A 3D method for modelling the fluid-structure interaction of heart valves
van Loon, R.

DOI:
10.6100/IR594604

Published: 01/01/2005

Document Version
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
A 3D method for modelling the fluid-structure interaction of heart valves
CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Loon, Raoul van

NUR 954
Subject headings: finite element modelling / computational fluid dynamics / solid mechanics / numerical modelling / cardiovascular mechanics / heart valves / fluid-structure interaction / fictitious domains / adaptive meshing / Arbitrary-Lagrangian-Eulerian / rigid-solid contact / Lagrange multipliers /

Copyright ©2005 by R. van Loon

All rights reserved. No part of this book may be reproduced, stored in a database or retrieval system, or published, in any form or in any way, electronically, mechanically, by print, photoprint, microfilm or any other means without prior written permission of the author.

Cover design: Raoul van Loon/ Jan-Willem Luiten (JWL Producties).

A 3D method for modelling the fluid-structure interaction of heart valves

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de Rector Magnificus, prof.dr.ir. C.J. van Duijn, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op woensdag 31 augustus 2005 om 16.00 uur

door

Raoul van Loon

geboren te Munstergeleen
Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. F.N. van de Vosse
en
prof.dr.ir. F.P.T. Baaijens

Copromotor:
dr.ir. P.D. Anderson
Contents

Summary ix

1 Introduction 1
  1.1 Native heart valves 2
  1.2 Heart valve disease 3
  1.3 Heart valve replacement 4
  1.4 Added value of computational methods for clinical practice 5
  1.5 Modelling fluid-structure interaction (FSI) 6
    1.5.1 FSI methods 6
    1.5.2 Heart valve models 9
  1.6 Objective and Outline of thesis 10

2 A combined 2D fictitious domain/adaptive meshing method 13
  2.1 Introduction 14
  2.2 Methods 15
    2.2.1 Governing equations 15
    2.2.2 Coupling 16
    2.2.3 Mesh adaptation 17
    2.2.4 Solution process 18
  2.3 Results 19
    2.3.1 Model problem 1: solid slab in pulsatile flow 19
    2.3.2 Model problem 2: fluid pressure drop across solid membrane 22
  2.4 Discussion 24

3 A fluid-structure interaction method with solid-rigid contact 27
  3.1 Introduction 28
  3.2 Governing equations 30
    3.2.1 Fluid and solid 30
    3.2.2 Fluid-structure coupling 30
    3.2.3 Contact 30
  3.3 Weak formulation 32
  3.4 Discretisation 33
    3.4.1 Discretisation of fluid and solid 33
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4.2</td>
<td>Discretisation of the Lagrange multiplier for fluid-structure interaction</td>
<td>38</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Discretisation of the Lagrange multiplier for solid-rigid contact</td>
<td>39</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Full set of discretised equations</td>
<td>39</td>
</tr>
<tr>
<td>3.4.5</td>
<td>Combined fluid-structure interaction/contact</td>
<td>40</td>
</tr>
<tr>
<td>3.4.6</td>
<td>Combined ALE and mapping</td>
<td>40</td>
</tr>
<tr>
<td>3.5</td>
<td>Numerical experiments</td>
<td>42</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Contact of solid slab in pulsatile flow</td>
<td>42</td>
</tr>
<tr>
<td>3.5.2</td>
<td>2D mitral valve</td>
<td>46</td>
</tr>
<tr>
<td>3.6</td>
<td>Conclusion</td>
<td>48</td>
</tr>
<tr>
<td>4</td>
<td>A 3D method for fluid-structure interaction in heart valves</td>
<td>51</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>52</td>
</tr>
<tr>
<td>4.2</td>
<td>Mathematical formulation</td>
<td>53</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Fluid</td>
<td>53</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Solid</td>
<td>53</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Fluid-structure coupling</td>
<td>54</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Discretisation</td>
<td>54</td>
</tr>
<tr>
<td>4.2.5</td>
<td>Adaptive meshing</td>
<td>55</td>
</tr>
<tr>
<td>4.2.6</td>
<td>Element choice</td>
<td>58</td>
</tr>
<tr>
<td>4.3</td>
<td>Results</td>
<td>59</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Model problem 1: movement of a solid slab</td>
<td>59</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Model problem 2: shear stresses along a rigid plate</td>
<td>59</td>
</tr>
<tr>
<td>4.3.3</td>
<td>Model problem 3: pressure drop in fluid domain (axisymmetric)</td>
<td>61</td>
</tr>
<tr>
<td>4.3.4</td>
<td>Model problem 4: pressure drop in fluid domain (3D)</td>
<td>61</td>
</tr>
<tr>
<td>4.4</td>
<td>Conclusion</td>
<td>64</td>
</tr>
<tr>
<td>5</td>
<td>3D dynamical analysis of an aortic heart valve in pulsatile flow</td>
<td>65</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>66</td>
</tr>
<tr>
<td>5.2</td>
<td>Computational solid model for the valve</td>
<td>67</td>
</tr>
<tr>
<td>5.2.1</td>
<td>An aortic heart valve mesh</td>
<td>67</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Solid model description</td>
<td>69</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Solid model results</td>
<td>70</td>
</tr>
<tr>
<td>5.3</td>
<td>Computational fluid-structure interaction model</td>
<td>71</td>
</tr>
<tr>
<td>5.3.1</td>
<td>FSI model description</td>
<td>71</td>
</tr>
<tr>
<td>5.3.2</td>
<td>FSI model results</td>
<td>72</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Discussion: some computational difficulties</td>
<td>74</td>
</tr>
<tr>
<td>5.4</td>
<td>Conclusion</td>
<td>75</td>
</tr>
<tr>
<td>6</td>
<td>Conclusions and Discussion</td>
<td>77</td>
</tr>
<tr>
<td>6.1</td>
<td>Introductory remarks</td>
<td>78</td>
</tr>
<tr>
<td>6.2</td>
<td>The fluid-structure interaction method</td>
<td>78</td>
</tr>
<tr>
<td>6.3</td>
<td>Recommendations</td>
<td>80</td>
</tr>
<tr>
<td>6.4</td>
<td>Model applications</td>
<td>81</td>
</tr>
</tbody>
</table>
A Linearisation

A.1 Fluid ................................................................. 83
A.2 Solid ................................................................. 84
   A.2.1 Linearisation of solid equations ......................... 85
   A.2.2 Remark on tensor $H$ in axisymmetry ................. 87
   A.2.3 Remark on the deformation gradient tensor for axisymmetry . 88
   A.2.4 Matrix notation for stiffness matrix .................. 89
A.3 Coupling solid and fluid ..................................... 91
A.4 Rigid-solid contact ............................................. 92

B Meshing

B.1 Robustness of meshing scheme ............................... 95
B.2 Intersections ..................................................... 96
B.3 Smoothing ......................................................... 97
   B.3.1 Laplacian smoothing .................................. 98
   B.3.2 Angle optimisation .................................. 98

C Arbitrary Lagrangian Eulerian ................................. 101

References ......................................................... 104

Samenvatting ....................................................... 111

Dankwoord .......................................................... 113

Curriculum Vitae .................................................. 115
Contents
Summary

In this study a numerical method is developed to enable the computation of physiologically realistic opening and closing behaviour of a heart valve with the associated mechanical loading. These computations could contribute to a better understanding of the healthy native heart valve function. In addition, the calculations might be applied for the analysis of ill-functioning valves. Finally, they could provide insight in valve dynamics of tissue engineered valves. The valve dynamics is determined by the interaction between the blood flow and the elastic deformation of the valve leaflets. Therefore, this study focusses on the fluid-structure interaction between valve and blood.

Using a finite element method, meshes are generated for the fluid and solid domain. A Lagrangian description is used for the solid with a hyperelastic Neo-Hookean model, while the fluid is described in an Eulerian manner by the unsteady Navier-Stokes equation. These meshes are overlapping but non-conforming, but velocities are coupled along the boundary of the solid by means of a Lagrange multiplier. Due to the non-conformity of the domains, interpolation of the velocities are required to apply the coupling, which potentially introduces loss of accuracy. To solve this problem a computationally inexpensive adaptive meshing scheme was developed that acts locally and provides a continuous update of the fluid mesh to the position of the solid each numerical time step. The result is an inner-fluid boundary that coincides with the solid boundary, but is still non-conforming. Therefore, only interpolation in the tangential direction of the boundary is required which, in combination with the properties of the elements, enables the computation of moving pressure discontinuities within the fluid. Additionally it allows for the shear stresses to be calculated along both sides of the valve. The method is combined with a solid-rigid contact algorithm in order to compute the interaction of the coapting leaflets. The result is a method that can provide insight in fluid dynamics, solid dynamics and solid mechanics during one or more cardiac cycles.

Summarising, this thesis discusses the development of an FSI method that is capable of computing large motions of a flexible solid in a fluid domain induced by an unsteady fluid flow. In this respect emphasis is on the accuracy of shear stresses and closing or closed behaviour of a valve. The method is first developed in two dimensions and extended to three dimensions. Finally, an application of the method to the aortic heart valve is presented.
Summary
Chapter 1

Introduction

**Summary:** The “hemodyn” project is a multi-disciplinary collaboration that strives for better understanding of cardiovascular applications. This work is part of the “hemodyn” initiative and will focus on the interaction between blood and tissue which plays an important role throughout the whole cardio-vascular system. The aortic root is chosen as a challenging area to analyse due to its complicated nature and, therefore, in this chapter first a general introduction will be given on heart valves, their function, associated diseases and replacements. The role of computational models is explained next and current fluid-structure interaction techniques with corresponding computational heart valve models are discussed, to provide a basis for the objective of this work.
This thesis falls within the framework of the “hemodyn” project that aims for a better understanding of cardiovascular pathologies, applications and interventions in order to enhance cardiovascular devices and decision making by simulation based analysis. By multidisciplinary research in the fields of animal experiments, monitoring patients, MR imaging/registration of patient specific data and scientific computing, a bridge between clinic and engineering should be established. One important aspect for a good understanding of the dynamical/mechanical behaviour in the cardiovascular system is the interaction between blood and the circumventing tissues. It is generally known that the blood pressure and blood flow influence the arterial or heart tissue behaviour and vice versa. One way to approach this continuous play between solid and fluid is computational modelling and one of the most challenging fluid-structure interaction problems in the human body are heart valves. A heart valve model system incorporates all different phenomena requiring computational techniques which subsequently can be used in other areas of the cardiovascular tree.

1.1 Native heart valves

There are four heart valves present in the human heart, the mitral valve, tricuspid valve, pulmonary valve and aortic valve. The former two are the atrioventricular valves and situated at the transition between left atrium to left ventricle and right atrium to right ventricle, respectively (Fig. 1.1).

Figure 1.1: The position of the valves in the heart. (A.D.A.M., inc.)

The latter two are the semilunar valves situated at the transition from the right ventricle to the pulmonary artery and from the left ventricle to the aorta, respectively. The mitral valve is the only bileaflet valve, the others are trileaflet valves. The general function of all these valves is to prevent retrograde blood flow or regurgitation. In order to provide good heart efficiency the valves should exert minor resistance to the flow in the opening phase and therefore the leaflets are extremely thin. Their
geometry and morphology are such that the leaflets have sufficient tangential stiffness to withstand the pressures to which they are exposed in the closed situation, although the mitral and tricuspid valve additionally need cords (chordae tendineae) to prevent the valves from prolapsing. Fig. 1.2 shows the pressure transients in time in the left atrium, left ventricle and aorta with indications for the opening and closing of the mitral and aortic valve.

![Diagram showing pressure transients in the left ventricle, left atrium, and aorta](image)

**Figure 1.2:** The pressure transients in the left ventricle, left atrium and aorta during one cardiac cycle. Closure and opening of the aortic and mitral valve are shown at the top. Also given is the flow curve in the aorta.

From this figure it can be seen that the valves are opened due to a fast increase and decrease of the pressure in the left ventricle, never establishing an open connection between the atrium and aorta, resulting in an efficient pump function.

### 1.2 Heart valve disease

There are several diseases which can cause malfunctioning of a heart valves, the most common ones being, rheumatic valve disease, infective endocarditis, myxomatous valves calcific degeneration and congenital anomalies [Olsen et al., 1984; Thubrikar, 1990], which are briefly addressed in the following. The immunologic complications of rheumatic fever can cause rheumatic heart valves, i.e. an inflammatory condition, induced by a bacterial infection can cause an auto-immune response that damages healthy tissue. Infective endocarditis is normally provoked by micro-organisms like bacteria or fungi, and also may lead to an inflammatory reaction. The infection can cause other complications, such as thrombotic vegetation development on the surfaces.
of the valve, an irregular heart beat (arrhythmia), valve damage or even destruction. Myxomatous valves or floppy valves have a connective tissue disorder that causes the heart valve tissue to weaken and lose elasticity. This condition leads to ballooning or prolapsing of the valves increasing the risk of regurgitation, chordal rupture and thrombosis. Most adults with aortic valve stenosis have a degenerative-calcific process that produces immobilisation of the valve cusps, while adolescents and young adults with isolated aortic valve stenosis almost always have congenital fusion of one or more commissures resulting in a bicuspid or unicuspidx valve. In general, one can conclude that each of these disorders can lead to severe stenosis or regurgitation, in either case clinical intervention is needed. In an end-stage valvular disease, a valve replacement is required, which will be briefly addressed next.

1.3 Heart valve replacement

When the valve has an end-stage disease and replacement is required, the native valve is surgically replaced by a mechanical or a bioprosthetic heart valve, each having its advantages and disadvantages. Over 250,000 heart valve replacements are performed worldwide each year and of all the implanted prosthetic valves about 55% are mechanical, while almost 45% are biological [Butany et al., 2003].

A mechanical valve usually consists of one or two carbon leaflets with a pyrolitic coating that are connected to a stiff aortic ring in which they can pivot, as shown in Fig. 1.3(a). Although mechanical prostheses have the advantage of being very durable, they are also prone to thrombus formations, that serve as excellent substrates for bacterial infection or that subsequently can develop into an embolism. Therefore patients are bound to a life-long anticoagulation treatment.

Bioprosthetic valves have either an animal (xenografts) or human (homografts) origin. Xenografts are made from porcine aortic valve tissue or bovine pericardial tissue. Homografts are taken from human donors. These valves are referred to as homografts.

![Heart Valve Replacements](image1.png)

_Figure 1.3: Different heart valve replacements: (a) Mechanical valve St. Jude Medical; (b) Hancock porcine bioprosthesis; (c) Tissue engineered heart valve [Mol, 2005]_
(see Fig. 1.3(b)). The function and geometry of both of these grafts resemble that of native aortic valves resulting in a higher haemodynamical performance than the mechanical valves. The flow is, however, still not completely physiological since mechanical properties are affected by the pre-treatment. The major drawback is the limited life-span with an average of 10 to 15 years, which makes these valves suitable for the elderly, but less attractive for the younger.

Special attention should be given to a developing field of research known as tissue engineering. This branch of biomedical science focusses on the generation of living tissue in bioreactors and quite some work has already been done with respect to heart valves [Schoen and Levy, 1999; Mol and Hoerstrup, 2004]. The general idea is to seed living cells on a scaffold and stimulate proliferation and matrix synthesis in vitro until a living construct is obtained that is ready for implantation. In vivo, the tissue engineered heart valve should be capable of remodelling its structure to the physiological situation or, in case of implant in children, to even grow, holding a large advantage with regard to current prostheses. Although this field of research has not yet reached a mature state, results look promising [Hoerstrup et al., 2000] (see Fig. 1.3(c)).

1.4 Added value of computational methods for clinical practice

Mechanical stimuli such as pressure, strain or shear are important with respect to damage or even rupture of tissue but are at a smaller level also involved in the onset of many biological processes. Endothelial cells, for instance, line the entire cardiovascular system including the heart valves, and are very sensitive to (changes in) shear stress, which influences their regulatory functions. These cells have been linked to the pathogenesis of atherosclerosis, hypertension and thrombosis [Leask et al., 2003]. Knowledge about the fluid dynamics in the aortic root, with special attention to shear stresses along the leaflets, might therefore give new insights. Since it is difficult to obtain this knowledge in vivo or in vitro due to the complex three dimensional shape and the high resolutions that are required, computational methods are useful to provide this information. Numerical analysis of wall or leaflet deformations in the situations before or after intervention might be another application to provide clinicians an extra handle for decision making [Stijnen, 2004; De Hart et al., 2003b]. With imaging as the bridge between the clinical practice and numerical modelling, patient specific data can be used in computations for improved judgment of a situation. In tissue engineering mechanical models may also be used to predict and control tissue and cell behaviour like e.g. Driessen et al. [2003], resulting in more mechanically and morphologically realistic tissue engineered constructs. Finally, it should be mentioned that the Food and Drug Administration (FDA) has acknowledged the importance of computational models and as a consequence numerical modelling is becoming a substantial part in the developing process of new prosthetic heart valves.
1.5 Modelling fluid-structure interaction (FSI)

There are many different scientific and industrial areas in which interaction between different substances plays an important role, like airflow along an aeroplane [Farhat et al., 2003], helicopter [Behr and Tezduyar, 2001], or parachute [Stein et al., 2000], the process of mixing [Bertrand et al., 1997; Hwang et al., 2005], particle flow or particle sedimentation [Glowinski et al., 1999a,b; Hwang et al., 2004], deformation of arteries [Zhao et al., 2000], the heart [Peskin and McQueen, 1989] or heart valves [Peskin, 1972; Makhijani et al., 1997; De Hart et al., 2003b], swimming mackerel [Gilmanov and Sotiropoulos, 2005] and waving motions of cilia [Dillon and Fauci, 2003a], which are only a number of examples from totally different fields. Although there are large differences between fluid-fluid, fluid-solid, solid-solid and gas-solid problems, they all treat interacting boundaries possibly making (parts of) the developed methods interchangeable. Therefore, a short overview will be given to provide insight in the ways to approach FSI problems.

1.5.1 FSI methods

In a finite element, finite difference or finite volume setting, the fluid domain is ideally described in an Eulerian frame of reference obtaining solutions in grid/mesh points that are fixed in space. Although this works well for computational fluid dynamics problems with fluid flow through rigid domains, difficulties arise when moving interfaces are present inside the domain or when the boundaries of the domain move. A typical example of a moving interface are two interaction fluids and typical examples of moving boundaries are free surface flows or compliant arterial walls. Many alternatives have been investigated for capturing interfacial motion in an Eulerian context, generally subdivided into boundary-fitted methods and non-boundary-fitted methods (Fig. 1.4).

![Schematic representation of boundary-fitted (b)-(c), and non-boundary-fitted (d) methods. Starting from (a) rotation of the gray body leads to (b), (c) or (d) depending on the method used.](image-url)
Boundary-fitted methods are distinguished from the non-boundary-fitted methods in that the Eulerian mesh is adapted to the moving interface in time. This adaptation is performed by an Arbitrary Lagrange Eulerian method (ALE) (Fig. 1.4(b)) or by remeshing (Fig. 1.4(c)). An Arbitrary Lagrangian Eulerian method enables translations of grid/mesh points in a Lagrangian way by taking the convection of these points into account as described in Appendix C [Hirt et al., 1974; Donea et al., 1982; Hughes et al., 1981]. Since the method is easy to implement, has low computational cost and is accurate, it is recommendable to use it if possible. However, for large translations, rotations or inhomogeneous movements of the grid/mesh points fluid elements tend to become ill-shaped, which reflects on the accuracy of the solution. Remeshing, in which the whole domain or part of the domain is spatially re-discretised, is then the common strategy. The process of mesh generation multiple times during a computation can, however, be a very troublesome and time consuming task. Furthermore, the transfer of solutions from the degenerated mesh to the new mesh may introduce artificial diffusion, causing loss of information.

As mentioned before, interfering with the Eulerian mesh can cause difficulties and therefore several methods have been developed, for which the fluid mesh remains unaltered throughout the computation. The application areas for these methods can roughly be grouped into multicomponent fluid flows, fluid/rigid-body problems and fluid/solid-body problems. It should be noted that, in this work the phrase fluid-structure interaction (FSI) will be used for the interaction of a rigid body or an elastic solid body with a fluid.

For a comprehensive overview containing large numbers of references with regard to free surface/interface fluid flows we refer to Floryan and Rasmussen [1989], Hou [1995] and Scardovelli and Zaleski [1999]. Typically used for capturing fluid-fluid interaction are the level-set method [Sussman et al., 1994; Sethian, 1996; Smolianski, 2005], front tracking [Glimm et al., 1986; Galaktionov et al., 2000], volume-of-fluid [Scardovelli and Zaleski, 1999; Sussman, 2003], diffuse interface method [Lowengrub and Truskinovsky, 1998; Verschueren et al., 2001; Keestra et al., 2003] and boundary integral method [Pozrikidis, 1992; Bazhlekov et al., 2004], all trying to capture the interface as accurate as possible without loss of mass. All methods can be subdivided into passive interfaces, that do not influence the flow and active interfaces, in which surface tension of the interface is incorporated. Another method that should be mentioned is eXtended FEM (XFEM) or Partition of Unity [Möes et al., 1999; Belytschko et al., 2001; Dolbow et al., 2001], which originates from the area of cracks and crack propagation for capturing discontinuities, but has also been introduced for fluids by Chessa and Belytschko [2003]. Although the methods used for multiple fluid flows will not be treated extensively here, one should be conscious of the similarities with FSI problems.

A widely used non-boundary fitted method for FSI applications is the Immersed Boundary Method, which was proposed by Peskin [Peskin, 1972, 2002]. The first models considered a finite difference grid for the fluid domain with an immersed set of non-conforming boundary points, that were mutually interconnected by some elastic law. This solid boundary interacted with the fluid by means of local body forces applied to the fluid at the position of the points. A representation is shown
in Fig. 1.4(a) and 1.4(d), with the solid points denoted by white circles and the fluid nodes by black dots. Fig. 1.4(a) shows two meshed domains with a coinciding interface, which is being rotated. Fig. 1.4(d) shows the rotation of the interface without altering the position of any of the fluid nodes. The velocity of each solid point is coupled to the interpolated fluid velocity at that point. The introduction of these body forces has become the basic idea behind several non-fitted-boundary FSI methods. Throughout the years, the Immersed Boundary Method has been successfully applied in many application fields [Peskin and McQueen, 1989; Dillon and Fauci, 2000a; Zhu and Peskin, 2002; Gilmanov and Sotiropoulos, 2005].

An interesting non-boundary-fitting method, that has some resemblance with the Immersed Boundary Method, is the so-called Fictitious Domain method [Glowinski et al., 1997]. The Immersed Boundary Method has been developed and applied within a finite difference framework, the Fictitious Domain method evolved from the field of finite elements. By constraining fluid and rigid body at the interface using a (distributed) Lagrange multiplier and extending this constraint to the inner body, coupling is established. In the strong form the Fictitious Domain method is not different from that of the Immersed Boundary Method method, only by applying the multipliers (which are in fact the body forces) in a Finite Element setting the forces are imposed 'weakly'. The method has been used successfully for modelling flow along particles describing their translations and rotations [Glowinski et al., 1999b; Patankar et al., 2000], the sedimentation of particles [Singh et al., 2000], particles in a shear flow [Hwang et al., 2004] or mixing blades of a kemix mixer [Bertrand et al., 1997].

As an extension of the Fictitious Domain methods Baaijens et al. proposed a fluid-solid interaction version suitable for slender bodies [Baaijens, 2001]. A fluid and solid mesh were generated independently from each other and both domains were coupled by means of a Lagrange multiplier along the boundary of the solid. The solid was described in a Lagrangian way and, therefore, deformed by the acting fluid forces, while the Eulerian fluid mesh did not require updating. This method was successfully applied in mechanical and flexible heart valve simulations by Stijnen et al. [2004] and De Hart et al. [2000, 2003b], respectively. Recently, coupling of the whole solid domain to the fluid instead of the boundary was published [Yu, 2005].

As shown, the number of methods for modelling interfacial interaction problems is large and there is not one single method that is applicable to every problem. Before choosing a method one should, therefore, firstly focus on the typical features of the problem like “Are there any discontinuities in the field variables?”, “Is only global movement of the interface important or should the local solution be very accurate?” “Are accurate approximations of stresses and strains required?” “Would 2D modelling suffice or is 3D modelling needed?”, “Should the solution be available in short time or can computations take days?”. The answers to these questions determine what method to choose for modelling. For each of the above methods, different solving strategies can be applied, which largely determine the complexity, accuracy, efficiency and robustness.

The separate equations for fluid and solid can be coupled directly (coinciding nodes for fluid and solid), by means of a penalty function or by introducing a Lagrange multiplier. The final set of equations can be solved uncoupled, weakly coupled or
fully coupled. Furthermore, a time-integration scheme should be chosen and many fractional step or operator splitting schemes are available that can be applied to facilitate the solution process. Therefore, with regard to these aspects a trade-off must be made between complexity, efficiency and accuracy.

1.5.2 Heart valve models

Based on the overview of the available methods to model FSI behaviour in computational fluid dynamics, a short survey will be given of the different heart valve models that have been developed over the years. These models can be subdivided into two groups: mechanical heart valve models and flexible heart valve models. The majority of published heart valve models discuss the fluid motion near monoleaflet or bileaflet mechanical heart valves in a steady flow.

If one is interested in the mechanical behaviour of a valve in closed position, it is not necessary to incorporate the interaction of the leaflets with the fluid dynamics. Black et al. was amongst the first to model a 3D geometry of an aortic heart valve and analyse the stress-strain distribution in the leaflets after applying pressure [Black et al., 1991]. A closer look at the morphology of a native valve shows that the collagen fiber reinforcement of the valves has a large influence on the mechanical behaviour of the leaflets [De Hart et al., 1998]. In a tissue engineering context adaptive models have been developed to predict the density and orientation of the fibers triggered by principal strains in the cusps of an aortic valve [Driessen et al., 2003; Boerboom et al., 2003]. Valve closure, valve motion and the corresponding fluid dynamics are the three

![Figure 1.5: Numerical heart valve modelling.](image)
main points of interest in dynamical heart valve analysis. When the analysis focuses
only on the fluid dynamics, leaflet movement can be prescribed from experimental
data [Makhijani et al., 2005; Krafczyk et al., 1998; Aluri and Chandran, 2001], but
when the focus also lies on valve motion then coupled FSI algorithms should be
used. For simplicity reasons the boundary-fitted weakly coupled methods are most
popular in the latter case [Makhijani et al., 2005; King et al., 1996; Lai et al., 2002;
Cheng et al., 2004; Dumont, 2004; Hose et al., 2005]. In these methods the fluid and
solid computations are solved separately, each computation generating the dynamical
or kinematical boundary conditions for the other computation. Another approach
has been proposed [Shi et al., 2003], where the fluid and solid mesh are generated
independently and every time step the overlapping part of the fluid and solid mesh
are removed from the fluid mesh and boundary conditions are applied at the generated
inner boundary. In addition, the method proposed in [Baaijens, 2001] was sucessfully
applied to analyse heart efficiency including the interaction of two mechanical valves
[Stijnen, 2004].

Few publications discuss the complex interaction of a flexible leaflet in a fluid,
although the first contribution in this field was already made in 1972 [Peskin, 1972].
As described before, Peskin succeeded to capture the movement of two flexible solids
in a two dimensional fluid domain using his Immersed Boundary Method. The
Immersed Boundary Method was presented in a finite difference context and as it
developed throughout the years it proved to be a suitable method for computing the
interaction between flexible bodies and fluid in the cardiovascular system [Peskin
and McQueen, 1980, 1989; McQueen and Peskin, 1997]. Many years later Makhijani
et al. presented their three dimensional model of a bioprosthetic aortic valve in a
finite volume setting [Makhijani et al., 1997]. Fluid and solid domain were coupled
by means of an implicit “influence coefficient”. A numerical validation of Baaijens’
method with finite elements was proposed by De Hart et al. [2000] and extended to a
three dimensional model of an aortic heart valve with compliant root [De Hart et al.,
2003b,a]. The importance of the collagen fiber network, present in the native aortic
valve, was anaalysed in a simulation using the same FSI technique [De Hart et al.,
2004]. Finally, a recent work should be mentioned that follows a trend that can be
observed from various other fields in biomedical engineering namely, patient specific
modelling. Nicosia et al. [2003] managed to compute a coupled dynamical analysis of
a patient specific aortic valve with root, including the closing behaviour of the valve.
Although the constitutive behaviour of the leaflets was still simplistic, this model well
indicates direction in which heart valve modelling will proceed in the future.

1.6 Objective and Outline of thesis

As can be deduced from the models developed in the past, there are quite some
complexities in modelling the FSI of a heart valve, which makes it a challenging
problem. The leaflets are very thin and, therefore, have very low bending stiffness,
allowing to open without hampering the flow too much. The collagen network on the
other hand provides for the necessary stiffness in tangential direction in case the valve
is closed and sustains a pressure. Modelling the FSI of such a body in a pulsatile flow at the physiologically high Reynolds numbers, does not contribute to the numerical stability, especially, when valve closure needs to be incorporated using a contact algorithm. In addition, the valve is attached to compliant walls, that influence leaflet motion and, therefore, it need to be modelled as well as the influence of the arterial system, which should be incorporated in the boundary conditions. Current models focus mainly on the motion of the valve and root and the corresponding global fluid dynamics, while to our knowledge no models yet exist that can sustain physiological transvalvular pressures and can provide accurate shear stress information along the leaflet surfaces.

Therefore, the objective of this work is to develop a method that is capable of describing the motion of a flexible heart valve in a fluid domain throughout a full cardiac cycle with sufficient accuracy within the fluid domain and to obtain quantitative shear stress information along both sides of the valve.

The outline of this work is as follows: In Chapter 2 a two-dimensional method is presented, founded on the fictitious domain method. This choice was made since from literature the method proved to be very flexible and give good results with respect to valve motion. Due to interpolation the accuracy of the fluid solution near the leaflets is low, which can cause severe numerical instabilities. Therefore, the method is extended with a simple adaptive meshing scheme to provide the necessary accuracy. The advantages of the method are shown by means of two numerical experiments in which shear stresses along the solid and a pressure drop across the solid are computed.

The numerical experiments in Chapter 2 can be considered to represent the systolic and diastolic part of a cardiac cycle and in order to combine these two experiments, a rigid-solid contact algorithm is incorporated into the method in Chapter 3. This algorithm should account for correct closing and closed behaviour of a valve allowing the solid to make contact with and slide along the boundary of the fluid domain and therefore introducing a moving pressure discontinuity through the domain. Furthermore, the adaptive meshing procedure is improved in this chapter, which makes the algorithm more robust especially with regard to its extension to 3D. The method is again illustrated with some numerical model problems, one problem combining those presented in Chapter 2 and one 2D representation of a mitral valve, both problems dealing with closure and opening.

Chapter 4 treats the straightforward extension of the methods proposed in Chapters 2 and 3 to 3D. There are no changes with respect to the strong and weak form of the governing equations. The difference should be sought in the discretisation and therefore in the adaptive meshing scheme which is now applied to a tetrahedral mesh instead of a triangular mesh. As in Chapters 2 and 3 some numerical experiments are performed, which subsequently treat the valve motion (plane strain), shear stresses along the solid (plane strain) and pressure drop across the solid (axi-symmetric and full 3D). The results of the last two problems were successfully compared.

Finally, the method developed is used to model the problem of a 3D aortic heart
valve that interacts fully coupled with a fluid domain in Chapter 5. The geometry of the valve is parameterised based on the descriptions by Thubrikar [1990]. This chapter gives a preliminary view on the application of the method in a heart valve model during opening and closure and discusses the difficulties still encountered.

In Chapter 6, some conclusions are drawn and recommendations are made with regard to the method developed and its application to heart valve modelling.

One should note that Chapters 2, 3 and 4 are based on the articles Van Loon et al. [2004], Van Loon et al. [2005b] and Van Loon et al. [2005a], respectively. Therefore it is possible that parts of these chapters show some overlap. This contributes, however, to the readability of the separate chapters.
Chapter 2

A combined fictitious domain/adaptive meshing method for fluid-structure interaction in heart valves

Summary: A new approach for modelling the fluid-structure interaction of flexible heart valves is proposed. Using a finite element method, a Lagrangian description of a non-linear solid and an Eulerian description of a fluid are coupled by a Lagrange multiplier. This multiplier allows the solid and fluid mesh to be non-conforming. To improve the accuracy with regard to pressures and shear stresses in the vicinity of the solid an inexpensive mesh-adaptation algorithm is applied, that adapts the fluid mesh to the position of the solid mesh every time step. This minor adjustment of the fluid mesh makes it possible to sustain a physiological pressure gradient across a solid leaflet. Furthermore, shear stresses can be computed at both sides of the leaflet. The method is demonstrated for a 2D example, however with a scope to 3D modelling.

2.1 Introduction

Computational methods can be of great help in understanding heart valve pathologies. The behaviour of the valves (mitral or aortic, mechanical or biological), which should cause no resistance during systole, but need to sustain large pressure gradients during diastole, is however not easy to capture. A variety of models have been proposed throughout the years in order to get better insight in the mechanical behaviour of heart valves. Application of these models can be distinguished into mechanical heart valve and biological (or polymeric) heart valve analysis.

The stiff mechanical valves will not change shape and hence the valve can be modelled as a rigid body if stresses and strains are not of interest. In an attempt to get information about the flow around a mechanical valve, a canal with rigid walls was considered with a rigid obstacle inhibiting the flow [King et al., 1997]. The angle of the obstacle can then easily be changed. However, the transient behaviour of valve and flow is not captured by such models. An ALE method combined with remeshing as used by Lai et al. [2002] or a marker method as proposed by Shi et al. [2003], overcome this problem and are capable of capturing transient behaviour.

Compared to the mechanical heart valve, modelling of a flexible heart valve is geometrically and numerically, more complex. The bending stiffness of the leaflets is very low compared to the tensile strength along the leaflet surface. This flexibility results in large geometrical changes of the leaflets when exposed to a pulsatile flow. Despite this flexibility the valve influences the flow greatly since it is attached to the wall and is impermeable. During diastolic phase the valve has to sustain a physiological transvalvular pressure gradient. It is obvious that a numerical code needs to be able to describe all phenomena during a heart cycle.

Compared to mechanical heart valves one is often interested not only in the flow pattern around the leaflets but also the strains and stresses inside the leaflets. A modelling approach was to apply a pressure force onto the leaflets disregarding the fluid [Black et al., 1991; De Hart et al., 1998]. These models show that the interaction between the blood flow and the valve leaflets can not be neglected. After that, two dimensional models emerged with a two dimensional fluid domain coupled to a solid leaflet. The ALE algorithm is often used as a way of coupling an Eulerian description of the fluid domain to a Lagrangian description of the solid domain. This method is combined with remeshing whenever the quality of the fluid mesh degenerates [Makhijani et al., 1997]. Since large rotations of the leaflets within the fluid domain are considered here, this combination of ALE and remeshing is obligatory. When meshes become larger and more complex, computing a new conforming mesh, as is required when using an ALE method, can be a difficult and time consuming task.

Another technique for modelling fluid-structure problems, called fictitious domain (FD), was used by Glowinski et al. [1997, 1999a,b] and Bertrand et al. [1997]. A Lagrange multiplier is used to couple an Eulerian fluid mesh to rigid solids. This Lagrange multiplier allows the meshes to be non-conforming and, more important, no remeshing is necessary. A differentiation of the FD method, in which slender Lagrangian solids are coupled to the Eulerian fluid domain, was introduced by Baaijens [2001].
This method was first applied in a two-dimensional fluid-leaflet interaction model by De Hart et al. [2000]. An extension to 3D was published recently [De Hart et al., 2003b,a]. In these models, the compliant solid walls are described using an ALE method while an FD method is used for the leaflets.

However, in general, FD methods require an interpolation to the immersed boundary. As a result, these methods do not allow for highly accurate descriptions of gradients in velocity field and pressure discontinuities across the immersed boundaries. Hence, in heart valves, where shear and substantial diastolic pressure gradients along the leaflets play an important role in their functioning, the application of FD solely to describe the interaction may not be sufficient.

The goal of this chapter is to develop a method that is able to accurately capture stresses along the leaflet boundary. Furthermore, physiological pressure gradients across a leaflet should be computed correctly. The presented method is an extension of the FD method [De Hart et al., 2000; Baaijens, 2001], with an inexpensive adaptive meshing technique. By creating an inner fluid curve, that coincides with the solid boundary, only interpolation along (and not across) the boundary is needed, resulting in more accurate solutions. Typical model problems are presented to show the method’s ability of describing transvalvular shear stress and pressure discontinuities.

2.2 Methods

2.2.1 Governing equations

Throughout the whole chapter fluid-structure problems are considered in which the fluid is described by the Navier-Stokes equation and the continuity equation,

\[
\rho \left( \frac{du^f}{dt} + u^f \cdot \nabla u^f \right) = \nabla \cdot \tau^f - \nabla p^f, \tag{2.1}
\]

\[
\nabla \cdot u^f = 0, \tag{2.2}
\]

in which \( u^f \) is the fluid velocity, \( \nabla \) the gradient operator, \( \rho \) the density and \( p^f \) the hydrostatic pressure in the fluid. The viscous part of the Cauchy stress tensor denoted as \( \tau^f \) reads,

\[
\tau^f = 2\eta D, \tag{2.3}
\]

in which \( \eta \) represents the dynamic viscosity and tensor \( D \) the rate of deformation tensor,

\[
D = \frac{1}{2} \left( \nabla u^f + (\nabla u^f)^T \right). \tag{2.4}
\]

An incompressible solid phase is considered, described by,

\[
\nabla \cdot \tau^s - \nabla p^s = 0, \tag{2.5}
\]

\[
\det(F) = 1, \tag{2.6}
\]
in which \( p^s \) is the hydrostatic pressure in the solid and \( F \) the gradient deformation tensor \( (\nabla_0 x^s)^T \) where \( x^s \) is the solid position vector and \( \nabla_0 \) the gradient operator with respect to the reference state. The extra stress tensor \( \tau^s \) is defined as,

\[
\tau^s = G \left( F \cdot F^T - I \right),
\]

where \( G \) is the shear modulus and \( I \) the unity tensor. Note that the superscripts \( f \) and \( s \) will be used in this thesis to distinguish fluid and solid, respectively.

### 2.2.2 Coupling

The fluid-structure problems presented throughout this chapter consist of a fluid domain \( (\Omega^f) \) and an immersed solid domain \( (\Omega^s) \). These two domains are coupled at the boundary \( (\partial \Omega^s) \) of the solid domain by the constraint, \( u^f - u^s = 0 \). This constraint is applied weakly by introducing a distributed Lagrange multiplier \( (\lambda) \). The weak forms for Eqs. (2.2), (2.6) and the constraint then become,

\[
\int_{\Omega^f} w^f \cdot \left( \rho \left( \frac{du^f}{dt} + u^f \cdot \nabla u^f \right) - \nabla \cdot \tau^f + \nabla p \right) \, d\Omega^f + \int_{\partial \Omega^s} w^f \cdot \lambda \, d\partial \Omega^s = 0 \tag{2.8}
\]

\[
\int_{\Omega^s} w^s \cdot (\nabla \cdot \tau^s - \nabla p) \, d\Omega^s - \int_{\partial \Omega^s} w^s \cdot \lambda \, d\partial \Omega^s = 0 \tag{2.9}
\]

\[
\int_{\partial \Omega^s} w^\lambda \cdot (u^f - u^s) \, d\partial \Omega^s = 0 \tag{2.10}
\]

in which \( w^s, w^f \) and \( w^\lambda \) are appropriate test functions. A same approach is used by Glowinski et al. [1999a,b] and De Hart et al. [2000] in an FD approach. So far, these methods are generally similar to the one proposed in this chapter. The way of applying the Lagrange multiplier is, however, different. The linearisation of these sets of equations can be found in Appendix A.

Consider a fluid domain, which is discretised into triangular elements, with a boundary \( \partial \Omega^s \) crossing the elements (Fig. 2.1(a)). In the model problems, that de Hart presented, the Lagrange multiplier is defined along the boundary, \( \partial \Omega^s \), to couple the velocity of this boundary to the velocity of the fluid elements in which the boundary is situated [De Hart et al., 2000]. Although such an approach gives satisfactory results for valve displacement and the general flow behaviour, it fails to provide an accurate description of shear stresses at either side of the valve. Since boundary \( \partial \Omega^s \) crosses the fluid elements, interpolation of the fluid velocity at this boundary results in less accurate solutions. To improve accuracy, mesh refinement is required, which can be expensive if it is not known \emph{a priori} where the solid phase is situated. Furthermore, during the diastolic phase of the heart cycle the valves are closed and due to the pressure decrease in the left ventricle, a large pressure gradient occurs across the valve leaflets. Since the leaflets are crossing the fluid elements, erroneous results for the pressures are obtained, which greatly influences the velocities.
The model proposed in this chapter is based on the idea of creating a boundary \( \partial \Omega_f \) inside the fluid domain that coincides with boundary \( \partial \Omega^s \) by performing an adaptation of the mesh in the vicinity of \( \partial \Omega^s \) as explained next.

### 2.2.3 Mesh adaptation

Consider a (fluid) mesh, \( \Omega_f \), with an arbitrary solid boundary curve, \( \partial \Omega^s \), crossing it as shown in Fig. 2.1(a).

In order to create a boundary \( \partial \Omega_f \) in the fluid domain that coincides with boundary \( \partial \Omega^s \), first the intersections of \( \partial \Omega^s \) with \( \Omega_f \) need to be found. When all the intersections are determined a selection of fluid nodes, that ensures mesh integrity, are shifted along the intersected curves from the nodes on this curve to boundary \( \partial \Omega^s \). The fluid nodes that lie on \( \partial \Omega^s \) now form a new inner fluid boundary called \( \partial \Omega_f \).

The repositioning of the nodes around the boundary \( \partial \Omega^s \) influences the element shapes of this boundary, which can lead to inaccurate results. Therefore, smoothing is applied in this region. The fluid nodes that lie on \( \partial \Omega^s \) are not allowed to shift position. The smoothing can be applied in several node layers around \( \partial \Omega^s \) and is based on connectivity. We implemented an algorithm as presented by Freitag et al. [1995]; Freitag and Ollivier-Gooch [1997]. First, one step of Laplacian smoothing is used, in which the new position of a node is calculated by determining the geometrical centre of the surrounding vertex nodes. Second, an angle optimisation algorithm is applied in which the angles of elements around \( \partial \Omega^s \) are enlarged if they are too small. Per node the surrounding elements and their angles are determined. The node is repositioned such that the smallest of these angles is maximised. Following, in the case of second-order extended elements, the midside nodes and centroid are repositioned. For a more extensive explanation on the smoothing algorithms used,
the reader is referred to Appendix B.3. The resulting adapted fluid mesh, which can be used for computing, is shown in Fig. 2.1(b).

Now that an inner fluid boundary \( \partial \Omega^f \) is obtained that coincides with the solid boundary \( \partial \Omega^s \), not only accuracy is gained. It is also possible to take the same discretisation for the Lagrange multiplier domain as for \( \partial \Omega^f \). From numerical experiments it was found that the weakly coupled system is very sensitive to the amount of coupling elements in which the Lagrange multiplier was discretised and to the order of the interpolation functions of these coupling elements. More or higher-order coupling elements clearly lead to stronger coupling between solid and fluid. However, if the discretisation is chosen too fine, the system tends to diverge, as is also observed for interpolation functions of order one and higher. The fluid is especially sensitive to the amount of coupling elements. If the discretisation of the Lagrange multiplier is based on the solid discretisation, it is difficult to control the number of coupling elements within one fluid element. It is therefore convenient to define it based on the discretisation of the newly created inner boundary \( \partial \Omega^f \).

### 2.2.4 Solution proces

If the combined fictitious domain/adaptive meshing approach is used in time-dependent problems, as will be presented in this chapter, the mesh needs to be generated only once. From this mesh, every timestep a second mesh will be created, which is adapted to current solid boundary position. This adapted mesh will be used for computational purposes. The non-linear set of equations is linearised and solved in a Newton-Raphson iterative scheme. Each time step after convergence of this iteration scheme the solid position is updated (an updated Lagrange formulation is used) and a new mesh is created based on the original mesh and the new position of the solid. Following, solutions are mapped from the second mesh onto the newly created mesh using the basis functions of the fluid elements. A graphical representation of the program is shown in Fig. 2.2. An implicit time-integration scheme is used for both the solid and fluid and a first-order approximation is used for the velocity (\( u^f = \frac{\Delta x^F}{\Delta t} \) and \( u^s = \frac{\Delta x^S}{\Delta t} \)), where \( \Delta t \) is the time step. For the fluid \( P^+ \) – \( P^- \) triangular elements are used and for the solid \( Q^+ \) – \( Q^- \) quadrilateral elements. The Lagrange multiplier domain has the same discretisation as the internal fluid boundary that is obtained from mesh adaptation and is integrated using discontinuous linear interpolation functions. Note, that in the model problems, presented in this chapter, the thickness and mass of the solid are negligible as far as the interaction with the fluid is considered. Therefore, fluid and solid velocity are coupled at only one boundary of the solid. With respect to the FD method only extra computation time is needed to perform the mesh adaptation and mapping. Since topology does not alter, the adaptation is a relatively small task, especially for large systems. The CPU time needed to perform the mesh adaptation is therefore negligible compared to the time needed for solving the system. The finite element package, SEPRAN [Segal, 2003], is extended for the computations in combination with a direct HSL solver [HSL, 2002].
2.3 Results

In order to show the value of the presented combined fictitious domain/adaptive meshing method in fluid-structure problems with flexible leaflets, two model problems are presented. The first model shows how an applied pulsatile flow will cause a flexible solid slab to deform and move. The flow field in its turn is highly influenced by the slab. As will be highlighted, shear stresses can vary significantly in space and time. The second problem will show the ability of the method to describe a transvalvular pressure drop. To emphasize the necessity of mesh adaptation, a comparison is made with an FD method.

2.3.1 Model problem 1: solid slab in pulsatile flow

An Eulerian fluid domain $\Omega_f$ is considered with an immersed Lagrangian solid domain $\Omega_s$ (Fig. 2.3). Along the walls, denoted with $\partial\Omega_{wall}^f$, a no-slip condition applies and at $\partial\Omega_{in}^f$ the velocity is prescribed as a function of time,

$$u^f = \sin(2\pi t),$$

over a dimensionless time period of $0.0 - 1.0$. The solid is attached to the upper wall. The fluid is described by the Navier-Stokes equation (Eq. (2.2)) and the solid by the Neo-Hookean relation (Eq. (2.6)). As stated earlier the Lagrange multiplier domain is defined along the internal fluid boundary that is obtained
from mesh adaptation and coincides with boundary $\partial \Omega^s$. The domain is integrated using linear, discontinuous interpolation functions. To get a stable tip displacement, coupling elements are defined along the solid boundary $\partial \Omega^s_{\text{tip}}$, which use constant, discontinuous interpolation functions.

For comparison, a fictitious domain computation has been performed using the same mesh. Only, the discretisation of the Lagrange multipliers is chosen equal to that of the solid at boundary $\partial \Omega^s$. Constant, discontinuous interpolation functions are used for integration.

For all computations concerning this model problem, time is discretised into 4000 time steps. A maximum Reynolds number of 1000 (height of the canal is used as characteristic length) and a maximum Strouhal of 0.1 are used. Large extensions for the inlet and outlet are needed to avoid boundary influences, which leads to a length/height ratio of the canal of 20:1. The fluid mesh is divided into 5004 elements. The solid mesh consists of a length/width ratio of 18:1 and is divided into $20 \times 2$ elements. Since the tip of the solid can move most freely, its movement is taken as a
comparison between both methods. In Figs. 2.4(a), 2.4(b) and 2.4(c) the tip position is plotted as well as the separate $x$ and $y$ displacements as a function of time (Note that $x$ is the axial direction). Only small differences are observed in axial direction while a comparison of the methods in $y$-direction shows larger differences. However, although the transient behaviour between maxima and minima is different for both methods, the value and time at which they take place correspond reasonably well. In Figs. 2.5(a), 2.5(b) and 2.5(c) the solution at $t=0.575$ is highlighted in a plot of a mesh, a vector plot of the velocity field and a contour plot of the velocity field, respectively.

![Mesh](image1)

(a)

![Velocity Field](image2)

(b)

![Contour Plot](image3)

(c)

Figure 2.5: The adapted mesh (a), a vector plot of the velocity field (b) and a streamline plot of the velocity field (c) at $t=0.575$.

For sake of clarity the long inlet and outlet are not shown in the figures, which leaves the most interesting middle part of the fluid domain to be presented. In this part of the time cycle, the prescribed velocity at the inlet has just changed sign and the solid starts to move from right to left. At this time several vortices are present, which immediately shows the necessity of the extended inlet and outlet. Time steps have proven to be sufficiently small, as has the element size in the spacial discretisation.
Computations have been performed for different mesh discretisations (1268, 2534, 5004 and 8548 fluid elements) of which the tip displacements are plotted in Figs. 2.6(a), 2.6(b) and 2.6(c).

![Graphs of tip position during 1 cycle](image)

Figure 2.6: Tip position of the solid during 1 cycle \((t=0.0-1.0)\) for different discretisations of the fluid mesh. Tip position in x- and y-direction as a function of time are shown in (b) and (c), respectively.

As mentioned before, one advantage of the FD method combined with mesh adaptation compared to the FD method without it, is the possibility of gaining information about the shear stresses along the solid. The choice of the \((P_2^+ - P_1)\) elements demands that the computed velocities is continuous across the element boundaries, however, derivatives can be discontinuous. Therefore, it is possible to compute different shear stresses at either side of the solid slab. Fig. 2.7(a) shows part of the mesh and velocity vector at \(t=0.188\) and Fig. 2.7(b) shows the corresponding shear stresses along both sides of the solid. The shear stresses in the midside nodes of the fluid element edges along the solid are plotted. Note that since the tip of the solid lies inside a fluid element, no shear stress information at this position is available.

### 2.3.2 Model problem 2: fluid pressure drop across solid membrane

The first model problem incorporated the movement of the valve, the complexity of the flow field and the determination of shear stresses. These are typical phenomena during systole. An important phenomenon during diastole is the moving pressure drop over the leaflets, which is captured in the following model problem. The setup, that is used, is similar to the one shown in the former example (Fig. 2.3). The only difference is that solid boundary \(\partial \Omega_{\text{tip}}\) is now connected to the bottom wall with homogeneous Dirichlet boundary conditions. The walls, denoted with \(\partial \Omega_{\text{wall}}^{f}\), have homogeneous Dirichlet conditions and at \(\partial \Omega_{\text{wall}}^{s}\) Neumann boundary conditions are prescribed. Coupling between the domains is established at \(\partial \Omega^{s}\). The equations and elements used for fluid and solid are the same as in the former example. Again, the adaptive meshing approach is compared with a fictitious domain approach. As the prescribed pressure builds up the membrane will start to bend and elongate and finally
Figure 2.7: *Velocity vector plot on fluid mesh at t=0.188* (a) and the *corresponding shear stresses along both sides of the solid* (b)
the membrane will find an equilibrium position in which it bears the total pressure. A similar computation is done using FD. However, erroneous results are obtained in the vicinity of the membrane when the pressure jump gets too large (Fig. 2.8(a)). The error in the solution accumulates, which after several time steps leads to divergence of the non-linear system. The FD method combined with adaptive meshing, on the other hand, captures the pressure drop accurately. The $P_2^+ - P_1$ elements are able to describe pressure discontinuities across the element edges. By mesh adaptation an inner fluid curve is created, which enables a drop of the pressure over the valve as would be expected. The corresponding velocity field will not be disturbed by a poor pressure solution in the vicinity of the valve as can be seen in Fig. 2.8(b). First, movement of the valve is induced by the applied pressure as shown in Fig. 2.8(b), but finally the velocity field will be zero with a pressure jump across the membrane.

2.4 Discussion

A method is presented for modelling fluid-structure problems in flexible heart valves, which is able to accurately capture shear stress along both sides of the leaflets as well as transvalvular pressure gradients. The FD method presented by De Hart et al. [2000] is extended with a computationally inexpensive adaptive meshing algorithm. Coupling of a separate Eulerian and Lagrangian mesh is established using a Lagrange multiplier, which allows for optimal choices of the discretisations of fluid and structure. The coupling between the meshes is enhanced by adapting the Eulerian mesh, such that an inner curve $\partial \Omega_f$ is created which coincides with a solid boundary. Since the fluid mesh is only adapted locally, where the solid crosses fluid elements and since topology remains unchanged, the adaptation algorithm is relatively inexpensive with respect to building and solving the set of equations. The use of a discontinuous pressure discretisation in combination with the inner fluid curve $\partial \Omega_f$ enables capturing of the
pressure drop across the leaflets. Furthermore, accurate solutions for the shear stress at both sides of the leaflets can be obtained. Two examples are presented in which the method proves to be a significant improvement with respect to the FD method in which the Eulerian mesh is kept unchanged. Since FD methods have proven to be an interesting numerical tool for 3D analysis of fluid-structure problems in heart valves [De Hart et al., 2003b,a], extension of the presented method to three dimensions is a step of great interest. However, in the 3D physiological situation, the three leaflets of the aortic heart valve interact with one another. This solid-solid contact problem was avoided in this chapter by considering two separate situations that could be seen as a representation for systole and diastole. In order to describe this transition phase, in which the coaptation areas of the leaflets interconnect, contact algorithms should be incorporated in the computations.
Chapter 3

A fluid-structure interaction method with solid-rigid contact

Summary: A computational method is proposed for problems where fluid-structure interaction is combined with solid-rigid contact. This combination is particularly important for the dynamics and impact of heart valves. The Navier-Stokes equation in an Eulerian setting is coupled to a Neo-Hookean solid model using a Lagrangian description. A fictitious domain method extended with a local mesh adaptation algorithm provides the required flexibility with respect to the motion and deformation of the valve. In addition, it ensures the solids ability of sustaining pressures present in the fluid. Making use of the fact that the fluid and solid mesh are not required to be connected conformly, it is shown that the model can be extended with a contact algorithm without introducing meshing complications near the contact surfaces.

3.1 Introduction

Fluid-structure interaction models are increasingly used in biomedical engineering applications and one of the most challenging fluid-structure problems that can be found in the human body involves the dynamics of heart valves. The most extensively studied valves are the mitral and the aortic valve. The former is a bileaflet valve located between the left atrium and left ventricle, the latter is a trileaflet and is located between the left ventricle and the aortic root. Both valves have extremely thin leaflets, which should hamper flow as little as possible when opened, but need to prevent blood backflow (regurgitation) when closed. In the closed state the leaflets are in mutual contact and a large transvalvular pressure gradient will occur. The tissue of which the heart valves are built has an anisotropic and heterogeneous structure. The arterial walls and heart muscle are compliant and therefore play an important role in the process of opening and closing. Altogether, the combination makes the problem extremely complex to model.

Different ways of modelling fluid-structure interaction (FSI) have been proposed in the past, each having its advantages and disadvantages. Arbitrary Lagrangian Eulerian (ALE) methods, as exploited by e.g. Hirt et al. [1974]; Donea et al. [1982], are most commonly used for FSI problems and have the advantage of providing a strong coupling. As long as rotations, translations and/or deformations of the solid remain within certain limits, this method works very well and is recommended. However, for problems in which these limits are violated, elements become ill-shaped and ALE alone does not suffice. As a solution to this problem an often-used combination is ALE with some form of remeshing. This can, however, be a difficult and time-consuming task.

A more elegant way to solve the system allowing free movements of a structure through a fluid domain was proposed by Peskin [1972]. He introduced a method that later became known as the Immersed Boundary Method (IBM) [Peskin, 2002] where flow-induced solid body motions could be computed without adjusting the fluid grid/mesh. By defining a set of interconnected points related to each other by some elastic law local body forces were enforced to the fluid. Extensions of this model to three dimensional heart (valve) problems were published in e.g. McQueen and Peskin [1997]; Peskin and McQueen [1989] and the method is still used in many fields.

A similar method that could cope with these large translations and rotations was used by Glowinski et al. [1997, 1998a,b]. Using a fictitious domain method the sedimentation of rigid particles in a fluid domain could be computed. The rigid particles are immersed on the fluid domain and are coupled to the fluid by applying constraints at the boundary of the particle using a Lagrange multiplier. This way of coupling allowed for large translations as well as rotations of the particles.

Recently, a method that resembles the above mentioned methods was introduced [De Hart et al., 2000; Baaijens, 2001] for slender bodies. In this method a fluid mesh is considered with an immersed solid mesh, and the solid mesh and fluid mesh are coupled by a Lagrange multiplier (or local body forces) at the boundary of the solid. With respect to heart valve modelling good results have been obtained in 2D [De Hart et al., 2000] as well as in 3D [De Hart et al., 2003b,a, 2004]. The elegance
of this method is its simplicity and flexibility. Finally, fictitious domain methods have been proposed that do not restrict to slender bodies by introducing Lagrange multipliers across the whole solid body instead of only along its boundaries [Yu, 2005; Shi and Phan-Thien, 2005]. This modification extends the method to a wider range of applications.

Drawbacks of these fictitious domain approaches is the inability of the method to sustain a transvalvular pressure difference and to compute accurate shear stresses along the solid boundaries. Stijnen et al. [2004] introduced a model for mechanical heart valves where the position of the closed valve was known \textit{a priori}, to circumvent the first of the two drawbacks mentioned above. By creating the fluid mesh such that a curve of fluid edges coincided with the solid boundary in the closed state, a drop in pressure across the valve could be described.

A combination of the fictitious domain method with adaptive meshing was proposed by Van Loon et al. [2004] to compute moving pressure discontinuities across a flexible leaflet without an \textit{a priori} known closed position. It was also shown that the method was able to compute shear stresses quantitatively at both sides of the solid. An extension of this method to three dimensions has also been published recently by Van Loon et al. [2005a] together with an improved adaptive meshing scheme. Although some of the flexibility of the original fictitious domain methods is lost by the introduction of adaptive meshing, accuracy with respect to shear stress is gained. Different discretisations for fluid and solid are, however, still allowed, solving the system fully coupled.

This chapter focusses on a fluid-structure interaction method as the one proposed in Chapter 2 extending it with a Lagrange multiplier based contact algorithm which enables interaction of the solid with walls of a fluid domain. The adaptive meshing scheme is a two-dimensional version of the one presented in Van Loon et al. [2005a]. The combination of the fictitious domain method with other established techniques like adaptive meshing and ALE methods allows not only for accurate motion of the solid body but also accurate fluid behaviour near the solid-fluid interface. The method presented is able to capture the opening and closing behaviour of a heart valve in a model problem sufficiently accurate, to be used for heart valve analysis during a complete cardiac cycle if extended to three dimensions.

The chapter is organised as follows. First, the governing equations are given, treating the equations for solid and fluid, followed by the constraints concerning the fluid-structure interaction and contact. Then the full set of equations is given in the weak form, after which the discretisation and adaptive meshing are treated. All techniques are combined in two illustrative finite element model problems, describing thin flexible solid bodies in a pulsatile flow, where the solid can make contact with the wall. Pressure fields, shear stresses and solid motion are observed closely.
3.2 Governing equations

3.2.1 Fluid and solid

First the governing equations that describe an incompressible Newtonian fluid and an incompressible Neo-Hookean solid, are presented. The set of equations for the fluid domain, denoted by $\Omega_f$, read,

\[ \rho \left( \frac{\partial u^f}{\partial t} + u^f \cdot \nabla u^f \right) = \nabla \cdot \sigma^f + \rho f^f, \]  
\[ \nabla \cdot u^f = 0, \]  
\[ \sigma^f = 2\eta D - p^f I \quad \text{with} \quad D = \frac{1}{2}(\nabla u^f + (\nabla u^f)^T), \]

and for the solid domain, $\Omega_s$,

\[ \nabla \cdot \sigma^s = \rho f^s, \]  
\[ \det(F) = 1, \quad \text{with} \quad F = (\nabla_0 x)^T \]  
\[ \sigma^s = G (B - I) - p^s I \quad \text{with} \quad B = F \cdot F^T \]

Eqs. (3.1)-(3.3) and (3.4)-(3.6) are the momentum balance, the continuity and the constitutive relation describing the rheological behaviour of the fluid and solid, respectively. The quantities $\rho$, $\eta$ and $G$ are the density, dynamic viscosity and shear modulus. The gradient operators with respect to the current and the initial configuration are denoted with $\nabla$ and $\nabla_0$, respectively. The vectors $x$ and $u$ denote the position and velocity vector, respectively, and $p$ the hydrostatic pressure. Body forces $f^s$ and $f^f$ are zero and gravity forces are negligible since a heart valve is almost neutrally buoyant.

3.2.2 Fluid-structure coupling

The fluid domain $\Omega_f$ with an imbedded solid domain $\Omega_s$ is considered. In order to capture the fluid-structure interaction, these two domains need to be coupled. This coupling is obtained by applying a no-slip condition,

\[ u^f - u^s = 0, \]

at the boundary of the solid $\partial \Omega_s$. Note that the partial sign $\partial$ will be used to denote the boundary of a domain.

3.2.3 Contact

Modelling the interaction of a deformable solid with a rigid contact surface is not very different from modelling the interaction between a fluid and a structure. The main difference lies in the fact that contact is a temporary state while it is assumed that the fluid and structure interact permanently. If a solid approaches a contact surface nothing will happen until the gap between them approaches zero. From that
moment a contact force, $\mathbf{X}^c$, between the bodies will be induced which inhibits the solid body from penetrating the contact surface and contact occurs until this contact force equals zero again. If we describe the contact surface as a function, $g(\mathbf{x}, t) = 0$, then for every point at the solid boundary, $\forall \mathbf{x}^s \in \partial \Omega^s$, a gap distance can be defined as $\delta^c = (\mathbf{x}^s - \mathbf{x}^c) \cdot \mathbf{n}^c$, where $\mathbf{x}^c$ is a position at the contact surface ($g(\mathbf{x}^c, t) = 0$) and $\mathbf{n}^c$ the corresponding normal vector with respect to this body in the direction of the solid body. For every $\mathbf{x}^s$ a point $\mathbf{x}^c$ is chosen such that the distance, $\min(|| \mathbf{x}^s - \mathbf{x}^c ||)$, between them is minimised with $|| \mathbf{x} || = \sqrt{\mathbf{x} \cdot \mathbf{x}}$. A second important quantity with respect to the contact problem is the contact force, which is defined as $\lambda^c = \lambda^c \cdot \mathbf{n}^c$. The constraints corresponding to the different contact states read in case of "contact",

$$\delta^c = 0,$$  \hspace{1cm} (3.8)

and in case of "no-contact",

$$\lambda^c = 0.$$  \hspace{1cm} (3.9)

The moment that the solid body penetrates the rigid surface (Fig. 3.1(b)), the gap distance will be forced to zero (Eq. (3.8)) and contact is established (Fig. 3.1(c)).

![Diagram](image_url)

Figure 3.1: Different stages of the contact algorithm.

When the contact force becomes negative (Fig. 3.1(d)) the solid body should be released from the contact surface and the trivial "no-contact" constraint (Eq. (3.9)) is
applied. When a node is in contact the switch of constraints is triggered by evaluating
the contact pressure and if a node is not in contact the switch is triggered by looking
at the gap distance. Note that frictionless contact is assumed since Eq. (3.8) only
restricts the displacement in the normal direction.

### 3.3 Weak formulation

Based on the governing set of equations and the set of constraints to which they
should comply as defined in the previous sections, the weak formulation for the total
set of equations will be derived. If we define the Lebesgue space \( L_2(\Omega) \) and Sobolev
space \( H^\alpha(\Omega) \) as,

\[
L_2(\Omega) = \{ v : \int_\Omega v^2 \, d\Omega < \infty \}
\]

\[
H^\alpha(\Omega) = \{ v \in L_2(\Omega) : D^\beta v \in L_2(\Omega) , \ |\beta| \leq \kappa \}
\]

with \( D^\beta v \) the weak derivatives of order \( \beta \), spaces of acceptable solutions for the solid
and fluid velocities can be defined as,

\[
\mathbb{V}^\alpha(\Omega^\alpha) = \{ v : v \in H^1(\Omega^\alpha)^2 , v = u_0^\alpha \text{ on } \partial\Omega^\alpha \} \text{ for } \alpha = f, s.
\]

The corresponding trial functions should lie in the space,

\[
\mathbb{V}_0^\alpha(\Omega^\alpha) = \{ \mathbb{V}(\Omega^\alpha) : u_0^\alpha = 0 \} \text{ for } \alpha = f, s.
\]

The kinematic constraints given by Eqs. (3.7), (3.8) and (3.9) are enforced by means
of the two Lagrange multipliers, \( \lambda^{f,s} \) and \( \lambda^c \). Both multipliers are typically defined
along the boundary of the solid domain \( \partial\Omega^s \), but depending on the problem alternative
domains can be chosen as will be done later in this chapter. In order to distinguish
the two Lagrange multiplier domains, \( \gamma^f \) and \( \gamma^c \) are introduced to denote the domain
responding to the fluid-structure interaction and the solid contact surface. By
using the momentum balance for fluid and solid, Eqs. (3.1) and (3.4), including the
multipliers and applying all the constraints Eqs. (3.2), (3.5), (3.7), (3.8) and (3.9),
the weak formulation for the total set of equations can be formed. Find \( u^f \in \mathbb{V}^f(\Omega^f) \),
\( u^s \in \mathbb{V}^s(\Omega^s) \), \( p^f \in L_2(\Omega^f) \), \( p^s \in L_2(\Omega^s) \), \( \lambda^{f,s} \in L_2(\gamma^f)^2 \) and \( \lambda^c \in L_2(\gamma^c)^2 \) such that,

\[
\int_{\Omega^f} w^f \cdot (\rho \frac{\partial u^f}{\partial t} + \rho u^f \cdot \nabla u^f) \, d\Omega^f - \int_{\Omega^f} 2\eta D(u^f) : D(u^f) \, d\Omega^f + \\
\int_{\Omega^f} p^f \nabla \cdot w^f \, d\Omega^f + \int_{\gamma^f} \sigma^f \cdot \lambda^{f,s} \, d\gamma^f = \int_{\partial\Omega^f} w^f \cdot (\sigma^f \cdot n^f) \, d\partial\Omega^f, \quad (3.10)
\]

\[
\int_{\Omega^f} q^f \nabla \cdot u^f \, d\Omega^f = 0, \quad (3.11)
\]
\[
\int_{\Omega^s} w^s \cdot \nabla \cdot \tau^s \, d\Omega^s - \int_{\Omega^s} p^s \nabla \cdot w^s \, d\Omega^s - \int_{\gamma^f} w^s \cdot \lambda^{fs} \, d\gamma^f \\
- \int_{\gamma^e} w^s \cdot \lambda^c \, d\gamma^c = \int_{\partial\Omega^s} w^s \cdot (\sigma^s \cdot n^s) \, d\partial\Omega^s,
\]

(3.12)

\[
\int_{\Omega^s} q^s (\det(F) - 1) \, d\Omega^s = 0,
\]

(3.13)

\[
\int_{\gamma^f} \mathbf{w}^f \cdot (\mathbf{u}^f - \mathbf{u}^s) \, d\gamma^f = 0,
\]

(3.14)

\[
\int_{\gamma^e} \mathbf{w}^e \, d\gamma^c = 0 \quad \text{with} \quad h = \begin{cases} \delta^c & \text{for } \lambda^c > 0, \delta^c \leq 0 \\ \lambda^c & \text{for } \lambda^c \leq 0, \delta^c \geq 0 \end{cases}
\]

(3.15)

for all trial functions \( \mathbf{w}^f \in \mathbb{V}^f(\Omega^f) \), \( \mathbf{w}^s \in \mathbb{V}^s(\Omega^s) \), \( q^f \in L_2(\Omega^f) \), \( q^s \in L_2(\Omega^s) \), \( \mathbf{w}^{fs} \in L_2(\gamma^f)^2 \) and \( \mathbf{w}^c \in L_2(\gamma^c) \). The normals at the solid boundary \( \partial\Omega^s \) and the fluid boundary \( \partial\Omega^f \) are denoted by \( \mathbf{n}^s \) and \( \mathbf{n}^f \), respectively. Note that the kinematic constraint with respect to fluid-structure interaction is applied in a “weak” manner, meaning that the integrals over the fluid and solid velocities are forced to be equal along boundary \( \gamma^f \). Similarly, the kinematic constraint for the contact problem is applied. As \( \lambda^c \) represents the contact force that acts between the solid and rigid body in case of contact, so the physical representation of \( \lambda^{fs} \) is the traction force between solid and fluid. These “additional body forces” are applied to the weak formulations of the balance of momentum for fluid and solid, Eq. (3.10) and (3.12), in order to satisfy the kinematic constraints. Finally, note that Eqs. (3.10) and (3.12) contain non-linear parts, and should therefore be linearised as presented in Appendix A.

### 3.4 Discretisation

Finite-element discretisations are applied for the fluid and solid domain and for the Lagrange multipliers corresponding to fluid-structure coupling and the contact law. First, we focus on the discretisations of the fluid and solid domain and the corresponding adaptive meshing scheme. Next, the discretisations of the Lagrange multipliers corresponding to the fluid-structure coupling and solid-rigid contact will be discussed, the first of which depends on the fluid discretisation and the latter on the solid discretisation.

#### 3.4.1 Discretisation of fluid and solid

For convenience, first some definitions concerning a finite element mesh are posed. Therefore, we consider a domain \( \Omega \subset \mathbb{R}^2 \) and its boundary \( \partial \Omega \subset \mathbb{R}^2 \). If \( \Omega \) is subdivided into \( N_\Omega \) elements \( \Omega_e \) with \( e = 1, \ldots, N_\Omega \), the following relations can be defined with
respect to the elements \( \Omega_e \), the elemental edges \( \Gamma_j \) and elemental vertex points \( x_k \),

\[
\Omega = \bigcup_{e=1}^{N_\Omega} \Omega_e \quad \text{and} \quad \Omega^g = \{ e \in \mathbb{N} : \Omega_{e \neq i} \cap \Omega_i = \emptyset \} \quad \forall i \in \mathbb{N}_0
\]

\[
\partial \Omega_e = \bigcup_{j=1}^{N_{\Gamma_e}} \Gamma_j \quad \text{and} \quad \Gamma^g = \{ k \in \mathbb{N} : \Gamma_k = e \Gamma_j, \Gamma_{k \neq l} \cap \Gamma_l = \emptyset \} \quad \forall e \Gamma_j, \forall l \in \mathbb{N}_0
\]

\[
\partial \Gamma_j = \bigcup_{i=1}^{2} \partial \Gamma_j \quad \text{and} \quad \partial^g = \{ k \in \mathbb{N} : x_k = j x_i, x_{k \neq l} \neq x_i \} \quad \forall j x_i, \forall l \in \mathbb{N}_0
\]

where \( N_{\Gamma} \) is the number of local edges \( \Gamma_j \) in element \( \Omega_e \). A graphical representation is presented in Fig. 3.2. Note that local edges are denoted as \( \Gamma_j \) and global edges as \( \Gamma_{e\Gamma_j} \) and similarly \( x_i \) and \( x_j \) are local and global points. The sets \( \Gamma^g \) and \( \partial^g \) contain the numbers of all elemental edges and nodal points of the element vertices in \( \Omega \), respectively.

![Graphical representation](image)

**Figure 3.2: Topological definitions.**

Every edge and vertex point in the mesh is unique. Now the domains \( \Omega^f \subset \mathbb{R}^2 \) and \( \Omega^s \subset \mathbb{R}^2 \) in the initial configuration can be discretised independently of one another to obtain the overlapping, conform meshes \( \Omega^g \) and \( \Omega^s \) respectively. Since these meshes are created independently, they will be non-conforming with respect to each other. Mesh \( \Omega^g \) is a triangulation while \( \Omega^s \) consists of quadrilaterals.

**Adaptive meshing**

An adaptive meshing procedure is now introduced for adapting the fluid mesh based on the position of the solid mesh, which is similar to the one proposed by Lock et al. [1998]. Although the shape of the solid elements can be chosen arbitrarily, the element shape of the fluid elements is restricted to a triangular shape for this meshing.
procedure. The fluid mesh will be adapted such that an inner fluid curve \( \gamma^f \subset \Omega^f \) is created based on the position of a solid boundary \( \gamma^s \subset \Omega^s \). Along this boundary the solid and fluid mesh will be coupled using Eq. (3.14). Unlike the adaptive meshing scheme proposed earlier in Section 2.2.3, local topological changes are permitted. We first consider the solid domain \( \Omega^s \) and fluid domain \( \Omega^f \) such that \( \overline{\Omega^s} \cap \overline{\Omega^f} \neq \emptyset \). A curve \( \gamma^s \) needs to be chosen as a collection of \( N^s \) elemental edges of the solid along which the fluid-structure coupling should be established:

\[
\gamma^s = \bigcup_{i=1}^{N^s} \Gamma^s_i \quad \text{with} \quad i \in \Gamma^s.
\]

Based on the position of this curve the fluid mesh can now be adapted. Therefore, the set of intersection points ought to be found of all the fluid edges that intersect \( \gamma^s \),

\[
S^f = \{ i \in \Gamma^s : \Gamma^f_i \cap \gamma^s \neq \emptyset \},
\]

\[
X_i = \{ x : x \in \Gamma^f_i \cap \gamma^s \} \quad \forall i \in S^f.
\]

Note that \( S^f \) contains the edge numbers of all edges that intersect \( \gamma^s \) and \( X_i \) contains all corresponding intersection points. Some comments on the numerical approach to determine the intersection points, are given in Appendix B.2. If more than one intersection point is stored in \( X_i \) only one, arbitrarily chosen, point is taken into account in the remainder of the adaptive meshing algorithm. Now, a parameter \( \varepsilon_i \) can be introduced, which describes the relative position of an intersection point along an edge:

\[
\varepsilon_i = \frac{\| x_i - x_1 \|}{\| x_2 - x_1 \|} \quad \forall x_i \in X_i.
\]

Based on this parameter together with the pre-defined parameter \( \tilde{\varepsilon} \), set \( S^f \) can be subdivided into the subsets,

\[
S^{f_1} = \{ i \in S^f : \varepsilon_i < \tilde{\varepsilon} \| \Gamma^f_i \| \},
\]

\[
S^{f_2} = \{ i \in S^f : \varepsilon_i > (1 - \tilde{\varepsilon}) \| \Gamma^f_i \| \},
\]

\[
S^{f_3} = \{ i \in S^f : \varepsilon_i < (1 - \tilde{\varepsilon}) \| \Gamma^f_i \| \},
\]

in which \( \| \Gamma^f_i \| = \| x_2 - x_1 \| \) is the length of an edge. The arbitrary parameter \( \tilde{\varepsilon} \) should have a value between 0 and 0.5 and is explained graphically in Fig. 3.3.

In this way every intersected fluid edge is subdivided into three parts and sets \( S^{f_1} \), \( S^{f_2} \) and \( S^{f_3} \) contain the edge numbers of the edges that are intersected in part one, two or three, respectively. Using these three sets the actual adaptation of the mesh is performed as follows:

\[
x_{n_1} = x_i \quad \text{with} \quad x_i \in X_i, \ n_1 \in \\^S^f \ \forall i \in S^{f_1},
\]

\[
x_{n_2} = x_i \quad \text{with} \quad x_i \in X_i, \ n_2 \in \\^S^f \ \forall i \in S^{f_2},
\]

\[
\hat{x}_{n_3} = x_i \quad \text{with} \quad x_i \in X_i, \ \forall i \in S^{f_3},
\]
in which $x_{n_1} = i \mathbf{x}_1$, $x_{n_2} = i \mathbf{x}_2$ and $\hat{x}_{n_2}$ is the position of a newly created node $\hat{n}_2$ on edge $\Gamma_i^j$. From the above it follows that fluid nodes are either repositioned or added to the mesh. At the intersection point of the edges in $S^{j_3}$, a new node is created. Consequently, each of these intersected edges is divided into two new edges. For all edges in $S^{j_1}$ and $S^{j_2}$ the begin or end node are shifted to the position of the intersection, respectively. We impose that the position of a node is allowed to be changed only once, even if this node is shared by several intersected edges. After the procedure of relocating and adding edges, topological changes are processed. Depending on the number of intersections a triangular fluid element is split into several elements. Elements with one, two and three intersected edges are subdivided into two, three and four new elements, respectively. There are two ways of subdividing a triangle with two intersections into three subtriangles. In our case the subdivision was chosen, that leads to the best-shaped subtriangles, i.e. most closely resembling an ideal triangular reference element. Note that for small values of parameter $\varepsilon$ more elements will be created (Fig. 3.4), however, the created elements will be increasingly ill-shaped.

A remark on the robustness of the adaptive meshing scheme can be found in Appendix B.1. Now, using the edges $\Gamma_i^j \subset \Omega^j$, of the adapted fluid mesh $\Omega^{j^*}$, the new
subdomain $\gamma^f \subset \Omega^f$ can be defined as,

$$\gamma^f = \{ x \in \Gamma^f : \mathbf{x}_1 \in \gamma^s, \mathbf{x}_2 \in \gamma^s \} \quad \forall i.$$  \hspace{1cm} (3.17)

Since $\gamma^s$ is part of the solid, which moves through the fluid domain, curve $\gamma^f$, which is based on $\gamma^s$, will also change every time step.

**Smoothing**

In order to improve the shape of the fluid elements near $\gamma^f$, Laplacian smoothing is applied [Field, 1988] to the fluid vertex nodes lying in the vicinity of, but not on $\gamma^f$, according to:

$$x_j = \frac{1}{w^N} \sum_{k=1}^{N} w_k x_k, \quad \text{with} \quad w^N = \sum_{k=1}^{N} w_k.$$  \hspace{1cm} (3.18)

Symbol $N$ denotes the number of mesh points $x_k = \mathbf{x}_1$ sharing an edge $\Gamma_i$ with point $x_j = \mathbf{x}_2$. The weight functions $w_k$ corresponding to $x_k$ can be used for introducing gradients in element size. In case $w_k = 1$, point $x_j$ will be repositioned to the geographical center of the surrounding points. However, if values larger than one are chosen for nodes that lie closer to $\gamma^f$, a refinement is obtained perpendicular to this boundary which is convenient since velocity gradients are largest in the boundary layer. This way the overall mesh size can be coarser reducing computational effort without compromising accuracy. The size of the area which will be smoothed is adjustable depending on the needs.

**Element choice**

The solid domain is discretised using quadratic nine noded quadrilaterals or so-called $Q_2 - P_1$ elements. As for the fluid domain, seven noded quadratic triangles or $P_2^+ - P_1$ elements are used. Both elements are so-called Crouzeix-Raviart type of elements, i.e. elements with a discontinuous pressure approximation. The spaces $Q_2$ and $P_2$ contain the polynomials of degree 2 or less used for the approximation of the velocity, and $P_1$ contains the polynomials of degree 1 or less. The $+$ sign denotes the enrichment of the elements by a third-order term which vanishes at the element boundary. Note that these elements are not arbitrarily chosen, since the discontinuous pressure interpolation for the fluid elements (across element borders) will be used to capture pressure gradients in the fluid domain across the solid as will be discussed in Section 3.5.1. Finite element spaces for velocity vectors $\mathbf{u}^f, \mathbf{u}^s$ and hydrostatic pressures $p^f$ and $p^s$ read, respectively,

$$V_h^f = \{ u_h^f \in C^0(\Omega)^2 : u_h^f|_e \in (P_2^+)^2, \forall e \in \Omega, u_h^f|_{\partial \Omega} = 0 \},$$

$$V_h^s = \{ u_h^s \in C^0(\Omega)^2 : u_h^s|_e \in (Q_2)^2, \forall e \in \Omega, u_h^s|_{\partial \Omega} = 0 \},$$

$$L_h^f = \{ q_h^f : q_h^f|_e \in P_1, \forall e \in \Omega \},$$

$$L_h^s = \{ q_h^s : q_h^s|_e \in P_1, \forall e \in \Omega \}.$$
Note that the introduction of second order elements for fluid and solid also allows for quadratically curved element edges. To achieve this, the mesh can be adapted as described in section 3.4.1 based on the vertex nodes of the triangular elements, performing only one additional step to change $\gamma^f$. The fluid edges $\Gamma^f$ that have $x_1$ and $x_2$ on $\gamma^f$ become curved by projecting their center nodes onto $\gamma^s$ in normal direction of edge $\Gamma^f$. This way the geometrical difference between $\gamma^s$ and $\gamma^f$ reduces, improving the accuracy of solutions.

3.4.2 Discretisation of the Lagrange multiplier for fluid-structure interaction

Since boundary $\gamma^f$ changes every time step, this has some implications for the discretisation of the Lagrange multiplier $\lambda^{fs}$, that is defined along $\gamma^f$. As derived earlier $\gamma^f$ consists of a set of fluid edges and the discretisation of $\lambda^{fs}$ is taken equal to discretisation of the fluid element edges of which $\gamma^f$ consists. Piecewise discontinuous linear polynomials are used for the multiplier resulting in the corresponding finite element space,

$$\mathbb{V}^{fs}_h = \{ \lambda^{fs}_h : \lambda^{fs}_h, e \in P_1, \forall e \in \gamma^f \},$$

where $\gamma^f$ contains the edge numbers of the fluid edges in $\gamma^f$. Although different kinds of interpolation polynomials were tested, discontinuous linear interpolation led to the best result (without spurious currents at the solid-fluid interface), which might be expected since internal stresses and pressures are similarly described.

By taking equal discretisations for fluid and multiplier, a fluid element never contains more than one coupling element defined along one of its edges. Note that this does not hold for the solid elements since fluid and solid mesh are non-conforming. Numerical experimenting showed, however, that solid elements are not as sensitive to the discretisation of $\lambda^{fs}$ as are fluid elements. Therefore, varying the discretisation of $\lambda^{fs}$ every time step based on the fluid mesh enhances the coupling between solid and fluid. Nonetheless, one should realise that for reasons of stability and accuracy, the mesh sizes of $\Omega^s$ and $\Omega^f$ should not differ too much as also stated by Glowinski et al. [1999a].

In case boundary $\gamma^s$ is not a closed boundary it is likely to occur that $\gamma^f$ will not be equal to $\gamma^s$. When $\gamma^s$ ends within a fluid element the adaptive meshing scheme does not accommodate for a coupling element in this area. A small part of $\gamma^s$, which depends on the element size of the fluid elements, will therefore not be coupled to the fluid, introducing leakage through this part. In the model problems, presented in Section 3.5, this error was reduced by applying a Lagrange multiplier at the free end of $\gamma^s$ and couple fluid and solid in a more global manner as proposed by De Hart et al. [2000].
3.4.3 Discretisation of the Lagrange multiplier for solid-rigid contact

The Lagrange multiplier corresponding to the contact problem is discretised somewhat differently to that in Section 3.4.2. For sake of simplicity with respect to the determination of the gap distance the multiplier is only evaluated in the nodal points of the solid contact surface $\gamma^c$. In this way the curvature of the quadratic solid elements is not fully taken into account in the case of contact, which is allowed for fine discretisations of the solid domain that result in low curvature per element. The corresponding finite element space for this collocation method then reads,

$$\mathbb{V}_h^c = \{ \lambda_h^c : \lambda_h^c|_i = \lambda_h^c \delta(x - x_i), \lambda_h^c \in \mathbb{R}, \forall i \in \gamma^c \},$$

with $\gamma^c$ a set containing the numbers of all solid nodes on boundary $\gamma^c$ and $\delta$ denoting the Dirac function.

3.4.4 Full set of discretised equations

An implicit Euler time integration scheme is used for the solid as well as the fluid. Taking a first order approximation for the solid velocity $u^s = \Delta x^s/\Delta t$ and applying it to Eqs. (3.12), (3.13) and (3.15) leads to a set of equations in which fluid and solid velocities are solved. The structure of the total set of discretised equations is then given by,

$$
\begin{pmatrix}
K^f & L^f & 0 & 0 & \Lambda^f_{fsi} & 0 \\
(L^f)^T & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \Delta tK^s & L^s & \Lambda^s_{fsi} & \Lambda^\text{cond} \\
0 & (L^s)^T & 0 & 0 & 0 & 0 \\
G^f_{fsi} & 0 & G^s_{fsi} & 0 & G^\text{no-fsi} & 0 \\
0 & 0 & \mathcal{C}^\text{cont} & 0 & 0 & \mathcal{C}^\text{no-cont}
\end{pmatrix}
\begin{pmatrix}
\mathbf{u}^f \\
\mathbf{p}^f \\
\mathbf{u}^s \\
\mathbf{p}^s \\
\lambda^{fsi} \\
\lambda^c
\end{pmatrix}
= 
\begin{pmatrix}
\mathbf{f}^f \\
\mathbf{f}^p \\
\mathbf{f}^s \\
\mathbf{f}^{fsi} \\
\mathbf{f}^c
\end{pmatrix}
$$

in which the solid and fluid system matrices $K^f$ and $K^s$ can be recognised with corresponding divergence matrices $L^f$ and $L^s$. With respect to fluid-interaction the matrices $\Lambda^f_{fsi}$ and $\Lambda^s_{fsi}$ are added to meet the dynamic constraint and corresponding matrices $G^f_{fsi}$, $G^s_{fsi}$ and $G^\text{no-fsi}$ in order to meet the kinematic constraint. This last matrix is filled in case there are nodes in contact. Similarly, $\mathcal{C}^\text{cont}$ is added to meet the
dynamic constraint and $C^\text{cont}$ and $C^\text{no-cont}$ take care of the kinematic condition for the contact problem in case of contact and no-contact, respectively. The finite element package SEPRAN [Segal, 2003] is used and extended for building the matrix and mapping the solutions. The set of equations is solved fully coupled and the asymmetric sparse matrix is solved using a direct method based on a sparse multifrontal variant of Gaussian elimination (HSL/MA41) [HSL, 2002].

### 3.4.5 Combined fluid-structure interaction/contact

In the case of contact, two contradicting boundary conditions are present in this region. Firstly, Eq. (3.7), which couples the fluid to the solid and secondly, Eq. (3.8) which provides for the contact between fluid wall and solid, but allows slip. Furthermore, Dirichlet boundary conditions are present at the fluid wall. Therefore a choice should be made as to which of these constraints should be the predominant one. If Eq. (3.7) 'overrules' Eq. (3.8), a no-slip condition along the fluid wall would also be enforced to the solid, preventing it from pulling itself off the wall. In the case Eq. (3.8) is predominant over Eq. (3.7) any essential boundary condition, applied to the wall will hold no effect on the solid. In the model presented here, fluid-structure coupling elements that have nodes with Dirichlet boundary conditions, are 'disabled'. For the degree(s) of freedom to which the boundary conditions are applied Eq. (3.7) is replaced by,

$$\chi^\text{frei} \cdot \mathbf{n} = 0$$

where $\mathbf{n}$ denotes the direction in which the boundary condition holds. This allows for free slip of the solid along and loosening from the fluid wall, although boundary conditions may enforce the opposite.

### 3.4.6 Combined ALE and mapping

Coupling a non-conforming fluid and solid mesh allows for a relatively simple way of extending the fluid-structure algorithm with contact. In ALE methods as often used for fluid-structure interaction problems, the Eulerian fluid mesh moves in a Lagrangian manner 'fixed' to the solid as shortly explained in Appendix C. If the solid approaches a boundary (e.g. fluid wall) fluid elements will be squeezed. Unlike these ALE methods, the boundary $\gamma^f$ at which the no-slip condition (Eq. (3.7)) is applied, changes in time. As shown earlier, $\gamma^f \subseteq \gamma^s$, which implies that the region along which no-slip holds can be smaller than the region along which it should hold. Although this introduces an inaccuracy as mentioned earlier, it allows for approaching boundaries and sliding along boundaries without the risk of squeezing fluid elements.

The reason for introducing an adaptive meshing algorithm was the need for accurate shear stresses and pressures at both sides of the solid. Since the Lagrangian solid moves in time, the fluid mesh requires continuous updating. Every time step an adapted fluid mesh $\Omega S^f$ is computed based on the initially generated fluid mesh $\Omega S^f$ using the adaptive meshing algorithm as described in Section 3.4.1. The set of equations is solved using the adapted mesh $\Omega S^f$ after which the solid is updated.
Another adapted mesh $\Omega S^{f*}$ is then generated based on the new position of the solid and on mesh $\Omega S^{f}$. Since $\Omega S^{f*}$ and $\Omega S^{f**}$ are both created from $\Omega S^{f}$, large parts of these meshes will coincide except near the solid position. Within one time step an ALE and mapping procedure are used for projecting the old solution onto the $\Omega S^{f**}$ as accurately as possible, which will be explained next and which is also drawn schematically in Fig. 3.5.

![Schematic representation of mapping procedure using 1D meshes](image)

Figure 3.5: Schematic representation of mapping procedure using 1D meshes. The velocity field is available on Mesh A and projected through the topological identical Mesh B to Mesh C. For every node $n_i$ a solution $u_i^f$ can be found by interpolation. The distance $\Delta x_i$ between the positions in Mesh A and Mesh B at which this solution is found can be divided by the timestep to obtain the grid velocity of node $n_i$.

Firstly, an auxiliary mesh is generated by repositioning all nodes $n_i$ that lie on curve $\gamma^f$ with step size, $\Delta x_i = u_i \cdot \Delta t$, resulting in a mesh which is topologically identical to $\Omega S^{f*}$, but has a set of fluid nodes that are shifted towards the new position of the solid. Applying these steps incrementally and alternating with a Laplacian smoothing procedure as described in Section 3.4.1, prevents elements with negative Jacobian. Note that near borders of the fluid domain squeezing elements to zero-volume is possible in case of contact. This does, however, not cause problems since this auxiliary mesh will not be used for computational purpose, but is only a tool for mapping solutions from $\Omega S^{f*}$ to $\Omega S^{f**}$ more correctly.

Secondly, for every node $n_i$ in $\Omega S^{f**}$ an old velocity $u_i^f$ and grid velocity $u_i^{grid}$ have to be determined. Therefore, we need to find in which element of the auxiliary mesh $n_i$ is situated and determine the relative position within this element, after which the old velocity $u_i^f$ is computed using the interpolation functions of the $P_2 - P_1$ elements. Note that the topological similarities between the meshes contribute to fast element searching. Since the old solution is derived from the auxiliary mesh and not from $\Omega S^{f*}$, convection of this point should be accounted for by determining the global position in $\Omega S^{f*}$ based on the local position within the element. The distance between this global position and current position of $n_i$ divided by the time step gives the grid velocity $u_i^{grid}$, which can be used to incorporate the convection of point $n_i$. 
[Donea et al., 1982]. The convective part in Eq. (3.3) can then be rewritten as,

$$ \rho (u^f - u^g) \cdot \nabla u^f $$

(3.20)

Some remarks can be made with respect to the projection of the velocity field from one mesh to another. Mapping solutions between two topologically different meshes by interpolation is known to result in a non-divergence free flow, especially when the velocity gradients are large. However, this introduced error is found to be small for several reasons. Firstly, an ALE step is undertaken in order to capture the large velocity gradients across the solid more correctly. Secondly, the parts of the meshes, which are topologically different, are small. Thirdly, the velocities at the solid boundary are constrained (Eq. (3.7)) and divergence free flow enforced every time step (Eq. (3.2)), which prevents error accumulation. In addition, we should note that although zero-volume elements are allowed in the auxiliary mesh, these elements should not be taken into account for the interpolation.

### 3.5 Numerical experiments

Two model problems are presented that treat the movement of a flexible slender solid body through a fluid domain induced by a pulsatile flow that is applied at the inlet of the fluid domain. The method allows for interaction of the solid with the walls of the fluid domain. The fluid dynamics (velocities and pressures) as well as the movement and deformation of the solid (displacements and pressures) are presented. Different velocity and pressure fields at either side of the solid are shown and furthermore the computation of derived quantities like shear stresses along the solid boundary or stresses and strains inside the solid body, which are important in heart valve analysis, is possible.

#### 3.5.1 Contact of solid slab in pulsatile flow

In the first model problem we consider a rectangular fluid domain with a thin rectangular solid inside as shown in Fig. 3.6, with the following essential boundary conditions,

$$
\begin{align*}
    u_1^f &= 0 \quad \text{at } \partial \Omega_1^f, \\
    u_1^f &= \alpha \sin \left(2\pi t / T \right) \quad \text{at } \partial \Omega_5^f, \\
    u_2^f &= 0 \quad \text{at } \partial \Omega_4^f, \quad \text{with } i = 1 \ldots 4 \\
    u^s &= 0 \quad \text{at } \partial \Omega_1^s,
\end{align*}
$$

(3.21)

with $u^f = [u_1^f, u_2^f]$. Note that domain $\Omega^p$ is not incorporated in this first model problem. The solid body has a length/width ratio of 40/1 and the fluid domain a length/height ratio of 20/1. The fluid domain has a finer discretisation in the area around the solid but for the long inlet and outlet, that are introduced to prevent boundary effects, a coarser discretisation is used. Solid and fluid mesh are discretised into 20×2 quadrilateral and 3402 triangular elements, respectively. Note that this is
Figure 3.6: Schematic representation of a mitral valve $\Omega^s$ in a fluid canal $\Omega^f$. The papillary muscle is denoted by $\Omega^{mp}$ which is attached to $\Omega^s$ at one side and can slide along $\partial \Omega^s_2$ at the other with point ‘A’ prohibited to trespass the contact wall.

the number of fluid elements prior to the computations, but that this amount increases due to adaptive meshing. In this model problem parameter $\varepsilon$ is set to 0.2, which causes the actual number of fluid elements to vary between 3424 and 3472. The number of coupling elements lies between 26 and 40. Since the width of the solid is very small compared to its length, it is assumed that only coupling along boundary $\partial \Omega^s_1$ is needed. As a consequence the mesh will be adapted only along this solid boundary. For the contact problem this boundary is the deformable contact body and boundary $\partial \Omega^s_2$ is defined as the rigid contact surface in the problem. The maximum Reynolds number for this problem is 1000 and the Strouhal number is 0.6, taking the height of the canal and the maximum applied velocity at the inlet as characteristic length and velocity, respectively. Streamline plots at several moments are given (Fig. 3.7(a)-3.7(d)) to get an idea of the flow field in time. The computation starts when the solid body is about to be in contact with $\partial \Omega^s_1$. After several time steps it is in contact with $\partial \Omega^s_2$ and, induced by the applied flow at $\partial \Omega^s_2$, the solid starts to slide along $\partial \Omega^s_2$. Due to its stiffness, the solid inhibits the flow and the pressure will start to rise in the left part of the fluid (Fig. 3.8(a)). Note that the model is able to describe this pressure drop, by the combination of adaptive meshing and use of the $P_2^+ - P_1$ fluid elements.

The adaptive meshing provides for a set of fluid edges coinciding with $\partial \Omega^s_1$ and the $P_2^+ - P_1$ fluid elements are able to compute the discontinuity of the pressure across this set of fluid edges [Van Loon et al., 2004]. The solid will slide along $\partial \Omega^s_2$ until it has bent sufficiently such that the contact pressure becomes 0. Due to the elastic energy stored in the solid during bending this loosening from the wall happens very quickly even inducing a negative pressure at the left side of the tip (Fig. 3.8(b)). Contact is released and pressure in the left part of the fluid domain will start to drop caused by the squeezing flow between solid and boundary. The solid bends further until an apparent stationary position is obtained (Fig. 3.8(c)). In the second part of the computation a similar movement of the solid is seen only in reversed direction (Fig. 3.8(d)).

In Fig. 3.9 the Lagrange multiplier in the x-direction is plotted at several time steps at the moment that the solid releases from the wall. This moment is interesting since there is a pressure difference present between the left and right part of the
Figure 3.7: The streamline plots and position of the solid slab at different moments in time (a) $t/T = 0.1$, (b) $t/T = 0.12$, (c) $t/T = 0.3$, (d) $t/T = 0.59$. 
Figure 3.8: Pressure contour bands and fluid mesh at different moments in time (a) $t/T = 0.1$, (b) $t/T = 0.12$, (c) $t/T = 0.3$, (d) $t/T = 0.59$. 
fluid domain. Furthermore elastic energy is stored in the solid due to bending. This "unbalance" in energy is released in a short period of time. As mentioned earlier the physical meaning of the multiplier in our problem is the traction stress between fluid and solid, and provides us with a good impression of the actual interaction. At $t/T = 0.1165$ and $t/T = 0.1175$ the solid still makes contact with the surface and it inhibits the fluid to flow in the positive $x$-direction. However, at $t/T = 0.1185$ the valve detaches from the wall and the solid starts to accelerate resulting in a change of sign of the multiplier. For a short period the solid pushes the flow in the positive $x$-direction. The last two time steps in Fig. 3.9 again show a change in sign.

### 3.5.2 2D mitral valve

It is difficult to mimic the behaviour of a three dimensionally shaped aortic valve into a two dimensional computational model, because the shape of the valve has large influence on its overall stiffness, which cannot be captured in two dimensions. In this respect a 2D model of a mitral valve is more representative. Since the shape of this bileaflet does not contribute sufficiently to the overall stiffness nature has "invented" the chordae tendinae. These muscles that are attached to the heart wall on one side and to the leaflet tip on the other side, prevent the leaflets from prolapsing. During opening of the valve the muscle contracts thereby not hampering the leaflets. In this second model problem we aimed to mimic this behaviour by introducing domain $\Omega^p$ as shown in Fig. 3.6. This rather stiff solid is attached to $\Omega^s$ at one end allowing free pivoting, and attached to the fluid wall at point $A$ along which it can slide freely. However, this point is not allowed to pass the contact wall denoted by the dotted line.

The dimensions of the different domains are similar to the ones defined for the first model problem as are the boundary conditions, except for the essential boundary condition at the inlet $\partial \Omega^f_3$ which is replaced by a natural one reading,

$$p^I = \alpha \sin \left(2\pi t/T \right) \text{ at } \partial \Omega^f_3.$$  \hspace{1cm} (3.22)
The motion of domain \( \Omega^s \) induced by the fluid is shown in Fig. 3.10(a) over one full sinus. First, pressure will build up in the left part of the fluid domain pushing the solid to the right as given by the solid lines. Since the leaflets are flexible the solid tends to prolapse as seen in the first model problem. However, at a certain moment in time point \( A \) reaches the contact wall (Fig. 3.6), which prohibits the solid tip to move further. If the pressure in the left part of the domain drops the solid will start moving to the left as denoted by the dash-dotted lines. In order to zoom in on the behaviour near the fluid wall the \( x \) and \( y \)-position of the tip plotted as a function of time, respectively, for different discretisations of the fluid mesh (Fig. 3.10(b) and Fig. 3.10(c)). The asterisks in these graphs denote the points in time at which the solid is plotted in Fig. 3.10(a). The “coarse” mesh and “coarse +” mesh have the same discretisation except for a very thin layer of fluid elements along the fluid wall.

Figure 3.10: Motion of a 2D mitral valve in a pulsatile flow (a). The solid and dashed lines represent the motion in time for \( t/T < 0.5 \) and \( t/T > 0.5 \), respectively. The corresponding points in time are given in (b) and (c) by the asterisks. In (b) and (c) the \( x \) and \( y \)-position of the tip in time is given for three different meshes, respectively. The difference between ‘coarse’ and ‘coarse +’ is that the latter has a thin layer of elements along its contact boundary, improving solutions.
Comparison of the results shows that this layer improves the solution largely, since it influences the moment of contact and release, which are important for the overall movement of the solid. In case a layer is present the solid motion is almost similar to that computed with a fine mesh. This fine mesh has an element size about 2.5 times as small as that of the coarse meshes.

Besides discontinuous pressures, the $P_2^+ - P_1$ elements also allow for discontinuities in the derivatives of the velocities across the element borders, what can be used to compute shear stresses at both sides of the solid. Fig. 3.11(a) and Fig. 3.11(b) show the shear stress along the solid at $t/T = 0.64$ and $t/T = 0.96$, respectively, at both sides of the solid. Although the coarser mesh hardly influenced the solid motion, a certain mesh resolution is needed for capturing the shear stresses accurately.

![Figure 3.11: Shear stresses along both sides of the solid for a coarse and fine mesh at $t/T = 0.64$ (a) and $t/T = 0.96$ (b) (mitral valve computation).](image)

The Reynolds number based on average inlet velocity and height of the domain is approximately 1750 at $t/T = 0.96$. The Strouhal number at this point is about 0.09.

### 3.6 Conclusion

A method is proposed to compute the motion of a heart valve in a pulsatile blood flow. Transvalvular pressure gradients can be described and shear stresses at either side of the valve can be accurately computed. Furthermore sliding of the valve along a symmetry axis is performed in a fully coupled FSI/contact problem. The method is tested in two model problems showing these phenomena.

Using a finite element method a fluid mesh and an immersed solid mesh are generated independently of one another. The fluid mesh is adapted every time step near an arbitrarily defined boundary of the solid at which fluid-structure coupling
A fluid-structure interaction method with solid-rigid contact

should take place. Depending on where this boundary intersects a fluid element, a vertex node of this element is shifted onto the boundary or the element is split by adding a node onto the boundary. This way an inner fluid curve is created which consists of fluid edges and coincides with the chosen solid boundary. Next, a Lagrange multiplier can be defined along this fluid curve that imposes that fluid and solid velocities should equal at this curve. By imposing this constraint weakly, discretisations for the solid and fluid mesh can be different.

To describe the closing and opening behaviour of a valve it is necessary to extend this fluid-structure part with contact. Therefore, an extra constraint was added to the set of equations. The part of the solid that was likely to make contact is defined as the contact surface and a rigid surface of which the position is known. The contact surfaces are a rigid surface of which the position is known and the solid boundary. The nodal points of the solid contact surface are constrained from penetrating the rigid surface. Free slip along the rigid surface is allowed.

The weak form of fluid-structure coupling allows not only for large movements of the solid through the fluid domain, but also simplifies the introduction of contact. Since fluid and solid meshes are non-conforming with respect to each other the solid can approximate the boundaries without squeezing elements. The method combines the benefits of techniques with Lagrange multipliers, adaptive meshing and ALE methods, resulting in an implicit code solving a fully coupled system and capturing difficult behaviour with good accuracy.

The combination of these techniques were shown and discussed in two model problems. The first problem treats the movement of a flexible solid in a channel induced by a pulsatile flow at the inlet of the canal. The thin long solid is attached to the upper wall of the canal and is allowed to make contact with the symmetry axis. Throughout the computation the solid makes contact, a pressure difference builds up in the fluid across the solid, the solid releases from the wall, which lowers the pressure difference. The second problem is a 2D representation of a mitral valve thereby introducing an extra solid denoting the papillary muscle, which prevents the leaflets from prolapsing. In this problem the movement near the boundary was studied more closely and shear stresses along both sides were presented.

In this way some complexities of heart valve analysis are overcome, but some critical notes are appropriate here. The model as presented in this chapter only addresses 2D computations as 3D influences are essential in heart valve analysis. Furthermore, the compliance of the wall has not been taken into account, which is known to be of importance for the closing behaviour of a valve and the material law of the solid is rather simplistic for modelling the fiber reinforced leaflets. However, the purpose of this work was to propose a method which is flexible enough to accurately capture the movement of a solid in a fluid including a contact problem and which has the potential to being extended to 3D.
Chapter 4

A 3D method for fluid-structure interaction in heart valves

Summary: A 3D algorithm is proposed, which can be used for modelling fluid-structure interaction of flexible heart valves. A Lagrangian description for the solid and an Eulerian description for the fluid are coupled using Lagrange multipliers. An adaptive meshing method is applied to the tetrahedral fluid mesh to improve accuracy. The mesh adaptation consists of a hybrid formulation, in which existing fluid nodes are shifted onto the solid surface or new nodes (and therefore elements) are created. Four model problems are presented to address the main advantages of the method.

4.1 Introduction

Many attempts have been made to capture the dynamics of an aortic heart valve, but although progress is booked in the past, still no models are available for computing one or more cardiac cycles with a realistic three dimensional geometry and realistic material parameters. Many techniques are involved in modelling this problem. The model should incorporate large rotations, translations and deformations of the valve. Furthermore, the compliant aortic root, to which the valve is attached, influences the opening and closing behaviour of the three leaflets by transient variations in radius. In order to capture closure of the valve, a contact algorithm should be incorporated into the model and finally, the interaction between blood and valve should be addressed.

A first attempt in modelling this fluid-structure problem was taken by Peskin [1972] introducing a method which is now known as the Immersed Boundary Method [Peskin, 2002]. In an Eulerian-Lagrangian framework, the valve was represented, introducing a set of points, in which forces were imposed on the fluid domain. These points were interconnected by a generalised Hooke’s law and were able to move across the fixed fluid mesh. This method was extended for 3D problems and applied to several valve and heart simulations in e.g. Peskin and McQueen [1989]; McQueen and Peskin [1997]. Makhijani et al. [1997] also proposed a three dimensional model in which fluid and solid were solved fully coupled. They used a combination of an Arbitrary Lagrangian Eulerian approach (ALE) and remeshing for describing the large movements of the solid. In an ALE method as exploited by e.g. Hirt et al. [1974]; Hughes et al. [1981]; Donea et al. [1982], the Eulerian grid is translated according to the movement of a Lagrangian grid and is corrected by taking into account the grid velocity in the convective term. Since the shape of the elements in the Eulerian grid degenerates in case of large solid body translations, a combination with remeshing is needed. A finite volume method was used for the fluid and combined with a finite element method for the solid. A finite element model of a 3D patient specific aortic root and valve was presented by Nicosia et al. [2003]. The computations were performed in an uncoupled manner and included the opening and closing behaviour. The aortic valve and root were modelled using Hughes-Liu shell elements with linear elastic material behaviour. For stability reasons the peak diastolic pressure was reduced.

Recently, a fully coupled 3D model of an aortic root and valve was published by De Hart et al. [2003a,b, 2004]. In this approach fluid and solid were described in an Eulerian and Lagrangian framework, respectively, and coupled using a Lagrange multiplier. This variation of the fictitious domain approach, as exploited in e.g. Glowinski et al. [1997]; Bertrand et al. [1997]; Patankar et al. [2000]; Baaijens [2001], considers the fictitious solid domain immersed in a fluid domain, with meshes that are non-conforming with respect to each other. The influence of a stented valve [De Hart et al., 2003b], a stentless valve [De Hart et al., 2003a] and constitutive behaviour of the solid [De Hart et al., 2004] on the solid movement and deformation was analysed. In this last work the influence of a fiber reinforced hyper elastic Neo-Hookean material law representing the leaflet structure of collagen fibers and matrix material was analysed. Although this fluid-structure interaction method looks promising with
respect to the solid movement during systole, capturing the transvalvular pressure drop during diastole is still a problem. In the work of Stijnen et al. [2004] this problem was overcome for a stiff mechanical valve. By introducing an inner fluid curve that coincides with the position of the solid boundary at the moment of closure, the coupling between fluid and solid is strong enough to describe a pressure drop in the fluid. Van Loon et al. [2004] extended this approach for flexible solid structures by introducing a local fluid mesh adaptation which provides for the creation of this inner fluid curve every time step. This method has the additional advantage that shear stresses can be computed at both sides of the solid.

In this chapter the method as proposed in Chapter 2 is extended to 3D. Furthermore, the adaptive meshing algorithm has changed in the sense that the fluid mesh does not keep its topology anymore, which improves the method's robustness in 3D. First the governing equations will be given after which the adaptive meshing algorithm is treated. Following, several model problems in 2D and 3D are presented which address the potential of the method.

4.2 Mathematical formulation

4.2.1 Fluid

Consider a fluid domain $\Omega^f$ with the governing equations for the fluid given by the instationary Navier-Stokes equation. The balance of momentum, continuity equation and constitutive law then read,

$$\rho \left( \frac{\partial u^f}{\partial t} + u^f \cdot \nabla u^f \right) = \nabla \cdot \sigma^f \quad \text{on } \Omega^f,$$

$$\nabla \cdot u^f = 0 \quad \text{on } \Omega^f,$$

$$\sigma^f = 2\eta D - p^f I,$$

with $\rho$, $\eta$, $u^f$, $p^f$, and $I$ the density, dynamic viscosity, unity tensor, fluid velocity and fluid pressure. The gradient operators with respect to the reference and current configuration are denoted by $\nabla_0$ and $\nabla$, respectively. Since modelling of the pulsatile blood flow in the aortic root is our goal, inertia and convection terms cannot be neglected, because the physiological Strouhal (Sr) and Reynolds (Re) number read approximately 0.06 and 4500 in this region of the arterial tree. Note that, although blood shows a non-Newtonian behaviour, it will be modelled as a Newtonian fluid since it is not of importance for the models presented in this work.

4.2.2 Solid

Next the solid domain $\Omega^s$ is considered. The theory in this chapter is developed for fluid-structure interaction problems concerning flexible solids. Hence, deformations will be too large to use an infinitesimal theory such that a hyperelastic, Neo-Hookean
constitutive model is chosen for describing the solid behaviour:

\[
\nabla \cdot \sigma^s = 0 \quad \text{on } \Omega^s, \\
\text{det}(F) = 1 \quad \text{on } \Omega^s, \\
\sigma^s = G \left( F \cdot F^T - I \right) - p^s I \quad \text{with } F = (\nabla \mathbf{x})^T. 
\]

(4.4)

(4.5)

(4.6)

The quantities \( \mathbf{x}, p^s \) and \( G \) denote the position, pressure and shear modulus. Since heart valves are neutrally buoyant, gravity effects can be neglected. Furthermore, the inertia and convection term in the solid are neglected with respect to the elastic term. In the scope of this chapter, it is not necessary to model the complex material structure and behaviour typically to heart valves. A homogeneous, isotropic material law is sufficient for purposes of presenting the fluid-structure interaction method.

4.2.3 Fluid-structure coupling

Finally, the coupling between fluid domain \( \Omega^f \) and the imbedded solid domain \( \Omega^s \) is considered. The interaction between fluid and solid is modelled by applying a no-slip condition \( \mathbf{u}^f - \mathbf{u}^s = 0 \) at the boundary of the solid. Eqs. (4.1) and (4.4) are coupled by applying this constraint in a ‘weak’ manner, by means of a Lagrange multiplier. This multiplier is defined over a surface \( \gamma^f \), where integrals over fluid and solid velocity are forced to be equal,

\[
\int_{\gamma^f} q^\lambda \cdot (\mathbf{u}^f - \mathbf{u}^s) \, d\gamma^f = 0,
\]

(4.7)

where \( q^\lambda \) is a trial function. Note that \( \gamma^f \) is defined along the boundary of the solid such that \( \gamma^f \subset \Omega^s \), but also \( \gamma^f \subset \Omega^f \). In this light the physical meaning of the multiplier is the traction force between the fluid and the solid. For a more extensive discussion on the weak formulation, linearisation and the exact coupling of solid and fluid, the reader is referred to Appendix A.

4.2.4 Discretisation

The solid domain, fluid domain and Lagrange multiplier domain now need to be discretised. In this section three dimensional domains for fluid and solid are considered. However, with respect to the two dimensional model problems in the next section, the approach treated in this section can easily be deduced to the two dimensional case.

For convenience, we will first introduce some definitions concerning the mesh. Noting that the boundary of a domain will be indicated by the prefix \( \partial \), we consider an open domain \( \Omega \subset \mathbb{R}^3 \) and its boundary \( \partial \Omega \subset \mathbb{R}^2 \). This domain can be subdivided into \( N_\Omega \) number of elements \( \Omega_e \), and the corresponding elemental faces \( \Lambda_i \), edges \( \Gamma_j \)
and nodes $n_k$, such that,
\[
\Omega = \bigcup_{e=1}^{N_0} \Omega_e \quad \Omega^i = \{ e \in \mathbb{N} : \Omega_{e>i} \cap \Omega_i = \emptyset \} \quad \forall i \in \mathbb{N}
\]
\[
\partial \Omega_e = \bigcup_{i=1}^{N_A} \partial \Omega_i \quad \Lambda^i = \{ k \in \mathbb{N} : \Lambda_k = e \Lambda_i, \Lambda_{k>i} \cap \Lambda_i = \emptyset \} \quad \forall e \Lambda_i, \forall i \in \mathbb{N}
\]
\[
\partial \Lambda_i = \bigcup_{j=1}^{N_r} \partial \Gamma_j \quad \Gamma^i = \{ k \in \mathbb{N} : \Gamma_k = i \Gamma_j, \Gamma_{k>i} \cap \Gamma_i = \emptyset \} \quad \forall i \Gamma_j, \forall i \in \mathbb{N}
\]
\[
\partial \Gamma_j = \bigcup_{k=1}^{N_x} x_k \quad \mathbf{x} = \{ k \in \mathbb{N} : x_{k\neq i} \neq x_i \} \quad \forall i \in \mathbb{N}
\]

where $N_A$ is the number of faces per element, $N_r$ is the number of edges per face and $N_x$ is the number of nodal points per edge. The above states that elements in the mesh are non-overlapping and all edges and faces are straight, except for $\Gamma_i \subset \partial \Omega$ and $\Lambda_i \subset \partial \Omega$, which can be curved. A local variable is denoted using two subscripts like $\Gamma_j^i$, while a global variable is denoted using only one subscript like $\Gamma_i$. The sets $\Omega^i, \Lambda^i, \Gamma^i$ and $\mathbf{x}$ consist of all global elements, faces, edges and nodal numbers, respectively.

Now that definitions with respect to the mesh are made, domains $\Omega^f \subset \mathbb{R}^3$ and $\Omega^s \subset \mathbb{R}^2$ can be discretised in their initial configurations independently from one another to obtain the conforming meshes $\Omega^i$ and $\Omega^s$, respectively. Mesh $\Omega^f$ consists of tetrahedral elements and $\Omega^s$ consists of hexahedral elements.

### 4.2.5 Adaptive meshing

In the following, a procedure is introduced for adapting the fluid mesh based on the position of the solid mesh. Note that this procedure is only applicable if $\Omega^f$ consists of tetrahedral elements. The shape of elements in $\Omega^s$, however, can be chosen arbitrarily. The objective of adapting the mesh is to create a surface $\gamma^f \subset \Omega^f$ at which the fluid and solid will be coupled.

Consider the solid domain $\Omega^s$ and fluid domain $\Omega^f$ such that $\Omega^s \cap \Omega^f \neq \emptyset$. An arbitrary surface in the solid domain can now be chosen as a collection of $N^s$ elemental faces,

\[
\gamma^s = \bigcup_{i=1}^{N^s} \gamma^s_i \quad \text{with } i \in \Lambda^s,
\]

at which coupling between fluid and solid should be established. The fluid mesh will be adapted based on the position of boundary $\gamma^s$. Therefore, a set of edges that intersect boundary $\gamma^s$ and the corresponding intersection points can be defined as,

\[
S^f = \{ i \in \Gamma^f : \Gamma_i^f \cap \gamma^s \neq \emptyset, \Gamma_i^f \cap \gamma^s \neq \Gamma_i^f \},
\]

\[
X_i = \{ x : x \in \Gamma_i^f \cap \gamma^s \} \quad \forall i \in S^f.
\]
Thus, the points found in $X_i$ are the intersection points of a fluid edge $\Gamma_i^f$ with $\gamma^a$. If $X_i$ contains more than one intersection point, only one, arbitrarily chosen, point is taken into account in the rest of this adaptive meshing algorithm. A parameter $\varepsilon_i$ is introduced such that,

$$
\varepsilon_i = \frac{||x_i - x_{n_{1,i}}||}{||x_{n_{2,i}} - x_{n_{1,i}}||} \quad \forall x_i \in X_i,
$$

where the begin node and end node of edge $\Gamma_i^f$ are indicated by $n_{1,i} \in xS^f$ and $n_{2,i} \in xS^f$. If the length of edge $\Gamma_i^f$ is defined as $||\Gamma_i^f||=||x_{n_{2,i}} - x_{n_{1,i}}||$, it can be stated that $\forall \varepsilon_i$,

$$
S_i^{f1} = \{i \in S^f : \varepsilon_i < \hat{\varepsilon} ||\Gamma_i^f||\},
S_i^{f2} = \{i \in S^f : \varepsilon_i > (1 - \hat{\varepsilon}) ||\Gamma_i^f||\},
S_i^{f3} = \{i \in S^f : \hat{\varepsilon} ||\Gamma_i^f|| < \varepsilon_i < (1 - \hat{\varepsilon}) ||\Gamma_i^f||\}.
$$

The arbitrary parameter $0 < \hat{\varepsilon} \leq 0.5$ should be defined by the user and is explained graphically in Fig. 4.1. After defining these three sets of edges the actual adaptation of the mesh can be initiated, by first repositioning and adding nodes such that,

$$
x_{n_{1,i}} = x_i, \quad \text{with } x_i \in X_i, \quad \forall i \in S_i^{f1},
x_{n_{2,i}} = x_i, \quad \text{with } x_i \in X_i, \quad \forall i \in S_i^{f2},
\hat{x}_{\hat{n}_i} = x_i, \quad \text{with } x_i \in X_i, \quad \forall i \in S_i^{f3},
$$

in which $\hat{x}_{\hat{n}_i}$ is the position of a newly created node $\hat{n}_i$ on edge $\Gamma_i^f$. With respect to the repositioning of the begin or end point of an edge, we impose that the position of a node is allowed to be changed only once, even if this node is shared by several intersected edges.

By adding new nodes, a cascade in topological changes takes place in the fluid mesh. Every edge, $\Gamma_i^f$, with a newly created node at the intersection point will be subdivided into two new edges, $\tilde{\Gamma}_i^f$ and $\tilde{\Gamma}_i^f$, such that,

$$
\tilde{\Gamma}_i^f = \{x \in \Omega^f : x = (1 - \alpha)x_{n_{1,i}} + \alpha x_{\hat{n}_i}, \quad 0 \leq \alpha \leq 1\},
\tilde{\Gamma}_i^f = \{x \in \Omega^f : x = (1 - \alpha)\hat{x}_{\hat{n}_i} + \alpha x_{n_{2,i}}, \quad 0 \leq \alpha \leq 1\}.
$$
Depending on the number of intersections (one or two per face), an elemental face can be subdivided into two or three subfaces as shown in Fig. 4.2.

![Subdivision Examples](image)

(a) (b) (c)

Figure 4.2: Different configurations for subdividing tetrahedrons with 1 intersection into 2 triangles (a), with 2 intersections into 3 triangles (b) and (c)

Note that there are two different ways of dividing a face into three subfaces (Fig. 4.2(b), 4.2(c)), one of which can be chosen arbitrarily (or based on, for example, angles). With respect to the tetrahedral elements, we consider five different configurations one of which can always be obtained by rotation of an intersected element (Fig. 4.3(a)-4.3(e)). More configurations are thinkable but in case surface $\gamma^*$ is smooth these will not occur and therefore not be treated here.

![Subdivision Variations](image)

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 4.3: Basic configurations for tetrahedrons with 1 intersection (a), 2 intersections (b), 3 intersections (c)-(d), and 4 intersections (e). Extra edges are created by subdividing the triangular faces. Since there are two ways of subdividing a face with two intersected edges, many tetrahedron subdivisions are possible, but for each configuration (a)-(e) one example is given (f)-(j), respectively.

Depending on the number and distribution of the intersections over a tetrahedron, two to six new non-overlapping subtetrahedrons will be created that replace the old tetrahedron. The edges of these tetrahedrons will be straight and the sum of their
volumes will equal the volume of the original tetrahedron. Every single subface will be the face of a newly created tetrahedron. As mentioned earlier, there are different ways of creating new subfaces and hence there are many different solutions for subdividing a tetrahedron into subtetrahedrons. For each of the presented configurations (Fig. 4.3(a)-4.3(e)) a corresponding example is shown in Fig. 4.3(f)-4.3(j). Note that for small values of parameter $\varepsilon$ more new elements will be created in the new mesh denoted by $\Omega^{s'}$. However, the created elements will be increasingly ill-shaped. Now, using the elemental faces $\Lambda^f_i \subseteq \Omega^f$, of the adapted fluid mesh, the new subdomain $\gamma^f \subseteq \Omega^f$ can be defined as,

$$\gamma^f = \{ x \in \Lambda^f_i : x_{n_{1,i}} \in \gamma^s, x_{n_{2,i}} \in \gamma^s, x_{n_{3,i}} \in \gamma^s \} \quad \forall i \in \Lambda^{s'},$$

(4.8)

where $n_{1,i} \in x^{s'}_i$, $n_{2,i} \in x^{s'}_i$ and $n_{3,i} \in x^{s'}_i$ are the nodes at the vertices of a triangular face $\Lambda^f_i$. Since $\gamma^s$ is part of the solid, which moves through the fluid domain during computations, domain $\gamma^f$, which is based on $\gamma^s$, will also change every time step.

### 4.2.6 Element choice

For the three dimensional model problems presented after this section, the 15 noded quadratic tetrahedrons ($P^+_2 - P_1$) and 27 noded quadratic hexahedrons ($Q^+_2 - Q_1$) of the Crouzeix-Raviart family are used. For the two dimensional model problems their 2D equivalents are used, i.e. the 7 noded triangle for the fluid and the 9 noded quadrilateral for the solid. In all of these elements the pressure is enforced to be continuous inside an element but can be discontinuous from one element to another. This property will be shrewdly used for capturing a moving pressure discontinuities in the fluid domain as will be demonstrated in the next section.

The fluid mesh and solid mesh are generated independently from each other. The solid is modelled with an updated Lagrange formulation which results in an update of the material points of the mesh every time step, i.e. the solid mesh moves across the Eulerian fluid mesh in time. The initially generated fluid mesh $\Omega^{s'}$ is adapted every timestep to the new position of the solid boundary $\gamma^s$, creating surface $\gamma^f$ at which Eq. (4.7) applies. The adapted mesh $\Omega^{s''}$ is used for the computation and after each time step the position of the solid is updated and a new adapted mesh $\Omega^{s'''}$ is created based on $\Omega^{s''}$ and the new position of the solid. Finally, solutions are mapped from $\Omega^{s''}$ to $\Omega^{s'''}$ using the basis functions of the tetrahedral elements.

By taking the discretisation of the coupling elements based on the discretisation in $\Omega^{s''}$, the coupling is enhanced, since the number of coupling of elements that are connected to a fluid element is one. Piecewise linear discontinuous basis functions are used to describe the Lagrange multiplier $\lambda$. Care should be taken with respect to the coarseness of the fluid and solid discretisations. The mesh size of the fluid and Lagrange multiplier $h^f$ should not differ too much from the mesh size of the solid $h^s$. If the ratio $h^f/h^s << 1$, accuracy improves but the condition of the algebraic system deteriorates and, vice versa, if $h^f/h^s >> 1$ the condition of the algebraic system improves but accuracy goes down.
An implicit Euler time integration scheme is used for the solid as well as the fluid. The finite element package 'Sepran' Segal [2003] is used and extended for building the matrix and mapping the solutions. The set of equations is solved fully coupled and the asymmetric sparse matrix is solved using a direct method based on a sparse multifrontal variant of Gaussian elimination (HSL/MA41) HSL [2002].

4.3 Results

By means of several model problems, the abilities and inabilities of the described method will be analysed. Firstly, a simple plane strain example is presented, showing large rotations and translations of a slender body in a pulsatile fluid flow. Secondly, shear stresses at both sides of a similar slender body are computed in two different ways. The third model problem considers the axisymmetric modelling of a fluid domain which is halfway divided by a thin solid membrane. A pressure applied to the fluid domain will cause the solid membrane to strain resulting in a sharp pressure drop when the problem reaches equilibrium state. Finally, a three dimensional problem is presented which mimics the third model problem. The adaptive meshing procedure is highlighted and results are compared to model problem three.

4.3.1 Model problem 1: movement of a solid slab

A rectangular plane strain fluid domain is considered with no-slip conditions at bottom and top wall. At the inlet the velocity (plug flow) is prescribed as a sinus function of time, which results in a pulsatile flow. The fluid domain is discretised into quadratic triangular elements and the solid domain into quadratic quadrilateral elements. Since a slender body is considered, only one of the solid boundaries is defined as \( \gamma' \) and the influence of the fluid underneath the fictitious solid body can be neglected. The flow induces the flexible slab, that is attached to the upper fluid wall, to move. The slab in its turn interacts with the fluid which highly influences the flow behaviour as shown by the streamline plots (Fig. 4.4). In this example the Reynolds number is 1000 based on the height of the channel and the average peak velocity at the inlet. Note that in order to circumvent boundary influences, the height:length ratio of the domain is taken 20:1. However, very coarse triangulations are used towards the inlet and outlet boundary.

4.3.2 Model problem 2: shear stresses along a rigid plate

This problem shows the ability of the method to capture the shear stresses at both sides of a slender body. Again a rectangular fluid domain is modelled but now with an immersed stiff solid plate skewly positioned in the middle of this domain (Fig. 4.5(a)). The same elements are used as in the former problem and again one boundary (in the length of the plate) is defined as \( \gamma' \). A stationary solution for the Navier-Stokes equation is found by applying a velocity plug flow at the inlet. The problem is solved twice, once using the method described in this chapter, and once by applying homogeneous Dirichlet boundary conditions in all fluid nodes of \( \gamma' \) and not taking
Figure 4.4: Streamline plots of the pulsatile fluid flow interacting with a thin solid slab.

Figure 4.5: Vector field of rigid plate in fluid domain (a) and the corresponding shear stress graphs at both sides of the plate (b). The problem is solved once as a fluid-structure interaction model and once as a fluid problem with Dirichlet BC’s in a set of internal nodes.
the solid equations into account. For both problems the shear stress is computed at both sides of the solid and the results are compared in Fig. 4.5(b). Although the computation with fluid-structure interaction shows some small mesh-dependent oscillations, the differences with the fluid computation are negligible.

4.3.3 Model problem 3: pressure drop in fluid domain (axisymmetric)

This model is used to emphasise the capability of the method to capture steep pressure gradients inside the fluid domain. A circular fluid domain is considered with a thin solid membrane dividing this domain halfway. This problem can be solved using an axisymmetric model, which results again in a rectangular fluid domain where the upper wall is the symmetry axis. The solid domain is defined from the axis to the bottom wall. The solid as well as the fluid are to prohibited from moving/flowing through the axis but are free to move along the axis. The solid is attached to the fluid wall at which a no-slip condition applies. Unlike the former model problems a stress in axial direction is applied depending on time, by taking the first quarter of a sinus function (with periodic time \(T=50\)) and then keeping the pressure at a constant level such that equilibrium is reached. The fluid will start to flow and the pressure will build up in the left side of the fluid domain inducing the solid to bend (Fig. 4.6).

![Figure 4.6: Axisymmetric model of a solid membrane in a fluid domain with applied pressure at the left wall. The solid stopped moving but the fluid still flows at \(t/T=22\).](image)

After some oscillating movement due to inertia forces, solid and fluid reach the equilibrium state. At this point in time the pressure in the left part of the fluid domain will be the applied stress, the pressure in the right part of the domain will be zero and the fluid flow in the whole domain will be zero.

4.3.4 Model problem 4: pressure drop in fluid domain (3D)

A similar circular fluid domain with thin solid membrane can be modelled in three dimensions. A quarter of the cylinder is discretised in 15 nodded quadratic tetrahedral elements, the 3D analogon of the 2D extended quadratic triangles as used in the former problems. The solid membrane is modelled by defining a mesh consisting hexahedral elements. The same boundary conditions apply as in the third model problem, which results in a fluid domain with two symmetry edges and a wall with no-slip condition and a solid fixed at the fluid wall. One of the two boundary surfaces of the solid
that divide the fluid domain in two parts is taken as $\gamma^s$ resulting in a fluid surface $\gamma^f$ after adapting the fluid mesh. In Fig. 4.7(a) it is clearly seen that these surfaces are non-conforming with respect to each other.

Figure 4.7: The non-conforming surfaces $\gamma^s$ and $\gamma^f$, which are part of the fluid and solid mesh, respectively. A Lagrange multiplier is defined over these surfaces by which fluid and solid are coupled (a). Clockwise from the upper right picture, the inner fluid surface, $\gamma^f$ is shown for time steps $t/T=0.6$, $t/T=4.8$, $t/T=10.8$ and $t/T=18$, respectively (b).

As mentioned earlier, the position of $\gamma^s$ changes in time coupling as will surface $\gamma^f$, which is based on the intersection of $\gamma^s$ with the fluid mesh (Fig. 4.7(b)). As for the axisymmetric example in equilibrium state the fluid domain is divided into a part with the applied pressure and a part with zero pressure as shown in Fig. 4.8.

Figure 4.8: The solid membrane surface, $\gamma^f$, and the newly created inner fluid surface, $\gamma^f$.

By following the point of the membrane at the axis in time, we can make a comparison between model problem three and model problem four. The axial displacements are graphically presented as a function of time in Fig. 4.9. There is hardly any difference
Figure 4.9: The z-displacement of the center of the solid membrane as a function of time

in these transients except for the equilibrium position. This is caused by the coarseness of the coupling surface near the axis point in the 3D mesh. The initial solid and fluid mesh consist of 80 and 3600 elements, respectively. During adaptation, with parameter \( \varepsilon = 0.2 \), a variation of 200 to approximately 400 fluid elements is added to this number as well as 150 to 175 coupling elements. As shown for the 2D case in Chapters 2 and 3, the introduction of an inner-fluid surface also allows for jumps in shear stress along this surface. Due to the solid the fluid domain is basically dissected into two separate parts with a moving no-slip boundary at position of the solid. The distribution of the shear stresses that act upon both sides of this boundary is shown in Fig. 4.10. As mentioned earlier the membrane has stopped moving but the fluid

Figure 4.10: Shear stress along the front (left) and back (right) surface of the membrane at \( t/T=22 \) (see also Fig. 4.6)
still flows at this point in time as was also shown in Fig. 4.6.

### 4.4 Conclusion

A method is proposed suitable for modelling the fluid dynamics of a flexible heart valve in a fluid domain. Large rotations and translations of a slender solid body, induced by a pulsatile fluid flow, can be captured at high Reynolds numbers. Furthermore, accurate shear stress information at both sides of the solid can be computed as well as a large pressure gradient across the solid. The potential of the method is demonstrated, considering three 2D and one 3D model problem.

We consider a thin solid body immersed in a fluid domain. The finite element meshes of solid and fluid are generated independently from one another and are therefore non-conforming with respect to each other, which makes meshing easier. Both domains are coupled by applying a Lagrange multiplier along the boundary of the solid, which enforces the interpolated fluid and solid velocities to be equal. In addition, an adaptive meshing scheme is introduced, which adapts the fluid mesh to the position of the solid. This mesh adaptation is only applied locally near the position of the solid and is computationally inexpensive and robust since the fluid and solid mesh do not have to be conforming. Furthermore, steep pressure gradients across the solid can be captured, making use of the discontinuous pressure description of the quadratic fluid elements. Similarly, gradients of the velocity and therefore shear stresses can be discontinuous as has been shown. In this chapter, the mathematical formulation is presented in 3D, but demonstrated using 2D and 3D model problems.

Before the framework, as presented here, can be used for analysing heart valve dynamics, some critical points should be addressed first. Although the different model problems show situations analogue to systole and diastole a combination as presented for two-dimensional problems in Chapter 3 should still be performed. To model the closing behaviour of a valve, the contact algorithm of Chapter 3.2.3 needs to be incorporated into the 3D model. Furthermore, the computing time for 3D problems is very large, which leads to concessions in mesh resolution. However, the boundary layer, that scales with $1/\sqrt{Re}$, will become very small at the leaflet, making it very hard to compute shear stresses accurately. The solution for this problem can be sought in different areas. A more appropriate solver could be used for solving the fully coupled set of equations. Currently, iterative solvers like GMRES and CG with an incomplete LU decomposition need much fill-in in order to converge and the direct solver used in this chapter is also clearly not ideal. Another way of saving CPU time, would be a more efficient way of meshing. The adaptive meshing could be extended such that the element size in the vicinity of the solid can be reduced to a level with which shear stresses can be computed accurately. Finally, more efficient alternatives might be found for the fifteen noded tetrahedrons.
Chapter 5

3D dynamical analysis of an aortic heart valve in pulsatile flow

Summary: A 3D model of a flexible aortic heart valve is presented, that is capable of capturing the dynamical interaction of a pulsatile blood flow with the slender leaflets. The aim of the study described in this chapter is to evaluate to what extent the developed FSI method is capable to accurately describe three-dimensional valve mechanics and dynamics.
5.1 Introduction

The most extensively studied heart valve is the aortic heart valve. To gain thorough understanding of its function (and dysfunction) by means of computational methods, the interaction between the valve, the aortic root and the blood should be incorporated. As already mentioned in Section 1.5.2, several FSI methods and 3D models for flexible aortic heart valves have been proposed in the past, the most cited ones addressed briefly in the following.

Peskin [1972] was the first to present a two-dimensional FSI method for heart valves that was called the Immersed Boundary Method. Later, three-dimensional FSI models have been presented that did not only include the aortic valve behaviour but also that of the left ventricular wall and mitral valve. In McQueen and Peskin [1997] a large finite difference grid for the fluid is considered in which the ventricular walls and valves are represented by interconnected fibers. At the end points of these fibers local body forces are introduced that provide the necessary coupling between solid and fluid. An advantage of this method is that the fibers can move through the fixed fluid grid without requiring alignment. The veins and arteries of the model heart are represented by hydraulic resistances to constant-pressure reservoirs. The sets of equations for fluid and solid are solved in a staggering scheme using dedicated solving techniques for the different parts.

Another method that was used to model the dynamics of heart valves was proposed by Makhijani et al. [1997]. He used a finite volume description with a moving grid capability for the fluid and a finite element description for the solid. Making use of symmetry, only 1/6th of the valve is modelled. Equilibrium between fluid and structure is obtained in an iterative loop, i.e. first the fluid velocities and pressures are computed after which a fluid force is applied to compute leaflet deformations and subsequently these solid displacements are applied to the fluid. This procedure is repeated until convergence. Contact of the leaflets with the wall of symmetry is incorporated by introducing non-linear springs that show increasing stiffness with decreasing gap distance.

In the models presented by De Hart et al. [2003b,a, 2004] a fictitious domain approach is applied, that can be seen as the finite element version of the Immersed Boundary Method. A fluid mesh is generated disconnected from an overlapping solid mesh, and the latter is allowed to freely move without altering the former. Coupling is obtained by the introduction of a Lagrange multiplier along the surface of the solid, which locally constrains the fluid and solid velocity to be equal. Several valve models with a flexible root have been proposed that all assume symmetry like in Makhijani et al. [1997]. Using these models the influence on the valve motion was analysed for stented [De Hart et al., 2003b], non-stented heart valves [De Hart et al., 2003b], and for fiber-reinforced leaflets [De Hart et al., 2004]. The set of equations was fully coupled and solved in a backward Euler scheme.

Finally, the computational model proposed by Nicosia et al. [2003] is mentioned, which is based on a three-dimensional patient specific valve and root. Using a standard explicit finite element package several cardiac cycles were computed. The root and leaflet tissue were modelled with elastic material properties. Coaptation of the leaflets
was incorporated by a contact algorithm using a penalty parameter.

To the writers knowledge no models are yet published that are able to provide information about the shear stresses along both sides of the leaflets. In Chapters 2, 3 and 4 a new method has been proposed that would be suitable for modelling the dynamical behaviour of the aortic valve and the blood during a full cardiac cycle and provide transient shear stress information. This chapter addresses the application of the developed method to an aortic heart valve model with FSI. It first briefly explains the generation of the leafet mesh after which two models are posed. The first treats the deformation of an aortic valve under pressure in a solid computation without including the fluid. For the second problem the fluid is incorporated in the model and systolic valve motion is presented. At the end of the chapter the current model is evaluated and associated difficulties are discussed.

5.2 Computational solid model for the valve

5.2.1 An aortic heart valve mesh

A finite element mesh is generated for the aortic valve according to the observations and descriptions of Thubrikar [1990]. A perfectly symmetrical valve shape can be created based on five parameters, i.e. the valve will consist of three identical leaflets and every leaflet is symmetrical to itself. The advantage of this approach is that only half a leaflet can be modelled and that the parameterised valve geometry can be altered easily by tuning the five parameters. The procedure for generating the aortic valve mesh will now be addressed shortly.

First, the surface $S_1$ of a topped cone is defined with an upper radius of $R_1$, a bottom radius of $R_2$ and a height $H$ as shown in Fig. 5.1(a). For clarity only one leaflet will be generated here, which can be copied twice around the cones axis in the end. Therefore only $1/3$ of the cone surface needs to be considered as indicated in Fig. 5.1(b).

The cone surface is intersected by a plane that is defined through points $C_1$, $C_2$ and $A$ generating the intersection line $L_1$ (Fig. 5.1(b)). Note, that the points $C_1$ and $C_2$ will become the commissure points of the leaflet. A vector can be defined as $\mathbf{v}_1 = \tan(\alpha)e_y + e_z$, with $e_x$, $e_y$ and $e_z$ the orthonormal vectors in $x$, $y$ and $z$ direction, respectively, and $\alpha$ the angle between the $z$-axis and vector $\mathbf{v}_1$. Now, another surface $S_2$ can be generated by projecting the intersection line $L_1$ in the direction of $\mathbf{v}_1$ onto plane $z = H + dh$ (Fig. 5.1(c)). Note that surface $S_2$ becomes leaflet surface and that parameter $dh$ is an offset parameter to enlarge the leaflets. Surface $S_2$ is then mirrored in the plane through $C_1$, $C_2$ and $A$ (Fig. 5.1(d)) and cut off by two planes that go through the $z$-axis and the two commissure points, respectively (Fig. 5.1(e)). One leaflet surface is now defined based on parameters $R_1$, $R_2$, $H$, $dh$ and $\alpha$, the values of which are shown in Table 5.1. By copying this leaflet two times around the $z$-axis the total valve is obtained (Fig. 5.1(f)). A uniform leaflet thickness $t$ is obtained by projection of the leaflet surface in normal direction, and discretisation of the volume between these leaflet surfaces with 27-noded hexahedral elements gives the finite element mesh as shown in Fig. 5.2 with one layer of 588 solid
Figure 5.1: Different stages for generation of an aortic heart valve: starting with a topped cone (a), one third of the cone is intersected by a plane through points $C_1$, $C_2$ and $A$ (b), projection of $L_1$ in direction of $v_1$ (c), mirroring in plane (d), cut off by planes through $C_1$ and $C_2$ (e) and multiply by three (f).
Table 5.1: Parameters used to generate the leaflet geometry.

<table>
<thead>
<tr>
<th>$R_1$</th>
<th>$R_2$</th>
<th>$H$</th>
<th>$\Delta h$</th>
<th>$\alpha$</th>
<th>$l$(mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>1.2</td>
<td>1.78</td>
<td>0.1</td>
<td>0.0</td>
<td>0.11</td>
</tr>
</tbody>
</table>

elements.

![Three dimensional view of the valve mesh.](image1)

![Boundary conditions for half a leaflet.](image2)

**5.2.2 Solid model description**

Heart valve models have been proposed that do not incorporate the interaction between solid and fluid [Black et al., 1991; De Hart et al., 1998; Driessen et al., 2005]. Although these types of models are not suitable for analysis of the opening and closing phase of the cardiac cycle, they give a good representation of the closed phase where valve motion is limited and mainly pressure driven. Therefore, a solid analysis is performed first, to get more insight in the behaviour of the leaflet geometry and to reduce the complexity of the model. Considering half a leaflet, i.e. one sixth of the total valve, three borders can be distinguished on which essential boundary conditions will be applied (Fig. 5.3). Since the aortic root is not incorporated in the model and walls are therefore rigid, one boundary will be fixed in space. The symmetry boundary of the mesh is restricted to move in normal direction to that boundary. Finally, the free edge of the leaflet normally coaps with another leaflet and is, therefore, in this model not allowed to penetrate the contact symmetry surface, but is allowed to pull itself off by introducing a contact algorithm as described in Section 3.2.3. The driving force in the model is a uniformly distributed stress applied in normal direction on the leaflet surface at the aortic side and is imposed incrementally. A total pressure of 5
kPa is applied in steps of 100 Pa. The valve material was considered incompressible with a shear modulus of 0.1 MPa.

5.2.3 Solid model results

When pressure is primarily applied, the part of the free edge that does not coapt is pushed towards the axis. This introduces a tensile strain along the leaflet symmetry boundary and will induce a lateral strain in circumferential direction. It is due to this lateral strain that part of the free edge that was initially in contact, is now pulled from the contact symmetry edge. Note that this is a relevant observation with regard to the FSI computation with the same aortic valve, that will be presented in Section 5.3. When pressure increases further, the free edge is pushed against the contact wall again and eventually resulting in a larger coaptation area as can be seen in Fig. 5.4(a). An indication of the deformation of the valve at this stage is given in Fig. 5.4(b), which shows the leaflet symmetry line of the valve initially and at pressure 2.7 kPa. As can be seen, deformations become large near the fixed wall at this pressure level causing numerical instabilities.

![Deformed valve mesh](image)

**Figure 5.4**: The deformed valve mesh after applying a pressure of 2.7 kPa in normal direction of the valve (a). The symmetry line A-B in initial and deformed position are denoted by the solid and dashed line, respectively (b).

At this point the first and second principal strains are in the radial and circumferential direction (Fig. 5.2.3), which is equal to the direction of the collagen architecture that is observed in native valves. Computational models for tissue adaptation have been developed that are based on this principal [Driessen et al., 2003].
5.3 Computational fluid-structure interaction model

5.3.1 FSI model description

Since solid modelling can only provide information about the mechanical behaviour in the closed phase of the cardiac cycle, the model proposed in Section 5.2.2 is extended with a fluid that interacts with the leaflets. For this purpose the FSI method as proposed in Chapters 3 and 4 is used. The fluid mesh is a straight channel, that is 13 cm long to avoid boundary effects and that consists of 3780 15-noded tetrahedral elements. The valve is positioned halfway the fluid domain and the fluid mesh is adapted to the ventricular leaflet surface in order to apply the coupling constraint. The boundary conditions for the valve equal those defined in Section 5.2.2 except for the applied normal stress, that is now redundant due to the FSI. The boundary conditions for the fluid domain are similar to those for the solid domain, prohibiting flow perpendicular to the symmetry surfaces and imposing a no-slip condition at the fixed wall surface. When modelling a native valve in the in-vivo situation, physiological pressure gradients would be ideally prescribed at the inlet and outlet. However, the model proposed here is based on a polyurethane valve (Fig. 5.6), for which pressure information is not available in advance. Therefore, it was chosen to prescribe the velocity as a plug flow at the inlet of the domain which is convenient for testing, since it provides \textit{a priori} information on the Reynolds and Strouhal numbers. For simplicity, the velocity at the inlet varies as a sinus function of time with periodic time $T=1$ s. At the outlet the pressure is set to zero. The values used for the maximum flow applied at the inlet, $v_{\text{max}}$, the shear modulus of the solid material, $G$, time step, $\Delta t$, the dynamic viscosity of fluid, $\eta$ and the density of the fluid, $\rho$, are shown in Table 5.2. A hyperelastic Neo-Hookean material law is used for the solid that allows large deformations. The fluid is assumed to behave in a Newtonian manner. Finally, with regard to the numerical method it should be mentioned that a backward Euler scheme is used for time integration and the set of equations is solved in a fully coupled technique.
Figure 5.6: Experimentally used poly-urethane valve (left) and corresponding computational geometry (right)

<table>
<thead>
<tr>
<th>$v_{\text{max}}$ (ml/s)</th>
<th>$G$ (MPa)</th>
<th>$\Delta t$ (s)</th>
<th>$\eta$ (Pa·s)</th>
<th>$\rho$ (kg/m$^3$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>180</td>
<td>10</td>
<td>$10^{-3}$</td>
<td>$4 \cdot 10^{-3}$</td>
<td>$10^3$</td>
</tr>
</tbody>
</table>

### 5.3.2 FSI model results

The different valve configurations during the opening and closing of the valve are shown in Fig. 5.7. The initial valve geometry has a coaptation area with perfectly aligned leaflets leaving only a small triangular orifice in the center of the valve (Fig. 5.2). When fluid starts to flow the path of least resistance is through this orifice, pushing the free edge of the valve open. However, near the commissures the leaflets obstruct the flow, causing pressure to rise and the valves to open (Figs. 5.7(a)-5.7(c)). The valve then opens further, first showing the typically triangular orifice (Fig. 5.7(d)) and subsequently developing to a more circular shape (Figs. 5.7(e)-5.7(g)). When the valve starts closing it deforms back along a rather similar path as can be seen by comparing Fig. 5.7(d) and 5.7(j) or Fig. 5.7(e) and 5.7(h).

In Chapter 4 the three-dimensional method was proposed, in which the fluid mesh is adapted to the solid position and such creating an inner fluid surface along the solid boundary. Looking at this adaptive meshing procedure throughout the computation, the leaflets are covered quite well with coupling elements depending on the coarseness of the initial fluid mesh (Fig. 5.8). With regard to the fluid mesh an additional amount of elements between 206 and 325 is generated and 85 to 143 coupling elements are created at each time step due to adaptive meshing. As seen for the two-dimensional problems presented in Chapter 2, only the free edge of the valve is not fully covered. To prevent blood flow through these uncoupled parts of the leaflet, a Lagrange multiplier is introduced along the free edge of the leaflet introducing coupling with the fluid similar to De Hart et al. [2003b].
Figure 5.7: Different stages of valve opening from \( t/T = 0.01 \) to \( t/T = 0.55 \)
5.3.3 Discussion: some computational difficulties

Capturing pressure gradients across the valve and describing shear stresses along the leaflet surfaces have been the main focus for developing the FSI method as presented in Chapters 2, 3 and 4. Although the method showed the potential to meet these two requirements in the various model problems proposed, several additional difficulties are encountered for a three-dimensional computational model of an aortic valve.

The three-dimensional numerical experiment, described in Section 4.3.4, showed the development of a fluid pressure gradient across a solid membrane in time. In this problem, the fluid domain was fully dissected by the membrane not allowing fluid to flow from one part to the other. In an aortic valve problem, however, the valve does not cover the whole cross-sectional fluid area having a small triangular opening in the middle. When the pressure gradient starts building up across the leaflet, the free edge at this opening is pushed to the center axis of the fluid domain in direction of the leaflet symmetry axis. Subsequently, this induces lateral contraction, which pulls the leaflets from each other in the coaptation area as observed in Fig. 5.7(1). Similar observations were performed in an experimental setup with an equally-shaped polyurethane valve with a straight compliant root (Fig. 5.6) and even the solid model proposed in Section 5.2.3 shows this behaviour, though to a lesser extent. Although the existing gap does with rising pressure on the leaflets in the presented solid model, it causes numerical problems in our FSI computation. The inner fluid surface that is computed every time step has a rather irregular shape along the free edge of the leaflet as seen in Fig. 5.8. The solid deformation is not significantly influenced by this irregular coupling due to its stiffness, but the fluid is far more susceptible to these irregularities. Since the fluid and solid are coupled in a weak sense as described in Section 3.3, the fluid solution does not follow the stable solid solution in every point, which can cause spurious solutions. In subsequent time steps the solid solution is affected by the introduced errors with divergence of the Newton-Raphson loop as result. One might conclude that the occurring instabilities are induced by the
prescribed velocity on the fluid domain in combination with the incompressibility of the fluid which forces the valve to fully close and deform or to leak. However, tests have been performed in which negative pressures instead of velocities were applied at the inlet. In these computations similar instabilities were obtained with rising pressure.

With regard to the shear stresses in this problem we can state that the current fluid mesh is too coarse to compute accurate shear stresses along the leaflet surfaces. As mentioned earlier, the errors in the velocity field near the free edge of the leaflets are a result of the large velocity gradients present in that area. These errors are restricted to a local area, not influencing the coupling towards the belly of the leaflet. However, the discretisation of the inner fluid surface and therefore that of the Lagrange multiplier is not fine enough to capture velocity gradients across the surface. Needless to say that the accuracy of the derivatives is poor. Expectations are, however, that refinement of the fluid mesh will result in more accurate shear stresses.

5.4 Conclusion

A heart valve model is proposed with FSI in a pulsatile flow at physiologically relevant Reynolds numbers. The FSI method used, was developed in Chapters 2, 3 and 4 and showed capable of computing the leaflet movements/deformations from which derived quantities like stresses and strains can be determined. With a prescribed fluid velocity or pressure as the only driving force a full opening and closing behaviour of the aortic heart valve is computed. Some remarks should however be made. The main advantages with regard to the FSI method, namely sustaining pressure in closed state and computing accurate shear stresses along the leaflet, could not be exploited in the current model. Although the method performed well for two-dimensional models in which a slender body moves through the fluid and coaps along a fluid symmetry wall as discussed in Section 3.5, difficulties arise if extended to three dimensions. Firstly, the velocities in the fluid domain near the free edge of the leaflet tend to be non-smooth, even causing divergence if the valve is nearly closed. Secondly, the fluid mesh is currently too coarse to compute accurate shear stresses along the cusps. Both these problems should be addressed before the model has an additional value to present models [Peskin and McQueen, 1980; Makhijani et al., 1997; De Hart et al., 2003b; Nicosia et al., 2003]. Additionally, it should be mentioned that the current model has a rigid root, no sinusses of Valsalva incorporated and no fiber reinforced leaflet material, although all have proven to highly influence the mechanical and dynamical behaviour of the valves.
Chapter 6

Conclusions and Discussion

Summary: In this chapter general conclusions on the study performed, are given. The different aspects of the developed FSI method are discussed and critical notes are made. In addition, some recommendations with respect to modelling and model application are proposed.
6.1 Introductory remarks

A limited number of FSI models has been proposed in the past that address the interactions in the aortic root [Peskin and McQueen, 1980; Makhijani et al., 1997; De Hart et al., 2003b; Nicosia et al., 2003; De Hart et al., 2004], each having its own advantages and disadvantages. The models proposed by Peskin and De Hart were based on methods with a fixed grid/mesh for the fluid domain with an immersed solid domain, which resulted in high flexibility of the method. Drawback of this approach was, however, that near the valve boundary the accuracy of the solutions for the velocity field and consequently for the shear stresses was poor. Furthermore, the diastolic transvalvular pressure gradient could not be captured unless unrealistic viscous flows were considered. These problems are of no concern if a method is applied like in the model of Makhijani et al. [1997], which incorporates continuous updating of the fluid mesh in order to keep the meshes for fluid and solid conforming. This way solutions near the valve can be computed more accurately, but, on the other hand, the remeshing can be a difficult and time consuming task, and may influence the accuracy of the solution.

The aim of this study was to combine the main advantages of the existing methods. The proposed method considers a fluid domain, described in an Eulerian manner, and an immersed solid domain with a Lagrangian description [De Hart et al., 2000], and additionally a local adaptive meshing scheme is incorporated, that adapts a basic fluid mesh to the position of the solid each time step. Since the fluid and solid mesh are coupled using a Lagrange multiplier, it is not required for these meshes to be conforming with regard to each other, which simplifies the required meshing scheme.

6.2 The fluid-structure interaction method

In Chapter 2 a two-dimensional FSI method has been introduced with an adaptive meshing scheme that only repositions fluid nodes but does not alter the topology of a basic mesh. In this procedure, the fluid mesh is altered near the solid position each time step such that a curve of elemental fluid edges coincides with the boundary of the solid. In combination with elements from the Crouzeix-Raviart family, i.e. continuous pressure description within an element but not across element borders, the transvalvular pressure gradient can be modelled as a sharp pressure drop. In addition, a similar explanation holds for the shear stresses, that can also be discontinuous across element borders. The two corresponding model problems show the advantages of the new method over the fictitious domain method as introduced by Baaijens [2001] and De Hart et al. [2000], by sustaining a (diastolic) fluid pressure across a membrane and by computing the shear stresses along both sides of a moving slender body.

Subsequently, in Chapter 3 the separate model problems proposed in Chapter 2 can be combined by extending the method with a rigid-solid contact algorithm, that enables the computation of the transition phase between the open and closed position of the valve. Similar to the fluid-solid coupling this contact algorithm is based on the introduction of a Lagrange multiplier. Secondly, the adaptive meshing scheme is enhanced in this chapter towards a hybrid formulation with combined node
repositioning and element adding, which makes the scheme more robust especially with regard to extension of the method to 3D. Thirdly, an improved mapping algorithm is introduced to transfer fluid velocities more correctly from the old adapted mesh onto the newly created one each time step. This enhanced method is successfully tested in two numerical experiments, the first treating a prolapsing slab and the second considering a two-dimensional model of a mitral valve including a chorda tendina to prevent prolapse. The solid bodies are exposed to a high-Reynolds pulsatile flow with \( Re=1000 \) for the former problem and \( Re=1750 \) for the latter and, additionally, unsteady inertial effects are substantial in both numerical experiments with \( Sr=0.6 \) and \( Sr=0.09 \), respectively. Note that the height of the channel and the maximum velocity at the inlet are the characteristic length scale and velocity for these dimensionless groups. When contact is established the flow is separated and a pressure drop across the solid is observed. In the process of seeking an equilibrium between the elastic forces within the solid and the pressures on the solid boundary, free sliding along the fluid walls is allowed. Due to a switch between contact and FSI near the boundary, the solid is free to pull itself from the boundary.

The extension of the method to three dimensions is addressed in Chapter 3. The triangular elements in the fluid mesh and the quadrilateral elements in the solid mesh are replaced by their three-dimensional counterparts, namely the tetrahedral and hexahedral elements, respectively. The adaptive meshing procedure, which is a similar hybrid formulation as in 2D, therefore, results in an inner-fluid surface consisting of elemental faces that coincide with the solid surface. This generated inner-fluid surface is presented at several time steps for a three-dimensional FSI model, in which a slender solid membrane halfway a fluid channel bulges due to the pressure applied at the inlet of the channel. The same problem is also solved using an axi-symmetric code and the time-dependent motion of the solid is compared successfully.

The method developed in Chapters 2, 3 and 4 are used to create an aortic heart valve model including FSI in Chapter 5. Based on the geometrical descriptions of Thubrikar [1990] a mesh is generated. Assuming a perfectly symmetrical valve, only half a leaflet mesh with a corresponding fluid mesh needs to be generated for the computations. A solid computation without FSI gives insight in the straining behaviour of the solid leaflet under increasing pressure in closed position with the first and second principal strains in perfect radial and circumferential direction as found in literature, e.g. Sacks et al. [1997]. The FSI model presented in Chapter 5 gives an impression of the large valve motions during systole, which is an ideal situation to test the robustness of the 3D adaptive meshing algorithm. Although opening and closing of the valve seem to be computed correctly, full closure of the valve still causes problems. A probable cause for this problem might be that the coaptation area of the valve is very unevenly coupled, not covering the full leaflet inherent to the adaptive meshing scheme. During opening this does not appear to introduce instabilities but in combination with the diastolic pressure gradient building up across the valve, the edge starts to flutter and finally solutions diverge. The 2D experiments in Chapter 3 showed that the movement of the valve near closure is very dependent on the fluid element size near the boundary. With regard to that, a possible solution to the problem would be to generate a fluid mesh with a set of
very thin tetrahedrons along the contact boundary. Another solution could be to introduce a thin coaptation area that is very flexible and would possibly coapt easily under pressure, but is impermeable and closes the gap entirely leaving a situation like the 3D membrane in Chapter 4. Furthermore, FSI computations for the aortic heart valve model with sufficiently fine meshes which would provide accurate shear stresses in the future, have not yet been performed, though to our believe this is only a matter of time.

Resuming, we can conclude that an FSI method has been developed that is capable of computing the dynamics of a heart valve in a pulsatile flow during a whole cardiac cycle with sufficiently accurate shear stresses along the leaflets. The method couples fluid and solid with a Lagrange multiplier for flexibility and combines it with a local adaptive meshing technique to improve accuracy. Enhancing the method with solid-rigid contact results in an appropriate tool for the FSI modelling of any kind of heart valve.

6.3 Recommendations

This study has mainly focussed on the development of an FSI method, and presented a aortic valve model using this method. Since in this model only FSI is included, several remarks with regard to more realistic modelling of aortic heart valve dynamics are necessary. A first remark can be made concerning the geometries of valve and fluid. The sinuses of Valsalva in the aortic root, that have a large influence on the fluid dynamics around the valve and have proved to play a crucial role in the onset of the valve closure, should be incorporated in the model. Also the assumption of symmetry is debatable since sinuses are not identical in size and coronary arteries are springing from two of them. Even for a perfectly symmetrical valve a 3D rotational twist can be observed experimentally which "locks" the valve in the closed position. With regard to the material behaviour we point out that native leaflets have a network of imbedded collagen fibers causing highly anisotropic material behaviour, that proved to influence the valve motion [De Hart et al., 2004]. Blood is known to show shear-thinning behaviour that could be easily included in the model, especially, if quantitative shear stresses are of interest. The aortic root has been modelled rigidly in our model as in reality it is compliant and such influences flow, pressure and valve motion. Boundary conditions are always of crucial importance in computational methods. The left ventricle in front and the whole arterial tree behind the aortic root determine the required boundary conditions that be obtained from patients data.

From a numerical point of view also some suggestions can be made to improve the current method. Currently, quadratic elements have been used successfully for the solid and fluid mesh up to Re=1750 in 2D. At physiological Reynolds numbers, the resolution of these finite element techniques are not adequate anymore to capture the complicated and separated flow field with associated shear stresses but, alternatively, high spatial accuracy techniques such as spectral/hp element methods might be used. These techniques have a higher-order polynomial expansion within the elemental domain and have already been used to simulate weakly turbulent cardiovascular flows
in complex geometry [Sherwin and Blackburn, 2005]. Another remark can be made with regard to the time integration scheme used. For all models presented in this study a backward Euler scheme was used for time integration, which is known to be $O(\Delta t)$ accurate and unconditionally stable. It is a robust scheme for testing, but when it comes to accuracy a Crank-Nicolson scheme would be the better option in case the set of equations is solved fully coupled. The currently used direct solver is clearly not the most efficient way to solve large systems and requires large amounts of memory. To that respect an iterative solver is a more appropriate option, but due to the Lagrange multipliers the set of equations becomes ill-posed, and a rather full ILU factorisation for the preconditioning matrix is required to let the solver converge. To reduce computational time and the amount of memory required, it would be logical to solve the FSI decoupled, solving a separate solid and fluid computation. This is, however, not as obvious as it seems. On the one hand a valve is a very slender and therefore flexible body, but on the other hand it can obstruct flow considerably as shown in several numerical experiments presented in this study. Due to this ambivalent behaviour decoupling of the set of equations results in an unstable system to solve. Nevertheless, some smart staggering or operator splitting schemes have been successfully used for decoupling the solid, fluid and coupling equations [Farhat et al., 1998; Felippa et al., 2001; Yu, 2005] and might be applicable to these kinds of models. Another useful reference to this regard would be Vierendeels et al. [2003] who treated the stability conditions in case of FSI with a rigid body for various time integration schemes.

6.4 Model applications

FSI models can provide information about stresses and strains within the valve and at the root, or information concerning the efficiency of the valve. Additionally, detailed knowledge about shear along valve or root can be gained. Several people might prosper from the outcome of the modelling like clinicians and developers or manufacturers of artificial or tissue engineered heart valves. We believe that modelling can improve our understanding of the in-vivo heart valve dynamics which have implications for the onset, as well as surgical and therapeutic treatment of associated diseases. One can think of diagnosis on several geometric parameters combined with blood pressure and flow, by predicting the amount of regurgitation and peak stresses in the leaflet or root tissue. Simulating the behaviour of newly developed or currently existing artificial heart valves can provide information about the valve efficiency or predict high shear areas in which blood could be damaged. The valve design could then be optimised by making geometrical adjustments. In the field of tissue engineering it is known that the endothelial cells that cover the leaflets sense shear and influence the underlying tissue. It is, however, not yet known if, how and when to add the endothelial cells to the tissue engineered construct and how to optimally stimulate the cells to attach to the construct. Modelling can provide knowledge about the shear stresses along the leaflets, which might be linked to cell activity.
Appendix A

Linearisation

A weighted residual formulation is used to approximate the strong form of the equations describing the solid fluid and their corresponding constraints. In the following part this formulation is given together with the linearisation of the non-linear set of equations analogue to the programming code as found in the finite element package SEPRAN. The fluid will be written in an Eulerian framework of reference while an updated Lagrangian formulation is used for the solid. In addition, the major differences between a 3D, plain strain and axisymmetric formulation are given.

A.1 Fluid

The weak formulation of the fluid including a Lagrange multiplier, $\lambda^{fsi}$, regarding the FSI coupling, is written as,

$$
\int_{\Omega^f} w^f \cdot \left( \rho \left( \frac{\partial u^f}{\partial t} + u^f \cdot \nabla u^f \right) - \nabla \cdot \sigma^f \right) \, d\Omega^f +
\int_{\Gamma^f} w^f \cdot \lambda^{fsi} \, d\Gamma^f = \int_{\Omega^f} w^f \cdot \rho f^f \, d\Omega^f \quad (A.1)
$$

$$
\int_{\Omega^f} w^f \cdot \nabla \cdot u^f \, d\Omega^f \quad (A.2)
$$

If the fluid velocity is subdivided in a last estimated velocity part $\hat{u}^f$ and a corresponding error $\delta u^f$ as, $u^f = \hat{u}^f + \delta u^f$, then Newton linearisation of the non-linear convective term in Eq. (A.1) reads,

$$
u^f \cdot \nabla u^f = (\hat{u}^f + \delta u^f) \cdot \nabla (\hat{u}^f + \delta u^f)$$

$$= \hat{u}^f \nabla u^f + (u^f - \hat{u}^f) \cdot \nabla \hat{u}^f + \delta u^f \nabla \delta u^f \quad (A.3)$$
the latter term of which is neglected. For the time integration a backward Euler scheme is applied in which each time step is defined as \( \Delta t = t_{n+1} - t_n \) and subsequently,

\[
\frac{\partial u^f}{\partial t} \approx \frac{u^f - u^f_n}{\Delta t}
\]  

(A.4)
a first order approximation for the velocity with \( u^f = u^f(t_{n+1}) \) and \( u^f_n = u^f(t_n) \). Substitution of Eqs. (A.3) and (A.4) into Eq. (A.1), applying partial integration and using the divergence theorem, gives in absence of body forces,

\[
\int_{\Omega^f} w^f \cdot \rho \left( \frac{u^f_n}{\Delta t} + \hat{u}^f \nabla u^f + u^f \cdot \nabla \hat{u}^f \right) \, d\Omega^f - \int_{\Omega^f} (\nabla w^f)^T : \sigma^f \, d\Omega^f + \\
\int_{\gamma_f} w^f \cdot \lambda^f \gamma \, d\gamma_f = \int_{\Omega^f} w^f \cdot \rho \left( \frac{u^f_n}{\Delta t} + \hat{u}^f \cdot \nabla \hat{u}^f \right) \, d\Omega^f + \\
\int_{\partial \Omega^f} w^f \cdot \nu^f \cdot n^f \, d\partial \Omega^f
\]  

(A.5)

### A.2 Solid

Similar to the fluid, a weak formulation for the solid can be derived, including the Lagrange multipliers that correspond to the FSI and contact problem.

\[
\int_{\Omega^s} (\nabla w^s)^T : [\tau^s - p^s] \, d\Omega^s - \int_{\gamma_f} w^s \cdot \lambda^f \gamma \, d\gamma_f - \int_{\gamma_f} w^s \cdot \lambda^s \nu \, d\gamma_f = \\
\int_{\partial \Omega^s} w^s \cdot (\sigma^s \cdot n^s) \, d\partial \Omega^s + \int_{\Omega^s} w^s \cdot \rho f^s \, d\Omega^s
\]  

(A.6)

\[
\int_{\Omega^s} w^s \cdot (\det(F_\Delta) - 1) \, d\Omega^s = 0
\]  

(A.7)

For the FSI problems presented in this thesis no external boundary or body forces were present, causing the right hand side to vanish. Note that, an updated Lagrangian formulation will be used for the solid, for which we can distinguish three different configurations, namely: the initial configuration (denoted by \( 0 \)), a reference configuration (denoted by \( n \)) and the current configuration (no subscript). The set of equations will be linearised in the reference configuration. It is therefore convenient to subdivide the deformation gradient tensor \( F = (\nabla x^s)^T \), that describes the total deformation from the initial to the current configuration, into two parts such that \( F = F_n \cdot F_\Delta \). Tensor \( F_n \) describes the deformation from the initial to the reference configuration and tensor \( F_\Delta \) describes that from the reference to the current configuration. Subsequently, we can introduce the following relations,

\[
J = \det(F) = \det(F_n \cdot F_\Delta) = \det(F_n) \det(F_\Delta) = J_n J_\Delta
\]  

(A.8)

\[
\nabla = F_\Delta^{-T} \cdot \nabla_n
\]  

(A.9)

\[
\Omega = J_\Delta \Omega_n
\]  

(A.10)
which are used to transform Eq. (A.6) to the reference configuration.

A.2.1 Linearisation of solid equations

If the position vector with respect to the reference configuration is linearised as \( \tilde{x}^s = \hat{x}^s + \delta x^s \), then from the definition of the deformation gradient follows that,

\[
F_\Delta (\hat{x}^s + \delta x^s) = (\nabla_n(\hat{x}^s + \delta x^s))^T = (\nabla_n \hat{x}^s)^T + (\nabla_n \delta x^s)^T
\]

\[
= \hat{F}_\Delta + \delta F_\Delta
\]  

(A.11)

Using the identity \( F_\Delta \cdot F_\Delta^{-1} = I \) it follows that,

\[
(\hat{F}_\Delta + \delta F_\Delta) \cdot (\hat{F}_\Delta^{-1} + \delta (F_\Delta^{-1})) = I + \hat{F}_\Delta \cdot \delta (F_\Delta^{-1}) + \delta F_\Delta \cdot F_\Delta^{-1} + \delta F_\Delta \cdot \delta (F_\Delta^{-1}) = I
\]

(A.12)

If \( \delta F_\Delta \) is sufficiently small then an approximation of \( \delta (F_\Delta^{-1}) \) is,

\[
\delta (F_\Delta^{-1}) = -\hat{F}_\Delta^{-1} \cdot \delta F_\Delta \cdot \hat{F}_\Delta^{-1} = -\hat{F}_\Delta^{-1} \cdot (\hat{F}_\Delta^{-T} \cdot \nabla \delta x^s)^T
\]

\[
= -\hat{F}_\Delta^{-1} \cdot (\nabla \delta x^s)^T = -\hat{F}_\Delta^{-1} \cdot H
\]

(A.13)

in which \( H \equiv (\nabla \delta x^s)^T \). Subsequently, for the volume ratio \( J_\Delta \) it can be seen that,

\[
J_\Delta = \det(\hat{F}_\Delta) = \det(\hat{F}_\Delta + \delta F_\Delta) = \det \left( (I + \delta F_\Delta \cdot \hat{F}_\Delta^{-1}) \cdot \hat{F}_\Delta \right)
\]

\[
= \det(I + \delta F_\Delta \cdot \hat{F}_\Delta^{-1}) \det(\hat{F}_\Delta) \approx \left( 1 + \text{tr}(\delta \hat{F}_\Delta \cdot \hat{F}_\Delta^{-1}) \right) \det(\hat{F}_\Delta)
\]

\[
= \hat{J}_\Delta + \text{tr}(\delta \hat{F}_\Delta \cdot \hat{F}_\Delta^{-1}) \hat{J}_\Delta
\]

(A.14)

from which it follows that,

\[
\delta J_\Delta = \text{tr}(\delta F_\Delta \cdot \hat{F}_\Delta^{-1}) \hat{J}_\Delta = \hat{J}_\Delta \text{tr}(H)
\]

(A.15)

Notice that in the expression for \( \delta \sigma^s \) the left Cauchy-Green strain tensor \( B \) appears. In an updated Lagrangian formulation the total deformation \( F \) is stored in the increment deformation tensor \( F_\Delta \) and the deformation from the initial configuration to the reference configuration \( F_n \). The strain tensor \( B \) is expressed in terms of \( F_\Delta \) and \( F_n \):

\[
B = F \cdot F^T = F_\Delta \cdot F_n \cdot F_n^T \cdot F_\Delta^T = F_\Delta \cdot B_n \cdot F_\Delta^T
\]

(A.16)

Therefore, with regard to the linearisation of the extra stress tensor \( \tau^s = \hat{\tau}^s + \delta \tau^s \) we can write,

\[
\delta \tau^s = G \left( \delta F_\Delta \cdot B_n \cdot \hat{F}_\Delta + \hat{F}_\Delta \cdot B_n \cdot \delta F_\Delta^T \right)
\]

(A.17)

\[
= G \left( H \cdot \hat{F}_\Delta \cdot B_n \cdot \hat{F}_\Delta^T + \hat{F}_\Delta \cdot B_n \cdot \hat{F}_\Delta^T \cdot H^T \right)
\]

(A.18)

\[
= G \left( H \cdot B + B \cdot H^T \right)
\]

(A.19)
For the integral concerning the contact part in Eq. (A.6), surface \( \gamma^c \) in \( \mathbb{R}^3 \) can be transformed back to a reference surface \( \mathbb{R}^2 \) with coordinates \( \xi \) and \( \eta \) as,

\[
d\gamma^c = \| \mathbf{t}_1 \times \mathbf{t}_2 \| \, d\xi d\eta
\]

(A.20)

with \( \mathbf{t}_1 = \frac{\partial \mathbf{F}^c}{\partial \xi} \) and \( \mathbf{t}_2 = \frac{\partial \mathbf{F}^c}{\partial \eta} \). Linearisation of \( \| \mathbf{t}_1 \times \mathbf{t}_2 \| \) gives \( \| \hat{\mathbf{t}}_1 \times \hat{\mathbf{t}}_2 \| + \delta \| \mathbf{t}_1 \times \mathbf{t}_2 \| \) with,

\[
\delta \| \mathbf{t}_1 \times \mathbf{t}_2 \| = \delta \left( \sqrt{(\mathbf{t}_1 \times \mathbf{t}_2) \cdot (\mathbf{t}_1 \times \mathbf{t}_2)} \right) = \frac{1}{\| \mathbf{t}_1 \times \mathbf{t}_2 \|} (\hat{\mathbf{t}}_1 \times \hat{\mathbf{t}}_2) \cdot \delta (\mathbf{t}_1 \times \mathbf{t}_2) = \mathbf{n}^s \cdot \delta \mathbf{n}^s
\]

(A.21)

Transformation of Eq. (A.6) using the identities (A.9), (A.10) and (A.20), and linearising the resulting set of equations gives,

\[
\int_{\Omega^*} (\nabla_n \mathbf{w}^s)^T : \hat{\mathbf{F}}_\Delta^{-1} \left[ -H : (\hat{\gamma}^s - \hat{p} \mathbf{I}) + \text{tr}(\mathbf{H}) \cdot (\hat{\gamma}^s - \hat{p} \mathbf{I}) \right] \, d\Omega^*_n + \\
\int_{\Omega^*} (\nabla_n \mathbf{w}^s)^T : \hat{\mathbf{F}}_\Delta^{-1} \left[ \delta \mathbf{r}^s - \hat{p} \mathbf{I} \right] \, d\Omega^*_n - \int_{\mathcal{C}_f} \mathbf{w}^s \cdot \mathbf{X}^{f*} \, d\gamma^f
\]

\[
- \int_{-1}^{1} \int_{-1}^{1} \mathbf{w}^s \cdot \left[ \delta \lambda^c \cdot \hat{n}^c + \hat{\lambda}^c \cdot \delta \mathbf{n}^c + \hat{\lambda}^c \cdot \hat{n}^c \cdot \frac{\hat{n}^s \cdot \delta \mathbf{n}^s}{\| \mathbf{t}_1 \times \mathbf{t}_2 \|} \right] \| \hat{\mathbf{t}}_1 \times \hat{\mathbf{t}}_2 \| \, d\xi d\eta = \\
\int_{\Omega^*} (\nabla_n \mathbf{w}^s)^T : \hat{\mathbf{F}}_\Delta^{-1} \cdot (\hat{\gamma}^s - \hat{p} \mathbf{I}) \, d\Omega^*_n + \int_{-1}^{1} \int_{-1}^{1} \mathbf{w}^s \cdot \hat{\lambda}^c \cdot \hat{n}^c \| \hat{\mathbf{t}}_1 \times \hat{\mathbf{t}}_2 \| \, d\xi d\eta
\]

(A.22)

and

\[
\int_{\Omega^*} J_\Delta \text{tr}(\mathbf{H}) \, d\Omega^*_n = \int_{\Omega^*} J_\Delta - 1 \, d\Omega^*_n
\]

(A.23)

Note that the linearisation of \( \mathbf{n}^c \) depends on the geometry of the rigid body used. For a straight line in 2D or a straight plate as used in Chapters 3 and 5, respectively, the normal of the rigid body does not change and is therefore \( \delta \mathbf{n}^c = \mathbf{0} \). Furthermore, the linearisation of \( J_\Delta \) for this set of equations might appear strange since an incompressible medium is considered (and therefore \( J_\Delta = 1 \)). However, the incompressibility constraint is enforced over an integral, which will be solved numerically and not linearising \( J_\Delta \) could therefore cause loss of convergence. This is reason that it is taken into account in most of the terms of Eq. (A.22). In SEPRAN the Jacobian of the FSI-coupling term and the incompressibility constraint are, however, currently not linearised. The linearised momentum equation and mass balance with
respect to the last known configuration can be derived as.

\[
\int_{\tilde{\Omega}^s} (\nabla \omega^s)^T : [-H \cdot (\tilde{\omega}^s - \tilde{\omega}) I + \text{tr}(H) \cdot (\tilde{\omega}^s - \tilde{\omega}) I] \, d\Omega^s + \\
\int_{\tilde{\Omega}^s} (\nabla \omega^s)^T : [\delta \tau^s - \delta p^s I] \, d\Omega^s - \int_{\gamma_f} \omega^s \cdot \lambda^f \, d\gamma^f \\
- \int_{\gamma_r} \omega^s \cdot [\delta \lambda^c \cdot \hat{n}^c + \lambda^c \cdot \delta n^c + \lambda^c \cdot \hat{n}^c \cdot \delta n^c \frac{\nabla^s \cdot \delta n^s}{||t_1 \times t_2||}] \, d\gamma^c = \\
\int_{\tilde{\Omega}^s} (\nabla \omega^s)^T : (\tilde{\omega}^s - \tilde{\omega}) I \, d\Omega^s + \int_{\gamma_r} \omega^s \cdot \hat{\lambda}^c \cdot \hat{n}^c \, d\gamma^c \tag{A.24}
\]

\[
\int_{\tilde{\Omega}^s} \text{tr}(H) \, d\Omega^s = \int_{\tilde{\Omega}^s} \frac{J_\Delta - 1}{J_\Delta} \, d\Omega^s \tag{A.25}
\]

A remark should be made when an axisymmetric approach is used. For a 3D or plain strain formulation the above linearisations for fluid and solid are equal, but in case of axisymmetry an extra \(2\pi r\) term should be added to each integral and taken into account in the linearisation. Additional differences between 3D, plain strain and axisymmetry are addressed in Section A.2.2 and A.2.3.

**A.2.2 Remark on tensor \(H\) in axisymmetry**

In 3D the gradient vector \(\nabla\) and position vector \(x\) are defined as,

\[
\nabla = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} \frac{\partial}{\partial x} + \begin{bmatrix} e_y \\ e_z \end{bmatrix} \frac{\partial}{\partial y} + \begin{bmatrix} e_z \end{bmatrix} \frac{\partial}{\partial z} \tag{A.26}
\]

\[
x = xe_x + ye_y + ze_z \tag{A.27}
\]

which leads to a matrix representation for tensor \(H\) as,

\[
H = (\nabla \delta x)^T \Rightarrow H = \begin{bmatrix} H_{11} & H_{21} & H_{31} \\ H_{12} & H_{22} & H_{32} \\ H_{13} & H_{23} & H_{33} \end{bmatrix} = \begin{bmatrix} \frac{\partial \delta x}{\partial x} & \frac{\partial \delta x}{\partial y} & \frac{\partial \delta x}{\partial z} \\ \frac{\partial \delta y}{\partial x} & \frac{\partial \delta y}{\partial y} & \frac{\partial \delta y}{\partial z} \\ \frac{\partial \delta z}{\partial x} & \frac{\partial \delta z}{\partial y} & \frac{\partial \delta z}{\partial z} \end{bmatrix} \tag{A.28}
\]

In axisymmetry the gradient vector and the solution vector are slightly different, namely:

\[
\nabla = \begin{bmatrix} e_r \\ e_z \end{bmatrix} \frac{\partial}{\partial r} + \frac{1}{r} \begin{bmatrix} e_\theta \end{bmatrix} \frac{\partial}{\partial \theta} \tag{A.29}
\]

\[
x = re_r + ze_z + \theta e_\theta \tag{A.30}
\]
We assume that $\theta$ is constant, i.e. no torsional effects, and that the variables $r$ and $z$ are constant over $\theta$. The linearisation of $x$ then reads,

$$x + \delta x = (r + \delta r)e_r + (z + \delta z)e_z$$  \hspace{1cm} (A.31)

and subsequently $\nabla \delta x = \nabla (\delta e_r + \delta e_z)$ can be computed as,

$$e_r \frac{\partial}{\partial r} (\delta x) = e_r \frac{\partial \delta r}{\partial r} e_r + e_r \frac{\partial \delta z}{\partial r} e_z$$
$$\frac{1}{r} e_\theta \frac{\partial}{\partial \theta} (\delta x) = \frac{1}{r} \delta r e_\theta e_\theta$$
$$e_z \frac{\partial}{\partial z} (\delta x) = e_z \frac{\partial \delta r}{\partial z} e_r + e_z \frac{\partial \delta z}{\partial z} e_z$$

For this axisymmetric case the matrix representation for tensor $H$ then reads,

$$H = (\nabla \delta x)^T$$ \Rightarrow \begin{bmatrix} H_{11} & H_{21} & H_{31} \\ H_{12} & H_{22} & H_{32} \\ H_{13} & H_{23} & H_{33} \end{bmatrix} = \begin{bmatrix} \frac{\partial \delta r}{\partial r} & \frac{\partial \delta r}{\partial z} & 0 \\ \frac{\partial \delta z}{\partial r} & \frac{\partial \delta z}{\partial z} & 0 \\ 0 & 0 & \frac{\delta r}{r} \end{bmatrix} \hspace{1cm} (A.32)$$

### A.2.3 Remark on $F_\Delta$ for axisymmetry

The gradient vector with respect to the original configuration reads,

$$\nabla_n = e_r \frac{\partial}{\partial n} + e_\theta \frac{\partial}{\partial \theta} + \frac{1}{r} e_n \frac{\partial}{\partial n}$$ \hspace{1cm} (A.33)

The correlation between the radial and tangential orthonormal vector in old and new configuration is,

$$e_r = e_r \cos(\theta - \theta_n) + e_\theta \sin(\theta - \theta_n)$$ \hspace{1cm} (A.34)
$$e_\theta = -e_r \sin(\theta - \theta_n) + e_\theta \cos(\theta - \theta_n)$$ \hspace{1cm} (A.35)

However, in case $\theta_n = \theta = \text{Constant}$, it follows that $e_r = e_{r_n}$ and $e_\theta = e_{\theta_n}$. Now the deformation gradient tensor $F_\Delta = (\nabla_n x)^T$ will be derived for the axisymmetric case,

$$e_{r_n} \frac{\partial}{\partial n} x = e_{r_n} \frac{\partial r}{\partial n} e_{r_n} + e_{r_n} \frac{\partial z}{\partial n} e_{z_n}$$ \hspace{1cm} (A.36)
$$e_{z_n} \frac{\partial}{\partial n} x = e_{z_n} \frac{\partial z}{\partial n} e_{r_n} + e_{z_n} \frac{\partial z}{\partial n} e_{z_n}$$ \hspace{1cm} (A.37)
$$e_{\theta_n} \frac{1}{r_n} \frac{\partial}{\partial \theta_n} x = e_{\theta_n} \frac{r}{r_n} e_{\theta_n}$$ \hspace{1cm} (A.38)
resulting in a matrix representation for $F_\Delta$ as,

$$F_\Delta = (\nabla_n x)^T \Rightarrow \mathcal{F} = \begin{bmatrix} \frac{\partial r}{\partial r_n} & \frac{\partial r}{\partial \sigma_n} & 0 \\ \frac{\partial z}{\partial r_n} & \frac{\partial z}{\partial \sigma_n} & 0 \\ 0 & 0 & \frac{r}{r_n} \end{bmatrix}$$

(A.39)

Note that in this axisymmetric formulation the variable $\theta$ is taken constant across the coordinates of the model. Due to this assumption the formulation is simplified greatly and $\theta$ can be eliminated as a solution variable resulting in a faster code. However, if one is interested in torsional effects $\theta$ should be incorporated as a function of $r$ and $z$ and should therefore, be taken into account when performing the linearisation. As a consequence linearisation of $e_r$ and $e_\theta$ is also required since these vectors are a function of $\theta$ and additionally the assumption that $e_r = e_{r_n}$ and $e_\theta = e_{\theta_n}$ would not hold anymore.

### A.2.4 Matrix notation for stiffness matrix

The relevant matrices are given for the stiffness part of Eq. (A.24) to give an impression of the translation from the tensor to matrix formulation. The 3D, plain strain and axisymmetric cases are considered. The components of a tensor $H^w \equiv \nabla w^w$ and a tensor $\mathcal{A}$ can be collected in columns $H^w$ and $\mathcal{A}$, such that

$$H^w : \mathcal{A} = (H^w)^T \mathcal{A}$$

(A.40)

which holds if,

$$H^w = \begin{bmatrix} H_{11}^w & H_{12}^w & H_{13}^w & H_{14}^w & H_{15}^w & H_{16}^w & H_{17}^w & H_{18}^w & H_{19}^w \\ H_{21}^w & H_{22}^w & H_{23}^w & H_{24}^w & H_{25}^w & H_{26}^w & H_{27}^w & H_{28}^w & H_{29}^w \\ H_{31}^w & H_{32}^w & H_{33}^w & H_{34}^w & H_{35}^w & H_{36}^w & H_{37}^w & H_{38}^w & H_{39}^w \end{bmatrix}$$

(A.41)

$$\mathcal{A} = \begin{bmatrix} A_{11} & A_{22} & A_{33} & A_{41} & A_{42} & A_{43} \end{bmatrix}$$

(A.42)

Notice that the locations of the components of $\mathcal{A}$ have different locations in the column $\mathcal{A}$ than the components of $H^w$ in the column $H^w$.

The first term originates from the geometrical nonlinearity of $F_\Delta^{-1}$. Evaluation of the product yields

$$\mathcal{A} = -H \cdot \sigma \quad \Rightarrow \quad \mathcal{A} = D_{\sigma} H$$

(A.43)

which can be rewritten such that,

$$\mathcal{A}_{3D} = -\begin{bmatrix} \sigma_{11} & 0 & 0 & \sigma_{12} & 0 & 0 & 0 & 0 & \sigma_{31} \\ 0 & \sigma_{22} & 0 & 0 & \sigma_{23} & 0 & \sigma_{12} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 & 0 & \sigma_{31} & 0 & \sigma_{23} & 0 \\ 0 & \sigma_{12} & 0 & 0 & \sigma_{31} & 0 & \sigma_{11} & 0 & 0 \\ 0 & 0 & \sigma_{23} & 0 & 0 & \sigma_{12} & 0 & \sigma_{22} & 0 \\ \sigma_{31} & 0 & 0 & \sigma_{23} & 0 & \sigma_{11} & 0 & \sigma_{33} & 0 \\ \sigma_{12} & 0 & 0 & \sigma_{22} & 0 & \sigma_{11} & 0 & \sigma_{23} & 0 \\ 0 & \sigma_{23} & 0 & 0 & \sigma_{33} & 0 & \sigma_{32} & 0 & 0 \\ 0 & 0 & \sigma_{31} & 0 & 0 & \sigma_{11} & 0 & \sigma_{12} & 0 \end{bmatrix} \begin{bmatrix} H_{11} \\ H_{22} \\ H_{33} \\ H_{12} \\ H_{23} \\ H_{31} \\ H_{21} \\ H_{32} \\ H_{13} \end{bmatrix}$$
\[
A_{2D} = - \begin{bmatrix}
\sigma_{11} & 0 & \sigma_{12} & 0 \\
0 & \sigma_{22} & 0 & \sigma_{12} \\
0 & \sigma_{12} & 0 & \sigma_{11} \\
\sigma_{12} & 0 & \sigma_{22} & 0
\end{bmatrix}
\begin{bmatrix}
H_{11} \\
H_{21} \\
H_{12} \\
H_{22}
\end{bmatrix}
\]

\[
A_{\alpha\xi} = - \begin{bmatrix}
\sigma_{11} & 0 & \sigma_{12} & 0 & 0 \\
0 & \sigma_{22} & 0 & \sigma_{12} & 0 \\
0 & \sigma_{12} & 0 & \sigma_{11} & 0 \\
\sigma_{12} & 0 & \sigma_{22} & 0 & 0 \\
0 & 0 & 0 & 0 & \sigma_{33}
\end{bmatrix}
\begin{bmatrix}
H_{11} \\
H_{22} \\
H_{12} \\
H_{21} \\
H_{33}
\end{bmatrix}
\]

The second term originates from the geometrical nonlinearity of \( J_\Delta \). Evaluation yields,

\[
A = \text{tr}(H) \sigma \quad \Rightarrow \quad A = \left( H_{11} + H_{22} + H_{33} \right) \begin{bmatrix}
\sigma_{11} & \sigma_{12} & \sigma_{13} \\
\sigma_{12} & \sigma_{22} & \sigma_{23} \\
\sigma_{31} & \sigma_{32} & \sigma_{33}
\end{bmatrix}
\]

(A.44)

\[
= D_r H
\]

(A.45)

which can be rewritten such that,

\[
A_{3D} = \begin{bmatrix}
\sigma_{11} & \sigma_{11} & \sigma_{11} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{22} & \sigma_{22} & \sigma_{22} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{33} & \sigma_{33} & \sigma_{33} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{12} & \sigma_{12} & \sigma_{12} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{23} & \sigma_{23} & \sigma_{23} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{31} & \sigma_{31} & \sigma_{31} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{21} & \sigma_{21} & \sigma_{21} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{32} & \sigma_{32} & \sigma_{32} & 0 & 0 & 0 & 0 & 0 & 0 \\
\sigma_{13} & \sigma_{13} & \sigma_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
H_{11} \\
H_{22} \\
H_{12} \\
H_{21} \\
H_{33}
\end{bmatrix}
\]

\[
A_{2D} = \begin{bmatrix}
\sigma_{11} & \sigma_{11} & 0 & 0 \\
\sigma_{22} & \sigma_{22} & 0 & 0 \\
\sigma_{12} & \sigma_{12} & 0 & 0 \\
\sigma_{21} & \sigma_{21} & 0 & 0
\end{bmatrix}
\begin{bmatrix}
H_{11} \\
H_{22} \\
H_{12} \\
H_{21}
\end{bmatrix}
\]

\[
A_{\alpha\xi} = \begin{bmatrix}
\sigma_{11} & \sigma_{11} & 0 & 0 & \sigma_{11} \\
\sigma_{22} & \sigma_{22} & 0 & 0 & \sigma_{22} \\
\sigma_{12} & \sigma_{12} & 0 & 0 & \sigma_{12} \\
\sigma_{21} & \sigma_{21} & 0 & 0 & \sigma_{21} \\
\sigma_{33} & \sigma_{33} & 0 & 0 & \sigma_{33}
\end{bmatrix}
\begin{bmatrix}
H_{11} \\
H_{22} \\
H_{12} \\
H_{21} \\
H_{33}
\end{bmatrix}
\]

The third term originates from the linearisation of the constitutive model. The linearised extra stress tensor \( \delta \tau \) is written in matrix form producing the material matrix \( D_r \) for the linearisation of \( B \).

\[
A = G \left( H \cdot B + B \cdot H^T \right) \quad \Rightarrow \quad \Delta = D_r H
\]

(A.46)
\[
\begin{align*}
D_{\tau,3D} &= G \begin{bmatrix}
2B_{11} & 0 & 0 & 2B_{12} & 0 & 0 & 0 & 0 & 2B_{31} \\
0 & 2B_{22} & 0 & 0 & 2B_{23} & 0 & 2B_{12} & 0 & 0 \\
0 & 0 & 2B_{33} & 0 & 0 & 2B_{31} & 0 & 2B_{23} & 0 \\
B_{12} & B_{12} & 0 & B_{22} & B_{31} & 0 & B_{11} & 0 & B_{23} \\
0 & B_{23} & B_{23} & 0 & B_{33} & B_{12} & B_{31} & B_{22} & 0 \\
B_{12} & 0 & B_{22} & 0 & B_{31} & 0 & B_{11} & 0 & B_{23} \\
0 & B_{23} & 0 & B_{23} & 0 & B_{12} & B_{31} & B_{22} & 0 \\
B_{31} & 0 & B_{31} & 0 & B_{31} & 0 & B_{11} & 0 & B_{12} & B_{33}
\end{bmatrix}
\end{align*}
\]

\[
D_{\tau,2D} = G \begin{bmatrix}
2B_{11} & 0 & 2B_{12} & 0 \\
0 & 2B_{22} & 0 & 2B_{12} \\
B_{12} & B_{12} & B_{22} & B_{11} \\
B_{12} & B_{12} & B_{22} & B_{11}
\end{bmatrix}
\]

\[
D_{\tau,\alpha 2D} = G \begin{bmatrix}
2B_{11} & 0 & 2B_{12} & 0 & 0 \\
0 & 2B_{22} & 0 & 2B_{12} & 0 \\
B_{12} & B_{12} & B_{22} & B_{11} & 0 \\
B_{12} & B_{12} & B_{22} & B_{11} & 0 \\
0 & 0 & 0 & 0 & B_{33}
\end{bmatrix}
\]

Summarising, the left hand side terms for the linearisation of \( J_\Delta \), \( \mathbf{F}_\Delta^{-1} \) and \( \tau \) in Eq. (A.24) reads in matrix form,

\[
\int_\Omega (\nabla w^\ast)^T [-\mathbf{H} \cdot (\hat{\tau} - \hat{\rho} \mathbf{I}) + \delta \tau + \text{tr}(\mathbf{H}) \cdot (\hat{\tau} - \hat{\rho} \mathbf{I})] \, d\Omega = \int_\Omega (\mathbf{H} w^\ast)^T [D_{\tau,3D} + D_{\tau,2D} + D_{\tau,\alpha 2D}] \mathbf{H} \, d\Omega - \text{A.47}
\]

For the discretisation of the equations concerning the FSI the reader is referred to De Hart [2002] and Stijnen [2004]. The discretisation of the equations related to the rigid-solid contact is described in more detail in an internal report \(^1\)

### A.3 Coupling solid and fluid

The coupling constraint for fluid and solid reads \( \mathbf{u}^f - \mathbf{u}^s = \mathbf{0} \) and needs no linearisation. However, from Section A.2.1 we know that the set of fluid equations is built such that velocities are to be solved, but the solution vector corresponding to the solid equations consists of iterative displacement variations. Therefore, a remark is made on the manipulation of the solid set of equations to enable the FSI coupling.

\(^1\)M.A.J. Cox, A. Steikelenburg, C.W.J. Oomens, Parameterized mesh generation of the lower hind limb of a rat., BMTE04.29, Internal Report, 2004
The finite element package SEPRAN [Segal, 2003] builds a linearised set of equations for the unsteady Navier-Stokes equations (“element 900-903” in SEPRAN) according to,

$$K_{n+1}^f u_{n+1,i+1}^f = f_{n+1}^f$$  \hspace{1cm} (A.48)

where subscripts $n$ and $i$ denote the time step and the Newton-Raphson iteration step within that time step, respectively. Every iteration velocity $u_{n+1,i+1}$ needs to be solved. The standard code for a (in)compressible hyperelastic solid (“element 200-205”) builds a system like,

$$K_{n+1}^s \delta x_{n+1,i+1}^s = f_{n+1}^s$$  \hspace{1cm} (A.49)

with $\delta x_{n+1,i}^s$ the iterative displacement, meaning the iterative variation on the incremental solution $\Delta x_{n+1,i}$ (total displacement within time step $n+1$).

$$\Delta x_{n+1,i} = \sum_{j=1}^{i+1} \delta x_{n+1,j}^s$$  \hspace{1cm} (A.50)

Based on the former definitions, we can manipulate the sets of equations for fluid, Eq. (A.48), and solid, Eq. (A.49), such that velocities can be coupled. First write the iterative solution as a function of the incremental solutions at iteration $i+1$ and $i$.

$$\delta x_{n+1,i+1}^s = \Delta x_{n+1,i}^s - \Delta x_{n+1,i+1}$$  \hspace{1cm} (A.51)

Substitution of Eq. (A.51) into Eq. (A.49) gives,

$$K_{n+1}^s \Delta x_{n+1,i+1}^s = f_{n+1}^s + K_{n+1}^s \Delta x_{n+1,i}^s$$  \hspace{1cm} (A.52)

and if a first order approximation $u_{n+1,i+1}^s = \Delta x_{n+1,i+1}/\Delta t$, is taken for the velocity,

$$\Delta t K_{n+1}^s u_{n+1,i+1}^s = f_{n+1}^s + K_{n+1}^s \Delta u_{n+1,i}^s$$  \hspace{1cm} (A.53)

which can be coupled to Eq. (A.48).

### A.4 Rigid-solid contact

The last set of equations that requires linearisation is the constraint corresponding to the rigid-solid contact problem. As discussed in Sections 3.2.3 and 3.3 there are two forms of the weak formulation for the contact constraint, namely,

$$\int_{\Omega} w^c(x^s - x^c) \cdot n^c \, d\Omega \quad \text{or} \quad \int_{\Omega} w^c \lambda^c \, d\Omega$$  \hspace{1cm} (A.54)

depending on a contact or no-contact situation. Transforming to a reference configuration and using $x^s = \hat{x}^s + \delta x^s$, $x^c = \hat{x}^c + \delta x^c$, $n^c = \hat{n}^c + \delta n^c$, $\lambda^c = \hat{\lambda}^c + \delta \lambda^c$,
and Eq. A.21, the the linearisation of these equations reads,

\[
\int_{\Omega_n} w^c \left( (\delta x^s - \delta x^c) \cdot \hat{n}^c + (\hat{x}^s - \hat{x}^c) \cdot (\delta n^c + \hat{n}^c \frac{\delta \hat{n}^c}{\|\hat{t}_1 \times \hat{t}_2\|}) \right) \|\hat{t}_1 \times \hat{t}_2\| d\Omega_n = \\
- \int_{\Omega_n} w^c (\hat{x}^s - \hat{x}^c) \cdot \hat{n}^c \|\hat{t}_1 \times \hat{t}_2\| d\Omega_n \tag{A.55}
\]

\[
\int_{\Omega_n} w^c \delta \lambda^c \|\hat{t}_1 \times \hat{t}_2\| d\Omega_n = - \int_{\Omega_n} w^c \hat{\lambda}^c \|\hat{t}_1 \times \hat{t}_2\| d\Omega_n \tag{A.56}
\]

for which one should note that \( n^c \) and \( x^c \) are a function of \( x^s \) depending on the geometry of the rigid body. Their variations will therefore also depend on the variation of \( x^s \).
Appendix B

Meshing

In this appendix several remarks are made with respect to the meshing procedure. First the robustness of the adaptive meshing scheme is addressed after which a short note on the numerical treatment of the intersections is given and finally two smoothing scheme are briefly discussed to improve the quality of the adapted mesh.

B.1 Robustness of meshing scheme

Whenever applying a adaptive meshing or remeshing scheme it is convenient to know if you will get a solution and if this solution is satisfactory. In our case, the hybrid nature of the meshing scheme with combined node-repositioning and node-adding provides the opportunity to adjust the methods robustness depending on the problem that is considered. As mentioned earlier, $\varepsilon = 0.5$ leads to repositioning only while $\varepsilon = 0$ leads to solely addition of nodes/elements. The disadvantage of the latter case is clear since it can result in ill-shaped or even zero-volume elements. On the other hand, element adding allows the initial mesh to remain unaltered facilitating and optimising the transfer of old solutions from one mesh to another. Furthermore, values of $\varepsilon < 0.5$ increases the algorithms robustness with regard to finding a continuously connected set of edges that cover interface $\gamma^a$. This is illustrated in Fig. B.1 in case $\varepsilon = 0.5$.

Consider the triangular fluid element with nodes A, B and C that is intersected by a curve. The arrows indicate in what direction the nodes will be repositioned based on the conditions defined in Eq. (3.16) (Note again that every node can be repositioned only once). Clearly, nodes A, B and C will be positioned onto the boundary resulting in an ill-shaped element. The change that a configuration as in Fig. B.1 will occur depends on the quality of the initial mesh and the curvature of the boundary. Two possible choices to reduce this risk are to choose a value for $\varepsilon$ smaller than 0.5 (= curvature is allowed to be larger) or to refine the mesh (= reduced curvature). As a
Figure B.1: Configuration in which adaptive meshing will result in ill-shaped elements

practical solution in the case that the meshing procedure goes wrong after computing many time steps in a time dependent problem, is to alter the position of the boundary slightly, e.g. by altering the time step.

Note that although illustrated in 2D similar problems occur for the 3D adaptive meshing procedure as presented in Chapter 4.

B.2 Intersections

In the adaptive meshing scheme the intersection points of the edges of a fluid edge $\Gamma_1^f$ with the curved solid boundary $\gamma^f$ (a line in 2D a surface in 3D) needs to be determined. Therefore, consider the straight fluid edge $\Gamma_2^f$ with begin and end points $x_1^f$ and $x_2^f$, for which the following equation holds,

$$x^f = x_1^f + \alpha(x_2^f - x_1^f), \quad (B.1)$$

with $\alpha$ an unknown factor. Since the position of this boundary is only known directly at the nodal points and should be interpolated with the quadratic basis functions in between, a formulation can be found for a point at the solid boundary element reading,

$$x^s = \sum_{i=1}^{N} \varphi_i(\xi) x_i^s \quad \text{with } N= \text{number of nodal points in element}, \quad (B.2)$$

with $\varphi_i$ one-dimensional basis functions of the edge as a function of local coordinate $\xi$ corresponding to a reference element. In case the solid edge is intersected by the straight line equations $x^f$ in Fig. B.1 and $x^s$ in Fig. B.2 should be equal (Fig. B.2). Therefore, a function can be defined which should be zero as,

$$f(\xi, \alpha) = x_1^f + \alpha(x_2^f - x_1^f) - \sum_i \varphi_i(\xi) x_i^s = 0 \quad (B.3)$$
This function can be linearised and solved in a Newton-Raphson loop.

\[
\frac{\partial f(\xi, \alpha)}{\partial g_i} g_i = -f(\xi, \alpha) \quad (B.4)
\]

\[
\left[ x'_2 - x'_1, \sum_i \frac{\partial \phi_i(\xi)}{\partial \xi} x'_i \right] \cdot g = -x'_1 - \alpha(x'_2 - x'_1) + \sum_i \phi_i(\xi) x'_i \quad (B.5)
\]

with \( g_i \) a component of vector \( g = [\delta \alpha \quad \delta \xi]^T \).

A similar approach can be followed for the intersection of a straight line with a curved element face. The only difference is that the coordinates lie in \( \mathbb{R}^3 \) rather than in \( \mathbb{R}^2 \) and Eq. (B.2) should be replaced by,

\[
x = \sum_i \phi_i(\xi, \eta) x_i \quad \text{with i=nr of nodal points in face} \quad (B.6)
\]

with \( \phi_i \) a function of two local coordinates \( \xi \) and \( \eta \) instead of one.

### B.3 Smoothing

After the adaptive meshing procedure has been applied, smoothing algorithms can be applied to improve mesh quality. These algorithms are based on repositioning of nodal points such that the shape of the elements is enhanced. Two simple schemes moving nodes are considered: Laplacian smoothing [Field, 1988] and angle optimisation [Freitag et al., 1995].
B.3.1 Laplacian smoothing

Probably the easiest and most commonly used way to improve the quality of the mesh is Laplacian smoothing. The new position of a node with position \( \mathbf{x}^* \) is calculated by determining the geometric centre of the surrounding nodes as graphically presented in Fig. B.3.

\[
\mathbf{x}^* = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i \quad \text{with} \ N=\text{number of nodes connected to} \ \mathbf{x}^* \quad \text{(B.7)}
\]

This procedure is often applied in an iterative loop over the nodes until the maximum number of iterations is exceeded or until some mesh quality criteria met. In case the mesh, on which the procedure is performed, has no “flipped” elements and changes in connectivity across the mesh are small it is a robust and simple method to improve the quality of the mesh. Note that in case of quadratic elements only the vertex nodes are considered in this procedure.

![Figure B.3: Move node \( \mathbf{x}^* \) to the geographical center of the connected nodes.](image1)

![Figure B.4: Move node \( \mathbf{x}^* \) such that the smallest angle of all connected elements is enlarged.](image2)

B.3.2 Angle optimisation

As mentioned, Laplacian smoothing does not necessarily lead to nicely shaped elements, depending on the local connectivity of the mesh. In that case the computationally somewhat more expensive algorithm of angle optimisation can be used. Consider a set of nodes with a set of elements attached to each of these nodes. The method focuses on the maximisation of the minimal angle per set of elements by changing the coordinate of the central node as depicted in Fig. B.4. Determine the elements corresponding to a nodal point \( \mathbf{x}^* \). Then the angles of these elements and
their derivatives with respect to $x^*$ can be computed as,

$$\theta_1 = \arccos \left[ \frac{-l_1 + l_2(x^*)^2 + l_3(x^*)^2}{2l_2(x^*)l_3(x^*)} \right]$$

and,

$$\frac{d\theta_1}{dx} = \frac{1}{\sqrt{1 - u_1^2}} \frac{du_1}{dx}, \quad (B.8)$$

$$\theta_2 = \arccos \left[ \frac{-l_1^2 + l_2(x^*) + l_3(x^*)^2}{2l_1l_3(x^*)} \right]$$

and,

$$\frac{d\theta_2}{dx} = \frac{1}{\sqrt{1 - u_2^2}} \frac{du_2}{dx}, \quad (B.9)$$

$$\theta_3 = \arccos \left[ \frac{-l_1^2 + l_2(x^*)^2 + l_3(x^*)}{2l_1l_2(x^*)} \right]$$

and,

$$\frac{d\theta_3}{dx} = \frac{1}{\sqrt{1 - u_3^2}} \frac{du_3}{dx}, \quad (B.10)$$

with $l_1$, $l_2(x^*)$, and $l_3(x^*)$ the edge lengths of an element. Note that two of the three edges are a function of $x^*$. Next step is to determine the minimal angle and the maximum derivative considering all $n$' angles corresponding to point $x^*$.

$$\varphi = \min \{ \theta_j \} \quad \text{with} \quad j = 1 \ldots n \quad (B.11)$$

$$\frac{\partial \psi}{\partial x} = \max \{ \theta_j \} \quad \text{with} \quad j = 1 \ldots n$$

(B.12)

Now a first order taylor polynomial for $\theta$ can be determined reading,

$$\theta = \varphi + \frac{\partial \psi}{\partial x} \delta x^* \quad (B.13)$$

which can be used to obtain a minimum step size by finding the intersection points of this polynomial with all first order polynomials of the other angles.

$$\delta x^* = \min \left( \frac{\varphi - \theta_j}{\frac{\partial \psi_j}{\partial x} - \frac{\partial \psi_j}{\partial x}} \right) \quad (B.14)$$

Note that this is the step size in direction of the steepest descent such that no other angle will become smaller than the currently smallest angle. Point $x^*$ can now be updated as,

$$x^* = x^* + \delta x^*$$

and the next node in the set can be considered. Subsequently, a new loop over the nodes can be done finding another minimal angle and maximum gradient until maximum number of iterations is reached or the step size becomes smaller than a defined critical step size.

In a similar manner the shape of tetrahedral elements can be optimised. One might optimise the angles between elemental faces resulting in six angles or the angles between an elemental edge and an elemental face, resulting in twelve angles.
Appendix C

Arbitrary Lagrangian Eulerian

The Arbitrary Lagrangian Eulerian (ALE) method is an ideal technique for computations with a moving interface or for preservation of good element shape. It has been developed in the early 70’s by Hirt et al. [1974], and has been exploited by many others ever since, popular references being Donea et al. [1982] and Hughes et al. [1981]. The general drive for the development of the method was that for fluid flow normally an Eulerian description is used, in which the mesh/grid is fixed in space and can therefore not capture movements of a boundary (inside the domain or at the border). This makes ALE a suitable method for FSI problems, but one should note that the method is not restricted to fluid problems.

In an ALE formulation, three reference systems can be distinguished, namely a spatial reference system (SRS), a material reference system (MRS) and a computational reference system (CRS). The SRS is a fixed reference system as is typical for an Eulerian description, the MRS moves with the material in a Lagrangian way and the CRS moves according to some prescribed displacement. In ALE methods, the finite element formulation is performed in the CRS. Following, the relation between the different reference systems will be explained shortly (Fig. C.1).

Consider a material point M at position $\mathbf{x}(t)$ with respect to the SRS. Over an infinitesimally small time step $\Delta t$ point M will move $d\mathbf{x}$ giving $\mathbf{x}(t+\Delta t) = \mathbf{x}(t)+d\mathbf{x}$. 

The material derivative of a time and place dependent variable $\varphi(x, t)$ then reads,

$$
\frac{D\varphi(x(t), t)}{Dt} = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left[ \varphi(x(t + \Delta t), t + \Delta t) - \varphi(x(t), t) \right]
$$

$$
= \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left[ \varphi(x + dx, t + \Delta t) - \varphi(x, t + \Delta t) + \varphi(x, t + \Delta t) - \varphi(x, t) \right]
$$

$$
= \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left[ dx \cdot \nabla \varphi \right] + \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left[ \varphi(x, t + \Delta t) - \varphi(x, t) \right]
$$

$$
= \vec{u} \cdot \nabla \varphi + \frac{\partial \varphi}{\partial t} \quad (C.1)
$$

The material derivative consists of a convective term, $\vec{u} \cdot \nabla \varphi$, that corresponds to the changes of $\varphi$ caused by the motion of the material, and an inertia term, $\frac{\partial \varphi}{\partial t}$, that incorporates the variations of $\varphi$ depending only on time. For example, if a fixed material point is considered in a steady flow, then it will not experience temporal changes and the latter term vanishes. If one considers a fluid particle moving with that flow, the former term will vanish also.

The above formulation is defined with respect to a fixed grid (SRS) like for an Eulerian approach, but a similar formulation can be performed with respect to a moving CRS. In Fig. C.1 is shown that at time $t$ the position of material point M with respect to the CRS is $x_3(t)$. Over a time period $\Delta t$, however, the CRS arbitrarily moves by $dx_3$ while point M moves by $dx$. Therefore, the position M with regard to the CRS at time $t + \Delta t$ is defined as,

$$
x_3(t + \Delta t) = x_3(t) + dx - dx_3 \quad (C.2)
$$
Replacing $x(t)$ by $x_g(t)$ and $dx$ by $dx - dx_g$ in Eq. (C.1) gives,

$$\frac{D\varphi(x,t)}{Dt} = \left( \mathbf{u} - \mathbf{u}_g \right) \cdot \nabla \varphi + \frac{\partial \varphi}{\partial t} \bigg|_{x} \quad (C.3)$$

This result makes sense since the convective term $(\mathbf{u} - \mathbf{u}_g) \cdot \nabla \varphi$ will vanish if a material point moves through the fluid with the velocity $\mathbf{u}$ of a fluid particle.
References


Samenvatting

Doel van het onderzoek is om numerieke methoden te ontwikkelen die het mogelijk maken fysioïstisch realistisch open en sluitgedrag en bijbehorende mechanische klepbelasting te kunnen doorrekenen. Deze berekeningen kunnen bijdragen aan een beter begrip van het functioneren van hartkleppen in de natuurlijke situatie. Verder kunnen de berekeningen toegepast worden bij de analyse van slechtfunctionerende hartkleppen. Tot slot moeten ze inzicht geven in de klepodynamica van 'tissue engineered' hartkleppen. De hartklepodynamica wordt bepaald door de interactie tussen het stromende bloed en de buiging van de klepbladen. De focus van het onderzoek ligt daarom op de vaste stof-vloeiostof interactie tussen klep en bloed.

De berekeningen worden uitgevoerd met behulp van een eindige elementen methode, waarbij een een vaste-stof mesh en een vloeiostof mesh gegenereerd wordt. De deformatie van het Lagrangian vaste-stof domein wordt beschreven met een hyperelastisch Neo-Hookean model, de stroming in het Eulerian vloeiostof domein wordt beschreven met de Navier-Stokes vergelijking. Beide domeinen worden aan elkaar gekoppeld met behulp van een Lagrange multiplier. Met deze multiplier wordt de voorwaarde afgedwongen dat de vloeiostof en vaste-stof dezelfde snelheid moeten hebben op de rand van de vaste-stof. Omdat er bij deze methode gebruik wordt gemaakt van interpolatie, wat verlies van nauwkeurigheid oplevert, is gekozen voor een uitbreiding met een adaptief meshing schema. Dit schema zorgt ervoor dat de vloeiostofmesh elke numerieke tijdstap wordt aangepast aan de positie van de vaste stof mesh, resulterend in een rand van vloeiostof-elementranden binnen het vloeiostofdomein die samenvalt met de rand van de vaste-stof. Hierdoor wordt er alleen geïnterpoeder tangentieel aan deze rand en wordt het mogelijk om, gebruik makend van de eigenschappen die inherent zijn aan de elementen, bewegende drukdiscontinuïteiten binnen het vloeiostofdomein te beschrijven. Verder geeft het de mogelijkheid om afschuifspanningen aan weerszijden van de klep te berekenen. Deze methode is gecombineerd met een contact probleem waarmee de interactie tussen de klep en de arteriële wand kan worden berekend. Zo is het mogelijk inzicht te krijgen in vloeiostofdynamica, vaste-stofdynamica en vaste-stofmechanica gedurende één of meerdere volledige hartcycli.

Samenvattend, behandelt dit proefschrift de ontwikkeling van een methode voor vaste stof-vloeiostof interactie, die het mogelijk maakt grote bewegingen en deformaties van een flexibel vaste-stof lichaam te beschrijven die veroorzaakt wordt door een pulsatiële vloeiostof stroming. De nadruk hierbij ligt op een nauwkeurige beschrijving
van de afschuifspanningen langs de vliezen en het sluit/gesloten gedrag van een klep. De methode is eerst ontwikkeld voor tweedimensionale problemen, waarna hij is uitgebreid naar drie dimensies. Uiteindelijk is de methode toegepast in een model van een aorta hartklep.
Dankwoord

Na meer dan vier jaar onderzoek bedreven te hebben in een goede atmosfeer is het op z’n plaats om even een aantal mensen te bedanken. Ten eerste wil ik Patrick Anderson bedanken die gedurende mijn gehele promotie de dagelijkse begeleiding op zich heeft genomen. Hoewel “dagelijks” een groot woord is, was hij er altijd als ik er behoefte aan had en ondanks zijn niet-biomedische achtergrond was hij de ideale begeleider voor mijn numerieke onderzoek.

Gedurende het onderzoek was het Frank Baaijens die de aanzet heeft gegeven voor mijn onderzoek en Frans van de Vosse die de afwerking voor zijn rekening heeft genomen. Frans wil ik hier speciaal bedanken voor de begeleiding in het laatste jaar waarin hij ondanks alle drukte en ondanks mijn niet altijd even fortuinlijke tijdsplanning toch de tijd nam om urgente stukken door te nemen en te bespreken. Ik heb onze gesprekken altijd als gezellig en leerzaam ervaren.

Verder heeft Jurgen de Hart natuurlijk zijn stempel gedrukt op mijn werk door de zinvolle gesprekken waarin zijn ervaringen mij veel inzicht hebben gegeven. De bereidwilligheid van Guus Segal heeft het onderzoek in belangrijke mate bevorderd evenals de inspanningen van Leo Wouters en Patrick van Brakel.

Ik ben blij met mijn betrokkenheid in het hemodynamic project, ten eerste uiteraard omdat mijn onderzoek vanuit dit project werd gefinancierd, maar tevens omdat het mijn blik heeft verbreed. De samenwerking in een groter geheel met mensen uit verschillende disciplines was uiterst leerzaam. De bijbehorende Hemodynamic-local bijeenkomsten met David, Berent, Tijmen en Sander waren altijd nuttig en, niet onbelangrijk, gezellig.

Dan zijn er nog de niet-onderzoek gerelateerde invloeden die wel degelijk belangrijk zijn. Naast de collega’s binnen MATE groep zijn er natuurlijk mijn kamergenoten Bram Sengers, Ralf Boerboom, Maria Stekelenburg, en Anke Stekelenburg die, met name in een pc-gebonden onderzoek als het mijne, zorgden voor het broodnodige sociale contact gedurende de dag. In het bijzonder bij Bram kon ik mijn opkomende problemen, frustraties en ideeën altijd kwijt en vice versa. Verder mag ook René Ubachs hier niet onttrekken met wie ik vele ontspannende koffiepauzes heb gedeeld.

Ik wil mijn ouders bedanken voor alle steun en vertrouwen die ze mij al die tijd hebben gegeven ondanks dat het geen “echte baan” was en tenslotte bedank ik natuurlijk mijn vriendin Roosmarie die, als mijn rots in de branding, de basis was waarop ik altijd kon terugvallen.
Curriculum Vitae


<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 12 1977</td>
<td>Born in Munstergeleen, the Netherlands</td>
</tr>
<tr>
<td>1989-1995</td>
<td>Secondary education, Scholengemeenschap Serviam, Sittard</td>
</tr>
<tr>
<td>1995-2001</td>
<td>Master’s degree, Mechanical Engineering, Eindhoven University of Technology</td>
</tr>
<tr>
<td>2001-2005</td>
<td>PhD, Biomedical Engineering, Eindhoven University of Technology</td>
</tr>
</tbody>
</table>