. 901–920.

[164] L. Radtke, A. Larena-Avellaneda, E. S. Debus, and A. Düster. “Simulation der Fluid-Struktur-Interaktion in arteriellen Bypässen”. In: *Gefässchirurgie* 22.6 (2017), pp. 400–406.

[165] L. Radtke, A. Larena-Avellaneda, T. Kölbel, E. S. Debus, and A. Düster. “Cardiovascular fluid-structure interaction: A partitioned approach utilizing the p-FEM”. In: *Proceeding in applied Mathematics and Mechanics* 14 (2014), pp. 493–494.

[166] A. Razavi, E. Shirani, and M. R. Sadeghi. “Numerical simulation of blood pulsatile flow in a stenosed carotid artery using different rheological models”. In: *Journal of Biomechanics* 44.11 (2011), pp. 2021–2030.

[167] C. M. Rhie and W. L. Chow. “Numerical study of the turbulent flow past an airfoil with trailing edge separation”. In: *AIAA Journal* 21 (1983), pp. 1525–1532.

[168] J. A. G. Rhodin. “Architecture of the Vessel Wall”. In: *Comprehensive Physiology.* John Wiley & Sons, Inc., 2011.

[169] A. Rohatgi. *WebPlotDigitizer – Web based tool to extract data from plots, images, and maps.* 2017. URL: `https://automeris.io/WebPlotDigitizer/` (visited on 11/27/2017).

[170] S. Roll, J. Müller-Nordhorn, T. Keil, H. Scholz, D. Eidt, W. Greiner, and S. N. Willich. “Dacron™vs. PTFE as bypass materials in peripheral vascular surgery – systematic review and meta-analysis”. In: *BMC Surgery* 8.1 (2008), p. 22.

[171] R. I. Rückert. “Experimentelle und klinische Untersuchungen zur Optimierung der Hämodynamik in termino-laterial Prothesenbypass-Anastomosen”. Postdoctoral thesis. Medizinischen Fakultät Charié, Humboldt-Universität zu Berlin, 2001.

[172] K. Sagawa, R. K. Lie, and J. Schaefer. “Translation of Otto Frank’s paper “Die Grundform des arteriellen Pulses”, Zeitschrift für Biologie 37: 483-526 (1899)”. In: *Journal of Molecular and Cellular Cardiology* 22.3 (1990), pp. 253–254.

[173] SAS VirtualExpo. *Medical EXPO - Die Online-Messe für medizinische Ausstattungen.* 2017. URL: `http://img.medicalexpo.com/images_me/photo-g/78422-7227067.jpg` (visited on 11/20/2017).

[174] S. Sathe, R. Benney, R. Charles, E. Doucette, J. Miletti, M. Senga, K. Stein, and T. E. Tezduyar. "Fluid-structure interaction modeling of complex parachute designs with the space-time finite element techniques". In: *Computers & Fluids* 36.1 (2007), pp. 127–135.

[175] R. F. Schmidt, F. Lang, and M. Hechmann. *Pysiologie des Menschen - 31. Auflage.* Berlin and Heidelberg: Springer-Verlag, 2010.

[176] O. Sepahi, L. Radtke, E. S. Debus, and A. Düster. "Anisotropic hierarchic solid finite elements for the simulation of passive-active arterial wall models". In: *Computers & Mathematics with Applications* 74.12 (2017), pp. 3058–3079.

[177] D. M. Sforza, C. M. Putman, and J. R. Cebral. "Computational fluid dynamics in brain aneurysms". In: *International Journal for Numerical Methods in Biomedical Engineering* 28.6-7 (2012), pp. 801–808.

[178] D. Shepard. "A Two-dimensional Interpolation Function for Irregularly-spaced Data". In: *Proceedings of the 1968 23rd ACM National Conference.* New York: Advancing Computing as a Science & Profession, 1968, pp. 517–524.

[179] S. J. Sherwin, L. Formaggia, J. Peiró, and V. Franke. "Computational modelling of 1D blood flow with variable mechanical properties and its application to the simulation of wave propagation in the human arterial system". In: *International Journal for Numerical Methods in Fluids* 43.6–7 (2003), pp. 673–700.

[180] X. Shi, P. Yu, and T. Wang. "G1 continuous conditions of biquartic B-spline surfaces". In: *Journal of Computational and Applied Mathematics* 144.1 (2002), pp. 251–262.

[181] S. S. Shibeshi and W. E. Collins. "The Rheology of Blood Flow in a Branched Arterial System". In: *Applied rheology* 15.6 (2005), pp. 398–405.

[182] P. D. Stein and H. N. Sabbah. "Turbulent blood flow in the ascending aorta of humans with normal and diseased aortic valves." In: *Circulation Research* 39.1 (1976), pp. 58–65.

[183] D. Strauch. *Classical Mechanics – An introduction.* Berlin and Heidelberg: Springer, 2009.

[184] B. A. Szabó and I. Babuška. *Introduction to Finite Element Analysis: Formulation, Verification, and Validation.* Chichester: Wiley-Blackwell, 2011.

[185] I. Tabas, G. García-Cardeña, and G. K. Owens. "Recent insights into the cellular biology of atherosclerosis". In: *The Journal of Cell Biology* 209.1 (2015), pp. 13–22.

[186] K. Takizawa, Y. Bazilevs, and T. E. Tezduyar. "Space-Time and ALE-VMS Techniques for Patient-Specific Cardiovascular Fluid-Structure Interaction Modeling". In: *Archives of Computational Methods in Engineering* 19.2 (2012), pp. 171–225.

[187] K. Takizawa, Y. Bazilevs, T. Tezduyar, C. C. Long, A. L. Marsden, and K. Schjodt. "Patient-Specific Cardiovascular Fluid Mechanics Analysis with the ST and ALE-VMS Methods". In: *Numerical Simulations of Coupled Problems in Engineering.* Ed. by S. Idelsohn. Vol. 33. Computational Methods in Applied Sciences. Springer International Publishing, 2014, pp. 71–102.

[188] K. Takizawa, J. Christopher, T. E. Tezduyar, and S. Sathe. “Space-time finite element computation of arterial fluid-structure interactions with patient-specific data”. In: *International Journal for Numerical Methods in Biomedical Engineering* 26.1 (2010), pp. 101–116.

[189] K. Takizawa, J. Christopher, T. E. Tezduyar, and S. Sathe. “Space-time finite element computation of arterial fluid-structure interactions with patient-specific data”. In: *International Journal for Numerical Methods in Biomedical Engineering* 26.1 (2010), pp. 101–116.

[190] K. Takizawa, C. Moorman, S. Wright, J. Christopher, and T. E. Tezduyar. “Wall shear stress calculations in space-time finite element computation of arterial fluid-structure interactions”. In: *Computational Mechanics* 46.1 (2010), pp. 31–41.

[191] K. Takizawa, H. Takagi, T. E. Tezduyar, and R. Torii. “Estimation of Element-based Zero-stress State for Arterial FSI Computations”. In: *Computational Mechanics* 54.4 (2014), pp. 895–910.

[192] K. Takizawa, R. Torii, H. Takagi, T. E. Tezduyar, and X. Y. Xu. “Coronary arterial dynamics computation with medical-image-based time-dependent anatomical models and element-based zero-stress state estimates”. In: *Computational Mechanics* 54.4 (2014), pp. 1047–1053.

[193] C. A. Taylor, T. J. R. Hughes, and C. K. Zarins. “Finite element modeling of blood flow in arteries”. In: *Computer Methods in Applied Mechanics and Engineering* 158.1 (1998), pp. 155–196.

[194] T. E. Tezduyar, S. Sathe, T. Cragin, B. Nanna, B. S. Conklin, J. Pausewang, and M. Schwaab. “Modelling of fluid-structure interactions with the space-time finite elements: Arterial fluid mechanics”. In: *International Journal for Numerical Methods in Fluids* 54.6-8 (2007), pp. 901–922.

[195] T. E. Tezduyar, S. Sathe, M. Schwaab, and B. S. Conklin. “Arterial fluid mechanics modeling with the stabilized space-time fluid-structure interaction technique”. In: *International Journal for Numerical Methods in Fluids* 57.5 (2008), pp. 601–629.

[196] T. E. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher. “Multiscale sequentially-coupled arterial FSI technique”. In: *Computational Mechanics* 46.1 (2010), pp. 17–29.

[197] The MathWorks, Inc. *MATLAB Documentaion: lsqnonlin - Solve nonlinear least-squares (nonlinear data-fitting) problems.* 2018. URL: `https://de.mathworks.com/help/optim/ug/lsqnonlin.html` (visited on 04/18/2018).

[198] The Open MPI Project. *Open MPI: Open Source High Performance Computing.* 2017. URL: `https://www.open-mpi.org` (visited on 12/20/2017).

[199] M. Thiriet. *Anatomy and physiology of the circulatory and ventilatory systems.* New York and other: Springer, 2014.

[200] M. J. Thubrikar. *Vascular Mechanics and Pathology.* Boston: Springer Science, 2007.

[201] L. H. Timmins, M. W. Miller, F. J. Clubb, and J. E. Moore. “Increased artery wall stress post-stenting leads to greater intimal thickening”. In: *LABORATORY INVESTIGATION* 91 (2011), pp. 955–967.

[202] R. Torii, M. Oshima, T. Kobayashi, and K. Takagi. “The hemodynamic study of the cerebral artery using numerical simulations based on medical imaging data”. In: *Journal of Visualization* 4 (2001), pp. 277–284.

[203] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, and T. E. Tezduyar. “Computer modeling of cardiovascular fluid-structure interactions with the deforming-spatial-domain/stabilized space-time formulation”. In: *Computer Methods in Applied Mechanics and Engineering* 195.13–16 (2006), pp. 1885–1895.

[204] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, and T. E. Tezduyar. “Fluid-structure interaction modeling of a patient-specific cerebral aneurysm: influence of structural modeling”. In: *Computational Mechanics* 43.1 (2008), p. 151.

[205] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, and T. E. Tezduyar. “Influence of wall elasticity in patient-specific hemodynamic simulations”. In: *Computers & Fluids* 36.1 (2007), pp. 160–168.

[206] R. Torii, N. B. Wood, N. Hadjiloizou, A. W. Dowsey, A. R. Wright, A. D. Hughes, J. Davies, D. P. Francis, J. Mayet, G. Z. Yang, S. A. M. G. Thom, and X. Y. Xu. “Fluid-structure interaction analysis of a patient-specific right coronary artery with physiological velocity and pressure waveforms”. In: *Communications in Numerical Methods in Engineering* 25.5 (2009), pp. 565–580.

[207] N. Tsilimparis, T. Hedrich, S. Yousefi, P. Alevizakos, U. Hanack, B. Blank, J. Hagemann, and R. I. Rückert. “Anastomotic engineering bei femorosupragenualen Bypässen – Erste Ergebnisse mit der Dynaflo™-Prothese”. In: *Läsionen der Becken- und Oberschenkelarterien*. Ed. by A. Gussmann and R. I. Rückert. Darmstadt: Steinkopff, 2008, pp. 121–129.

[208] S. Turek and J. Hron. “Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow”. In: *Fluid-Structure Interaction*. Ed. by H. J. Bungartz and M. Schäfer. Berlin and Heidelberg: Springer Berlin Heidelberg, 2006, pp. 371–385.

[209] B. Uekermann, H. J. Bungartz, B. Gatzhammer, and M. Mehl. “A parallel, black-box coupling algorithm for fluid-structure interaction”. In: *Proceedings of the V International Conference on Computational Methods for Coupled Problems in Science and Engineering*. Ed. by S. Idelsohn, M. Papadrakakis, and B. Schrefler. Ibiza, 2013, pp. 1–12.

[210] P. A. Ullrich and M. A. Taylor. “Arbitrary-Order Conservative and Consistent Remapping and a Theory of Linear Maps: Part I”. In: *Monthly Weather Review* 143.6 (2015), pp. 2419–2440.

[211] A. Updegrove, N. M. Wilson, J. Merkow, H. Lan, A. L. Marsden, and S. C. Shadden. “SimVascular: An Open Source Pipeline for Cardiovascular Simulation”. In: *Annals of Biomedical Engineering* 45.3 (2017), pp. 525–541.

[212] A. Valencia, A. Zarate, M. Galvez, and L. Badilla. “Non-Newtonian blood flow dynamics in a right internal carotid artery with a saccular aneurysm”. In: *International Journal for Numerical Methods in Fluids* 50.6 (2006), pp. 751–764.

[213] J. van der Slegt, S. L. Steunenberg, J. M. W. Donker, E. J. Veen, G. H. Ho, H. G. W. de Groot, and L. van der Laan. “The current position of precuffed expanded polytetrafluoroethylene bypass grafts in peripheral vascular surgery”. In: *Journal of Vascular Surgery* 60.1 (2014), pp. 120–128.

[214] J. G. G. Vázquez. “Nonlinear analysis of orthotropic membrane and shell structures including fluid-structure interaction”. Doctoral thesis. Universitat Politècnica de Catalunya, 2007.

[215] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck. “Implicit coupling of partitioned fluid-structure interaction problems with reduced order models”. In: *Computers & Structures* 85.11 (2007), pp. 970–976.

[216] I. E. Vignon-Clementel, C. A. Figueroa, K. E. Jansen, and C. A. Taylor. “Outflow boundary conditions for 3D simulations of non-periodic blood flow and pressure fields in deformable arteries”. In: *Computer Methods in Biomechanics and Biomedical Engineering* 13.5 (2010), pp. 625–640.

[217] I. E. Vignon-Clementel, C. A. Figueroa, K. E. Jansen, and C. A. Taylor. “Outflow boundary conditions for three-dimensional finite element modeling of blood flow and pressure in arteries”. In: *Computer Methods in Applied Mechanics and Engineering* 195.29-32 (2006), pp. 3776–3796.

[218] A. Vlachos, J. Peters, C. Boyd, and J. L. Mitchell. “Curved PN Triangles”. In: *Proceedings of the 2001 Symposium on Interactive 3D Graphics*. New York: Advancing Computing as a Science & Profession, 2001, pp. 159–166.

[219] J. J. Wang and K. H. Parker. “Wave propagation in a model of the arterial circulation”. In: *Journal of Biomechanics* 37.4 (2004), pp. 457–470.

[220] T. Wang, S. Sicklinger, R. Wuchner, and K. U. Bletzinger. “Concept and realization of the coupling software empire in multiphysics co-simulation”. In: *Computational Methods in Marine Engineering*. 2013, pp. 289–298.

[221] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. “A Tensorial Approach to Computational Continuum Mechanics Using Object-oriented Techniques”. In: *Computers in Physics* 12.6 (1998), pp. 620–631.

[222] J. Wen, T. Zheng, W. Jiang, X. Deng, and Y. Fan. “A comparative study of helical-type and traditional-type artery bypass grafts: numerical simulation”. In: *American Society for Artificial Internal Organs Journal* 57.2 (2011), pp. 399–406.

[223] N. Westerhof, F. Bosman, C. J. De Vries, and A. Noordergraaf. “Analog studies of the human systemic arterial tree”. In: *Journal of Biomechanics* 2.2 (1969), pp. 121–143.

[224] G. Westgaard and H. Nowacki. “Construction of Fair Surfaces over Irregular Meshes”. In: *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications*. SMA ’01. New York: ACM, 2001, pp. 88–98.

[225] Wikipedia. *Heart*. 2017. URL: `https://en.wikipedia.org/wiki/Heart` (visited on 11/06/2017).

[226] N. M. Wilson, F. R. Arko, and C. A. Taylor. “Predicting changes in blood flow in patient-specific operative plans for treating aortoiliac occlusive disease”. In: *Computer Aided Surgery* 10.4 (2005), pp. 257–277.

[227] J. R. Womersley. *An elastic tube theory of pulse transmission and oscillatory flow in mammalian arteries*. WADC technical report. Ohio: Wright Air Development Center, 1957.

[228] J. R. Womersley. "Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known". In: *The Journal of Physiology* 127.3 (1955), pp. 553–563.

[229] World Health Organization (WHO). *Fact sheet 317: Cardiovascular diseases*. 2017. URL: `http://www.who.int/mediacentre/factsheets/fs317/en/` (visited on 10/21/2017).

[230] P. Wriggers. *Nonlinear Finite-Element-Methods*. Berlin and Heidelberg: Springer-Verlag, 2008.

[231] R. Wüchner, A. Kupzok, and K. U. Bletzinger. "A framework for stabilized partitioned analysis of thin membrane-wind interaction". In: *International Journal for Numerical Methods in Fluids* 54.6-8 (2007), pp. 945–963.

[232] N. Xiao, J. D. Humphrey, and C. A. Figueroa. "Multi-scale computational model of three-dimensional hemodynamics within a deformable full-body arterial network". In: *Journal of Computational Physics* 244.Supplement C (2013), pp. 22–40.

[233] T. Yamada and S. Yoshimura. "Line Search Partitioned Approach for Fluid-structure Interaction Analysis of Flapping Wing". In: *Computer Modeling in Engineering and Sciences* 24.1 (2008), pp. 51–60.

[234] Z. Yosibash, I. Manor, I. Gilad, and U. Willentz. "Experimental evidence of the compressibility of arteries". In: *Journal of the Mechanical Behavior of Biomedical Materials* 39.Supplement C (2014), pp. 339–354.

[235] Z. Yosibash and E. Priel. "Artery active mechanical response: High order finite element implementation and investigation". In: *Computer Methods in Applied Mechanics and Engineering* 237 (2012), pp. 51–66.

[236] Z. Yosibash and E. Priel. "p-FEMs for hyperelastic anisotropic nearly incompressible materials under finite deformations with applications to arteries simulation". In: *International Journal for Numerical Methods in Engineering* 88.11 (2011), pp. 1152–1174.

[237] C. Zhang, S. Xie, S. Li, F. Pu, X. Deng, Y. Fan, and D. Li. "Flow patterns and wall shear stress distribution in human internal carotid arteries: The geometric effect on the risk for stenoses". In: *Journal of Biomechanics* 45.1 (2012), pp. 83–89.

[238] W. Zhang, C. Herreraie, S. N. Atluri, and G. S. Kassab. "Effect of Surrounding Tissue on Vessel Fluid and Solid Mechanics". In: *Journal of Biomechanical Engineering* 126.6 (2005), pp. 760–769.

[239] O. C. Zienkiewicz and A. Craig. "Adaptive Refinement, Error estimates, multigrid solution, and hierarchic finite element method concepts". In: *Accuracy estimates and adaptive refinements in finite element computations*. Ed. by I. Babuška, O. Zienkiewicz, J. Gago, and E. d. A. Olivera. John Wiley & Sons, 1986, pp. 25–59.

[240] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method – Solid Mechanics*. 5th. Vol. 2. Barcelona: Butterworth-Heinemann, 2000.

[241] Zygote Media Group, Inc. *Zygote Body*. 2017. URL: `http://www.zygotebody.com` (visited on 10/23/2017).

Die Reihen der Fortschritt-Berichte VDI:

1 Konstruktionstechnik/Maschinenelemente
2 Fertigungstechnik
3 Verfahrenstechnik
4 Bauingenieurwesen
5 Grund- und Werkstoffe/Kunststoffe
6 Energietechnik
7 Strömungstechnik
8 Mess-, Steuerungs- und Regelungstechnik
9 Elektronik/Mikro- und Nanotechnik
10 Informatik/Kommunikation
11 Schwingungstechnik
12 Verkehrstechnik/Fahrzeugtechnik
13 Fördertechnik/Logistik
14 Landtechnik/Lebensmitteltechnik
15 Umwelttechnik
16 Technik und Wirtschaft
17 Biotechnik/Medizintechnik
18 Mechanik/Bruchmechanik
19 Wärmetechnik/Kältetechnik
20 Rechnerunterstützte Verfahren (CAD, CAM, CAE CAQ, CIM ...)
21 Elektrotechnik
22 Mensch-Maschine-Systeme
23 Technische Gebäudeausrüstung

ISBN 978-3-18-3**35318**-7